

Цифровая обработка 2D- и 3D-изображений







Н. Н. Красильников

Цифровая обработка 2D- и 3D-изображений

Допущено Учебно-методическим объединением вузов по университетскому политехническому образованию в качестве учебного пособия для студентов высших учебных заведений, обучающихся по направлению подготовки 230400 Информационные системы и технологии

Санкт-Петербург «БХВ-Петербург» 2011 УДК 681.3.06 ББК 32.973.26-018.2 К78

К78

Красильников Н. Н.

К78

Цифровая обработка 2D- и 3D-изображений: учеб. пособие. —

СПб.: БХВ-Петербург, 2011. — 608 с.: ил. — (Учебная литература для вузов)

ISBN 978-5-9775-0700-4

Изложены теоретические основы цифровой обработки 2D- и 3D-изображений. Подробно рссмотрены статистические характеристики изображений и помех, колориметрические системы и управление цветом, зрительная система и восприятие изображений, оцифровка изображений, линейная и нелинейная обработка изображений, геометрические преобразования изображений, морфологические операции и их применение, сегментация изображений, компрессия изображений без потери и с потерей информации, объемные изображения, виртуальная реальность и трехмерное сканирование, цифровое видео и телевидение. Наряду с традиционными методами обработки и компрессии неподвижных изображений и видеопоследовательностей, рассмотрены новейшие.

Для студентов, аспирантов и специалистов, изучающих методы цифровой обработки изображений и занимающихся разработкой алгоритмов и прикладных программ их обработки

> УДК 681.3.06 ББК 32.973.26-018.2

Рецензенты:

М. О. Колбанев, д-р техн. наук, проф., завкафедрой информационных управляющих систем Санкт-Петербургского государственного университета телекоммуникаций им. М. А. Бонч-Бруевича С. А. Яковлев, д-р техн. наук, проф., проф. Санкт-Петербургского государственного электротехнического университета "ЛЭТИ"

Группа подготовки издания:

Главный редактор	Екатерина Кондукова	
Зам. главного редактора	Евгений Рыбаков	
Зав. редакцией	Григорий Добин	
Редактор	Анна Кузьмина	
Компьютерная верстка	Натальи Караваевой	
Корректор	Виктория Пиотровская	
Дизайн серии	Инны Тачиной	
Оформление обложки	Елены Беляевой	
Фото	Кирилла Сергеева	
Зав. производством	Николай Тверских	

Лицензия ИД № 02429 от 24.07.00. Подписано в печать 28.03.11. Формат 70×100⁷/16. Печать офсетная. Усл. печ. л. 49,02. Тираж 1500 экз. Заказ № "БХВ-Петербург", 190005, Санкт-Петербург, Измайловский пр., 29.

Санитарно-эпидемиологическое заключение на продукцию № 77.99.60.953.Д.005770.05.09 от 26.05.2009 г. выдано Федеральной службой по надзору в сфере защиты прав потребителей и благополучия человека.

> Отпечатано с готовых диапозитивов в ГУП "Типография "Наука" 199034, Санкт-Петербург, 9 линия, 12.

Оглавление

Введение	1
Глава 1. Изображения и помехи	5
1.1. Объекты реального мира и их свойства	5
1.2. Двумерная растровая модель изображения сцен и составляющих	
их объектов	6
1.3. Векторная модель представления двумерных изображений	8
1.4. Виды изображений	10
1.5. Модели изображений	15
1.6. Пространственные спектры изображений	16
1.7. Спектральные интенсивности изображений	18
1.8. Функции автоковариации изображений	22
1.9. Плотность вероятности распределения яркости в изображениях	26
1.10. Коэффициент автокорреляции оболочек трехмерных объектов	30
Вывод формулы для коэффициента автокорреляции оболочек	
трехмерных объектов	32
Приближенный способ измерения коэффициентов автокорреляции оболочек	
трехмерных объектов	32
Зависимость коэффициента автокорреляции оболочек от угла θ	35
Зависимость коэффициента автокорреляции оболочек от их взаимного	
смещения вдоль оси z на величину ζ	36
1.11. Гистограммы оболочек реальных трехмерных изображений	36
1.12. Характеристики цвета	37
1.13. Помехи и их статистические характеристики	39
1.14. Источники флуктуационного шума в цифровых фото- и видеокамерах	
на ПЗС	44
1.15. Модель флуктуационного шума матрицы ПЗС	46
1.16. Технология маркирования изображений	47
Глава 2. Воспроизведение цвета на изображениях и управление цветом	49
2.1. Аксиомы Грассмана и законы смешения цветов	49
Опыты по уравниванию цветов	49
Аксиомы уравнивания цветов	51
2.2. Принципы построения колориметрической системы	52
2.3. Колориметрическая система RGB	55

2.4. Колориметрическая система XYZ	58
2.5. Цветовые расчеты в колориметрической системе XYZ	61
Определение цвета смеси двух цветов, заданных координатами цвета	62
Определение цвета смеси трех цветов, заданных цветовыми	
коэффициентами	62
Определение цвета смеси трех цветов, заданных координатами цветностей	
и величинами яркостей	63
Определение доминирующей длины волны и чистоты цвета	63
2.6. Цветовая температура	65
2.7. Другие колориметрические системы	66
Колориметрическая система Манселла	66
Колориметрическая система Мак-Адама	67
Колориметрическая система L*a*b*	68
Колориметрические системы HLS и HSB	69
Колориметрические системы СМУ и СМУК	69
2.8. Вывод на экран и печать цветных изображений	70
2.9. Управление цветом, калибровка, основы систем CMS	74
2.10. Баланс белого	77
Автоматический способ установки баланса белого	78
Способ с использованием ручной предустановки	79
Ручная установка баланса белого	79
2.11. Каналы, слои и изображения с прозрачным фоном	79
Каналы	79
Изобразительные слои изображения	80
Изображения с прозрачным фоном	81
Цветовая обработка изображения в области, ограниченной произвольным	
контуром	81
2.12. Работа с цветом в графических редакторах	82
F	05
тлава 5. зрительная система и восприятие изооражении	03
3.1. Проблема оценки качества воспроизведения изображений	85
3.2. Краткие сведения о строении зрительной системы	88
Оптика глаза	88
3.3. Адаптация зрительной системы к освещенности	
и контрастная чувствительность	92
3.4. Разрешающая способность зрительной системы в пространстве	
(острота зрения)	97
3.5. Инерционность зрения	101
3.6. Восприятие движения	104
3.7. Восприятие цвета	104
3.8. Восприятие объема	108
3.9. Обнаружение и опознавание изображений объектов в пороговых	
условиях наблюдения	109

3.11. Обобщенная функциональная модель зрительной системы	3.10. Функциональные модели зрительной системы	111
3.12. Особенности восприятия семантических изображений зрительной системой человека. 12 3.13. Методы оценки качества воспроизведения изображений. 12 3.14. Корреляционно-энергетический критерий качества изображения, кодированного по стандарту MPEG-2 12 Глава 4. Оцифровка изображений 13 4.1. Представление изображений в памяти компьютера. 13 4.2. Теорема Котельникова 13 4.3. Помеха пространственной дискретизации 14 4.4. Метод уменьшения помехи пространственной дискретизации изображений. 14 4.5. Структуры расположения отсчетов при дискретизации изображений. 14 4.6. Интерполация при воспроизведении изображений 15 4.7. Квантование изображений по яркости. 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространствелной дискретизации 15 4.11. Сканеры. 15 4.12. Цифровые фото- и видеокамеры 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульской характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Алертурные искажения изображения, обусловленные емазом вследст	3.11. Обобщенная функциональная модель зрительной системы	113
системой человека	3.12. Особенности восприятия семантических изображений зрительной	
3.13. Методы оценки качества воспроизведения изображений 12 3.14. Корреляционно-энергетический критерий качества изображения, 12 Глава 4. Оцифровка изображений 13 4.1. Представление изображений в памяти компьютера. 13 4.1. Представление изображений в памяти компьютера. 13 4.2. Теорема Котельникова 13 4.3. Помеха пространственной дискретизации 13 4.4. Метод уменьшения помехи пространственной дискретизации изображений 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости. 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.11. Сканеры. 15 4.12. Цифровые фото- и видеокамеры 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 17 Антертурные искажения изображения, обусловленные есмехрамой области 16 5.3. Метод фильтрации цифровых изображен	системой человека	120
3.14. Корреляционно-энергетический критерий качества изображения, 12 Кодированного по стандарту MPEG-2 12 Глава 4. Оцифровка изображений 13 4.1. Представление изображений в памяти компьютера. 13 4.2. Теорема Котельникова 13 4.3. Помеха пространственной дискретизации 14 4.4. Метод уменьшения помехи пространственной дискретизации изображений 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 15 4.7. Квантование изображений по яркости 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображения, обусловленные несовершенством пресобразователей изображения, обусловленные несовершенством пресобразователей изображения, обусловленные несовер	3.13. Методы оценки качества воспроизведения изображений	127
кодированного по стандарту MPEG-2 12 Глава 4. Оцифровка изображений 13 4.1. Представление изображений в памяти компьютера 13 4.2. Теорема Котельникова 13 4.3. Помеха пространственной дискретизации 13 4.4. Метод уменьшения помехи пространственной дискретизации изображений 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости. 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений. 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 16 5.1. Цели применения линейной фильтрации изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображения, обусловленные	3.14. Корреляционно-энергетический критерий качества изображения,	
Глава 4. Оцифровка изображений 13 4.1. Представление изображений в памяти компьютера 13 4.2. Теорема Котельникова 13 4.3. Помеха пространственной дискретизации 13 4.4. Метод уменьшения помехи пространственной дискретизации изображений 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости. 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота 15 пространственной дискретизации 15 4.12. Цифровые фото- и видеокамеры 16 5.1. Цели применения линейной фильтрации изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений 17 Апертурные искажения изображений 17 Апертурные искажения изображений, обусловленные енесовершенством 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные и	кодированного по стандарту MPEG-2	129
4.1. Представление изображений в памяти компьютера	Глава 4. Оцифровка изображений	133
4.2. Теорема Котельникова 13 4.3. Помеха пространственной дискретизации 13 4.4. Метод уменьшения помехи пространственной дискретизации изображений 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.11. Сканеры 15 4.12. Цифровые фото- и видеокамеры 16 Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений в спектральной области 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображения, обусловленные неточной фокусировкой оптической системы 17 Апертурные искажения изображения, обусловленные есмазом вследствие недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения, обусловленные несовершенством воспроизоб экспозиции при съемке движущ	4.1. Представление изображений в памяти компьютера	133
4.3. Помеха пространственной дискретизации 13 4.4. Метод уменьшения помехи пространственной дискретизации 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости. 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота 15 пространственной дискретизации 15 4.11. Сканеры. 15 4.12. Цифоровые фото- и видеокамеры 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифоровых изображений путем их свертки 16 5.4. Апертурные искажения изображений в спектральной области 16 5.4. Апертурные искажения изображения, обусловленные неточной фокусировкой 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, об	4.2. Теорема Котельникова	134
4.4. Метод уменьшения помехи пространственной дискретизации 14 4.5. Структуры расположения отсчетов при дискретизации изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости. 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.11. Сканеры. 15 4.12. Цифровые фото- и видеокамеры 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображения, обусловленные смазом вследствие необразователей изображения, обусловленные смазом вследствие необранством преобразователей изображения, обусловленные несовершенством преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством воспроизводящии устройств. 17 Апертурные искажения изображения, обусловленные несовершенством воспроизводящих устройств. <	4.3. Помеха пространственной лискретизации	
4.5. Структуры расположения отсчетов при дискретизации изображений. 14 4.6. Интерполяция при воспроизведении изображений. 14 4.6. Интерполяция при воспроизведении изображений. 14 4.7. Квантование изображений по яркости. 15 4.8. Ложные контуры и методы их ослабления. 15 4.9. Моды представления изображений. 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.11. Сканеры. 15 4.12. Цифровые фото- и видеокамеры 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений, обусловленные смазом вследствие необразователей изображения, обусловленные смазом вследствие необразователей изображения, обусловленные несовершенством преобразователей изображения, обусловленные несовершенством преобразователей изображения собусловленные несовершенством 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения сигнал 17 Апертурные искажения изображения, обусловленные несовершенством воспроизводящих устройств. 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения	4.4. Метол уменьшения помехи пространственной лискретизации	
4.6. Интерполяция при воспроизведении изображений 14 4.7. Квантование изображений по яркости 15 4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.11. Сканеры 15 4.12. Цифровые фото- и видеокамеры 16 Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений. 17 Апертурные искажения изображения, обусловленные неточной фокусировкой оптической системы. 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения, обусловленные несовершенством 17 Апертурные искажения в сигнал	4.5. Структуры расположения отсчетов при дискретизации изображений	146
4.7. Квантование изображений по яркости	4.6. Интерполяция при воспроизведении изображений	149
4.8. Ложные контуры и методы их ослабления 15 4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота 15 пространственной дискретизации 15 4.11. Сканеры 15 4.12. Цифровые фото- и видеокамеры 16 Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображен	4.7. Квантование изображений по яркости	150
4.9. Моды представления изображений 15 4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации 15 4.11. Сканеры 15 4.12. Цифровые фото- и видеокамеры 16 Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения, обусловленные неточной фокусировкой оптической системы 17 Апертурные искажения изображения, обусловленные смазом вследствие недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения в сигнал 17 Апертурные искажения изображения собусловленные несовершенством преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения в сигнал 17 Апертурные искажения изображения сигнал 17	4.8. Ложные контуры и методы их ослабления	154
4.10. Контраст, детальность, разрешение принтера и частота 15 пространственной дискретизации 15 4.11. Сканеры 15 4.12. Цифровые фото- и видеокамеры 16 Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения изображений, обусловленные неточной фокусировкой 17 оптической системы 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Асертурны искажения изобра	4.9. Моды представления изображений	156
пространственной дискретизации 15 4.11. Сканеры	4.10. Контраст, детальность, разрешение принтера и частота	
4.11. Сканеры	пространственной дискретизации	157
4.12. Цифровые фото- и видеокамеры 16 Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения изображений 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения сигнал 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения в сигнал 17 Апертурные искажения изображения обусловленные несовершенством 17	4.11. Сканеры	159
Глава 5. Линейная фильтрация изображений 16 5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения, обусловленные неточной фокусировкой 17 Оптической системы 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные несовершенством 17 преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством 17 Боспроизводящих устройств. 17 Кажения, обусловленные турбулентностью атмосферы 17 <td< td=""><td>4.12. Цифровые фото- и видеокамеры</td><td> 161</td></td<>	4.12. Цифровые фото- и видеокамеры	161
5.1. Цели применения линейной фильтрации изображений 16 5.2. Метод фильтрации цифровых изображений путем их свертки 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения, обусловленные неточной фокусировкой 17 оптической системы 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные песовершенством 17 Апертурные искажения изображения, обусловленные насовершенством 17 Боспроизводящих устройств. 17 5.5. Фильтрация изображений, искаженных гауссовым б	Глава 5. Линейная фильтрация изображений	163
5.2. Метод фильтрации цифровых изображений путем их свертки 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения, обусловленные неточной фокусировкой 17 оптической системы 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 <	5.1. Пели применения линейной фильтрации изображений	163
с импульсной характеристикой 16 5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения, обусловленные неточной фокусировкой 17 оптической системы 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные несовершенством 17 преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством 17 преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством 17 Лопртурные искажения изображения, обусловленные несовершенством 17 Лопртурные искажения изображения, обусловленные песовершенством 17 Лороизводящих устройств. 17 Лороизводящих устройств. 17 5.5. Фильтрация изображений, искаженных гауссовым белым шумом 17 5.6. Коррекция апертурных искажений в изображени	5.2 Метол фильтрации цифровых изображений путем их свертки	
5.3. Метод фильтрации цифровых изображений в спектральной области 16 5.4. Апертурные искажения изображений 17 Апертурные искажения, обусловленные неточной фокусировкой 17 оптической системы 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 недостаточно короткой экспозиции при съемке движущегося объекта 17 Апертурные искажения изображения, обусловленные несовершенством 17 преобразователей изображения в сигнал 17 Апертурные искажения изображения, обусловленные несовершенством 17 Лертурные искажения изображения, обусловленные песовершенством 17 Лертурные искажения изображения, обусловленные несовершенством 17 Лертурные искажения изображения, обусловленные песовершенством 17 Лопроизводящих устройств 17 Л.б. Коррекция апертурных искажений в изображениях при наличии шума<	с импульсной характеристикой	164
5.4. Апертурные искажения изображений. 17 Апертурные искажения, обусловленные неточной фокусировкой 17 Оптической системы. 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Апертурные искажения изображения, обусловленные песовершенством 17 Апертурные искажения изображения, обусловленные песовершенством 17 Боспроизводящих устройств. 17 Упранция изображений, искаженных гауссовым белым шумом 17 5.6. Коррекция апертурных искажений в изображениях при наличии шума 18 5.7. Принципы обработки изображений,	5.3. Метол фильтрации цифровых изображений в спектральной области	
Апертурные искажения, обусловленные неточной фокусировкой 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные смазом вследствие 17 Апертурные искажения изображения, обусловленные несовершенством 17 Боспроизводящих устройств. 17 Искажения, обусловленные турбулентностью атмосферы 17 5.5. Фильтрация изображений, искаженных гауссовым белым шумом 17 5.6. Коррекция апертурных искажений в изображениях при наличии шума 18 5.7. Принципы обработки изображений, основанные на учете особенностей их восприятия зрительной системой 18	5.4. Апертурные искажения изображений	
оптической системы	Апертурные искажения, обусловленные неточной фокусировкой	
Апертурные искажения изображения, обусловленные смазом вследствие недостаточно короткой экспозиции при съемке движущегося объекта	оптической системы	171
недостаточно короткой экспозиции при съемке движущегося объекта	Апертурные искажения изображения, обусловленные смазом вследствие	
Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения в сигнал	недостаточно короткой экспозиции при съемке движущегося объекта	174
преобразователей изображения в сигнал	Апертурные искажения изображения, обусловленные несовершенством	
Апертурные искажения изображения, обусловленные несовершенством воспроизводящих устройств	преобразователей изображения в сигнал	176
воспроизводящих устройств	Апертурные искажения изображения, обусловленные несовершенством	
Искажения, обусловленные турбулентностью атмосферы	воспроизводящих устройств	178
 5.5. Фильтрация изображений, искаженных гауссовым белым шумом	Искажения, обусловленные турбулентностью атмосферы	178
 5.6. Коррекция апертурных искажений в изображениях при наличии шума	5.5. Фильтрация изображений, искаженных гауссовым белым шумом	179
5.7. Принципы обработки изображений, основанные на учете особенностей их восприятия зрительной системой	5.6. Коррекция апертурных искажений в изображениях при наличии шума	183
их восприятия зрительной системой18	5.7. Принципы обработки изображений, основанные на учете особенностей	
1 1	их восприятия зрительной системой	189

5.8. Ослабление шума на изображениях методом адаптивной анизотропной	
фильтрации	191
5.9. Фильтрация изображений в ограниченной пространственной области	195
5.10. Подчеркивание световых границ на изображениях	197
5.11. Метод нерезкого маскирования	198
5.12. Другие линейные методы подчеркивания границ на изображениях	202
5.13. Ослабление дефектов изображения путем размытия	205
5.14. Пределы увеличения разрешения	206
5.15. Линейная фильтрация изображений в графических редакторах	211
Глава 6. Нелинейная обработка изображений	215
6.1. Искажения воспроизведения градаций яркости	215
Фотоматериалы	216
Преобразователи изображения в видеосигнал	217
Воспроизводящие устройства	217
6.2. Коррекция искажений воспроизведения градаций яркости	218
6.3. Видоизменение гистограмм	220
6.4. Методы согласования динамического диапазона изображений	
с динамическим диапазоном дисплея	222
6.5. Проблема разрежения шкалы квантования при коррекции	
световых характеристик	230
6.6. Тоновая коррекция	233
6.7. Фотографирование сцен с большим динамическим диапазоном яркостей	234
6.8. Методы отображения сцен с большим динамическим диапазоном	
яркостей на дисплеях с ограниченным динамическим диапазоном	237
6.9. Билатеральная фильтрация изображений	241
Применение билатеральной фильтрации для улучшения воспроизведения	
небольших малоконтрастных деталей на изображениях сцен с большим	
динамическим диапазоном на носителях, у которых динамический	
диапазон сравнительно невелик	244
Применение билатеральной фильтрации для ослабления заметности	
ложных контуров на проквантованном изображении	245
Применение билатеральной фильтрации для ослабления заметности	
гауссова шума на изображении	246
Применение билатеральной фильтрации для выделения текстур	- · -
из изображений	247
6.10. Методы коррекции апертурных искажений изображений, основанные	
на использовании их семантической структуры	248
6.11. 1 омоморфная фильтрация изображений	253
6.12. Ослаоление искажений фотокопии, обусловленных неравномерностью	•
освещенности поверхности фотографируемого оригинала	256
6.13. Медианная и ранговая фильтрация изображений	259
6.14. Адаптивная медианная фильтрация изображений	263

6.15. Некоторые другие виды нелинейной фильтрации изображений	267
Среднегеометрический фильтр	267
Среднегармонический фильтр	267
Контргармонический фильтр	268
6.16. Методы фильтрации гауссова шума, основанные на учете особенностей	
восприятия семантических изображений зрительной системой	268
Метод, базирующийся на использовании кусочно-гладкой модели	
изображения	269
Метод, базирующийся на регенерации контурных, градиентных	
и фактурных компонентов	272
6.17. Нелинейные преобразования изображений в области, ограниченной	
произвольным контуром	276
6.18. Методы нелинейной обработки изображений	
в графических редакторах	277
Г 7 Г	101
1 лава 7. 1 еометрические преооразования изооражении	281
7.1. Геометрические искажения изображений	281
7.2. Аффинные преобразования изображений	282
Сдвиг изображения	283
Поворот изображения относительно начала координат	283
Изменение масштаба изображения	284
7.3. Дисторсия и методы ее коррекции	285
7.4. Перспективные искажения изображений и их коррекция	287
7.5. Коррекция геометрических искажений изображений в общем случае	289
7.6. Метод определения функций, описывающих геометрические искажения	292
7.7. Наложение текстуры на полигональную оболочку объекта	293
7.8. Увеличение изображений	296
Интерполяция нулевого порядка	296
Билинейная интерполяция	299
Бикубическая интерполяция	300
Интерполяция посредством функций sin x/x	302
7.9. Метод триангуляции	304
7.10. Уменьшение изображений	306
7.11. Интерполяция при использовании матриц Байера	308
Глава 8. Морфологические операции и их применение	315
8.1. Морфологические операции	315
8.2. Бинарные изображения и их характеристики	315
8.3. Логические операции над бинарными изображениями	317
8.4. Понятие связности	318
8.5. Логическая апертура	320
8.6. Морфологические операции дилатации и эрозии бинарных изображений	322
8.7. Использование шаблонов при выполнении морфологических операций	326

8.8. Морфологические операции бинарного открытия	
и бинарного закрытия	329
8.9. Морфологические операции утончения и утолщения бинарных	
изображений	333
8.10. Морфологические операции: усечение, мост и удаление центрального	
пиксела в конфигурациях с <i>Н</i> -связностью бинарных изображений	335
8.11. Заполнение (заливка) областей бинарных изображений	339
8.12. Построение скелета объекта на бинарном изображении	340
8.13. Производные морфологические операции над бинарными	
изображениями	341
8.14. Использование "интегральных" шаблонов при выполнении	
морфологических операций над бинарными изображениями	343
8.15. Морфологические операции дилатации, эрозии, открытия	
и закрытия полутоновых изображений	344
8.16. Морфологические операции tophat, bothat и морфологический градиент	
при фильтрации полутоновых изображений	347
8.17. Фильтрация полутоновых изображений посредством	
логической апертуры	350
Глава 9. Сегментация изображений	353
9.1. Сегментация изображений на отдельные области	353
9.2. Обнаружение на изображении точек и отрезков прямых линий	354
9.3. Обнаружение перепадов яркости на изображении	355
Перекрестный градиентный оператор Робертса	357
Градиентный оператор Превитта	358
Градиентный оператор Собела	358
9.4. Методы выделения контуров на изображениях	359
9.5. Метод сегментации путем связывания пирамиды	365
9.6. Пороговая обработка	368
9.7. Метод выращивания областей	370
9.8. Метод разделения и слияния областей	371
9.9. Метод сегментации на основе движения	372
9.10. Сегментация, основанная на применении морфологических операций	
эрозии и наращивания	374
9.11. Особенности сегментации изображений в случае, когда признак,	
по которому она выполняется, является векторной величиной	376
9.12. К-метод сегментации изображений	378
9.13. Сегментация изображений на основе статистических характеристик	
текстуры	378
9.14. Интерактивная сегментация изображений	380
9.15. Гранулометрия	383
9.16. Обнаружение и различение на изображении объектов	
известной формы	385

Глава 10. Объемные изображения	
10.1. Стереоскопические изображения	
Методы получения стереоизображений	390
Методы наблюдения стереоизображений	391
10.2. 2,5D-изображения	395
10.3. Воксельная модель представления трехмерных изображений	396
10.4. Векторная полигональная модель представления трехмерных	
изображений	398
10.5. Модель освещения	401
10.6. Определение нормали к поверхности и вектора отражения	403
10.7. Методы закраски граней	406
Метод Гуро	407
Метод Фонга	408
10.8. Трассировка лучей	408
Метод прямой трассировки лучей	408
Метод обратной трассировки лучей	409
10.9. Проекции трехмерных изображений на плоскость	410
10.10. 3D-сканеры	411
Контактный метод сканирования и реализующие его сканеры	412
Бесконтактные методы сканирования и реализующие их сканеры	412
10.11. Трехмерное сканирование, основанное на диффузном отражении	
света сканируемыми объектами	420
10.12. Определение координаты глубины по 2D-изображению	426
10.13. 3D-печать	431
10.14. Обработка оболочек трехмерных изображений	434
Глава 11. Цифровое видео и телевидение	439
11.1. Динамические изображения	439
11.2. Съемка исхолного материала.	440
11.3. Форматы растров цветных изображений	443
Формат 4:4:4	444
Формат 4:2:2	444
Формат 4:2:0	445
Формат 4:1:1	446
11.4. Элементы нелинейного монтажа	446
11.5. Разбивка видеоматериала на отдельные сцены	449
11.6. Создание переходов	450
11.7. Морфинг и варпинг	453
11.8. Принципы анимации	455
11.9. Спецэффекты	457
11.10. Цифровая запись видеоданных	460
11.11. Мобильное телевидение и видео	464

11.12. Объемное телевидение и видео	466
11.13. Виртуальные студии	468
11.14. Аппаратные средства и пакеты программ для работы	
с цифровым видео	470
Глава 12. Сжатие изображений без потерь информации	473
12.1. Проблема сжатия изображений	473
12.2. Избыточность изображений	474
12.3. Декорреляция сигнала изображения	477
12.4. Кодирование длин серий	478
12.5. Кодирование методом LZW	480
12.6. Метод кодирования Хаффмена	482
12.7. Арифметическое кодирование	484
12.8. Некоторые детали алгоритмов арифметического кодирования	488
Сжатие	488
Протяженность цепочек кодируемых символов	490
Адаптивное арифметическое кодирование	490
12.9. Разделение кодируемого сигнала изображения на контексты	491
12.10. Проблема накопления ошибок преобразования при сжатии изображений	494
12.11. Предварительная логическая фильтрация изображений для увеличения	
степени их сжатия кодерами без потерь информации	497
12.12. Экспериментальное исследование метода предварительной	
обработки изображений	498
12.13. Кодирование битовых плоскостей	502
Глава 13. Сжатие изображений с потерями информации	507
13.1 Лифференциальная колово-импульсная молуляция	507
13.2. Колирование с использованием ортогональных преобразований	510
13.3 Лискретное косинусное преобразование	514
13.4 Метол оптимального распределения двоичных единиц кола между	
спектральными коэффициентами	516
13.5 Сжатие изображений в формате IPEG	519
13.6. Вейвлет-преобразование	522
13.7. Сжатие изображений на основе вейвлет-преобразования	526
13.8. Сжатие изображений в формате IPEG 2000	530
13.9 Фрактальное колирование	531
13.10 Сжатие изображений в формате MPEG-2	533
13.11. Определение векторов движения	536
13.12 Сжатие изображений в формате MPFG-4	538
13.13. Сжатие изображений в формате Н 264	540
13.14 Метол сжатия основанный на использовании 3D-молелей	540
лля представления движущихся объектов на 2D-изоблажениях	541
13 15 Обзор форматов записи изображений	543
12.12. Cosop wopmarob Jamen noopancennin	

Глава 14. Виртуальная реальность	547
14.1. Виртуальная реальность и области ее применения	547
14.2. Объектно-ориентированный подход	548
14.3. Простейшие сцены в виртуальном мире	550
14.4. Более сложные сцены в виртуальном мире	553
14.5. Объекты, обеспечивающие путешествие в виртуальном мире	556
14.6. Объекты, обеспечивающие взаимодействие пользователя с объектами	
виртуального мира	559
14.7. Алгоритмические языки, применяемые в системах виртуальной	
реальности	564
14.8. Имитация искусственного интеллекта	565
14.9. Тренажеры	566
Тренажер NASA, имитирующий работу аэропортов	566
Тренажер, имитирующий взлет шаттла	567
Виртуальный тренажер бензопилы, основанный на принципе	
смешанной реальности	567
Тренажер "комната виртуальной реальности"	568
14.10. Компьютерные игры	568
14.11. Периферийные устройства	569
Шлем виртуальной реальности	569
Системы трекинга	570
Джойстик и руль автомобиля	570
Перчатки виртуальной peaльности (VR Gloves)	570
Всенаправленная беговая дорожка	571
14.12. Методы записи движений человеческого тела	572
Заключение	575
Литература	577
Предметный указатель	589

Введение

В основу данной книги положены курсы лекций, поставленные и читаемые автором в Санкт-Петербургском государственном университете аэрокосмического приборостроения, а также результаты научных исследований, выполненных автором и руководимым им коллективом в данной области за последние более чем 40 лет.

Эта книга, по мнению автора, отличается от других книг на аналогичную тему, в основном переводных, тремя особенностями:

- изложение материала ведется с учетом свойств получателя информации, как правило, это зрительная система человека;
- в одном контексте рассматривается обработка и отображение двумерных и трехмерных изображений;
- в книге наряду с уже ставшими классическими методами рассматриваются новейшие методы обработки изображений.

Впервые изображения в цифровой форме были переданы по трансокеанскому подводному кабелю между Лондоном и Нью-Йорком в начале 1920-х годов посредством системы "Бартлейн", однако интенсивное развитие методов цифровой обработки и передачи изображений начинается лишь с 60-х годов прошлого века [23]. Связано это с появлением и быстрым развитием вычислительной техники, которая в это время становится доступной. Первыми разработчиками и исследователями этой новой технологии стали специалисты, работающие в области прикладной математики, программирования и теории связи, что в значительной степени определило направление исследований. Если обратиться к наиболее значимым публикациям того времени, например, к [83], то легко заметить, что постановка задач была примерно такой. Имеется искаженное изображение, которое рассматривается как случайная функция двух переменных (координат) и не более, статистические характеристики которой известны. Требуется найти такое преобразование, в результате которого восстановленное изображение минимально бы отличалось от неискаженного изображения, считающегося известным. В качестве критерия отличия при этом принимался средний квадрат попиксельной разности изображений.

Чрезмерная упрощенность такого подхода, неизбежная на начальном этапе исследований, заключается в том, что при этом не учитываются ни свойства получателя

информации (как правило, это зрительная системы человека), ни важные для зрителя характеристики обрабатываемого изображения, например, свойства контурного компонента изображения, т. е. его гладкость, непрерывность и т. д. Отсутствие учета свойств получателя информации и характеристик изображений, важных для зрителя при таком подходе, не позволяет синтезировать оптимальные алгоритмы их обработки не только в системах, предназначенных для визуального наблюдения, но и в автоматических системах наблюдения, а это создает серьезные трудности для дальнейшего развития эффективных алгоритмов обработки изображений.

В отличие от такого подхода в предлагаемой книге изложение материала проводится применительно не к абстрактным двумерным массивам чисел, над которыми совершаются некоторые математические преобразования, как это часто практикуется при изложении подобных вопросов, а применительно к реальным семантическим изображениям, характеризуемым вполне определенными свойствами. При этом целью обработки является улучшение изображений в рамках того или иного критерия качества, полученного на основе модели зрения. Такой подход обусловил включение в книгу глав, посвященных описанию характеристик семантических изображений, зрительной системы человека и ее моделей, а также основных положений колориметрии. При изложении материала показаны имеющиеся ограничения в достижении результатов обработки, обусловленные самой природой используемых процессов.

В книге рассмотрен широкий круг вопросов, начиная от ввода изображений в память компьютера, их обработки и вывода либо на экран монитора, в случае мультимедиа и цифрового видео, либо на печать в случае полиграфии. Большое внимание уделено вопросам оцифровки изображений, вопросам сжатия, линейной и нелинейной обработкам, вопросам коррекции геометрических искажений, а также трехмерным изображениям.

В этой книге дается систематическое изложение теоретических основ цифровой, в основном компьютерной, обработки изображений с иллюстрацией их практического применения в наиболее популярных графических редакторах. Такой подход к теме позволяет не только сознательно использовать существующие методы цифровой обработки изображений, применяемые в графических редакторах, но и создавать новые, а также готовить квалифицированных специалистов в этой области.

Кроме того, в книге рассмотрены методы получения 3D-изображений путем сканирования трехмерных объектов, а также методы их обработки. Значительное внимание уделено практическому применению рассматриваемых методов в цифровом видео и системах виртуальной реальности. Это особенно актуально, поскольку в последнее время имеет место значительное усиление интереса к трехмерным кино и телевидению.

И, наконец, в данной книге приведено описание эффективных методов обработки изображений, развитых в самое последнее время, таких как:

методы ослабления аддитивного гауссового шума на изображении путем его адаптивной фильтрации, основанной на учете свойств зрительной системы;

- методы фотографирования и отображения сцен с большим динамическим диапазоном яркости;
- 🗖 методы интерполяции, применяемые при использовании матриц Байера;
- методы интерполяции, направленные на устранение ступенчатости границ на изображениях при их увеличении,

а также ряда других.

На протяжении всей книги изложение материала иллюстрируется практическими примерами обработки изображений посредством наиболее распространенных в нашей стране компьютеров IBM с использованием наиболее популярных графических редакторов.

Книга ориентирована на специалистов, работающих в области обработки изображений, аспирантов и студентов старших курсов соответствующих специальностей.

Автор считает своим долгом поблагодарить Красильникову Ольгу Ивановну за участие в обсуждении и редактировании рукописи книги на этапе ее подготовки к изданию. Автор также выражает благодарность руководству Санкт-Петербургского государственного университета аэрокосмического приборостроения и руководству Северо-западной академии государственной службы за активную поддержку издания книги.

Глава 1



Изображения и помехи

1.1. Объекты реального мира и их свойства

Посредством зрения мы воспринимаем образ окружающего нас мира. Этот мир трехмерный и состоит из множества, как правило, трехмерных объектов. Каждый объект имеет оболочку (поверхность), которая отделяет его от окружающего пространства. Оболочка каждого объекта имеет присущие ей отражательные характеристики, характеристики прозрачности и характеристики излучения, которые по ее поверхности могут быть неоднородными. Для того чтобы мы могли наблюдать окружающие нас объекты, они должны быть освещены, если только не являются самосветящимися.

В процессе наблюдения свет от рассматриваемых объектов попадает на сетчатки глаз и формирует на них изображения, которые представляют собой центральные проекции рассматриваемых объектов. Распределение интенсивностей для всего спектра видимого излучения по поверхностям сетчаток определяется отражательными характеристиками, характеристиками прозрачности и характеристиками излучения поверхностей объектов, их положением относительно точки наблюдения, спектральными характеристиками источников освещения, а также положением источников освещения, относительно точки наблюдения.

Поскольку положение наблюдателя, положение источников освещения и спектральные характеристики источников освещения могут изменяться в широких пределах, то одному и тому же объекту может соответствовать бесчисленное количество проекций на сетчатках глаз.

Наиболее устойчивыми признаками объектов на изображениях, формирующихся на сетчатках глаз, являются их контуры, поскольку они представляют собой проекции оболочек объектов, которые не зависят от условий освещения. Вот почему контурный, или силуэтный, рисунок позволяет легко узнавать изображаемый объект [46, 47, 72].

Зрительная система при наблюдении трехмерной сцены формирует ее *трехмерное представление* в коре головного мозга. Это необходимо уже хотя бы для того, чтобы мы имели возможность манипулировать различными объектами, составляющими сцену, и не сталкиваться с ними при своем перемещении в пространстве, а также распознавать их при наблюдении под различными ракурсами [169]. С самого начала отметим, что для формирования в зрительной системе объемного представления объектов, составляющих сцену, наличие бинокулярного зрения не является обязательным. Объемное представление трехмерных объектов способны формировать и люди, не имеющие возможности бинокулярно рассматривать наблюдаемую сцену, а также многие животные и птицы, вообще лишенные бинокулярного зрения. Поскольку при формировании объемного представления незнакомого трехмерного объекта в зрительной системе единственным источником информации о нем являются его двумерные проекции на сетчатки глаз, то для этого необходимо располагать рядом таких проекций, полученных, например, при различных ракурсах наблюдения.

Из сказанного следует, что при наблюдении сцен, изображенных, например, на фотографиях, на сетчатках глаз создаются проекции подобные тем, что создаются при непосредственном наблюдении самих изображаемых сцен, благодаря чему создаваемое впечатление близко к тому, которое имеет место при наблюдении изображаемой натуры. На этом, собственно, и базируются такие виды изобразительного искусства, как живопись, фотография и кино.

Впечатление реальности еще больше усиливается, если зрителю предлагают для наблюдения трехмерное изображение сцены, поскольку при этом у него появляется возможность его рассматривать при различных ракурсах наблюдения, как в жизни. Трехмерные изображения лежат в основе развивающихся в настоящее время новых технологий — технологий виртуальной реальности.

1.2. Двумерная растровая модель изображения сцен и составляющих их объектов

В компьютерной графике используют растровую и векторную модели двумерного изображения [82]. В основе растровой модели лежит *растр* — матрица пикселов, которые представляют интенсивность соответствующих участков изображения. В качестве поясняющего примера на рис. 1.1 приведен фрагмент растрового изображения с ортогональным расположением пикселов, линейные размеры которых в обоих направлениях одинаковы. Помимо растра с ортогональным расположением пикселов возможны и другие растры, например, растр с шахматным расположением пикселов, однако они применяются редко [98].



Рис. 1.1. Растровое изображение

Растровое изображение характеризуется разрешением, которое определяется количеством пикселов на единицу длины. Чем больше пикселов приходится на единицу длины, тем выше разрешение и тем более мелкие детали могут быть воспроизведены на изображении. Для того чтобы изображение можно было обрабатывать посредством компьютера, его представляют в цифровой форме. В случае полутонового черно-белого, т. е. не цветного изображения, это означает, что интенсивность (яркость) каждого пиксела представляется числом, которое обычно лежит в пределах от 0 до 255. Другими словами, полутоновое черно-белое изображение представляется в виде двумерной матрицы, состоящей обычно из 8-разрядных двоичных чисел. Для представления цветного изображения в цифровой форме используют уже три матрицы, каждая из которых, как правило, состоит из 8-разрядных двоичных чисел, реже из 16-разрядных. При этом элементы каждой из этих матриц представляют интенсивности красного, зеленого и синего компонентов цвета пиксела, поскольку цвет каждого из пикселов растра синтезируется путем смешения красного, зеленого и синего компонентов цвета пиксела, поскольку цвет каждого из пикселов растра синтезируется путем смешения красного, зеленого и синего компонентов цвета пиксела.

При выборе разрешения изображения исходят из того, чтобы зрителю изображение казалось непрерывным, т. е. чтобы зритель не видел на изображении растровой структуры. С этой целью количество пикселов на единицу длины в изображении выбирают, исходя из остроты (разрешающей способности) зрения. Известно, что при наблюдении двух объектов, угловое расстояние между которыми меньше одной угловой минуты, они сливаются в один объект. Из этого следует, что для отсутствия заметности растровой структуры угловое расстояние между формирующими ее пикселами β должно быть меньше одной угловой минуты или, в крайнем случае, равняться ей. Рисунок 1.2, на котором представлен растр, поясняет изложенное.



Рис. 1.2. К определению выбора разрешения

Поскольку угол β зависит от расстояния наблюдения, линейное расстояние между центрами пикселов Δ (шаг пространственной дискретизации изображений) также зависит от этого угла. Записывая очевидное геометрическое соотношение $\Delta/L = tg\beta$, имеем $\Delta = Ltg\beta$, при этом общее число пикселов N, составляющих растр, будет равно $N = hl/\Delta^2$, что после подстановки значения Δ дает $N = hl/(Ltg\beta)^2$. Поскольку tg от одной минуты равен 2,909 · 10⁻⁴, получим окончательно

$$N = \frac{11817146hl}{L^2},$$
 (1.1)

а число строк z в изображении, как не трудно видеть, будет равно

$$z = \frac{3437h}{L} \,.$$

Из выражения (1.1) следует, что минимально необходимое количество пикселов, составляющих растр изображения, определяется не только его размером, но также в сильной степени условиями наблюдения изображения, т. е. расстоянием, с которого ведется наблюдение. В телевидении расстояние наблюдения принимается равным $L \cong (5..6)h$. Учитывая, что согласно телевизионному стандарту отношение ширины изображения к его высоте l/h принято близким к 4/3, число строк в телевизионном изображении в соответствии с приведенными расчетами должно быть $z \cong 574..689$, что близко к принятому по стандарту значению 625 [93]. Аналогичным образом решается задача применительно к компьютерным дисплеям, но с учетом специфики наблюдения изображения на экране монитора.

Количество растровых элементов в изображении наряду с разрядностью двоичного кода, посредством которого представляется яркость (в случае цветного изображения интенсивности красного, зеленого и синего цветов), определяет необходимый объем памяти для хранения изображения. В случае черно-белого полутонового изображения (grayscale) необходимый объем памяти равен M = 8N бит, в случае цветного изображения с использованием моды RGB (True Color) необходимый объем памяти составит уже M = 24N бит. Так, например, для того чтобы сохранить цветное RGB-изображение размером 1000×1000 пикселов, потребуется около 3 Мбайт памяти.

1.3. Векторная модель представления двумерных изображений

В отличие от растровых изображений, основой которых являются пикселы, основой векторных изображений являются контуры, представляемые кривыми, называемыми *векторами* [82]. Каждый контур векторного изображения представляет собой отдельный объект, который можно независимо от других редактировать.

В соответствии с этим векторную графику также иногда называют объектноориентированной графикой. Файлы векторных изображений содержат набор инструкций, их основа — формулы, по которым строятся контуры. Благодаря тому, что форма контуров задается этими инструкциями, точность воспроизведения векторного изображения, его четкость, определяется только разрешающей способностью воспроизводящего устройства (принтера, экрана дисплея и т. д.). По этой причине векторное изображение может быть неограниченно увеличено без потери качества, в то время как в случае увеличения растровых изображений мы встречаемся с серьезными проблемами. Другим достоинством векторных изображений является компактность их хранения. Действительно, для того чтобы сохранить изображение отрезка прямой линии в векторном формате, необходимо сохранить координаты, определяющие его начало и конец, а также данные о цвете и толщине. Для этого потребуется всего несколько байтов. В то же время для хранения изображения того же отрезка в растровом режиме потребуется значительно больший объем информации, поскольку при этом необходимо хранить цвет каждого из множества пикселов, составляющих отрезок. Отсюда следует, что векторный формат, как правило, более компактен, чем растровый, хотя байтовый размер векторного изображения в случае сложных рисунков может превышать байтовый размер этих же рисунков в растровом формате. Векторное изображение легко экспортируется в растровый формат, но не наоборот. Вместе с тем векторная графика не пригодна для хранения полутоновых изображений, например, фотографий.

Векторное изображение состоит из объектов. Каждый объект состоит из одного или нескольких контуров. Контуры могут быть замкнутыми или открытыми, т. е. незамкнутыми. Примером замкнутого контура является окружность, примером открытого — отрезок линии. Закрытые контуры могут быть заполненными заливкой. В свою очередь, каждый контур, в зависимости от сложности, состоит из одного или нескольких сегментов, представляющих собой отрезки прямых или кривых линий, заканчивающихся узлами (опорными точками). Узлы определяют положение сегмента в контуре. На рис. 1.3 показаны элементы векторного изображения.



Рис. 1.3. Векторная модель представления двумерных изображений

Рассмотрим кратко математические основы векторной графики. В векторной графике точке соответствует узел.

Для описания отрезка прямой линии в векторной графике используют уравнение

$$y = ax + b,$$

где x и y — декартовы координаты, а a и b — постоянные коэффициенты. Кроме того, начало и конец отрезка задаются координатами x_1 и x_2 .

Для описания кривых второго порядка — окружностей, эллипсов, парабол и гипербол — используют кривые второго порядка

$$x^{2} + a_{1}y^{2} + a_{2}xy + a_{3}x + a_{4}y + a_{5} = 0$$

где a_1 , a_2 , a_3 , a_4 и a_5 — постоянные коэффициенты. Поскольку сегмент представляет собой отрезок линии, в данном случае кривой, то необходимо задать координаты, определяющие его начало и конец, например, x_1 и x_2 . Эти кривые в векторной графике используются для построения базовых примитивов, в частности эллипсов и окружностей. Особенностью этих кривых является то, что они не имеют точек перегиба.

В отличие от кривых второго порядка, кривые третьего порядка могут иметь точки перегиба, что дает возможность их использовать для представления в векторной форме различных природных объектов. Уравнение, применяемое для описания кривых третьего порядка, имеет вид

$$x^{3} + a_{1}y^{3} + a_{2}x^{2}y + a_{3}xy^{2} + a_{4}x^{2} + a_{5}y^{2} + a_{6}xy + a_{7}x + a_{8}y + a_{9} = 0$$

где a_1 , a_2 , a_3 , a_4 , a_5 , a_6 , a_7 , a_8 и a_9 — постоянные коэффициенты. Так же, как и ранее, поскольку сегмент представляет собой отрезок линии, то необходимо задать координаты, определяющие его начало и конец — x_1 и x_2 .

Широкое применение в векторной графике получили *кривые Безье третьего порядка*. Особенностью этих кривых является то, что они позволяют удобно регулировать не только положение узлов на плоскости изображения, но также величины первой производной линии (угла наклона) и ее второй производной (кривизну) в этих точках. Это, в частности, обеспечивает возможность соединять отдельные сегменты без изломов в точках соединения и тем самым аппроксимировать отрезками кривых Безье контуры любой сложности.

1.4. Виды изображений

Неподвижное ахроматическое изображение представляет собой функцию, описывающую распределение яркости L_c на плоскости, т. е. $L_c(x, y)$, где x и y — декартовы координаты. Для представления ахроматического движущегося изображения в написанное выражение добавляется еще одна независимая переменная — время t, а запись принимает вид $L_c(x, y, t)$. Цветные изображения натурных сцен,

получаемые в результате их оптической проекции на какую-либо поверхность, например на светочувствительную поверхность датчика телевизионного сигнала, будут в числе независимых переменных содержать еще и длину волны светового излучения λ . В этом случае для цветного неподвижного изображения будем иметь $L_c(x, y, \lambda)$, а для движущегося — $L_c(x, y, t, \lambda)$, где L_c следует рассматривать как интенсивность излучения на длине волны λ , в точке с координатами x и y, в момент времени t. Аналогичным образом можно перейти к описанию "объемных" изображений, добавив еще одну пространственную координату z.

Однако современная техника передачи, консервации и демонстрации изображений основана на их представлении в виде ряда компонентов. Так, например, цветное изображение, предназначенное для вывода на экран телевизора или на экран монитора компьютера, представляется в виде 3-х цветоделенных изображений: красного, зеленого и синего. В полиграфии по причинам, которые будут рассмотрены в дальнейших разделах книги, число цветоделенных изображений может быть большим. Аналогичным образом обстоит дело с движущимися изображениями, которые, как известно, представляются последовательностью неподвижных, быстро сменяющих друг друга изображений, на каждом из которых зафиксирована соответствующая фаза движения. Быстрая смена этих изображений создает иллюзию движения.

В общем случае, последовательность неподвижных цветоделенных изображений, которой представляются реальные цветные движущиеся изображения, обладает одной замечательной особенностью — все они описываются очень похожими характеристиками. Эта особенность позволяет в дальнейшем сосредоточиться на рассмотрении свойств и методов обработки неподвижных ахроматических изображений, распространяя полученные результаты на цветные изображения, и лишь в необходимых случаях выходить за эти рамки.

Продолжая изложение, отметим, что изображения принято разделять на два класса: *семантические*, т. е. смысловые, и *текстурные*. Примеры этих изображений приведены на рис. 1.4 и 1.5 соответственно.



Рис 1.4. Пример семантического изображения





Рис. 1.5. Пример текстурного изображения

В процессе длительной эволюции зрительная система человека приспособилась обнаруживать, опознавать и классифицировать не любые произвольные распределения яркости, спроецированные зрачком на сетчатку глаза, а лишь те, которые создаются объектами внешнего мира. В этом легко убедиться, если попытаться обнаружить шумовой узор ("шумовой объект"), показанный на рис. 1.5, δ на фоне шумового поля, которое приведено на рис. 1.5, *a*. Эта задача решается с трудом путем поэлементного сравнения обоих изображений, тогда как на рис. 1.4 любой объект находится легко и быстро.

Отмеченная особенность зрения широко используется в природе для целей камуфляжа. Так, например, неправильной формы полосы на шкуре тигра делают его плохо различимым в зарослях.

Характерной особенностью изображений реальных объектов является то, что они состоят из областей, разделенных более или менее резкими световыми границами, внутри которых яркость и цвет изменяются сравнительно медленно. Эти световые границы (контуры) передают форму объекта и являются основой для его опознавания. Из опыта известно, что информации, содержащейся в контурах, как правило, вполне достаточно для безошибочного узнавания объекта. Так, например, мы легко узнаем лицо знакомого человека по контурному рисунку.

Найдем связь между структурой изображений реальных объектов и их пространственными спектрами, полученными в результате интегрального преобразования Фурье. С этой целью рассмотрим спектры трех различных по резкости световых границ, ориентированных перпендикулярно к оси x. Поскольку в данном случае яркость изображений не зависит от координаты y, задачу можно существенно упростить, сведя ее к одномерной. На рис. 1.6 приведены три различные зависимости изменения яркости на границе от координаты x, которые описываются выражениями

$$L_{1}(x) = \begin{cases} 0 & \text{при } x \le x_{0}, \\ L & \text{при } x > x_{0}, \end{cases}$$

$$L_{2}(x) = \begin{cases} \frac{L}{2} - \frac{L}{2} \exp[\alpha(x - x_{0})] & \text{при } x \le x_{0}, \\ \frac{L}{2} + \frac{L}{2} \exp[-\alpha(x - x_{0})] & \text{при } x > x_{0}, \end{cases}$$
$$L_{3}(x) = \begin{cases} \frac{L}{2} \exp[\alpha(x - x_{0})] & \text{при } x \le x_{0}, \\ L - \frac{L}{2} \exp[-\alpha(x - x_{0})] & \text{при } x > x_{0}. \end{cases}$$

Определяя спектры функций L(x, y), в результате преобразований получим

$$M_{1}(\omega_{x}) = (L/\omega_{x}) \exp\left[-\mathbf{i}(\pi/2 + \omega_{x}x_{0})\right],$$

$$M_{2}(\omega_{x}) = L\pi\delta(\omega_{x}) + \left[(L\omega_{x})/(\alpha^{2} + \omega_{x}^{2})\right] \exp\left[-\mathbf{i}(\pi/2 + \omega_{x}x_{0})\right],$$

$$M_{3}(\omega_{x}) = \left\{(L\alpha^{2})/\left[\omega_{x}(\alpha^{2} + \omega_{x}^{2})\right]\right\} \exp\left[-\mathbf{i}(\pi/2 + \omega_{x}x_{0})\right],$$

где ω_x — круговая пространственная частота, $\mathbf{i} = \sqrt{-1}$. (Подробно о спектрах изображений см. в разд. 1.6.)



Рис. 1.6. Примеры распределения яркости на световых границах

Первое, что обращает на себя внимание, — идентичность фазовых спектров для всех трех границ. Нетрудно видеть, что информация о наличии и положении световой границы заключена в фазовом спектре. Амплитудный спектр не содержит информации о положении границы, однако в нем заключена информация о резкости изображения [49]. Из опыта работы с изображениями известно, что изображение может быть подвергнуто значительным линейным и нелинейным искажениям, но если при этом искажения фазового спектра будут невелики, так что они не вызовут исчезновения существующих или появления новых световых границ, изображение будет оставаться легко узнаваемым. К таким искажениям относятся интегрирование, которое приводит к потере четкости изображений, дифференцирование, приводящее к подчеркиванию границ на изображении, поэлементное преобразование вида u = f(v) (где f(v) — монотонная функция), приводящее к изменению контраста, и ряд других. Если же в результате преобразования изображения существенно искажается его фазовый спектр, то может иметь место потеря узнаваемости изображаемого объекта.

Наглядный пример тому — потеря различения изображения при воздействии на него флуктуационным шумом, имеющим большую дисперсию. В результате такого воздействия плотность вероятности распределения фазы частотных составляющих зашумленной реализации приближается к равномерной на интервале 0...2π, что влечет за собой полное размытие световых границ на изображении.











Рис. 1.7. Исходные (*a*) и (*б*) и синтезированные (*в*) и (*г*) изображения. Изображения заимствованы из [79]

Наиболее убедительным доказательством приведенных ранее положений служит эксперимент с восстановлением изображений по "перепутанным" амплитудным и фазовым спектрам [79]. Эксперимент заключался в том, что для изображений, показанных на рис. 1.7, *а* и *б*, находились амплитудные и фазовые спектры, по которым составлялись новые спектры следующим образом:

$$M_{\rm B}(\omega_x,\omega_y) = |M_{\rm G}(\omega_x,\omega_y)| \exp\left[-i\varphi_{\rm a}(\omega_x,\omega_y)\right],$$
$$M_{\rm F}(\omega_x,\omega_y) = |M_{\rm a}(\omega_x,\omega_y)| \exp\left[-i\varphi_{\rm G}(\omega_x,\omega_y)\right],$$

после чего по спектрам $M_{\rm B}(\omega_x, \omega_y)$ и $M_{\rm r}(\omega_x, \omega_y)$ синтезировались изображения, показанные на рис. 1.7, *в* и *г*. Из рисунка видно, что замена ("перепутывание") амплитудных спектров привела лишь к некоторому зашумлению изображений без потери их различимости.

На основании изложенного можно сделать вывод, что *при передаче и консервации* изображений особое внимание следует уделять точности передачи фазового спектра. В телевидении к этому выводу пришли давно, чисто опытным путем, заметив, что амплитудно-частотные и амплитудные (нелинейные) искажения менее заметны на изображении, чем фазо-частотные.

При решении ряда задач прикладного характера приходится иметь дело с различением текстурных полей на изображении. В последнее время много внимания уделяется анализу изображений, полученных при дистанционном зондировании Земли, где вопрос различения текстур занимает ведущее место. В результате экспериментальных исследований было найдено, что человек способен различать текстурные поля, если они различаются между собой одномерными плотностями вероятностей распределения яркости в них или если при одинаковых одномерных плотностях распределения вероятностей имеется различие в функциях автокорреляции. Если же текстурные поля различаются только плотностями распределения вероятностей претьего или более высокого порядка, то они визуально не различимы. Это положение носит название гипотезы Юлеша. Впоследствии рядом исследователей были найдены примеры текстур, противоречащих этой гипотезе, хотя, как отмечают сами исследователи, зрительно эти текстуры трудно различимы [83].

1.5. Модели изображений

В теории цифровой обработки изображений в зависимости от решаемой задачи используют различные модели изображений. Под *моделью изображения* понимается комплекс характеристик, описывающих распределение яркости на плоскости, которыми аппроксимируется рассматриваемый класс изображений. Модель должна удовлетворять противоречивым требованиям достаточной близости к реальным изображениям и простоты теоретического анализа. При выборе модели существенное значение имеет также степень общности результатов, которые могут быть получены при ее использовании. В зависимости от того, какое из требований приобретает большее значение, используют ту или иную модель изображения различной степени сложности. В качестве примера приведем одну из моделей, применяемую при синтезе алгоритмов нелинейной обработки изображений:

$$L_{c}(x, y) = kE(x, y)r_{c}(x, y),$$

где $r_{\rm c}(x, y)$ — коэффициент отражения различных участков сцены, соответствующих ее проекции на светочувствительную поверхность датчика сигнала изображения с координатами (x, y); E(x, y) — освещенность различных участков сцены, соответствующих ее проекции на светочувствительную поверхность датчика сигнала изображения с координатами (x, y); k — коэффициент, согласующий размерности. Важными свойствами функции $r_{c}(x, y)$ и E(x, y) являются следующие. Коэффициент отражения $r_{c}(x, y)$ — это функция, характеризующаяся наличием резких скачков, возникающих на контурах, в то время как E(x, y) представляет собой в основном очень медленно изменяющуюся функцию координат (x, y). Благодаря этому спектр $r_{c}(x, y)$ является широкополосным, а спектр функции E(x, y)узкополосным, что используется, например, в целях "улучшения" качества изображения при его гомоморфной фильтрации. Другим важным свойством функций $r_{\rm c}(x, y)$ и E(x, y) является их положительность, что обеспечивает положительность $L_{c}(x, y)$. Свойство положительности $L_{c}(x, y)$ накладывает сильные ограничения на выбор возможных алгоритмов обработки, поскольку результат обработки — обработанное изображение — также должен быть положительной функцией, т. к. отрицательные значения яркости физически не реализуемы. В дальнейшем мы познакомимся и с другими моделями изображений.

Хорошая модель изображения является надежной основой для синтеза эффективных алгоритмов обработки изображений, и наоборот, неудачные модели часто оказывались причиной неудач при разработке таких алгоритмов.

1.6. Пространственные спектры изображений

При анализе линейных искажений изображений, а также при решении задач, связанных с обнаружением и опознаванием изображений объектов на репродукции, чрезвычайно полезно использовать понятие спектров изображений и их попарных разностей при "наилучшем" совмещении. Под *"наилучшим" совмещением изображений* понимается такое совмещение, при котором средний квадрат их поточечной

разности $\left[L_{c}(x, y, i) - L_{c}(x, y, j)\right]^{2}$ достигает минимума, где $L_{c}(x, y, i)$ и $L_{c}(x, y, j)$ — распределения яркости в *i*-ом и *j*-ом изображениях [49].

Спектр *j-го изображения* $M_{c}(\omega_{x}, \omega_{y}, j)$ по определению представляет собой комплексную функцию, связанную с распределением яркости на изображении $L_{c}(x, y, j)$ парой преобразований Фурье:

$$M_{c}(\omega_{x}, \omega_{y}, j) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L_{c}(x, y, j) \exp\left[-\mathbf{i}\left(\omega_{x}x + \omega_{y}y\right)\right] dxdy, \qquad (1.2)$$

$$L_{\rm c}(x, y, j) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{\rm c}(\omega_x, \omega_y, j) \exp\left[i(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y, \qquad (1.3)$$

где $\mathbf{i} = \sqrt{-1}$, ω_x и ω_y — круговые пространственные частоты спектра в направлении осей *x* и *y*.

Аналогично определим спектр разности двух изображений (j-го и i-го) при их "наилучшем" совмещении:

$$M_{\Delta}(\omega_{x},\omega_{y},j,i) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[L_{c}(x,y,j) - L_{c}(x,y,i) \right] \exp\left[-\mathbf{i}\left(\omega_{x}x + \omega_{y}y\right)\right] dxdy, \quad (1.4)$$
$$L_{c}(x,y,j) - L_{c}(x,y,i) =$$
$$= \frac{1}{4\pi^{2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{\Delta}(\omega_{x},\omega_{y},j,i) \exp\left[\mathbf{i}\left(\omega_{x}x + \omega_{y}y\right)\right] d\omega_{x}d\omega_{y}. \quad (1.5)$$

Спектры разности двух изображений при их "наилучшем" совмещении используются при решении задач, связанных с опознаванием (различением) изображений объектов при наличии шума на изображениях. Определенные таким образом спектры содержат полную информацию как об амплитудах, так и о фазах частотных составляющих.

Распределение яркости на репродукции $L_{c\Omega}(x, y, j)$, воспроизводимой линейной системой, вносящей искажения, может быть найдено, исходя из известного распределения яркости в исходном изображении $L_c(x, y, j)$ посредством интеграла свертки (интеграла Дюамеля):

$$L_{c\Omega}(x, y, j) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L_{c}(\xi, \eta, j) h(x - \xi, y - \eta) d\xi d\eta, \qquad (1.6)$$

где ξ и η — переменные интегрирования; h(x, y) — импульсная характеристика системы, которая с точностью до постоянного множителя совпадает с распределением яркости на репродукции при передаче линейной системой изображения точки. Импульсная характеристика h(x, y), или, что то же самое, функция рассеяния точки, в терминологии оптических систем (в дальнейшем мы будем использовать и этот термин) полностью характеризует искажения, вносимые линейной системой. При этом спектры исходного и воспроизводимого линейной системой изображений связаны между собой соотношением

$$M_{c\Omega}(\omega_x, \omega_y, j) = M_c(\omega_x, \omega_y, j) K(\omega_x, \omega_y), \qquad (1.7)$$

где $M_{c\Omega}(\omega_x, \omega_y, j)$ — спектр репродукции $L_{c\Omega}(x, y, j)$; $K(\omega_x, \omega_y)$ — частотнопередаточная функция рассматриваемой линейной системы, связанная с импульсной характеристикой парой преобразований Фурье:

$$K(\omega_x, \omega_y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y) \exp\left[-\mathbf{i}\left(\omega_x x + \omega_y y\right)\right] dxdy, \qquad (1.8)$$

$$h(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(\omega_x, \omega_y) \exp\left[i(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y.$$
(1.9)

Аналогично находится спектр разности изображений, воспроизводимых линейной системой при их "наилучшем" совмещении:

$$M_{\Delta\Omega}\left(\omega_x, \omega_y, j, i\right) = M_{\Delta}\left(\omega_x, \omega_y, j, i\right) K\left(\omega_x, \omega_y\right).$$
(1.10)

1.7. Спектральные интенсивности изображений

При решении некоторых задач информация о фазовых соотношениях частотных составляющих оказывается излишней, и тогда используют понятия *спектральной интенсивности изображений*. В отличие от спектра спектральная интенсивность содержит информацию только о распределении энергии (или величины, пропорциональной ей) по пространственным частотам.

Определим спектральную интенсивность *j*-го изображения:

$$S_{c}\left(\omega_{x},\omega_{y},j\right) = \frac{1}{S_{H3}} \left| M_{c}\left(\omega_{x},\omega_{y},j\right) \right|^{2}, \qquad (1.11)$$

 $\underbrace{ \begin{array}{c} & & \\ &$

Рис. 1.8. К определению площади изображения

где

$$s_{\mu_3} = x_0 y_0, \tag{1.12}$$

площадь прямоугольника, в который вписано изображение (рис. 1.8).

Из соображений удобства будем считать, что для всех изображений, составляющих данный алфавит, *s*_{из} одинаково и что за пределами прямоугольника яркость равна нулю.

Аналогично определим спектральную интенсивность разности двух изображений (*j*-го и *i*-го) при их наилучшем совмещении:

$$S_{\Delta}\left(\omega_{x},\omega_{y},j,i\right) = \frac{1}{s_{_{H3}}} \left| M_{\Delta}\left(\omega_{x},\omega_{y},j,i\right) \right|^{2}.$$
(1.13)

Спектральные интенсивности изображений и их разностей позволяют вычислить соответствующие им значения энергий:

$$E_{\rm c}(j) = \frac{s_{\rm H3}}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S_{\rm c}(\omega_x, \omega_y, j) d\omega_x d\omega_y, \qquad (1.14)$$

$$E_{\Delta}(j,i) = \frac{s_{\text{H3}}}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S_{\Delta}(\omega_x, \omega_y, j, i) d\omega_x d\omega_y . \qquad (1.15)$$

Чем больше значения $E_{c}(j)$ и $E_{\Delta}(j,i)$, тем выше вероятности правильного обнаружения и опознавания изображений объектов в присутствии шума.

Экспериментальные исследования показывают, что основная доля энергии изображений сосредоточена в области низких пространственных частот, при этом примерно половина ее приходится на постоянную составляющую. Последнее утверждение, в частности, вытекает из плотности вероятности распределения яркости в изображении [49]. Действительно, полная энергия изображения

$$E_{\rm c} = s_{\rm H3} \overline{L_{\rm c}^2} = \frac{s_{\rm H3}}{\overline{L_{\rm c}}} \int_0^\infty L_{\rm c}^2 \exp\left(-\frac{L_{\rm c}}{\overline{L_{\rm c}}}\right) dL_{\rm c} = 2s_{\rm H3} \left(\overline{L_{\rm c}}\right)^2, \qquad (1.16)$$

в то время как энергия, приходящаяся на постоянную составляющую, $E_{\rm c=} = s_{\rm H3} \left(\overline{L_{\rm c}}\right)^2$.

Спектральная интенсивность разности изображений при их "наилучшем" совмещении также имеет максимум в области низких пространственных частот, но в отличие от спектральной интенсивности изображения с ростом частоты она убывает медленнее. Кроме того, в среднем вся энергия разности изображений приходится на переменные составляющие спектра. *Несмотря на то, что энергия, приходящаяся на область высоких пространственных частот в изображении, невелика, эти составляющие спектра играют большую роль, обеспечивая четкость изображения.*



Рис. 1.9. Зависимости $S_{c}(\omega_{x}, 0)/S_{c}(0, 14, 0)$ и $S_{\Delta}(\omega_{x}, 0)/S_{\Delta}(0, 14, 0)$ от $|\omega_{x}|$

В качестве примера на рис. 1.9 приведены отмеченные квадратиками экспериментальные точки, представляющие усредненные по алфавиту зависимости нормированных значений спектральных интенсивностей изображений и их попарных разностей при "наилучшем" совмещении от $|\omega_x|$, полученные для изображений теста, представляющего собой набор из 7 двухградационных стилизованных изображений самолетов.

Экспериментальные исследования показывают также, что спектральные интенсивности изображений анизотропны [98]. Это находит свое выражение в более медленном их убывании вдоль осей ω_x , ω_y , чем в направлениях, расположенных к этим осям под углами 45°.

Анизотропия спектральных интенсивностей реальных изображений обусловлена их структурой, которая, в конечном счете, определяется структурой самой Вселенной,

например, формой земной поверхности, направлением сил тяготения и т. п. Вследствие указанных причин в "естественных" изображениях вертикальные и горизонтальные контуры встречаются чаще, чем контуры, ориентированные в других направлениях.

Спектральные интенсивности, полученные экспериментальным путем, хорошо аппроксимируются выражениями:

$$S_{c}\left(\omega_{x},\omega_{y}\right) = \frac{2a_{s}^{2}\overline{E_{c}}}{\left(a_{s}^{2}+\omega_{x}^{2}\right)\left(a_{s}^{2}+\omega_{y}^{2}\right)s_{\mu_{3}}} + \frac{2\pi^{2}}{s_{\mu_{3}}}\overline{E_{c}}\delta\left(\omega_{x}\right)\delta\left(\omega_{y}\right), \qquad (1.17)$$

$$S_{\Delta}\left(\omega_{x},\omega_{y}\right) = \frac{4a_{s\Delta}^{2}\overline{E_{\Delta}}}{\left(a_{s\Delta}^{2}+\omega_{x}^{2}\right)\left(a_{s\Delta}^{2}+\omega_{y}^{2}\right)s_{_{\rm H3}}},\qquad(1.18)$$

где $\delta(.)$ — дельта-функция, удовлетворяющая условиям

$$\int_{0}^{\infty} \delta(\omega_{x}) d\omega_{x} = 1, \ \delta(\omega_{x}) = \begin{cases} \infty & \text{при } \omega_{x} = 0, \\ 0 & \text{при } \omega_{x} \neq 0, \end{cases}$$

 a_s и $a_{s\Delta}$ — параметры, определяемые характером и размерами изображений, составляющих алфавит.

Поскольку выражения (1.17) и (1.18) описывают усредненные по алфавиту спектральные интенсивности, в них устранены символы j и i, которыми в (1.11), (1.13)—(1.15) обозначалось соответствие $S_{\rm c}(\omega_x, \omega_y, j)$ и $S_{\Delta}(\omega_x, \omega_y, j, i)$ определенным изображениям алфавита.

На рис. 1.9 приведены аппроксимирующие кривые, рассчитанные по формулам (1.17) и (1.18). Сопоставление результатов расчета и эксперимента указывает на их хорошее совпадение.

Аппроксимирующие формулы (1.17) и (1.18) также хорошо отражают анизотропию спектральных интенсивностей реальных изображений.

Рисунок 1.10, на котором приведено семейство линий постоянной спектральной интенсивности (изолиний) $S_{\rm c}(\omega_x, \omega_y) = {\rm const}$, иллюстрирует сказанное.

Параметры изображений a_s и $a_{s\Delta}$, как будет показано в *разд. 1.8*, представляют собой величины, обратные интервалам автокорреляции, поэтому при изменении площади изображений $a_s^2 s_{\mu 3} = \text{const.}$

В табл. 1.1 в качестве примера приведены усредненные по алфавиту изображений значения $a_s^2 s_{_{\rm H3}}$ и $a_{\Delta}^2 s_{_{\rm H3}}$, полученные для 3-х алфавитов изображений экспериментальным путем.



Рис. 1.10. Семейство линий постоянной спектральной интенсивности S_c (...)

Тип изображения	$a_s^2 s_{\mu_3}$	$a_{\Delta}^2 s_{{}_{\mathrm{H}3}}$
Тест 1	17,0	42,0
Тест 2	14,4	20,4
Тест 3	2,5	14,4

Таблица 1.1. Значения $a_s^2 s_{_{\rm H3}}$ и $a_{\Lambda}^2 s_{_{\rm H3}}$ для 3-х алфавитов изображений

Первый алфавит (тест 1) включал в себя 20 полутоновых изображений портретов, второй (тест 2) — 20 полутоновых изображений пейзажей и третий (тест 3) — 7 двухградационных стилизованных изображений самолетов.

1.8. Функции автоковариации изображений

Для описания статистических связей между элементами изображений часто используют *функцию автоковариации*, которую для *j*-го изображения определяют следующим образом:

$$R_{\rm c}(\xi,\eta,j) = \frac{1}{s_{_{\rm H3}}} \int_{0}^{x_0} \int_{0}^{y_0} L_{\rm c}(x,y,j) L_{\rm c}(x+\xi,y+\eta,j) dxdy, \qquad (1.19)$$

где ξ и η — смещения реализации *j*-го изображения $L_c(x + \xi, y + \eta, j)$ относительно реализации этого же изображения $L_c(x, y, j)$ по осям *x* и *y* соответственно.

Распространение пределов интегрирования до бесконечности не изменит значения интеграла, поскольку за пределами прямоугольника с координатами (0, 0), $(x_0, 0)$, $(0, y_0)$, (x_0, y_0) , в которых заключено изображение, яркость принята равной нулю *(см. разд. 1.7).*

Приведенное определение функции автоковариации будет тем ближе к строгому определению, используемому в математической статистике, чем меньше *площадь* автокорреляции изображения $\xi_{\kappa}\eta_{\kappa}$, которую мы определим соотношением

$$\xi_{\kappa}\eta_{\kappa} = \frac{1}{4\left\{R_{c}(0,0,j) - \left[\overline{L_{c}(j)}\right]^{2}\right\}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left\{R_{c}(\xi,\eta,j) - \left[\overline{L_{c}(j)}\right]^{2}\right\} d\xi d\eta,$$

по сравнению с s_{из} и чем однороднее алфавит изображений. Аналогично определяется функция автоковариации разности j-го и i-го изображений при их наилучием совмещении

$$R_{\Delta}(\xi,\eta,j,i) = \frac{1}{s_{_{H3}}} \int_{0}^{x_{_{0}}} \int_{0}^{y_{_{0}}} \left[L_{c}(x,y,j) - L_{c}(x,y,i) \right] \times \left[L_{c}(x+\xi,y+\eta,j) - L_{c}(x+\xi,y+\eta,i) \right] dxdy.$$
(1.20)

Отметим ряд важных свойств функции автоковариации.

Функция автоковариации обладает свойством четности

$$R_{c}(\xi,\eta,j) = R_{c}(-\xi,\eta,j) = R_{c}(\xi,-\eta,j) = R_{c}(-\xi,-\eta,j).$$

Φункция автоковариации принимает максимальное значение при ξ=0, η=0, причем это значение согласно формуле (1.19) равно

$$R_{\rm c}(0,0,j) = E_{\rm c}/s_{_{\rm H3}} . \tag{1.21}$$

C ростом ξ и η функция автоковариации убывает вследствие ослабления статистических связей с увеличением расстояния между элементами в изображении. При этом в пределе для неограниченно больших изображений R_c(ξ, η, j)

стремится к
$$\left[\overline{L_{c}(\xi,\eta,j)}\right]^{2}$$
, а $R_{\Delta}(\xi,\eta,j,i)$ — к нулю.

Функция автоковариации изображения R_c (ξ, η, j) связана с его спектральной интенсивностью парой преобразований Фурье:

$$R_{\rm c}(\xi,\eta,j) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S_{\rm c}(\omega_x,\omega_y,j) \exp\left[i(\omega_x\xi+\omega_y\eta)\right] d\omega_x d\omega_y, \qquad (1.22)$$

$$S_{c}(\omega_{x},\omega_{y},j) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} R_{c}(\xi,\eta,j) \exp\left[-i(\omega_{x}\xi + \omega_{y}\eta)\right] d\xi d\eta. \qquad (1.23)$$

Аналогичная связь имеет место между $R_{\Delta}(\xi,\eta,j,i)$ и $S_{\Delta}(\omega_x,\omega_y,j,i)$.

Часто в рассмотрение вводят коэффициент автокорреляции изображения

$$\rho_{c}(\xi,\eta,j) = \frac{R_{c}(\xi,\eta,j) - \left[\overline{L_{c}(x,y,j)}\right]^{2}}{R_{c}(0,0,j) - \left[\overline{L_{c}(x,y,j)}\right]^{2}}.$$

Одним из первых исследователей, получивших экспериментальным путем коэффициент автокорреляции изображений, был Кречмер [139]. При проведении экспериментов Кречмер использовал прибор, в котором два диапозитива одного и того же изображения, наложенные друг на друга со сдвигом ξ и η , просвечивались параллельным пучком световых лучей. Суммарный световой поток, прошедший через диапозитивы и пропорциональный $R_c(\xi, \eta, j)$, регистрировался фотоэлектронным умножителем. Перемещая один диапозитив относительно другого и измеряя ток фотоэлектронного фотоумножителя, он получил зависимости коэффициентов автокорреляции от смещений ξ и η для ряда изображений.

На рис. 1.11 сплошными линиями показаны зависимости коэффициентов автокорреляции от смещения в направлении оси y, представленного числом растровых элементов (строк), найденные экспериментальным путем для двух изображений. Из рисунка видно, что в первом случае, когда изображение содержит небольшое число мелких деталей и, следовательно, статистические связи между элементами изображения более сильны, коэффициент автокорреляции убывает медленнее, чем во втором случае. Аналогичный вид имеют кривые, представляющие зависимость $\rho_c(\xi, \eta)$ от смещения ξ в направлении оси x.

Введем в рассмотрение усредненные по алфавиту изображений функции автоковариации $R_{\rm c}(\xi,\eta)$ и $R_{\Delta}(\xi,\eta)$, которые связаны с соответствующими спектральными интенсивностями $S_{\rm c}(\omega_x,\omega_y)$ и $S_{\Delta}(\omega_x,\omega_y)$ через преобразования Фурье.

.

Используя аппроксимирующие формулы (1.17) и (1.18) для $S_{\rm c}(\omega_x, \omega_y)$ и $S_{\Delta}(\omega_x, \omega_y)$, получаем

$$R_{\rm c}\left(\xi,\eta\right) = \frac{\overline{E_{\rm c}}}{2s_{\rm H3}} \left\{1 + \exp\left[-a_s\left(\left|\xi\right| + \left|\eta\right|\right)\right]\right\},\tag{1.24}$$

$$R_{\Delta}(\xi,\eta) = \frac{\overline{E_{\Delta}}}{s_{_{\rm H3}}} \left\{ \exp\left[-a_{s\Delta}\left(\left|\xi\right| + \left|\eta\right|\right)\right] \right\}.$$
(1.25)



Рис. 1.11. Зависимость ρ_c от смещения вдоль оси у, выраженного
 в числе растровых элементов η/Δ_д; 1 — изображение лица крупным планом;
 2 — изображение общего плана (трибуны стадиона, заполненные зрителями);
 Δ_д — линейный размер растрового элемента

При этом усредненный по алфавиту коэффициент автокорреляции изображений

$$\rho_{\rm c}(\boldsymbol{\xi},\boldsymbol{\eta}) = \exp\left[-a_s\left(|\boldsymbol{\xi}| + |\boldsymbol{\eta}|\right)\right]. \tag{1.26}$$

Параметр a_s в формуле (1.26) представляет собой величину, обратную интервалу автокорреляции ξ_{κ} (η_{κ}), который по определению равен

$$\xi_{\kappa}\Big|_{\eta=0} = \frac{1}{2} \int_{-\infty}^{\infty} \rho_{c}(\xi,0) d\xi.$$

При этом произведения $a_s^2 s_{\mu_3}$ и $a_{s\Delta}^2 s_{\mu_3}$ могут рассматриваться как отношения площади изображения к введенным выше площадям автокорреляции изображения ξ_{κ} η_{κ}
и разности изображений при их наилучшем совмещении. На рис. 1.11 штриховыми линиями представлены зависимости ρ_c от η/Δ_{π} , рассчитанные по формуле (1.26).

Примечание

То обстоятельство, что коэффициенты автокорреляции изображений достаточно точно аппроксимируются экспоненциальными функциями, позволяет в качестве стохастической модели изображений использовать случайный двумерный марковский процесс первого порядка.

Нетрудно видеть, что функции автоковариации видеосигналов также описываются экспоненциальными функциями. Более того, как показали экспериментальные исследования, экспоненциальными являются и функции взаимной ковариации видеосигналов от двух различных изображений.

1.9. Плотность вероятности распределения яркости в изображениях

Изображение "ахроматической сцены" $L_{c}(x, y)$, представляющее собой распределение яркости на плоскости, можно рассматривать как результат перемножения коэффициента отражения различных ее участков на их освещенность.

При этом часто бывает удобно рассматривать распределение яркости в изображении $L_{c}(x, y)$ как реализацию двумерного марковского процесса первого порядка, где яркость L_{c} является случайной функцией декартовых координат (x, y). В этом случае распределение яркости по уровням в изображении может быть описано многомерной плотностью вероятности.

В настоящее время накоплен большой экспериментальный материал по статистике изображений, из которого, в частности, следует, что распределение яркости по уровням в изображении существенно не равномерно. Наиболее вероятны малые уровни яркости в изображении и менее вероятны большие. Однако благодаря тому, что характеристика восприятия яркости человеком имеет вид насыщающейся зависимости, плотность вероятности визуально воспринимаемой яркости оказывается близкой к равномерной.

Хорошей аппроксимирующей функцией двумерной плотности вероятности распределения яркости в изображении является приведенная ниже функция

$$W(L_{c}, L_{c\xi\eta}) = \frac{1}{\left(\overline{L_{c}}\right)^{2} \left[1 - \rho_{c}\left(\xi, \eta\right)\right]} \times \left\{ \exp\left[-\frac{L_{c} + L_{c\xi\eta}}{\overline{L_{c}} \left[1 - \rho_{c}\left(\xi, \eta\right)\right]}\right] \right\} I_{0}\left\{\frac{2\sqrt{L_{c}L_{c\xi\eta}\rho_{c}\left(\xi, \eta\right)}}{\overline{L_{c}} \left[1 - \rho_{c}\left(\xi, \eta\right)\right]}\right\},$$
(1.27)

которая известна в литературе как плотность вероятности распределения квадрата огибающей стационарного нормального случайного процесса [49]. В выражении (1.27) использованы следующие обозначения: L_c и $L_{c\xi\eta}$ — величины яркости на изображении в двух точках, смещенных одна относительно другой по осям координат x, y соответственно на интервалы ξ и η ; $\overline{L_c}$ — средняя яркость изображения; $\rho_c(\xi,\eta)$ — коэффициент автокорреляции изображения (подробнее см. в разд. 1.8); $I_0\{$ } — функция Бесселя 1-го рода нулевого порядка от мнимого аргумента.

Выражение для одномерной плотности вероятности распределения яркости в изображении получается путем интегрирования $W(L_c, L_{c\xi\eta})$ по переменной $L_{c\xi\eta}$ в пределах от 0 до ∞ :

$$W(L_{\rm c}) = \int_{0}^{\infty} W(L_{\rm c}, L_{\rm c\xi\eta}) dL_{\rm c\xi\eta} = \frac{1}{L_{\rm c}} \exp\left(-\frac{L_{\rm c}}{L_{\rm c}}\right).$$
(1.28)

Это распределение хорошо согласуется с экспериментальными данными, которые впервые были получены нами путем фотометрирования натурных сцен и изображений, снятых на кинопленку [40].

При выполнении одного из таких экспериментов стандартная 35-миллиметровая кинопленка с линейной сенситометрической характеристикой, взятая из нескольких случайных цветных кинофильмов отечественного производства, была подвергнута фотометрированию. Луч микрофотометра имел квадратное сечение в плоскости исследуемого изображения со стороной квадрата 0,04 мм. В микрофотометре использовался селеновый фотоэлемент, кривая спектральной чувствительности которого близка к кривой видности (спектральной чувствительности глаза). Фотометрированию было подвергнуто 14 кинокадров в 120 точках каждого из них, расположенных на одинаковых расстояниях в шахматном порядке. В результате обработки фотометрических измерений и усреднения их по всем кадрам было получено усредненное распределение, которое приведено в табл. 1.2. Результаты эксперимента представлены также в виде гистограммы $p(L_c)$ на рис. 1.12 сплошной линией. Для сравнения на этом же рисунке штриховой линией нанесена гистограмма распределения яркости, рассчитанная с использованием аппроксимирующей формулы (1.28). Как следует из рисунка, рассчитанная гистограмма вполне удовлетворительно аппроксимирует гистограмму, полученную экспериментальным путем.

<i>L</i> _с , отн. ед.	010	1020	2030	3040	4050	5060
$p(L_{\rm c})$	0,569	0,138	0,072	0,075	0,057	0,025
<i>L</i> _с , отн. ед.	6070	7080	8090	90100	100110	110120
$p(L_{\rm c})$	0,023	0,010	0,010	0,012	0,007	0,000
L _c , отн. ед.	120130	_	_	_	_	—
$p(L_{\rm c})$	0,002		_	_		

Таблица 1.2. Усредненное распределение яркости в изображениях



Рис. 1.12. Усредненная гистограмма распределения яркости в изображении

В табл. 1.3 приведены результаты экспериментальных исследований гистограммы распределения яркости в изображении для семи типичных изображений из другой серии экспериментов, включавших в себя фотометрирование 68 случайных кинокадров, а на рис. 1.13 — одно из этих изображений [49]. Эта серия экспериментов отличалась от описанной выше тем, что фотометрировались изображения, снятые на черно-белой 35-миллиметровой кинопленке, *сенситометрическая характеристика* которой нелинейна и в первом приближении может быть описана степенной функцией с показателем степени, равным 1,3—1,7. Фотометрирование проводилось в 120 точках, расположенных на одинаковых расстояниях в шахматном порядке. Сечение луча света микрофотометра в плоскости изображения представляло собой квадрат со стороной менее 0,1 мм. Как видно из таблицы, результаты этой серии экспериментов имеют тот же характер, что и предыдущей.

L _с , отн. ед.	<i>р</i> (<i>L</i> _c) для изображений									
	1	2	3	4	5	6	7			
010	0,160	0,434	0,042	0, 292	0,366	0,266	0,240			
1020	0,140	0,192	0,058	0,092	0,125	0,092	0,150			
2030	0,110	0,133	0,092	0, 025	0,075	0,058	0,092			
3040	0,025	0,044	0,058	0,008	0,075	0,042	0,033			
4050	0,067	0,033	0,067	0,025	0,042	0,050	0,033			
5060	0,083	0,025	0,092	0,025	0,075	0,058	0,050			
6070	0,067	0,042	0,058	0,000	0,025	0,050	0,042			
7080	0,025	0,008	0,058	0,041	0,041	0,075	0,042			
8090	0,050	0,017	0,053	0,025	0,033	0,017	0,025			
90100	0,025	0,008	0,033	0,025	0,008	0,050	0,017			
100110	0,033	0,000	0,025	0,025	0,033	0,025	0,017			
110120	0,033	0,008	0,008	0,008	0,009	0,050	0,050			
120130	0,050	0,008	0,025	0,025	0,008	0,000	0,008			
130140	0,008	0,008	0,083	0,017	0,008	0,017	0,000			
140150	0,016	0,000	0,075	0,033	0,008	0,017	0,033			
150160	0,008	0,017	0,016	0,050	0,016	0,008	0,017			
160	0,100	0,023	0,150	0,284	0,062	0,135	0,151			

Таблица 1.3. Распределения яркости в 7 типичных изображениях



Рис. 1.13. Изображение с типичным распределением яркости

Характерной особенностью распределения яркости в изображении $p(L_c)$, как уже отмечалось, является наличие максимума в области малых значений яркости и спадание $p(L_c)$ с ее увеличением. При небольших значениях яркости L_c из формулы (1.28) путем разложения экспоненты в ряд и отбрасывания членов со степенями отношения $L_c/\overline{L_c}$ выше первой получаем

$$W(L_{\rm c}) = \frac{1}{\overline{L_{\rm c}}} \exp\left(-\frac{L_{\rm c}}{\overline{L_{\rm c}}}\right) \approx \frac{1}{\overline{L_{\rm c}}\left(1 + L_{\rm c}/\overline{L_{\rm c}}\right)}.$$

Заметим, что приближенное выражение для $W(L_c)$ по своей структуре совпадает с известным аппроксимирующим выражением [49]. Аналогичные результаты, но значительно позже, были получены и опубликованы Стокхемом [91]. Следует обратить внимание на то, что в старых публикациях можно встретить утверждение, что в типичных для телевизионного вещания изображениях наиболее сильно выражены серые тона. Существенно то, что в этих работах исследовалось распределение по уровням телевизионного сигнала в видеотракте, и этим все объясняется. Распределение такого сигнала по уровням значительно отличалось от распределения по уровням яркости в передаваемом изображении, поскольку преобразователи свет-сигнал во времена выполнения этих работ имели нелинейную световую характеристику с сильно выраженным насыщением в области светлых полутонов. С отмеченным обстоятельством автор встречался при проведении аналогичных экспериментов. Заметим, что у современных преобразователей "свет — сигнал" световая характеристика близка к линейной. Обратим также внимание на то, что в ряде случаев при редактировании сознательно, посредством графического редактора изменяют распределение яркости в изображении, выделяя те или иные области динамического диапазона в целях придания определенной выразительности репродукции. Изменяется также гистограмма распределения яркости в изображении при сжатии его динамического диапазона. Разумеется, во всех этих случаях распределение яркости будет сильно отличаться от распределения приведенного выше.

В заключение отметим, что при работе со специфическими изображениями можно встретиться и с другими типами распределений. Так, например, для изображений облачного покрова характерно распределение с двумя максимумами, а для двух-градационных изображений (чертежи, печатный текст) — дельтаобразное.

1.10. Коэффициент автокорреляции оболочек трехмерных объектов

До сих пор мы рассматривали статистические характеристики двумерных изображений, т. е. проекций реальных трехмерных объектов на поверхность, как правило, на плоскость. Разработка новых методов сжатия трехмерных изображений объектов и сцен привели к необходимости исследования их статистических характеристик и, в частности, коэффициента автокорреляции оболочки объекта. Как известно, под оболочкой, определяющей форму объекта, понимается поверхность, которая отделяет тело объекта от окружающего его пространства. Оболочка может быть описана функцией, вид которой зависит от выбранной системы координат [43]. Для описания оболочек трехмерных объектов удобно использовать цилиндрическую систему координат. Такой выбор системы координат при работе с трехмерными объектами может быть мотивирован тем, что в своей жизни мы по существу используем именно эту систему координат (справа, слева, ближе, дальше, выше, ниже), помещая себя в ее начало. В этом случае оболочка определяется функцией

$$r = f(\alpha, z), \tag{1.29}$$

где r — длина радиус-вектора, α — угол между радиус-вектором и осью координат x, z — смещение вектора относительно начала координат по оси z. Рисунок 1.14 поясняет сказанное. (Переход из одной системы координат в другую не создает проблем, это просто преобразование координат.)



Рис. 1.14. Цилиндрическая система координат

Коэффициент автокорреляции характеризует силу статистических связей между различными точками оболочки. В рассматриваемом случае коэффициент автокорреляции определяется следующим образом

$$k(\theta,\zeta) = \frac{\left[f(\alpha,z) - \overline{f(\alpha,z)}\right] \left[f(\alpha+\theta,z+\zeta) - \overline{f(\alpha,z)}\right]}{\left[f(\alpha,z) - \overline{f(\alpha,z)}\right]^{2}},$$
(1.30)

где ζ и θ — смещения вдоль оси z и поворот вокруг нее реализации оболочки трехмерного объекта $f(\alpha + \theta, z + \zeta)$ относительно реализации этой же оболочки $f(\alpha, z), \overline{f(\alpha, z)}$ — среднее значение функции (радиус-вектора). Обратим внимание на то, что и в случае трехмерных изображений коэффициент автокорреляции является двумерной функцией взаимных смещений, что обусловлено двумерностью оболочек.

Для проведения эксперимента посредством графических редакторов 3ds Max и Poser были изготовлены трехмерные изображения различной степени сложности, от простых геометрических примитивов до скульптурных изображений голов людей. Эти изображения были лишены текстуры и были окрашены в белый цвет.

Вывод формулы для коэффициента автокорреляции оболочек трехмерных объектов

Приведем выражение (1.30) к более удобному виду. С этой целью введем новые обозначения $f_1 = \left[f(\alpha, z) - \overline{f(\alpha, z)} \right], f_2 = \left[f(\alpha + \theta, z + \zeta) - \overline{f(\alpha, z)} \right]$ и перепишем выражение для коэффициента автокорреляции оболочек (1.30) в следующем виде:

$$k(\theta,\zeta) = \frac{\overline{f_1 f_2}}{\overline{f_1^2}},$$

где черта обозначает усреднение. Преобразуем это выражение к виду

$$k(\theta,\zeta) = \frac{\overline{2f_1f_2} - \overline{f_1^2} - \overline{f_2^2} + \overline{f_1^2} + \overline{f_2^2}}{2f_1^2}.$$

Принимая во внимание, что

$$\overline{2f_1f_2} - \overline{f_1^2} - \overline{f_2^2} = -\overline{\Delta(\theta,\zeta)^2}$$

представляет собой средний квадрат разностей радиус-векторов оболочки при аргументах ζ и θ , а $\overline{f_1^2}$ и $\overline{f_2^2}$ — средние квадраты переменной составляющей радиус-векторов, а также учитывая, что

$$\overline{f_1^2} = \overline{f_2^2} \quad \text{if } \overline{f_1^2} = \left[f(\alpha, z) - \overline{f(\alpha, z)} \right]^2,$$

запишем выражение для коэффициента автокорреляции в следующем виде:

$$k(\theta,\zeta) = 1 - \frac{\overline{\Delta(\theta,\zeta)^2}}{2\left[f(\alpha,z) - \overline{f(\alpha,z)}\right]^2}.$$
 (1.31)

Приближенный способ измерения коэффициентов автокорреляции оболочек трехмерных объектов

Поскольку измерение коэффициентов автокорреляции оболочек трехмерных объектов представляет собой трудоемкую задачу, нами для ее реализации был разработан приближенный способ.

Существо этого способа поясним на примере измерения зависимости коэффициента автокорреляции от взаимного углового смещения двух реализаций оболочки θ при отсутствии взаимного их смещения вдоль вертикальной оси, т. е. при значении $\zeta = 0$.

Особенность способа заключалась в следующем. Значение среднего квадрата разностей радиус-векторов $\overline{\Delta(\theta,\zeta)^2}$ и значения среднего квадрата переменной составляющей радиус-вектора $\left[f(\alpha,z)-\overline{f(\alpha,z)}\right]^2$ измерялись не по всей поверхности оболочки объекта, а только на контурах, получающихся при ее сечении плоскостями, проведенными, например, так, как это показано на рис. 1.15, *а*. На рис. 1.15, *б*—*д* приведены эти контуры [43]. Такой подход существенно уменьшал количество точек, в которых производились измерения.

С этой целью для оболочки трехмерного объекта, а также для оболочки этого же объекта, но повернутой относительно исходного положения на угол θ , изготавливались серии пар контуров для разных значений координаты z. На рис. 1.16, a-в показана пара таких контуров для одного из сечений.



Рис. 1.15. Пример сечения объекта плоскостями и контуры, соответствующие этим сечениям при измерениях зависимости коэффициента автокорреляции от угла поворота θ



Рис. 1.16. К пояснению приближенного способа измерения коэффициента автокорреляции

Далее, определялись центры тяжести этих контуров (на рисунке они показаны крестиками). После этого определялось среднее значение радиус-вектора, $\overline{f(\alpha, z)}$ по всей серии изготовленных контуров. Затем для всей серии пар контуров вычислялся средний квадрат разности радиус-векторов оболочки $\overline{\Delta(\theta, \zeta)^2}$ и средний квадрат

переменной составляющей радиус-вектора $\overline{\left[f(\alpha,z)-\overline{f(\alpha,z)}\right]^2}$, которые подставлялись в выражение (1.31).

Экспериментальные исследования показали, что результат измерения оказывается достаточно точным уже при использовании всего шести секущих плоскостей. Аналогичным образом находилась зависимость коэффициента автокорреляции оболочки объекта $k(\zeta)$ от взаимного смещения двух ее реализаций вдоль вертикальной оси при отсутствии взаимного углового смещения этих реализаций, т. е. при $\theta = 0$. Единственным отличием этой серии измерений от предыдущей серии было то, что в рассматриваемом случае положение секущих плоскостей выбиралось таким, как показано на рис. 1.17.



Рис. 1.17. Положение секущих плоскостей при измерении зависимости коэффициента автокорреляции от смещения ζ

Измерения коэффициента автокорреляции оболочек, как уже было отмечено, выполнялись для трехмерных изображений объектов различной степени сложности. При выполнении экспериментов были измерены зависимости коэффициента автокорреляции: от взаимного углового смещения двух реализаций оболочки объекта θ при отсутствии их смещений вдоль вертикальной оси, т. е. при $\zeta = 0$, а также от взаимных смещений вдоль вертикальной оси ζ при отсутствии взаимных угловых смещений, т. е. при $\theta = 0$.

Зависимость коэффициента автокорреляции оболочек от угла θ

На рис. 1.18, *а* приведена экспериментальная зависимость коэффициента автокорреляции оболочки куба от угла θ. Из рисунка видно, что эта зависимость носит периодический характер, что объясняется структурой самой оболочки. Аналогичная зависимость, но для оболочки скульптурного изображения головы человека, показана на рис. 1.18, *б*.

Интересной особенностью этой зависимости является то, что она напоминает своей формой два периода синусоидальной функции. Общим для оболочек самых различных трехмерных объектов является то, что зависимость коэффициента автокорреляции от угла θ далека от экспоненциальной функции. Кроме того, в этом случае коэффициент автокорреляции не зависит от размеров объекта.

Следует отметить, что в зависимости от формы объекта зависимость коэффициента автокорреляции от угла может иметь много максимумов, как, например, это мы можем видеть на рис. 1.18, *а*. В случае тел вращения, когда линия наблюдения перпендикулярна оси вращения, значение коэффициента автокорреляции не зависит от этого угла и равно единице.



Рис. 1.18. Зависимость коэффициента автокорреляции от угла поворота $\boldsymbol{\theta}$

Зависимость коэффициента автокорреляции оболочек от их взаимного смещения вдоль оси *z* на величину ζ

На рис. 1.18, *в* приведена экспериментальная зависимость коэффициента автокорреляции для оболочки скульптурного изображения головы человека от взаимного смещения ее реализаций ζ. Из рисунка видно, что эта зависимость носит экспоненциальный характер.

1.11. Гистограммы оболочек реальных трехмерных изображений

В разработанной методике измерения гистограммы распределения радиусвекторов значение среднего радиус-вектора, так же, как и ранее, измерялось не по всей поверхности оболочки, а только на контурах, получающихся при ее сечении плоскостями, проведенными, например, так, как это показано на рис. 1.15. Такой подход существенно уменьшал количество точек, в которых производились измерения.



Рис. 1.19. Гистограмма распределения радиусов-векторов: сплошная линия — для оболочки скульптурного изображения головы; пунктирная линия — для оболочки куба

Методика измерения гистограммы распределения радиус-векторов включала в себя два этапа. На первом этапе измерялась зависимость вероятности p_0 события, заключающегося в том, что радиус-вектор $f(\alpha, z)$ не превышает установленного порога r_n , т. е. измерялась зависимость $p_0(r_n)$. Измерения выполнялись по всем сечениям. Затем находилось усредненное значение этой зависимости $\overline{p_0(r_n)}$. На втором этапе вычислялась собственно гистограмма распределения по формуле $P(r_n) = \overline{p_0(r_{n1})} - \overline{p_0(r_{n2})}$, где r_{n1} и r_{n2} — значения верхней и нижней границ, определяющих шаг гистограммы.

При измерении вероятности p_0 на изображение контура, образованного сечением плоскости с трехмерным объектом, накладывалось кольцо, образованное двумя окружностями с радиусами r и $r + \Delta r$. Рисунок 1.16, c поясняет изложенное. Отношение площади, которая не перекрывается контуром трехмерного объекта (на рис. 1.16, c зачерненная область) к площади кольца дает значение p_0 . Гистограмма распределения радиус-векторов, полученная описанным выше методом, приведена на рис. 1.19 для оболочек двух трехмерных объектов: для оболочки скульптурного изображения головы человека (сплошная линия) и оболочки куба (пунктирная линия). Особенностью приведенных зависимостей является, как и следовало ожидать, нулевая вероятность нулевых значений радиус-вектора. Не трудно видеть, что для цилиндра зависимость $P(r_n)$ вырождается в дельтафункцию.

Полученные результаты могут быть использованы в системах, предназначенных для сжатия цифровых потоков при передаче 3D-сцен, и, в частности, в системах виртуальной реальности.

1.12. Характеристики цвета

До сих пор речь в основном шла о так называемых полутоновых черно-белых изображениях, для описания которых было достаточно задать функцию $L_c(x, y)$, представляющую распределение яркости L_c на плоскости с координатами (x, y). Черно-белые изображения являются частным случаем более широкого класса цветных изображений (более полно эта проблема рассматривается в последующих разделах книги).

Цвет в отличие от спектральной плотности излучения представляет собой понятие субъективное, при этом его можно характеризовать тремя величинами: *светлотой*, *цветовым тоном* и *насыщенностью* [66, 76]. Из этого следует, что цвет следует рассматривать как векторную величину.

Светлота играет ведущую роль в восприятии изображений. Одни и те же цвета, но обладающие разной светлотой, вызывают различные зрительные ощущения. Так например, белый цвет в зависимости от светлоты может восприниматься

как белый, светло-серый, серый, темно-серый и т. д. Цвет, воспринимаемый при большой светлоте как желтый, при малой светлоте воспринимается как коричневый и т. п. Более того, в зависимости от яркости соседних участков изображения нейтральный цвет, характеризуемый одной и той же яркостью, может восприниматься нами и как белый, и как черный. В подтверждение этого парадокса приведем пример, заключающийся в том, что яркость черных букв газетного текста на солнце (соответственно, количество отраженного от них света) выше, чем яркость белой бумаги, на которой они напечатаны, в тени. Из изложенного следует, что светлота, представляющая собой зрительное ощущение, является нелинейной и достаточно сложной функцией яркости, которая определяется свойствами зрения. Яркость же представляет собой физическую величину и от свойств зрения не зависит.

Цветовой тон излучения характеризует свойство цвета, отличающее его от белого и серого. Названия цветов — красный, желтый, синий и т. д. — можно рассматривать как приблизительные обозначения цветового тона. Количественно цветовой тон излучения характеризуется доминирующей длиной волны, т. е. длиной волны такого спектрально-чистого цвета, который необходимо смешать с белым цветом в определенной пропорции для того, чтобы получить описываемый цвет. Между цветовым тоном и доминирующей длиной волны существует нелинейная зависимость. Так например, желтый цвет занимает на шкале длин волн интервал приблизительно 30 нм, в то время как красный — около 75 нм.

Насыщенность цвета характеризует степень разбавленности спектрально чистого цвета белым цветом. Если к спектрально чистому цвету добавить белый, то насыщенность уменьшится. Доля спектрального цвета в его смеси с белым называется чистотой цвета р

$$p = \frac{L_{\lambda}}{L_{\lambda} + L_E} = \frac{L_{\lambda}}{L},$$

где L_{λ} — яркость спектрально чистого цвета, L_E — яркость равноэнергетического белого цвета, т. е. излучения, спектральная интенсивность которого на всех длинах волн видимого излучения постоянна, L — яркость смеси. Из написанного выражения следует, что для спектрально чистых излучений чистота цвета максимальна (p=1), а для равноэнергетического белого минимальна (p=0). Следует отметить, что в силу особенностей нашей зрительной системы связь между насыщенностью цвета (т. е. зрительным ощущением насыщенности) и чистотой цвета нелинейная. Более того, при одной и той же чистоте цвета, красный и синий спектрально чистые цвета воспринимаются как высоконасыщенные, а желтый и голубой, тоже спектрально чистые, как цвета, имеющие меньшую насыщенность. Таким образом, между характеристиками восприятия цвета: светлотой, цветовым тоном и насыщенностью, с одной стороны, и характеристиками цвета: яркостью, доминирующей длиной волны и чистотой цвета, с другой стороны, имеют место нелинейные зависимости.

1.13. Помехи и их статистические характеристики

Помехи, воздействуя на сигнал изображения, вызывают появление на изображении флуктуаций яркости, которые искажают его, а при высоком уровне помех делают изображение неразличимым. Наиболее распространенным видом помех являются аддитивные, при которых сигнал U_c и помеха U_m , суммируясь алгебраически, образуют зашумленную реализацию $U = U_c + U_m$. Воздействие помехи на сигнал может также выражаться в их перемножении (мультипликативная nomexa), при этом $U = c_1 U_c U_m$, где c_1 — множитель, введенный для согласования размерностей. Возможны и другие случаи.

Помеху можно рассматривать как *случайную функцию* времени *t* при воздействии ее на электрический сигнал или как случайную функцию координат *x* и *y* при воздействии на изображение. Если помеха является функцией дискретного времени или дискретных координат, как это имеет место при передаче изображений по цифровому каналу, говорят о *случайной последовательностии*. В противном случае случайную функцию называют *случайным процессом*. Случайные функции описываются многомерными плотностями вероятности

$$W_n(U_{111}, U_{1112}, ..., U_{111k}, ...),$$

где $U_{\rm m1}$, $U_{\rm m2}$, ..., $U_{\rm mk}$ — мгновенные значения помехи, которые она принимает в моменты времени t_1 , t_2 , ..., t_k . Подобным же образом записывают многомерную плотность вероятности помехи, представляющую флуктуации яркости на изображении. Помехи называются *стационарными*, если их статистические характеристики не зависят от времени (от координат).

Для вычисления среднего значения помехи $\overline{U_{\rm m}}$ и ее среднего квадрата $\overline{U_{\rm m}^2}$, которые называют соответственно первым и вторым начальными моментами распределения, достаточно знать ее одномерную плотность вероятности $W_n(U_{\rm m})$, при этом

$$\overline{U_{\mathfrak{II}}} = \int_{-\infty}^{\infty} U_{\mathfrak{II}} W_n (U_{\mathfrak{II}}) dU_{\mathfrak{II}} , \qquad (1.32)$$

$$\overline{U_{\mathfrak{III}}^{2}} = \int_{-\infty}^{\infty} U_{\mathfrak{III}}^{2} W_{n} (U_{\mathfrak{III}}) dU_{\mathfrak{III}} . \qquad (1.33)$$

Чертой над $U_{\rm m}$ и $U_{\rm m}^2$ обозначено усреднение этих величин по ансамблю. В ряде случаев более удобным бывает использовать так называемые *центральные моменты распределения*, т. е. моменты, центрированные относительно среднего значения помехи $\overline{U_{\rm m}}$. Так выражение

$$\overline{\left(U_{\mathfrak{m}}-\overline{U_{\mathfrak{m}}}\right)^{2}} = \int_{-\infty}^{\infty} \left(U_{\mathfrak{m}}-\overline{U_{\mathfrak{m}}}\right)^{2} W_{n}\left(U_{\mathfrak{m}}\right) dU_{\mathfrak{m}}$$
(1.34)

представляет собой второй центральный момент распределения, или *дисперсию*, которую в дальнейшем будем обозначать через

$$\sigma^2 = \overline{\left(U_{\rm III} - \overline{U_{\rm III}}\right)^2} \,. \tag{1.35}$$

Среди случайных помех особое положение занимает *стационарная флуктуацион*ная помеха (или, как ее часто называют, *флуктуационный шум*), которая возникает в датчиках сигнала изображения (например, в передающих телевизионных матрицах), в первых каскадах предварительных усилителей передающих телевизионных камер, во входных цепях радиоприемных устройств и т. д., и распределена по нормальному закону. Флуктуационная помеха обладает свойством эргодичности, заключающемся в том, что средние по ансамблю $\overline{U_{\rm m}}$, $\overline{U_{\rm m}^2}$ и т. д. с вероятностью, сколь угодно близкой к единице, совпадают со средними по времени. Статистические свойства стационарной флуктуационной помехи полностью описываются двумерной плотностью вероятности

$$W_{n}(U_{\rm m},U_{\rm m\tau}) = \frac{1}{2\pi\sigma^{2}\sqrt{1-\rho_{\rm m}^{2}(\tau)}} \exp\left[-\frac{U_{\rm m}^{2}-2\rho_{\rm m}(\tau)U_{\rm m}U_{\rm m\tau}+U_{\rm m\tau}^{2}}{2\sigma^{2}\left[1-\rho_{\rm m}^{2}(\tau)\right]}\right], \quad (1.36)$$

где $\rho_{\rm m}(\tau)$ — коэффициент автокорреляции помехи, τ — интервал времени, на который смещены значения $U_{\rm m}$ и $U_{\rm m\tau}$.

Коэффициент автокорреляции помехи $\rho_{\rm m}(\tau)$ связан с ее функцией автокорреляции $R_{\rm m}(\tau)$ соотношением

$$R_{\rm III}(\tau) = \rho_{\rm III}(\tau)\sigma^2 \tag{1.37}$$

и устанавливает статистическую связь между значениями помехи $U_{\rm m}$, разнесенными во времени на интервал τ . Не останавливаясь на свойствах функции автокорреляции, поскольку о них говорилось в *разд. 1.8*, напомним лишь о том, что она является четной функцией времени τ и связана со спектральной интенсивностью помехи $S_{\rm m}(\omega)$ парой преобразований Фурье. Обратим внимание, что

$$S_{\rm III}(\omega) = \int_{-\infty}^{\infty} R_{\rm III}(\tau) \exp(-i\omega\tau) d\tau \qquad (1.38)$$

является четной функцией частоты ω, при этом

$$\sigma^{2} = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{\text{III}}(\omega) d\omega. \qquad (1.39)$$

Иногда вместо спектральной интенсивности помехи $S_{\mu\nu}(\omega)$ используют ее энергетический спектр $F_{\mu\nu}(\omega)$, который определяют следующим образом:

$$F_{\rm III}(\omega) = \begin{cases} S_{\rm III}(\omega)/\pi & \text{при } \omega \ge 0, \\ 0 & \text{при } \omega < 0. \end{cases}$$
(1.40)

В этом случае

$$\sigma^2 = \int_0^\infty F_{\rm III}(\omega) d\omega. \qquad (1.41)$$

Как известно, спектральная интенсивность дробового и теплового шумов, являющихся первопричиной флуктуаций, не зависит от частоты в очень широких пределах. В связи с этим вводят понятие (модель) так называемого *белого шума*, который при нулевом среднем значении, т. е. при $\overline{U_{\rm m}} = 0$, и нормальном распределении по уровням имеет спектральную интенсивность, не зависящую от частоты в пределах от $-\infty$ до ∞ .

Наряду с этим часто пользуются понятием *квазибелого шума*, который отличается от белого шума тем, что его спектральная интенсивность постоянна в интервале частот от $-\omega_{rp}$ до ω_{rp} и равна нулю за его пределами.

Можно показать путем применения преобразования Фурье к его спектральной интенсивности, что коэффициент автокорреляции квазибелого шума равен

$$\rho_{\rm III} = \sin \omega_{\rm rp} \tau / \omega_{\rm rp} \tau \,. \tag{1.42}$$

Эта зависимость представлена графически на рис. 1.20.



Рис. 1.20. Зависимость коэффициента автокорреляции квазибелого шума от ω_{гр}τ

В отличие от квазибелого шума функция автокорреляции белого шума выражается через дельта-функцию

$$R_{\rm III}(\tau) = \frac{S_{\rm III}(0)}{2\pi} \int_{-\infty}^{\infty} \exp(\mathbf{i}\omega\tau) d\omega = S_{\rm III}(0)\delta(\tau),$$

поскольку

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}\exp(\mathbf{i}\omega\tau)d\omega=\delta(\tau),$$

из чего следует, что любые два отсчета, взятые на конечном интервале времени τ , оказываются некоррелированными.

Примечание

Особенностью модели белого шума является также то, что при конечном значении спектральной интенсивности дисперсия помехи стремится к бесконечности.

Все изложенное остается справедливым и для случая, когда квазибелый и белый шумы представляют собой флуктуации яркости на изображении, т. е. являются двумерными. При этом спектральная интенсивность двумерного квазибелого шума

$$S_{\rm III}(\omega_x, \omega_y) = \begin{cases} S_{\rm III}(0,0) \text{ при } |\omega_x| \le \omega_{\rm xrp} \text{ и } |\omega_y| \le \omega_{\rm yrp}, \\ 0 \text{ при невыполнении этих условий,} \end{cases}$$
(1.43)

а его функция автокорреляции

$$R_{\rm III}(\xi,\eta) = \overline{L_{\rm III}^2} \frac{\sin\omega_{\rm xrp}\xi}{\omega_{\rm xrp}\xi} \frac{\sin\omega_{\rm yrp}\eta}{\omega_{\rm yrp}\eta} \,. \tag{1.44}$$

Этот вид шума проявляется на изображении в виде беспорядочных флуктуаций яркости, как, например, показано на рис. 5.12, б.

Другой полезной моделью помехи, с которой приходится иметь дело на практике, является помеха, описываемая равномерным законом распределения по уровням на некотором интервале $-\delta_{\rm kB}/2...\delta_{\rm kB}/2$. Помехи такого рода возникают при квантовании изображений по интенсивности с шагом $\delta_{\rm kB}$ при их оцифровке.

Шум на изображениях, получаемых с использованием лазеров, может иметь как экспоненциальное распределение, так и распределение Эрланга [23]. В первом случае плотность вероятности распределения шума описывается выражением

$$W(L_{\rm III}) = \begin{cases} \frac{1}{L_{\rm III}} \exp\left(-\frac{L_{\rm III}}{L_{\rm III}}\right) & \text{при } L_{\rm III} \ge 0, \\ 0 & \text{при } L_{\rm III} < 0, \end{cases}$$

где $\overline{L_{III}} > 0$ — среднее значение шума. Дисперсия шума при этом равна

$$\sigma^2 = \left(\overline{L_{\rm III}}\right)^2.$$

При распределении Эрланга, плотность вероятности шума описывается выражением

$$W(L_{\rm m}) = \begin{cases} \frac{a^b L_{\rm m}^{b-1}}{(b-1)!} \exp(-aL_{\rm m}) & \text{при } L_{\rm m} \ge 0, \\ 0 & \text{при } L_{\rm m} < 0, \end{cases}$$

где *a* > 0, *b* — положительное целое число, а символ "!" обозначает факториал. Среднее значение и дисперсия шума при этом равны соответственно

$$\overline{L_{\text{III}}} = b/a$$
, $\sigma^2 = b/a^2$

Следующим видом помехи, который мы рассмотрим, является *аддитивная бипо-лярная импульсная помеха*. Возникновение импульсной биполярной помехи часто обусловлено коммутационными процессами в электрических схемах, наводками, а также процессами электрических разрядов. Распределение вероятности p_n биполярной импульсной помехи описывается выражением

$$p_n = \begin{cases} p_a \text{ при } L_{\rm III} = L_a, \\ p_6 \text{ при } L_{\rm III} = -L_6, \\ 0 \text{ в остальных случаях} \end{cases}$$

где p_a — вероятность возникновения положительного импульса помехи, величина которого равна L_a , p_5 — вероятность возникновения отрицательного импульса помехи, величина которого равна $-L_5$. В том случае если одна из вероятностей p_a или p_5 равна нулю, биполярная помеха превращается в униполярную. Положи-

тельные импульсы помехи выглядят на изображении в виде белых точек, а отрицательные — в виде черных точек. Биполярную импульсную помеху в переводной литературе, например в руководствах по MATLAB, часто называют помехой типа "coль u nepeų", поскольку она напоминает рассыпанные на изображении крупицы соли и перца. Одной из особенностей импульсной биполярной помехи, с которой довольно часто приходится встречаться, является то, что величина ее импульсов оказывается больше размаха сигнала изображения. В результате этого возникает ограничение интенсивности тех пикселов изображения, на которые попали импульсы помехи и они становятся либо белыми, либо черными.

В заключение остановимся еще на так называемой *структурной помехе*, которая обусловлена тем, что отдельные участки светочувствительной поверхности датчика сигнала изображения (например, ПЗС) имеют различные чувствительности и различные значения "сигнала фона". Структурная помеха принципиально может быть полностью скомпенсирована, если предварительно записать в запоминающее устройство данные о ее локальных значениях и использовать эти значения в процессе соответствующей обработки сигнала.

1.14. Источники флуктуационного шума в цифровых фото- и видеокамерах на ПЗС

Фактором, ограничивающим световую чувствительность цифровых фото- и видеокамер, является флуктуационный шум. Этот шум слагается из двух компонентов: компонента шума, возникающего в преобразователе изображения в сигнал, и компонента шума, имеющегося в световом потоке (фотонный шум). Поскольку в настоящее время в качестве преобразователей изображений в сигнал применяются приборы с зарядовой связью (ПЗС), то в дальнейшем речь пойдет о них [120]. Первый компонент шума (шум матрицы ПЗС), в свою очередь, включает в себя шум, обусловленный темновым током, (он, как правило, невелик), и шум считывания. Рассмотрим более подробно эти компоненты шума.

Фотонный шум обусловлен квантовой природой света. Воздействующие на ПЗС фотоны света порождают в кремниевом слое прибора фотоэлектроны. Количество фотоэлектронов, накапливаемых за время экспозиции, в каждой ячейке ПЗС описывается распределением Пуассона. При высоких уровнях освещенности число фотонов велико и закон Пуассона может быть аппроксимирован Гауссовым законом. Среднеквадратичное значение флуктуирующей (шумовой) составляющей заряда, накопленного в ячейке ПЗС, $Q_{\phi m}$ и его среднее значение Q_{ϕ} , представляющее полезный сигнал, взаимосвязаны

$$Q_{\rm dun} \approx \sqrt{Q_{\rm d}} \; .$$

Темновой шум возникает под воздействием тепла, вследствие генерации электронов в кристаллической решетке кремния ПЗС, поскольку процесс генерации термоэлектронов является случайным процессом. Величина темнового тока не зависит от величины светового потока, который проецируется на ПЗС, а следовательно, не зависит от величины светового потока и темновой шум. Количество термоэлектронов, накапливаемых в каждой ячейке ПЗС за время экспозиции, так же, как и количество фотоэлектронов, описывается распределением Пуассона. Поэтому средне-квадратичное значение темнового шума $Q_{\text{ти}}$ подчиняется соотношению

$$Q_{\text{THI}} \approx \sqrt{Q_{\text{T}}}$$
,

где $Q_{\rm T}$ — среднее значение накопленного за время экспозиции заряда, обусловленного термоэлектронами.

В настоящее время типовые значения темнового тока при комнатной температуре для лучших ПЗС-матриц составляют сотые доли нА/см². Для вещательного и бытового телевидения такой темновой ток практически незаметен, а значит, мал и темновой шум. В тех же случаях, когда величину темнового шума необходимо уменьшить, например, в астрономических системах, прибегают к охлаждению кристалла матрицы ПЗС либо путем применения термоэлектронного охлаждения с использованием батарей Пельтье, либо посредством применения азотных криостатов.

Шум считывания (его часто называют шумом предусилителя) в основном обусловлен предусилителем на чипе. Среднеквадратичное значение шума считывания определяется параметрами матрицы ПЗС и не зависит от величины считываемого сигнала, а следовательно, не зависит и от светового потока. Шум считывания распределен по гауссову закону. Эта составляющая шума, строго говоря, имеет два компонента: компонент, не зависящий от частоты, и компонент, спектральная интенсивность которого убывает обратно пропорционально частоте. Второй компонент в значительной степени подавляется схемой двойной коррелированной выборки, которая служит фильтром верхних частот.

Отношение "сигнал — шум" Ψ (эта величина для каждого цветового канала рассчитывается отдельно) является одним из важнейших параметров, определяющих качество изображения. Отношение "сигнал — шум" определяется как отношение размаха сигнала от минимума до максимума к результирующему среднеквадратичному значению шума. Выражение для отношения "сигнал — шум" рассчитывается по формуле

$$\Psi = \frac{IK_{\text{KB}} t}{\sqrt{IK_{\text{KB}} t + N_{\text{T}} + N_{\text{CY}}^2}},$$

где I — количество фотонов, попадающих в ячейку матрицы ПЗС за одну секунду, $K_{\text{KB}\Im}$ — квантовая эффективность (величина, близкая к единице), t — время накопления заряда, N_{T} — количество термоэлектронов, генерируемых в ячейке матрицы за время накопления заряда, $N_{\text{сч}}$ — среднеквадратичное значение шума считывания.

Из приведенного выражения видно, что при низких уровнях освещенности, т. е. при малых значениях I, отношение сигнала к шуму определяется главным образом

шумом считывания. Это следует из того, что величина заряда, $IK_{\rm KB}$, накопленного в ячейке ПЗС при низких освещенностях, мала, а значит, небольшим будет и значение среднего квадрата фотонного шума. При увеличении освещенности увеличивается и отношение сигнала к шуму. При этом величина среднеквадратичного значения фотонного шума возрастает пропорционально квадратному корню от освещенности и при больших уровнях освещенности отношение сигнала к шуму определяется в основном фотонным шумом.

Что касается спектрального состава шума на выходе матрицы ПЗС, то отметим, что в первом приближении можно считать, что его спектральная интенсивность не зависит от частоты для всех трех цветовых каналов.

1.15. Модель флуктуационного шума матрицы ПЗС

Из изложенного следует, что дисперсия флуктуационного шума на выходе матрицы зависит от величины полезного сигнала. При этом модель зашумленного изображения для каждого цветового канала одинакова и может быть представлена как

$$L(k,n) = L_{\rm c}(k,n) + L_{\rm III}(k,n),$$

где L(k,n) — интенсивность пиксела, расположенного в *k*-ой строке и *n*-ом столбце зашумленного изображения, $L_{c}(k,n)$ — интенсивность соответствующего пиксела гипотетического незашумленного изображения, $L_{uu}(k,n)$ — величина флуктуационного шума, добавляемая к яркости пиксела гипотетического незашумленного изображения. Другими словами, $L_{uu}(k,n)$ представляет собой компонент изображения, обусловленный флуктуационным шумом.

Модель флуктуационного шума на выходе матрицы ПЗС также одинакова для каждого цветового канала и может быть представлена в виде

$$L_{\mathrm{III}}(k,n) = \sqrt{L_{\mathrm{c}}(k,n)}\sigma_{0} + L_{\mathrm{III}}(k,n),$$

где $\sqrt{L_c(k,n)}\sigma_0$ — флуктуирующий компонент, определяемый фотонным шумом, σ_0 — коэффициент, определяющий составляющую дисперсии фотонного шума, $L_{\rm m1}(k,n)$ — флуктуирующий компонент, определяемый темновым шумом и шумом считывания. Так как эти составляющие шума взаимно не коррелированны, то выражение для дисперсии результирующего шума можно записать в виде

$$\sigma_{\Sigma}^2 = L_{\rm c}\sigma_0^2 + \sigma_1^2 \,,$$

где σ_1^2 — составляющая дисперсии шума, не зависящего от величины сигнала, т. е. составляющая дисперсии, обусловленная темновым шумом и шумом считывания. При этом, как не трудно видеть,

$$\sigma_1^2 = \overline{\left[L_{\rm III1}(k,n)\right]^2}.$$

Заканчивая этот раздел, отметим, что в первом приближении флуктуационный шум на выходе матрицы ПЗС можно рассматривать как белый аддитивный. Однако поскольку в цифровых фото- и видеокамерах всегда производится предварительная обработка сигнала, например, гамма-коррекция, то шум на выходе фото- и видеокамер обычно отличается от белого.

1.16. Технология маркирования изображений

Цифровые водяные знаки используются для защиты авторских или имущественных прав на цифровые изображения, фотографии или другие оцифрованные произведения искусства. Технология цифровых водяных знаков является одним из разделов технологии стеганографии — раздела техники скрытной передачи сообщений. При использовании этой технологии в изображение встраивается так называемый водяной знак — набор символов, присутствие которого незаметно зрителю, но может быть обнаружено посредством специального алгоритма, располагающего стегоключом. К водяным знакам предъявляются следующие требования:

- □ водяной знак, как уже отмечалось, не должен вносить в маркированное изображение заметных зрителю изменений, т. е. он должен оставаться невидимым;
- водяной знак должен быть устойчивым к искажениям и преобразованиям изображения, таким как: изменение масштаба, увеличение или уменьшение контраста или яркости, обрезка, размытие и подчеркивание границ, преобразование в другой формат, сжатие с потерей информации и т. д.

В настоящее время разработано много различных методов встраивания цифровых водяных знаков в изображения и другие сообщения, информацию о некоторых из них читатель найдет в [111, 127, 134, 162].

Наиболее распространенным, но и наименее устойчивым к искажениям является метод замены битов младших разрядов, представляющих яркость/цвет пиксела или так называемый LSB-метод. При этом модификация младших разрядов, как правило, не вызывает значительной трансформации изображения и не обнаруживается визуально [69].

В настоящее время при формировании цифровых водяных знаков применяется принцип встраивания метки, которая представляет собой узкополосный сигнал, в широком диапазоне частот маркируемого изображения. Этот метод реализуется посредством использования двух различных алгоритмов.

В случае применения первого алгоритма информация передается посредством фазовой модуляции "несущей", представляющей собой псевдослучайную последовательность чисел.

В случае применения второго алгоритма весь диапазон частот делится на несколько субдиапазонов и передача производится между этими субдиапазонами. В отношении маркируемого изображения метку можно рассматривать как некоторый дополнительный шум, который к нему добавляется. Однако поскольку в изображении всегда присутствует шум, то его незначительное возрастание за счет добавления метки не приводит к заметному для зрителя увеличению искажений. Кроме того, и это принципиально важно, сигнал, представляющий метку, распространяется по всему изображению, благодаря чему достигается устойчивость к обрезке изображения.



Воспроизведение цвета на изображениях и управление цветом

2.1. Аксиомы Грассмана и законы смешения цветов

В основе цветовых расчетов, без которых были бы невозможны ни редактирование изображений, ни вывод их на экран монитора или на печать, лежит трехкомпонентная теория зрения. Согласно этой теории сетчатка глаза содержит три типа светочувствительных клеток — колбочек, которые различаются между собой спектральной чувствительностью. Соотношение возбуждений этих трех типов колбочек, зависящее от спектрального состава облучающего их света, и определяет цвет, который мы ощущаем.

Гипотеза о трехкомпонентной природе зрения впервые была выдвинута в 1756 г. М. В. Ломоносовым в работе "Слово о происхождении света, новую теорию о цветах представляющем". Спустя 46 лет к этой гипотезе обратился Юнг, затем в 1854 г. Грассман и, наконец, Гельмгольц, которым была детально разработана и опубликована в 1911 г. теория цветового зрения в современном виде.

В основу этой теории положены данные о строении сетчатки глаза, а также опыты по уравниванию цветов.

Опыты по уравниванию цветов

Эти опыты выполнялись по схеме, приведенной на рис. 2.1. На поверхность идеального диффузного отражателя проецируется пятно света C' с произвольным спектральным составом $C(\lambda)$, где λ — длина световой волны. Рядом на ту же поверхность проецируются пятно опорного белого цвета W', спектральный состав в котором обозначим $W(\lambda)$, а также три взаимно перекрывающихся пятна, P'_1 , P'_2 , P'_3 трех основных цветов, спектральный состав которых обозначим $P_1(\lambda)$, $P_2(\lambda)$ и $P_3(\lambda)$.

Сначала интенсивности трех основных цветов устанавливаются таким образом, чтобы их общая часть имела бы те же самые светлоту, цветовой тон и насыщенность, что и опорный белый цвет. Интенсивности трех основных цветов $A_1(W)$,

 $A_2(W)$ и $A_3(W)$ измеряют в каких-либо физических величинах, например, в ваттах. После этого переходят к уравниванию измеряемого цвета *C*' путем подбора интенсивностей основных цветов $A_1(C)$, $A_2(C)$ и $A_3(C)$.



Рис. 2.1. Уравнивание цветов

В процессе уравнивания в зависимости от измеряемого цвета C' возможны следующие три ситуации.

1. При выполнении измерений удается подобрать интенсивности трех взаимно перекрывающихся пятен основных цветов таким образом, что цвет той части пятен, где они перекрываются, не отличим от измеряемого цвета. В этом случае измеряют значения $A_1(C)$, $A_2(C)$ и $A_3(C)$, а затем вычисляют величины

$$T_{1}(C) = A_{1}(C) / A_{1}(W), \ T_{2}(C) = A_{2}(C) / A_{2}(W), \ T_{3}(C) = A_{3}(C) / A_{3}(W),$$
(2.1)

которые называются координатами цвета C'. Координаты цвета однозначно определяют цвет.

2. Измеряемый цвет уравнять не удается. В этом случае по очереди пятно одного из основных цветов накладывают на измеряемый цвет, как это показано на рис. 2.1 во втором ряду, и пытаются подобрать такие интенсивности, при которых достигается уравнивание по цвету общей части пятен справа и слева. Если это удается сделать, измеряют значения интенсивностей основных цветов, при которых получено уравнивание, при этом интенсивность основного цвета, пятно которого перекрывает пятно измеряемого цвета, берется со знаком "минус". На рис. 2.1 таким цветом является цвет P'_3 , поэтому запись применительно к примеру, приведенному на рисунке, будет выглядеть следующим образом:

$$T_1(C) = A_1(C)/A_1(W), \ T_2(C) = A_2(C)/A_2(W), \ T_3(C) = -A_3(C)/A_3(W)$$

3. Наконец, возможна такая ситуация, когда для достижения уравнивания необходимо на пятно измеряемого цвета накладывать не одно, а два пятна основных цветов. В этом случае необходимо: во-первых, определить, какие именно два основных цвета должны быть наложены на пятно измеряемого цвета для достижения уравнивания по цвету, а во-вторых, подобрать и измерить интенсивности основных цветов, при которых достигается это уравнивание. Как и в предыдущем случае, знаки интенсивностей основных цветов, перемещенных слева направо, должны быть взяты отрицательными. В примере, показанном на рис. 2.1, такими основными цветами являются цвета P₂ и P₃, и, следовательно, координаты цвета определятся как

$$T_1(C) = A_1(C)/A_1(W), \ T_2(C) = -A_2(C)/A_2(W), \ T_3(C) = -A_3(C)/A_3(W)$$

На основании этих опытов Грассманом были сформулированы аксиомы уравнивания цветов, положенные в дальнейшем в основу колориметрии.

Аксиомы уравнивания цветов

- 1. Любой цвет может быть уравнен путем смешивания не более чем трех линейно независимых цветов, т. е. таких цветов, когда ни один из них не является результатом смешения двух других. Эта аксиома известна как *первый закон смешения цветов*.
- 2. Уравнивание, достигнутое при одной интенсивности света, сохраняется в широком диапазоне изменения интенсивностей.
- 3. Смесь цветов не может быть разделена зрительной системой человека на отдельные компоненты. Или, другими словами, цвет смеси нескольких излучений определяется только цветами смешиваемых компонентов, не зависимо от того, как они получены. Эта аксиома известна как *третий закон смешения цветов*.
- 4. Яркость смеси цветов равна сумме яркостей ее компонентов.
- 5. Закон сложения. Если цвет M' эквивалентен цвету N', а цвет P' цвету Q', т. е. если они для человека не различимы, то смесь цветов M' и P' эквивалентна смеси цветов N' и Q'. Записывается это утверждение следующим образом:

если
$$M' = N'$$
 и $P' = Q'$, то $M' + P' = N' + Q'$.

Здесь и далее в аналогичных выражениях знак "=" следует понимать как знак эквивалентности.

6. Закон вычитания. Если смесь цветов M' и P' эквивалентна смеси цветов N' и Q', а цвет P' эквивалентен цвету Q', то цвета M' и N' также эквивалентны друг другу. Это утверждение может быть записано следующим образом:

если
$$M' + P' = N' + Q'$$
, а $P' = Q'$, то $M' = N'$.

- 7. Закон транзитивности. Согласно этому закону: если цвет M' эквивалентен цвету N', а цвет N' эквивалентен цвету P', то цвет M' эквивалентен цвету P'.
- 8. Уравнивание цветов. Для любого цвета С' справедливо одно из трех соотношений:
 - цвет C' уравнивается смесью $A_1(C)$ единиц единичного количества цвета M, $A_2(C)$ единиц единичного количества цвета N и $A_3(C)$ единиц единичного количества цвета P;
 - смесь цвета C' с $A_1(C)$ единицами единичного количества цвета M уравнивается смесью $A_2(C)$ единиц единичного количества цвета N и $A_3(C)$ единиц единичного количества цвета P;
 - смесь цвета C' с $A_1(C)$ единицами единичного количества цвета M и $A_2(C)$ единицами единичного количества цвета N уравнивается $A_3(C)$ единицами единичного количества цвета P.

Какое из трех соотношений будет выполнено, зависит от смешиваемых цветов.

Аксиомы Грассмана дополняют еще одним экспериментально установленным законом, который известен как *второй закон смешения цветов*. Этот закон утверждает, что непрерывному изменению физических характеристик излучения соответствует непрерывное изменение цвета.

Перечисленные соотношения являются основой колориметрии — науки об измерении цвета.

2.2. Принципы построения колориметрической системы

Из восьмой аксиомы Грассмана следует, что цвет C' может быть уравнен смесью трех основных цветов P_1 , P_2 и P_3 , т. е.

$$C' = A_1(C)P_1 + A_2(C)P_2 + A_3(C)P_3, \qquad (2.2)$$

где $A_1(C)$, $A_2(C)$, $A_3(C)$ — интенсивности основных цветов, которые могут принимать как положительные, так и отрицательные значения (о физическом смысле отрицательных значений этих величин было сказано ранее).

Поскольку интенсивности некогерентных источников света складываются линейно, то спектральная интенсивность (плотность) смеси равна сумме спектральных плотностей ее компонентов.

Из формулы (2.2) следует, что спектральная плотность $C(\lambda)$ может быть заменена эквивалентной спектральной плотностью смеси основных цветов, т. е.

$$C(\lambda) = A_1(C)P_1(\lambda) + A_2(C)P_2(\lambda) + A_3(C)P_3(\lambda), \qquad (2.3)$$

где λ — длина световой волны. Другими словами, излучение со спектральной плотностью $C(\lambda)$ и излучение со спектральной плотностью $A_1(C)P_1(\lambda) + A_2(C)P_2(\lambda) + A_3(C)P_3(\lambda)$ вызывают одинаковые цветовые ощущения.

Выражая интенсивности основных цветов в формуле (2.3) через координаты цвета в соответствии с формулой (2.1), получим

$$C(\lambda) = T_1(C)A_1(W)P_1(\lambda) + T_2(C)A_2(W)P_2(\lambda) + T_3(C)A_3(W)P_3(\lambda).$$
(2.4)

Из четвертой аксиомы Грассмана следует, что яркость цвета L(C) равна сумме яркостей его компонентов

$$L(C) = \int_{\lambda_{\text{MHH}}}^{\lambda_{\text{MARC}}} C(\lambda)V(\lambda)d\lambda = \int_{\lambda_{\text{MHH}}}^{\lambda_{\text{MARC}}} \left[A_1(C)P_1(\lambda) + A_2(C)P_2(\lambda) + A_3(C)P_3(\lambda)\right]V(\lambda)d\lambda,$$

где $V(\lambda)$ — функция относительной спектральной световой эффективности монохроматического излучения для дневного зрения (кривая видности); $\lambda_{\text{мин}}$, $\lambda_{\text{макс}}$ длины волн светового излучения, определяющие интервал электромагнитного излучения, воспринимаемого как световое излучение. Выражая интенсивность основных цветов через координаты цвета, находим

$$L(C) = T_1(C)L(P_1) + T_2(C)L(P_2) + T_3(C)L(P_3), \qquad (2.5)$$

где

$$L(P_{1}) = \int_{\lambda_{\text{MARC}}}^{\lambda_{\text{MARC}}} A_{1}(W) P_{1}(\lambda) V(\lambda) d\lambda,$$
$$L(P_{2}) = \int_{\lambda_{\text{MARC}}}^{\lambda_{\text{MARC}}} A_{2}(W) P_{2}(\lambda) V(\lambda) d\lambda,$$
$$L(P_{3}) = \int_{\lambda_{\text{MARC}}}^{\lambda_{\text{MARC}}} A_{3}(W) P_{3}(\lambda) V(\lambda) d\lambda$$

— яркостные коэффициенты основных цветов. Формулы (2.4) и (2.5) являются количественной основой колориметрии. Для создания колориметрической системы необходимо выполнить следующее:

- 1. Выбрать основные цвета и определить единицы их измерения *P*₁, *P*₂, *P*₃. Поскольку выбор основных цветов ничем не ограничен, число колориметрических систем, которые возможно создать, также не ограничено.
- 2. Определить для выбранных основных цветов координаты цвета спектральных излучений единичной мощности $\overline{T_1(\lambda)}$, $\overline{T_2(\lambda)}$, $\overline{T_3(\lambda)}$, которые называются удельными координатами или кривыми смешения.

Знание удельных координат позволяет определять координаты цвета излучения с произвольной спектральной плотностью $C(\lambda)$ путем взвешенного суммирования $\overline{T_1(\lambda)}$, $\overline{T_2(\lambda)}$, $\overline{T_3(\lambda)}$, при этом взвешивающей функцией является спектральная плотность

$$T_{1}(C) = \int_{\lambda_{\text{MRC}}}^{\lambda_{\text{MRC}}} C(\lambda) \overline{T_{1}(\lambda)} d\lambda,$$

$$T_{2}(C) = \int_{\lambda_{\text{MRC}}}^{\lambda_{\text{MRC}}} C(\lambda) \overline{T_{2}(\lambda)} d\lambda,$$

$$T_{3}(C) = \int_{\lambda_{\text{MRC}}}^{\lambda_{\text{MRC}}} C(\lambda) \overline{T_{3}(\lambda)} d\lambda.$$
(2.6)

Из формул (2.6) следует, что любой цвет может быть полностью описан, если задать три его координаты $T_1(C)$, $T_2(C)$, $T_3(C)$, а значит, он может быть представлен положением точки в трехмерном пространстве T_1 , T_2 , T_3 , или же вектором, как это показано на рис. 2.2. При этом некоторые координаты цвета могут принимать отрицательные значения, однако результирующая яркость, определяемая формулой (2.5), всегда будет положительной. Как уже отмечалось ранее, отрицательные значения координат цвета интерпретируются следующим образом: основной цвет, соответствующий отрицательному значению координаты, должен быть прибавлен к измеряемому цвету для достижения уравнивания смесью двух оставшихся основных цветов. Это замечание носит принципиальный характер. Так, например, *цветм, у которого хотя бы одна координата отрицательна, не может быть получен путем физического смешения трех основных цветов*.



Рис. 2.2. Представление цвета в цветовом пространстве

Часто при описании цвета его яркость не представляет интереса, и тогда от координат цвета переходят к координатам цветности (координатам единичных цветов)

$$t_1 = \frac{T_1}{T_1 + T_2 + T_3}, \ t_2 = \frac{T_2}{T_1 + T_2 + T_3}, \ t_3 = \frac{T_3}{T_1 + T_2 + T_3}$$

которые полностью определяют цветовой тон и насыщенность цвета. Соотношение $t_1 + t_2 + t_3 = 1$, в справедливости которого легко убедиться, определяет плоскость единичных цветов, показанную на рис. 2.2 штриховкой, а координаты единичных цветов — точку прокола этой плоскости вектором цвета. Поскольку $t_3 = 1 - t_1 - t_2$, то для полного описания цветности достаточно использовать всего две переменные t_1 и t_2 , что удобно при представлении результатов цветовых расчетов на двумерном графике. Переход к такому графику достигается путем проекции плоскости единичных цветов, показанный на рис. 2.2 штриховкой, преобразуется в прямоугольный треугольник.

Рассмотрев принципы, положенные в основу колориметрии, перейдем к описанию колориметрических систем, наиболее часто применяемых на практике.

2.3. Колориметрическая система RGB

В 1931 г. международной комиссией по освещению (МКО), или как ее еще называют СІЕ, от французского названия Commission Internationale de L'Eclairage, была разработана и рекомендована к применению *колориметрическая система RGB*. Отличительной особенностью этой колориметрической системы является использование в качестве основных цветов трех монохроматических излучений со следующими длинами волн: для красного (R) — 700 нм, для зеленого (G) — 546,1 нм и для синего (B) — 435,8 нм.

Единицы измерения координат цвета, которые в этой системе обозначают через r', g', b', были выбраны таким образом, чтобы для опорного белого цвета типа E, характеризуемого постоянной спектральной интенсивностью в видимой части спектра, они были бы одинаковы, т. е. r' = g' = b'.

Использование в качестве основных — реальных цветов R, G, B удобно, поскольку позволяет при колориметрических измерениях применять *метод уравнивания цветов*. При этом измеряемый цвет C' может быть записан как

$$C' = r'R + g'G + b'B$$

а его яркость

$$L = r' + 4,5907\,g' + 0,0601b',$$

где *R*, *G* и *B* — единицы красного, зеленого и синего соответственно. Из написанной формулы следует, что наибольший вклад в яркость вносит зеленый компонент цвета, а наименьший — синий компонент.



Рис. 2.3. График удельных координат колориметрической системы RGB

Для колориметрической системы RGB путем экспериментальных измерений были получены зависимости удельных координат $\overline{r(\lambda)}$, $\overline{g(\lambda)}$, $\overline{b(\lambda)}$ от длины волны λ , график которых для поля зрения 2° показан на рис. 2.3. Для любого монохроматического цвета, за исключением основных цветов R, G, B, одна из удельных координат $\overline{r(\lambda)}$, $\overline{g(\lambda)}$ или $\overline{b(\lambda)}$ всегда отрицательна. Из этого следует, что монохроматические цвета, кроме самих основных, не могут быть физически реализованы путем смешения основных цветов.

Удельные координаты (кривые смешения) служат связующим звеном между физической характеристикой излучения — его спектральной интенсивностью $C(\lambda)$ и координатами цвета r', g' и b', которые определяют его цветовое восприятие.

$$r' = \int_{\lambda_{\text{MHH}}}^{\lambda_{\text{MARC}}} C(\lambda) \overline{r(\lambda)} d\lambda, \ g' = \int_{\lambda_{\text{MHH}}}^{\lambda_{\text{MARC}}} C(\lambda) \overline{g(\lambda)} d\lambda, \ b' = \int_{\lambda_{\text{MHH}}}^{\lambda_{\text{MARC}}} C(\lambda) \overline{b(\lambda)} d\lambda.$$

Рассчитав координаты цвета, легко найти координаты цветности (координаты единичных цветов)

$$r=\frac{r'}{m}, g=\frac{g'}{m}, b=\frac{b'}{m},$$

где m = r' + g' + b' — цветовой модуль.



Рис. 2.4. График цветности колориметрической системы RGB

На рис. 2.4 приведен график цветностей, на котором показан цветовой треугольник R, G, B, образованный основными цветами, принятыми в колориметрической системе RGB, а также *локус* — геометрическое место точек, определяющих на этом графике положение спектрально чистых цветов. Подчеркнем, что, используя основные цвета R, G, B, можно определить координаты любого реально существующего цвета, т. е. описать его, при этом некоторые из них могут оказаться отрицательными. Однако, смешивая их в любых пропорциях, возможно синтезировать только те цвета, которые лежат внутри цветового треугольника. Объясняется это тем, что для синтеза цветов, находящихся вне цветового треугольника, потребовалось бы смешивать не только положительные, но и отрицательные цвета, а это физически не реализуемо. По этой причине невозможно путем смешивания основных

цветов получать спектрально чистые цвета за исключением цветов, совпадающих с основными. Этот вывод особенно важен при выборе основных цветов воспроизводящего устройства, например, кинескопа или жидкокристаллической панели при их разработке, которые могут и не совпадать с основными цветами колориметрической системы.

Чем больше площадь цветового треугольника воспроизводящего устройства, т. е. треугольника, в вершинах которого лежат его основные цвета, тем больше цветов оно способно воспроизвести. Однако не все цвета, существующие в природе, могут быть воспроизведены на экранах воспроизводящих устройств.

Колориметрическая система RGB в настоящее время широко применяется для описания цвета при вводе цветных изображений в память компьютеров посредством сканеров, видеокамер, цифровых фотоаппаратов, а также при их выводе на экран монитора.

2.4. Колориметрическая система XYZ

Одновременно с колориметрической системой RGB международной комиссией по освещению в 1931 г. была разработана и рекомендована другая колориметрическая система, которая была названа колориметрической системой XYZ (CIE). Вызвано это было тем, что наряду с достоинствами колориметрической системы RGB, основным из которых является удобство ее применения при измерении цвета, она имеет два существенных недостатка, делающих ее неудобной при цветовых расчетах. Первый из них заключается в наличии отрицательных коэффициентов, второй — в необходимости определять яркость цвета путем взвешенного суммирования трех координат цвета.

При выборе основных цветов колориметрической системы XYZ было потребовано, чтобы:

- 🗖 координаты цвета для любого излучения всегда оставались положительными;
- 🗖 яркость излучения определялась одной координатой цвета;
- □ точка, соответствующая опорному белому цвету типа *E* (белому равноэнергетическому излучению), находилась в центре тяжести цветового треугольника.

В результате удовлетворения этим условиям была разработана система, в которой основные цвета *X*, *Y*, *Z* оказались фиктивными, т. е. не существующими в природе, а представляющими собой некоторые математические обозначения. Цвет в этой колориметрической системе определяется в соответствии с выражением

$$C' = x'X + y'Y + z'Z,$$

где x', y', z' — координаты цвета. На рис. 2.5 представлен график, на котором показаны координатные оси x', y', z' и вектор C', определяющий положение цвета в цветовом пространстве разработанной колориметрической системы, а также плоскость единичных цветов, выделенная штриховкой. Яркость цвета в этой колориметрической системе определяется одной координатой y'.



Рис. 2.5. Представление цвета в колориметрической системе ХҮZ

Переход от одной колориметрической системы к другой осуществляется посредством линейных преобразований координат цвета

$$\begin{aligned} x' &= 0,49000r' + 0,31000g' + 0,20000b', \\ y' &= 0,17700r' + 0,81238g' + 0,01062b', \\ z' &= 0,00002r' + 0,01002g' + 0,98996b', \\ r' &= 2,36467x' - 0,89658y' - 0,46809z', \\ g' &= -0,51523x' + 1,42647y' + 0,08877z', \\ b' &= 0,00518x' - 0,01442y' + 1,00925z'. \end{aligned}$$

Для колориметрической системы XYZ, как и для колориметрической системы RGB, были определены удельные координаты $\overline{x(\lambda)}$, $\overline{y(\lambda)}$, $\overline{z(\lambda)}$, графики зависимостей которых от длины волны λ для поля зрения 2° приведены на рис. 2.6. Особенность определения удельных координат в этом случае заключалась в том, что в отличие от колориметрической системы RGB они определялись не опытным путем, это было бы невозможно сделать, поскольку основные цвета в этой системы RGB в колориметрической систему XYZ.

Обратим внимание, что в отличие от удельных координат для колориметрической системы RGB, показанных на рис. 2.3, в колориметрической системе XYZ удельные координаты положительны для всех длин волн монохроматического излучения.

Удельные координаты используются для вычисления координат цвета x', y', z' на основании известной спектральной интенсивности излучения $C(\lambda)$





Рис. 2.6. График удельных координат колориметрической системы ХҮZ

Как уже отмечалось, часто при описании цвета его яркость не представляет интереса, и тогда от координат цвета переходят к координатам цветности (координатам единичных цветов) посредством преобразований

$$x = \frac{x'}{m}, \quad y = \frac{y'}{m}, \quad z = \frac{z'}{m},$$

где т — цветовой модуль,

$$m = x' + y' + z'.$$

Координаты цветности полностью определяют цветовой тон и насыщенность цвета. Очевидное соотношение x + y + z = 1 определяет плоскость единичных цветов, показанную на рис. 2.5 штриховкой, а координаты единичных цветов — точку прокола этой плоскости вектором цвета. Поскольку z = 1 - x - y, то для полного описания цветности достаточно использовать всего две переменных: x и y, что удобно при представлении результатов цветовых расчетов на двумерном графике. Переход к такому графику достигается путем проекции плоскости единичных цветов на плоскость x'0y', при этом равносторонний треугольник единичных цветов, показанный на рис. 2.5 штриховкой, преобразуется в прямо-угольный треугольник. На рис. 2.7 приведен график цветности для колориметрической системы XYZ.

На этом графике показано местоположение цветов монохроматических излучений (локус), которое легко вычисляется по формулам

$$x(\lambda) = \frac{\overline{x(\lambda)}}{\overline{x(\lambda)} + \overline{y(\lambda)} + \overline{z(\lambda)}}, \quad y(\lambda) = \frac{\overline{y(\lambda)}}{\overline{x(\lambda)} + \overline{y(\lambda)} + \overline{z(\lambda)}},$$



Рис. 2.7. График цветности колориметрической системы ХҮХ

а также отмечены точки и указаны соответствующие длины волн в микрометрах для некоторых из них. Все реально существующие цвета, воспринимаемые нашим зрением, расположены на части графика, ограниченной локусом и прямой, соединяющей его крайние точки. В центре тяжести цветового треугольника расположен белый цвет типа E. Область графика, ограниченная треугольником, в вершинах которого расположены концы отрезка прямой, замыкающей локус, и точка белого цвета типа E, соответствует пурпурным цветам. На этом же рисунке показан цветовой треугольник колориметрической системы RGB, в вершинах которого расположены основные цвета колориметрической системы — R, G, B.

2.5. Цветовые расчеты в колориметрической системе XYZ

Колориметрическая система XYZ была специально разработана как система, в которой удобно производить цветовые расчеты, поэтому она часто используется как система-посредник между другими колориметрическими системами. Проиллюстрируем это на примере решения нескольких конкретных задач.
Определение цвета смеси двух цветов, заданных координатами цвета

Координаты цвета C_{Σ} , получаемого в результате смешения двух цветов C_1 и C_2 , определяются путем суммирования соответствующих координат цвета смешиваемых компонентов x'_1 , y'_1 , z'_1 и x'_2 , y'_2 , z'_2 , т. е.

$$x'_{\Sigma} = x'_1 + x'_2, \ y'_{\Sigma} = y'_1 + y'_2, \ z'_{\Sigma} = z'_1 + z'_2,$$

при этом цветовой модуль результирующего цвета будет равен сумме цветовых модулей исходных цветов

$$m_{\Sigma} = m_1 + m_2$$

а координаты цветности смеси определятся по формулам

$$x_{\Sigma} = \frac{m_1 x_1 + x_2 m_2}{m_1 + m_2}, \ y_{\Sigma} = \frac{m_1 y_1 + y_2 m_2}{m_1 + m_2}$$

или после несложных преобразований

$$x_{\Sigma} = \frac{x_1 + x_2 m_2 / m_1}{1 + m_2 / m_1}, \quad y_{\Sigma} = \frac{y_1 + y_2 m_2 / m_1}{1 + m_2 / m_1}.$$
 (2.7)

Формулы (2.7) представляют собой уравнение прямой, заданное в параметрической форме, где параметром является отношение цветовых модулей смешиваемых цветов m_2/m_1 . Из этого следует, что на графике цветностей точка, характеризующая цветность полученной смеси, лежит на отрезке прямой, соединяющем две точки цветностей смешиваемых цветов C_1 и C_2 . Конкретное положение точки определяется отношением модулей смешиваемых цветов. Чем больше значение m_2/m_1 , тем ближе цветность смеси будет к цветности цвета C_2 .

Определение цвета смеси трех цветов, заданных цветовыми коэффициентами

Рассуждая аналогичным образом, легко видеть, что координаты цвета C_{Σ} , полученного в результате смешения трех цветов C_1 , C_2 , C_3 , определятся как

$$x'_{\Sigma} = x'_1 + x'_2 + x'_3, \ y'_{\Sigma} = y'_1 + y'_2 + y'_3, \ z'_{\Sigma} = z'_1 + z'_2 + z'_3,$$

при этом координаты цветности будут равны

$$x_{\Sigma} = \frac{m_1 x_1 + x_2 m_2 + x_3 m_3}{m_1 + m_2 + m_3}, \quad y_{\Sigma} = \frac{m_1 y_1 + y_2 m_2 + y_3 m_3}{m_1 + m_2 + m_3}, \quad (2.8)$$

где x'_1 , y'_1 , z'_1 , x'_2 , y'_2 , z'_2 , x'_3 , y'_3 , z'_3 , m_1 , m_2 , m_3 — координаты цвета и цветовые модули соответственно первого, второго и третьего смешиваемых компонентов. Точка, характеризующая цветность смеси на цветовом графике, будет расположена внутри треугольника, в вершинах которого лежат точки, определяющие цветности смешиваемых цветов. Поскольку локус (кривая спектрально чистых цветов) имеет выпуклую форму, то невозможно выбрать три цвета, смешивая которые можно было бы получать любые существующие в природе цвета, поскольку площадь треугольника, вписанного в фигуру, образуемую локусом и замыкающей его прямой, меньше площади этой фигуры. Это ограничение носит принципиальный характер, и его надо иметь в виду при выборе основных цветов устройств, воспроизводящих изображения.

Определение цвета смеси трех цветов, заданных координатами цветностей и величинами яркостей

Поскольку нам известны яркости трех смешиваемых компонентов L_1 , L_2 , L_3 и их координаты цветностей, то вначале определим их цветовые модули, которые будут равны $m_1 = L_1/y_1$, $m_2 = L_2/y_2$, $m_3 = L_3/y_3$. После того как найдены значения цветовых модулей, по формулам (2.8) находятся координаты цветности смеси.

Определение доминирующей длины волны и чистоты цвета

Координаты цветности не связаны непосредственно с субъективными параметрами, характеризующими цветность, т. е. цветовым тоном и насыщенностью. Для характеристики цветового тона и насыщенности иногда используют объективные параметры: доминирующую длину волны и чистоту цвета. В основе определения этих параметров лежит представление о любом цвете как цвете, полученном в результате смешения в соответствующей пропорции двух цветов: спектрально чистого и белого типа Е. В этом случае цвет, для которого определяются доминирующая длина волны λ_{π} и чистота цвета *p*, будет находиться на отрезке прямой, соединяющей на графике цветности точку белого цвета типа Е и точку на кривой спектральных цветов, соответствующую длине волны λ_{π} . Другими словами, для определения λ_л необходимо на график цветности нанести точку цвета C₀, для которого производятся измерения, затем через эту точку и точку белого типа Е провести прямую до ее пересечения с кривой спектрально чистых цветов и в месте пересечения прочитать λ_n . Рисунок 2.8 поясняет изложенное. Если цвет C_n , для которого определяется цветовой тон, относится к пурпурным цветам, доминирующая длина волны не может быть для него определена, поскольку она не существует. В этом случае вместо доминирующей длины волны определяют так называемую дополнительную длину волны λ_c , которая находится на пересечении отрезка прямой, проведенной из точки С_n через точку Е до ее пересечения с линией спектральных цветов.



Рис. 2.8. К определению доминирующей длины волны

Чистота цвета p по определению равна отношению яркости спектрально чистого цвета L_{λ} к яркости измеряемого цвета L, который может рассматриваться как результат смешения спектрально чистого цвета и опорного белого цвета типа E, а, следовательно, его яркость равна сумме яркостей образующих его цветов L_{λ} и L_F , поэтому

$$p = \frac{L_{\lambda}}{L_{\lambda} + L_E}.$$

Исходя из этого выражения, нетрудно получить следующие формулы для расчета чистоты цвета:

$$p = \frac{y_{\lambda}}{y_C} \frac{y_C - y_E}{y_{\lambda} - y_E},$$
(2.9)

$$p = \frac{y_{\lambda}}{y_C} \frac{x_C - x_E}{x_{\lambda} - x_E},$$
(2.10)

где x_C , y_C , x_E , y_E , x_λ , y_λ — координаты цветности измеряемого цвета C, опорного белого цвета типа E и спектрально чистого цвета, соответствующего доминирующей длине волны. Какую из этих формул использовать для расчета, зависит от конкретных значений координат цветности. Если знаменатель в формуле (2.9) принимает малое значение, то расчет чистоты цвета следует вести по формуле (2.10), если же мал знаменатель в формуле (2.10), для расчета следует использовать формулу (2.9). Из написанных формул следует, что для спектрально чистых цветов p = 1, а для опорного белого p = 0.

В заключение подчеркнем, что колориметрическая система XYZ часто используется для пересчета данных из одной колориметрической системы в другую. Такая необходимость возникает, в частности, когда нужно согласовать между собой работу ряда устройств, используемых в настольных издательских системах, например, работу сканера, монитора и печатного станка. Обычно такой пересчет выполняется графическим редактором каждый раз при переходе из одного цветового пространства, например RGB, в другое цветовое пространство, например СМҮК.

2.6. Цветовая температура

В основе всех колориметрических измерений и расчетов, как это следует из приведенного ранее материала главы, лежит понятие опорного белого цвета. Именно относительно опорного белого цвета определены единицы основных цветов в любой колориметрической системе. Наиболее часто в качестве опорного белого используют равноэнергетический белый цвет типа E, однако его использование при измерениях неудобно. Дело в том, что в природе отсутствуют источники, спектр излучения которых в световом диапазоне не зависел бы от длины волны, хотя излучение с таким спектром и можно получить искусственно. Это привело к тому, что в качестве опорного белого цвета стали применять источники, спектральный состав излучения которых близок к спектральному составу излучения абсолютно черного тела, нагретого до температуры Т_п, называемой цветовой температурой. При этом цветность этих источников стали характеризовать величиной цветовой температуры в градусах Кельвина, что оказалось удобным, т. к. для характеристики их цветности вместо двух чисел (например, x = 0,310, y = 0,316) достаточно было назвать только одно ($T_{\rm H} = 6674$ K). На графике (см. рис. 2.7) приведена кривая цветовой температуры, которая показывает, как изменяется цветность излучения абсолютно черного тела при изменении его температуры, спектральный состав которого определяется по формуле Планка.

В тех случаях, когда точка цветности расположена не на кривой цветовой температуры, но близко от нее, используют понятие коррелированной цветовой температуры [67].

В 1931 г. МКО рекомендовала использовать при измерениях один из трех стандартных источников белого цвета, которые были обозначены A, B и C соответственно. В дальнейшем к ним были добавлены источники D_{55} , D_{65} , D_{75} и 9300 К + 27 МРСД, а затем еще ряд источников. В табл. 2.1 приведены основные характеристики излучений некоторых источников. В качестве источника A применяется газонаполненная лампа накаливания с вольфрамовой нитью накала, цветовая температура которой составляет $T_{\rm u}$ = 2856 K, остальные источники излучения, приведенные в таблице, получаются путем добавления к источнику A цветофильтров.

Излучение	x	у	λ_{d} , мкм	p, %	Цветовая температура <i>Т</i> _ц , К
А (стандартизованный свет лампы накаливания)	0,4476	0,4075	0,583	65	2856
В (стандартизованный прямой солнечный свет)	0,3484	0,3516	0,574	15	4874
С (стандартизованный усредненный дневной свет)	0,310	0,316	0,482	5	6774
D ₅₅					5500
D ₆₅	0,313	0,329			6504
D ₇₅	0,3003	0,3103			7500
9300 К + 27 МРСД	0,281	0,34			
Е (свет равноэнергетического источника)	0,333	0,333		0	5700

Таблица 2.1. Характеристики излучений стандартных источников белого цвета

2.7. Другие колориметрические системы

Наряду с колориметрическими системами RGB и XYZ в настоящее время разработаны и применяются другие колориметрические системы, которые при решении некоторых задач оказываются более удобными. Недостатком колориметрических систем RGB и XYZ является то, что в этих системах одинаковым расстояниям на графиках цветности не соответствуют одинаковые визуальные различия. Рисунок 2.9, на котором показаны эллипсы равнозаметных различий цветности, поясняет это. Для большей наглядности эллипсы на рисунке изображены с десятикратным увеличением. Из рисунка следует, что в области синих цветов размер эллипсов много меньше, чем в области зеленых цветов, в результате чего в области синего число различимых цветностей, приходящихся на единицу площади диаграммы, оказывается в 400 раз большим, чем в области зеленого. Стремление избавиться от этого недостатка привело к разработке ряда так называемых равноконтрастных систем, некоторые из них рассматриваются далее.

Колориметрическая система Манселла

В основу этой системы положены три характеристики цвета, обозначаемые терминами: hue — цветовой тон, chroma — насыщенность и value — светлота. Система реализована в виде атласа образцов с приближенно равными ступенями изменения цвета, охватывающих подавляющую часть области возможных цветов. Эти образцы были сгруппированы в пространственное цветовое тело, причем ахроматические цвета (от черного до белого) располагались по вертикальной оси, цветовые тона — по кругу, а насыщенность — по радиусам.



Рис. 2.9. Эллипсы равнозаметных различий цветности

При определении цвета заданного образца его цвет сравнивался с образцами цветов, помещенными в атласе (выкрасками атласа) и в атласе находился такой образец, от которого заданный образец меньше всего отличался по цвету. Результат измерения выражался в цифровых индексах по шкалам тела цветового пространства Манселла [1].

Система Манселла в настоящее время находит применение при определении цвета различных материалов. Серьезным недостатком этой системы является сложность пересчета результатов измерений, полученных в ней, в другие колориметрические системы. Похожая система была разработана Оствальдом.

Колориметрическая система Мак-Адама

Колориметрическая система Мак-Адама является равноконтрастной системой, в которой одинаковым расстояниям на графике цветностей соответствуют примерно одинаковые визуальные различия, что делает ее удобной для цветовых расчетов. Эта система получена из колориметрической системы XYZ путем нелинейного преобразования координат цветностей. Новые равноконтрастные координаты цветности *u*, *v*, определяющие цветность излучения, находятся следующим образом:

$$u = 4x/(-2x + 12y + 3),$$

$$v = 6y/(-2x + 12y + 3).$$



Рис. 2.10. Равноконтрастный график цветности

На рис. 2.10 приведен равноконтрастный график цветности.

Колориметрическая система Мак-Адама была рекомендована МКО в 1960 г. для временного использования.

Колориметрическая система L*a*b*

Эта система также является равноконтрастной и получается из колориметрической системы XYZ путем преобразования ее цветовых координат по формулам

$$L^{*} = 116 (y'/y'_{n})^{\frac{1}{3}} - 16,$$

$$a^{*} = 500 (x'/x'_{n})^{\frac{1}{3}} - (y'/y'_{n})^{\frac{1}{3}},$$

$$b^{*} = 200 (y'/y'_{n})^{\frac{1}{3}} - (z'/z'_{n})^{\frac{1}{3}},$$
(2.11)

где x'_n , y'_n , z'_n — цветовые координаты опорного белого цвета в системе XYZ. Координата L^* в этой системе представляет визуальную яркость (светлоту) цвета, a^* — соотношение красного и зеленого цветов, b^* — соотношение синего и желтого цветов. В этой колориметрической системе полное цветовое различие между двумя цветами вычисляется по формуле

$$\Delta E_{ab}' = \sqrt{\left(\Delta L^*\right)^2 + \left(\Delta a^*\right)^2 + \left(\Delta b^*\right)^2}, \qquad (2.12)$$

где ΔL^* , Δa^* , Δb^* — расстояния между сравниваемыми цветами, вычисленное для каждой из цветовых координат. Формулы (2.11) и (2.12) применимы при условии, что $x'/x'_n \ge 0,01$, $y'/y'_n \ge 0,01$, $z'/z'_n \ge 0,01$.

Колориметрическая система L*a*b* была рекомендована МКО в 1976 г. Многие колориметры, выпускаемые промышленностью, измеряют значения цветовых координат в этой системе. Эта колориметрическая система используется в основных графических редакторах, таких как, например Photoshop, Corel PHOTO-PAINT, CorelDRAW. При использовании этой системы в графических редакторах значения координат цвета приводятся в условных единицах, при этом в различных редакторах они могут несколько различаться между собой. Так в графическом редакторе Corel PHOTO-PAINT $0 \le L^* \le 100$, $-60 \le a^* \le 60$, $-60 \le b^* \le 60$, а в графическом редакторе Photoshop $0 \le L^* \le 100$, $-128 \le a^* \le 127$, $-128 \le b^* \le 127$. Для белого цвета $a^* = b^* = 0$.

Колориметрические системы HLS и HSB

Названия этих систем, наиболее известных дизайнерам и художникам, представляют собой сокращения от слов: hue, lightness, saturation — цветовой тон, светлота, насыщенность, в первом случае, и hue, saturation, brightness — цветовой тон, насыщенность, яркость, во втором случае. Значение цветового тона в этих системах задается в градусах, как в колориметрической системе Манселла, а насыщенность и светлота (яркость в случае HSB) — в процентах. Обе колориметрические системы, имея много общего, различаются в деталях. Так в системе HSB все три координаты цвета являются взаимно независимыми, что следует отнести к ее достоинству, в то время как в системе HSL координата L зависит от насыщенности S. Взаимная независимость координат цвета создает определенные удобства при обработке изображений, т. к. позволяет по отдельности корректировать яркостные и цветовые искажения. Подробнее о методах независимой коррекции яркостных и цветовых искажений см. в [96].

Колориметрические системы СМУ и СМУК

Эти колориметрические системы предназначены для цветовых расчетов при работе с устройствами, выводящими изображения на бумагу. К таким устройствам относятся струйные и лазерные принтеры и печатные станки. Название колориметрической системы СМҮ образовано из начальных букв названий цветов, принятых в качестве основных в этой системе: Cyan — голубой, Magenta — пурпурный, Yellow — желтый. Эти цвета являются дополнительными к основным цветам, используемым в колориметрической системе RGB: красному, зеленому и синему. В колориметрической системе СМҮК дополнительно введена еще одна координата цвета К — ключевой черный цвет. Необходимость введения дополнительной координаты (избыточной, с точки зрения описания цвета) обусловлена стремлением расширить диапазон воспроизводимых при печати цветов. Кроме того, это позволило улучшить качество воспроизведения черных и серых тонов. Эти колориметрические системы также представлены в современных графических редакторах, предназначенных для подготовки к печати цветных изображений.



Рис. 2.11. Компоненты цветного изображения в колориметрической системе RGB

В заключение заметим, что мы рассмотрели далеко не все применяемые в настоящее время колориметрические системы, а лишь те из них, которые, с нашей точки зрения, являются наиболее важными при цифровой обработке изображений. Так за рамками приведенного описания остались колориметрические системы: U*V*W*, S0W*, L*U*V*, NTSC, Карунена — Лоэва, колориметрическая система колбочек сетчатки и некоторые другие, сведения о которых можно найти в [1, 84].

Несмотря на взаимные различия, все известные колориметрические системы обнаруживают одно общее для них свойство, а именно наличие сильной корреляционной связи между контурами на различных цветовых компонентах изображения. Рисунок 2.11 иллюстрирует сказанное. Объясняется это тем, что цвет сравнительно слабо изменяется в пределах областей, ограниченных контурами, которые являются проекциями оболочек объектов, и сильно изменяется за их пределами, т. е. при переходе с одного объекта на другой.

2.8. Вывод на экран и печать цветных изображений

При наблюдении реальных сцен на сетчатки глаз проецируются изображения, которые могут включать в себя все существующие в природе цвета. Обусловлено это тем, что спектральный состав излучений, формирующих эти изображения, ничем не ограничен. Совсем иначе обстоит дело в случае наблюдения цветных изображений: фотографий, печатных репродукций или изображений на экране телевизора. Во всех этих случаях цвет синтезируется из ограниченного числа компонентов, например трех, как это имеет место в случае цветных фотографий или цветных изображений на экране монитора компьютера. Ограниченное количество компонентов, используемых при воспроизведении таких изображений, позволяет воспроизводить на них лишь ограниченный цветовой диапазон, определяемый используемыми компонентами. Рассмотрим эту проблему более детально.

Как известно, при воспроизведении изображений используют два метода получения цвета: субтрактивный и аддитивный. *Субтрактивный метод* образования цвета заключается в том, что белый цвет пропускается через окрашенную среду (цветофильтр), в результате часть его спектрального состава поглощается. Соответственно этому яркость образованного цвета будет меньше яркости исходного белого цвета.

Существо *аддитивного метода* образования цвета состоит в том, что окрашенные световые потоки смешиваются (суммируются), образуя новый цвет. В отличие от субтрактивного метода при аддитивном методе яркость результирующего цвета больше яркости смешиваемых цветов и равна их сумме.

Субтрактивный метод применяется в цветной фотографии как при изготовлении фотокарточек, так и при изготовлении слайдов, а также в цветном кино. Цветной слайд можно рассматривать как множество элементарных цветофильтров, образующих мозаичное цветное изображение, нечто вроде витража. Элементарные цветофильтры состоят из трех образующих их слоев: голубого, пурпурного и желтого, каждый из которых поглощает свет приблизительно в 1/3 части спектра отличной от областей поглощения двух остальных слоев. Значения коэффициентов поглощения окрашенных слоев определяют цвет такого элементарного цветофильтра. Цветовой охват, т. е. область воспроизводимых цветов в цветной фотографии в первом приближении можно оценить с помощью цветового треугольника, в вершинах которого расположены основные цвета фотоматериала. В рассматриваемом случае все цвета, которые лежат внутри треугольника, воспроизводятся, а цвета, которые оказались вне треугольника, воспроизведены быть не могут. Цветами, располагающимися в вершинах цветового треугольника, при используемых в настоящее время красителях (голубом, пурпурном и желтом) будут красный, зеленый и синий. Более точный анализ показывает, что в рассматриваемом случае область воспроизводимых цветов определяется не прямолинейным, а криволинейным треугольником, показанным на рис. 2.12. Объясняется это тем, что при субтрактивном методе образования цвета, в отличие от аддитивного, результирующий цвет в общем случае располагается не на прямолинейном отрезке, соединяющем точки исходных цветов, а на криволинейном отрезке. Подробную информацию об этом можно найти в [30].

При выводе изображений на экран кинескопа или жидкокристаллическую панель, а также при их печати используется аддитивный метод образования цвета.

В случае вывода изображения на экран, например, кинескопа, цвет в изображении образуется путем смешения красного, зеленого и синего цветов, создаваемых свечением трех типов пятен люминофоров его мозаичного экрана под воздействием трех сканирующих электронных лучей [93]. Благодаря тому, что расстояния между светящимися пятнами люминофоров составляют десятые доли миллиметра, а сами пятна имеют примерно такой же размер, для зрителя они сливаются, что приводит

к пространственному смешению их цветов. Область воспроизводимых цветов в рассматриваемом случае определяется цветовым треугольником, в вершинах которого расположены цвета, определяемые спектрами используемых люминофоров. На рис. 2.12 показан цветовой треугольник, образуемый цветами люминофоров, принятыми для системы цветного телевидения *NTSC*. Из рисунка видно, что значительная часть зеленых цветов не может быть воспроизведена на экране кинескопа, однако это практически не сказывается на визуальном качестве изображений, поскольку в этой области цветностей цветоразличимость зрения понижена (см. рис. 2.9).



Рис. 2.12. Цветовой график: *1* — цветовой охват фотоматериала; 2 — цветовой охват кинескопа NTSC

В целях удобства выполнения цветовых расчетов, связанных с выводом изображений на экран кинескопа, иногда создают его колориметрическую систему, где в качестве основных цветов используют цвета свечения люминофоров. Так была создана колориметрическая система NTSC, в которой в качестве опорного используется белый цвет типа *C*, координаты цветности: красный $r_N = 0.67$, y = 0.33, зеленый $g_N = 0.21$, y = 0.71, синий $b_N = 0.14$, y = 0.08, а яркость определяется по формуле

$$L = 0, 3r_N + 0, 59g_N + 0, 11b_N$$
.

Аналогичным образом обстоит дело при использовании жидкокристаллических и плазменных панелей.

При выводе изображений на печать используется метод растрирования, который рассмотрен в разд. 4.10. В основе этого метода лежит разложение изображения в растр, каждая ячейка которого представляет собой растровый элемент (пиксел). Обычно необходимый цвет при печати изображений получается путем смешения четырех цветов: голубого, пурпурного, желтого и черного. Добавление черного цвета к трем другим обусловлено тем, что благодаря наличию примесей в красителях, применяемых при печати, получить серые и черный тона путем их смешения не удается, вместо этого получаются грязно-коричневые тона. Насыщенность каждого из смешиваемых цветов на растровом элементе определяется размером точки, состоящей из пятен, и количеством добавленного черного цвета. Чем больше размер точки, т. е. чем большим количеством пятен она образована, и чем меньше добавлено черного, тем насыщеннее данный цвет. И наоборот, если размер точки, образованной данным красителем, нулевой, что соответствует случаю, когда в данном растровом элементе бумага не покрыта данным красителем или покрыта точкой, созданной черным красителем, насыщенность данного цвета будет равна нулю. Аналогичным образом обстоит дело и в более сложном случае, когда цвет образуется в результате смешения ряда красителей.

Растровая структура отпечатанного цветного изображения незаметная в нормальных условиях наблюдения хорошо заметна, если изображение рассматривать под увеличительным стеклом.

При печати цветных изображений на печатных машинах изготавливаются цветоделенные печатные формы, каждая из которых переносит на бумагу свой цветовой компонент изображения, представленный в виде растра. Для уменьшения заметности растровой структуры отпечатанного изображения растры голубого, пурпурного, желтого и черного поворачивают относительно горизонтальной оси на различные углы. Так, например, растр черного компонента как самого заметного поворачивают на угол 45°, при этом растр черного приобретает шахматную структуру, т. е. наиболее близкую к оптимальной при пространственной дискретизации изображений (*см. разд. 4.5*). Рекомендуемыми углами поворота остальных растров являются: для голубого — 105°, для пурпурного — 75°, для желтого — 90° или 0°. При этом возможны как взаимные перекрытия растров, так и отсутствие этих перекрытий. В силу описанных особенностей воспроизведения цвета при четырехцветной печати область воспроизводимых цветов на бумаге оказывается существенно меньше области воспроизводимых цветов в цветной фотографии и на экране монитора.

Расчеты при преобразовании цвета из колориметрической системы RGB в колориметрическую систему CMYK достаточно сложны.

Для получения цветоделенных изображений (компонентов), из которых формируются конечные цветные изображения, используются специальные программы цветоделения, некоторые из них разработаны под графические редакторы, например, под Photoshop, Corel PHOTO-PAINT и др. При этом возможны следующие варианты преобразования.

□ *Перцепционный*. Обеспечивает сохранение соотношения цветовых значений исходного изображения, хотя сами цвета при этом могут измениться.

- □ *С сохранением насыщенности*. Обеспечивает сохранение соотношения насыщенности исходного изображения, хотя сами цвета при этом могут измениться.
- Относительный колориметрический. Обеспечивает сохранение цветов исходного изображения, которые входят в область воспроизводимых цветов печатающего устройства. Те цвета, которые оказываются вне этой области, заменяются другими с теми же значениями яркости.

□ Абсолютный колориметрический. Обеспечивает отключение белой точки.

Четырехцветная печать в состоянии хорошо воспроизводить красные цвета, но не может воспроизвести ярких синих, зеленых, розовых, фиолетовых и коричневых цветов.

С целью устранения этого недостатка была разработана новая технология печати, названная *HiFiColor*. Имеется ряд вариантов ее реализаций, но общей чертой для всех них является расширение числа используемых для печати цветов. Так, например, в системе HiFiColor 300 для получения более ярких красных, зеленых и синих цветов использовано семь различных красителей. В системе Hexachrome к цветам СМҮК добавлены еще два настраиваемых цвета: оранжевый и зеленый. В системе, использованной фирмой Hewlett Packard для печати фотографий на струйных принтерах, применяется шесть цветов и т. д. [2].

2.9. Управление цветом, калибровка, основы систем CMS

Развитие компьютерных технологий привело к разработке и внедрению настольных издательских систем, в которых большое место занимает компьютерная графика. Использование графических редакторов при подготовке изображений к печати расширяет возможности полиграфии, повышая качество напечатанных репродукций, особенно цветных, и сокращая требующееся для этого время. При этом в одну технологическую цепочку оказываются включенными:

- 🗖 устройства ввода изображений: сканеры, видеокамеры, цифровые фотоаппараты;
- мониторы, которые используются для контроля в процессе компьютерной обработки изображений;
- □ устройства вывода изображений на печать, в качестве которых могут быть использованы струйные и лазерные принтеры, а также печатные станки различных типов.

В силу ряда причин все эти устройства работают в различных колориметрических системах, или, как принято говорить, в различных цветовых моделях. Другими словами, основные цвета этих устройств, которыми определяются цветовые пространства воспроизводимых цветов в изображениях, не совпадают. Так, например, основные цвета кинескопов, применяемых в мониторах компьютеров, определяются типом используемых люминофоров, причем в различных типах мониторов используются люминофоры, отличающиеся между собой спектральными характеристиками излучения. Такая же ситуация имеет место при использовании жидкокристал-

лических панелей. Аналогичным образом обстоит дело и в полиграфии, где цветовые пространства воспроизводимых цветов могут варьироваться в зависимости от используемых полиграфических красок и сортов бумаги, на которой производится печать. Кроме того, характеристики (цветовые профили) каждого из устройств, входящих в эту цепочку, с течением времени изменяются. Если все это не учитывать, то нельзя рассчитывать на то, что отпечатанное изображение будет правильно передавать цвета исходного изображения. Отсюда возникает необходимость управления цветом на всех этапах подготовки изображения к печати. Существо проблемы управления цветом заключается во взаимном согласовании и поддержании стабильными цветовых профилей каждого из звеньев системы.

Первым и главным компонентом технологии управления цветом является калибровка всех устройств, используемых при подготовке изображений к печати. Под *калибровкой* понимают процесс настройки каждого устройства системы так, чтобы устройства передавали цвет в соответствии со спецификациями изготовителя.

Рассмотрим процесс калибровки вначале на примере системы, которая включает в себя монитор, использующий электронно-лучевую трубку. Калибровку рекомендуется начинать с монитора, поскольку монитор является наименее стабильным устройством системы, и его калибровку следует делать не реже одного раза в месяц. Перед началом калибровки необходимо дать ему поработать в течение получаса с тем, чтобы его характеристики стабилизировались. Используют как аппаратный, так и ручной способы калибровки. Устройства аппаратной калибровки обычно состоят из датчика — присоски, который укрепляется на экране, и программы калибровки. Посредством этих устройств настраивают "белое" монитора на заданную цветовую температуру, что приводит к автоматической настройке всех других цветов, воспроизводимых монитором. Кроме того, посредством этих устройств можно настроить цветовой баланс и передаточную функцию (световую характеристику). Правильная настройка позволяет обеспечить максимальную нейтральность серых полутонов. Устройства аппаратной калибровки позволяют также сохранять ее результаты в виде ІСС-профиля, который записывается на диск. Сохранение профиля дает возможность использовать его для настройки других мониторов.

При ручном способе калибровки используют специальный программный модуль, который обычно прилагается к графическому редактору — в случае Adobe Photoshop это Adobe Gamma.

Ручную калибровку следует начинать со стандартизации цветовой среды, в которой осуществляется работа. Прежде всего, следует позаботиться о том, чтобы были нейтральными освещение и цвет стен. Яркие цвета рядом с изображениями будут искажать восприятие цвета. Начинать собственно калибровку нужно с установок яркости и контраста монитора по тестовому изображению, которое выводит на его экран калибровочный программный модуль. Установка эта делается с помощью кнопок управления монитора. Далее необходимо в списке, который предлагает калибровочная программа, выбрать тип монитора, а если нужного типа в списке нет, но пользователю известны технические данные люминофоров, используемых в кинескопе монитора, ввести эти данные в диалоговом режиме. После этого необхо-

димо установить значение гамма, которым определяется передаточная функция (световая характеристика) кинескопа. Эта регулировка также осуществляется по специальному тестовому изображению, которое выводится на экран калибровочным модулем. При регулировке значения гамма предусмотрены две возможности, а именно можно подобрать его нужные значения либо для серой составляющей, т. е. сразу для трех компонентов цвета, либо для каждого компонента отдельно. И, наконец, устанавливается цветовая температура опорного белого цвета. Часто устанавливают цветовую температуру 6500 К, хотя стандарт для работы с цветными изображениями предусматривает цветовую температуру 5000 К. Результаты калибровки сохраняют в виде цветового профиля ICC.

Рассмотрим теперь особенности ручной калибровки монитора, в котором использована жидкокристаллическая панель. В этом случае, как и в случае монитора на электроннолучевой трубке, вначале следует установить уровень яркости в белом, используя тестовую таблицу и руководствуясь следующими соображениями. При регулировке этого параметра за счет изменения яркости ламп подсветки одновременно изменяются уровни яркости во всем динамическом диапазоне. Понижение уровня яркости приводит к неразличимости деталей в глубоких тенях изображения вследствие того, что их контраст становится ниже дифференциального порога зрения, определяющего способность зрительной системы различать малые контрасты (см. разд. 3.3). Далее следует установить контраст. При регулировке контраста сигнал, соответствующий уровню черного, не изменяется, сигналы же, соответствующие другим уровням, изменяются, сохраняя пропорциональность своим цифровым значениям. Уменьшение контраста влечет за собою уменьшение яркости в белом со всеми вытекающими последствиями, а чрезмерное увеличение контраста приводит к ограничению яркости в области белого, поскольку световая характеристика жидкокристаллической панели имеет S-образную форму. Следствием ограничения в белом является потеря различимости светлых деталей. И, наконец, необходимо установить цветовую температуру опорного белого цвета путем поканальной регулировки интенсивности основных (базовых) цветов RGB. Здесь так же, как и в случае мониторов на электронно-лучевых трубках, часто устанавливают цветовую температуру 6500 К, хотя стандарт для работы с цветными изображениями предусматривает цветовую температуру 5000 К. Результаты калибровки сохраняют в виде цветового профиля ІСС.

После того как калибровка монитора завершена, переходят к калибровке сканера. Барабанные сканеры, планшетные сканеры высокого класса, а также сканеры, предназначенные для ввода изображений с диапозитивов, калибруются автоматически при каждом включении. Планшетные сканеры среднего класса необходимо калибровать вручную, используя для этой цели специальные калибровочные программы и эталонные изображения, поставляемые вместе со сканерами в виде серых полутоновых или цветных шаблонов.

Завершающим этапом является калибровка устройств вывода изображений на печать, которая сводится по существу к учету свойств полиграфических красок и бумаги, а также к учету увеличения растровой точки ("растискивание") в результате впитывания и растекания краски на бумаге, которые возникают при печати. Все это должно быть учтено при преобразовании цветов из одной цветовой модели (колориметрической системы) в другую. Поэтому в процессе калибровки делают пробную печать. Изображение, используемое для пробной печати, должно содержать в себе все комбинации цветов и быть представлено в цветовой модели СМҮК. Такие изображения обычно в виде файла поставляются вместе с графическими редакторами (в случае редактора Photoshop поставляется файл CMYK Colors.tif). Выполнив пробную печать, необходимо проанализировать оттиск и на основе этого анализа внести коррективы в настройку.

Калибровка всех устройств системы вручную с последующим их согласованием занимает много времени. Для ускорения этого процесса и стандартизации были разработаны системы управления цветом CMS (Color Management System), которые устанавливают правила обмена цветом между устройствами. Способность системы управления цветом описывать и записывать характеристики цвета любого устройства позволяет обеспечить согласованность цвета.

2.10. Баланс белого

Как известно, фотографии, снятые посредством фотокамеры на обычную фотопленку в комнате с включенным светом и без фотовспышки, имеют искаженную цветопередачу. В этих фотографиях преобладают желто-оранжево-красные тона. Происходит это потому, что цветовая температура ламп накаливания (около 3000 K) значительно ниже цветовой температуры солнечного излучения (около 5500 K). Другими словами, спектральный состав света, излучаемый лампами накаливания, имеет значительную интенсивность в области красно-желтых лучей. Зрительная система человека адаптируется к условиям освещения и практически не замечает этого, однако фотопленка, сбалансированная под цветовую температуру 5500 K, чувствительна к подобным отклонениям в спектральном составе освещения. Для того чтобы избежать подобных цветовых искажений, в цифровых фотоаппаратах и видеокамерах выполняется так называемый *баланс белого*.

Идеальный способ установки баланса белого заключается в предварительном определении спектрального состава освещения фотографируемой сцены и спектральной чувствительности используемой камеры, на основании которых рассчитываются параметры обработки отснятых изображений, обеспечивающей компенсацию первоначальных цветовых искажений. В результате будет достигнута правильная цветопередача независимо от спектрального состава освещения. Однако это не всегда возможно, поэтому применяют следующие способы установки баланса белого:

- □ автоматический способ;
- способ с использованием ручной предустановки, при котором учитывается вид источника освещения, например, солнце, тень, лампы накаливания, люминесцентные лампы различных типов и т. д.;
- ручная установка.

Рассмотрим кратко эти способы.

Автоматический способ установки баланса белого

Он реализуется посредством ряда методов.

При реализации первого метода исходят из предположения, что в цветном изображении отснятой сцены в среднем присутствуют в равных количествах компоненты трех основных цветов: красного, зеленого и синего (R, G, B), а также полагают, что спектральная чувствительность используемой камеры не вносит цветовых искажений. Существо метода заключается в том, что вначале рассчитываются средние значения интенсивностей $\overline{R}, \overline{G}, \overline{B}$ по всем трем каналам

$$\overline{R} = \frac{1}{KN} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} R(k,n), \ \overline{G} = \frac{1}{KN} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} G(k,n), \ \overline{B} = \frac{1}{KN} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} B(k,n),$$

где K и N — количество строк и столбцов (количество пикселов в строке) в изображении, k и n — номера строки и столбца, определяющие положение пиксела в изображении. Заключительным этапом обработки является масштабирование интенсивностей цветовых компонентов каждого пиксела в соответствии с формулами

$$R_{\text{pe3}} = R_n \frac{C}{\overline{R}}, \ G_{\text{pe3}} = G_n \frac{C}{\overline{G}}, \ B_{\text{pe3}} = B_n \frac{C}{\overline{B}},$$

где R_n , G_n , B_n — корректируемые значения интенсивностей пикселов, R_{pes} , G_{pes} , B_{pes} — результирующие (скорректированные) значения интенсивностей пикселов, C — постоянный для всех трех каналов коэффициент. Значения интенсивностей пикселов обычно выражаются соответствующими им номерами уровней квантования, при этом интервал, в котором они лежат, как правило (но не обязательно) составляет 0...255.

При реализации другого метода исходят из предположения, что наиболее яркие области в изображении являются бликами на поверхностях изображаемых объектов, т. е. представляют собой результат зеркального отражения света источника освещения, а, следовательно, их спектр совпадает со спектром источника освещения. Исходя из этого предположения и полагая, что спектральная чувствительность камеры не является источником цветовых искажений, поступают следующим образом. Вначале находят максимумы интенсивностей в каждом из цветовых каналов: $R_{\text{макс}}$, $G_{\text{макс}}$, $B_{\text{макс}}$, после чего масштабируют интенсивности цветовых компонентов в соответствии с формулами

$$R_{\text{pe3}} = R_n \frac{255}{R_{\text{makc}}}, \ G_{\text{pe3}} = G_n \frac{255}{G_{\text{makc}}}, \ B_{\text{pe3}} = B_n \frac{255}{B_{\text{makc}}}.$$

Таким образом, при автоматическом режиме баланса белого непрерывно отслеживается изменение спектрального состава освещения и выполняется соответствующая коррекция. Недостатком этого метода является то, что при этом возможны ошибки. Например, при фотографировании лица крупным планом большая часть поля оказывается занятой изображением лица, которое имеет розовый цвет. Поскольку алгоритму, реализующему баланс белого, не известно, чем вызван этот цветовой тон, цветовым ли тоном фотографируемого объекта или же цветовым тоном освещения, при котором производится фотографирование, он попытается несколько скомпенсировать возможные цветовые искажения. В результате этого в изображение алгоритмом будут внесены искажения, и цвет лица окажется излишне холодным — голубоватым. Таким образом, система автоматического баланса белого, эффективная во многих случаях, сама создаст цветовые искажения.

Способ с использованием ручной предустановки

Этот способ баланса белого путем предустановок также может оказаться неэффективным, поскольку предустановки рассчитаны на коррекцию строго определенных цветовых искажений. Так, предустановка "лампы накаливания" компенсирует разницу между цветовой температурой лампы накаливания 3200 К и дневной — 5500 К. Однако цветовая температура лампы накаливания может быть как 3200 К, так и 2800 К, поскольку лампы бывают разные. Вследствие этого предустановка также может оказаться неэффективной в подобных случаях.

Ручная установка баланса белого

Наилучший результат может быть получен при ручной установке баланса белого, однако эта процедура требует выполнения ряда операций. Для ручной установки баланса белого необходимо в фотографируемой сцене выбрать объект белого цвета и по нему производить регулировку. В том случае если в снимаемой сцене такой объект отсутствует, для регулировки можно использовать лист белой бумаги. В качестве примера рассмотрим случай, когда при съемке пейзажного изображения установка белого выполняется по изображению облаков, оказавшихся в кадре. Вначале оператор по отснятому изображению находит значения интенсивностей цветовых компонентов для одного из пикселов, принадлежащих изображению облака (R_0 , G_0 , B_0). Затем, полагая цвет облаков белым, оператор назначает интенсивности цветовых компонентов этого же пиксела равными (R_1 , G_1 , B_1). После этого по приводимым далее формулам рассчитываются значения интенсивностей цветовых компонентов для всех пикселов корректируемого изображения:

$$R_{\text{pe3}} = R_n \frac{R_1}{R_0}, \ G_{\text{pe3}} = G_n \frac{G_1}{G_0}, \ B_{\text{pe3}} = B_n \frac{B_1}{B_0}.$$

2.11. Каналы, слои и изображения с прозрачным фоном

Каналы

Как уже отмечалось, изображения в памяти компьютера записываются в виде массивов чисел. В простейшем случае, когда изображение является двухградационным (штриховым) или полутоновым черно-белым, оно может быть представлено двумерным массивом чисел. Для представления цветного изображения в колориметрической системе (цветовой модели) RGB потребуются три двумерных массива для того, чтобы записать в них информацию об интенсивности красного, зеленого и синего компонентов каждого пиксела. Еще больший объем памяти необходим для записи цветного изображения, представленного в цветовой модели СМҮК, а именно: четыре двумерных массива с тем, чтобы представить данные о голубом (cyan), пурпурном (magenta), желтом (yellow) и ключевом черном (black) цветах. При разработке графических редакторов, и в частности, такого широко применяемого редактора, как Photoshop, было введено понятие каналов обработки изображения, под которыми понимается не что иное, как возможность независимого доступа к каждому из компонентов цветного изображения (например, красному, синему или зеленому в модели RGB). Такие каналы называются иветовыми и служат в случае цветного изображения для хранения данных об интенсивностях цветовых компонентов каждого пиксела. В случае черно-белых изображений имеется один канал, который служит для хранения информации о яркости. Таким образом, для представления полутонового черно-белого изображения необходим всего один цветовой канал, а для представления цветного изображения в цветовой модели СМҮК — четыре. При создании изображения в любой цветовой модели цветовые каналы создаются автоматически. Любые изменения, вносимые в изображения при их редактировании, фиксируются в цветовых каналах. Помимо цветовых каналов некоторые графические редакторы, например Photoshop, позволяют создавать в документе (в файле изображения, за исключением случая черно-белого штрихового изображения) дополнительные каналы, которые называются альфа-каналами. Альфа-каналы используются для хранения и редактирования масок, предназначенных для защиты определенной части изображения от воздействия рисующих инструментов и фильтров. Каждое изображение при обработке его в редакторе Photoshop может включать не более 24 каналов, при этом в каждом канале для представления данных о каждом пикселе расходуется по одному байту. Из этого следует, что чем больше каналов будет содержать файл изображения, тем большим будет его размер. Подробную информацию о приемах работы с каналами изображения, и в частности, с альфа-каналами, можно найти в [28].

Изобразительные слои изображения

Под слоями понимаются прозрачные изображения, которые, накладываясь друг на друга, подобно изображениям, нарисованным на прозрачных кальках, создают результирующее изображение, вид которого зависит от прозрачности слоев. При создании исходного изображения, например, путем сканирования фотографии, образуется всего один-единственный слой, который называется задним планом (*background*). На этот единственный слой можно наложить целый ряд дополнительных слоев, что создает большие удобства при редактировании изображений.

Слой — это обыкновенное растровое изображение, представленное в определенной цветовой модели. Обратим внимание, что задний план и накладываемые на него слои могут быть представлены в разных цветовых моделях (например, задний план

может быть представлен в цветовой модели RGB, а слой — в цветовой модели CMYK), а также могут иметь разные размеры. В отношении работы с изображением слои ничем не отличаются от заднего плана, на них также можно рисовать, их можно фильтровать и выполнять над каждым из них другие виды преобразований. Существенным является также то, что изменения, вносимые в один слой, не влияют на другие слои и на задний план. При окончании обработки (редактирования) изображения и записи его в файл все слои и задний план обычно склеиваются, что позволяет сократить размер результирующего файла. Однако если необходимо временно прервать работу и при этом сохранить слои в не склеенном виде, то это можно сделать, используя формат файла Adobe Photoshop. Кроме *изобразительных слоев* графический редактор Photoshop позволяет использовать также так называемые *корректирующие слои*. Подробнее о корректирующих слоях см. в [28].

Изображения с прозрачным фоном

В ряде случаев требуется на имеющееся фоновое изображение наложить сюжетную часть другого изображения, т. е. осуществить своего рода электронный фотомонтаж. Достигается это путем специальной подготовки сюжетного изображения, которое должно быть наложено на фоновое изображение. Существо этой технологии заключается в том, что вначале из того изображения, которое должно быть наложено, удаляется все ненужное и заменяется фоном, имеющим такие цветовые компоненты, которые отсутствуют в сохраняемой сюжетной части. Делается это с той целью, чтобы программа, осуществляющая "фотомонтаж", могла бы отличить фоновый компонент сюжетного изображения, который должен быть заменен компонентом фонового изображения, от сюжетного компонента, который должен быть сохранен. После этого объединение изображений осуществляется программой в соответствии с алгоритмом, который, анализируя сюжетное изображение на каждом пикселе, принимает решение о включении в результирующее изображение либо сюжетного компонента, если его цветовые компоненты отличаются от цветовых компонентов его фона, либо компоненты фонового изображения в противном случае. В настоящее время имеется целый ряд графических редакторов, позволяющих изготавливать изображения с прозрачным фоном.

Цветовая обработка изображения в области, ограниченной произвольным контуром

В ряде случаев при подготовке изображений к печати возникает необходимость в избирательной цветовой обработке изображенных на нем отдельных объектов, например их цветовой коррекции или даже кардинального изменения цвета. Для цветовой обработки изображений в ограниченной области, которая задается пользователем, применяют, так называемую, *маску*. При обработке маска накладывается на изображение. Маска представляет собой бинарное изображение, на котором показаны области изображения, доступные для обработки (маска прозрачна), и области изображения, которые маска делает недоступными для обработки, т. е. зацищает от нее (маска не прозрачна). В процессе обработки изображение сканируется "окном" размером в один пиксел. Все отсчеты изображения, попавшие в область прозрачности маски, подвергаются обработке, а отсчеты изображения, оказавшиеся вне этой области, копируются из массива исходного изображения в массив обработанного изображения. Рисунок 2.13 и ЦВ.1 (на цветной вклейке) поясняет изложенное. На рис. 2.13, *а* приведено исходное изображение, подлежащее обработке, целью которой является изменение цвета одного из цветков тюльпана. На рис. 2.13, *б* показано изображение с наложенной на него маской. В целях наглядности часть изображения, расположенная под непрозрачной областью маски, показана в тусклых цветах, а та часть изображения, которая расположена в области прозрачности маски и, следовательно, подлежит обработке, показана в своем естественном цвете. На рис. 2.13, *в* приведено обработанное изображение, на котором изменен цвет одного из цветков на обработанном изображении с красного на желтый.



Рис. 2.13. Цветовое преобразования изображений в области, ограниченной произвольным контуром: *a* — исходное изображение; *б* — изображение с наложенной на него маской; *в* — преобразованное изображение

2.12. Работа с цветом в графических редакторах

При получении исходных изображений, например, путем фотографирования натуры, сканирования цветной репродукции или ввода в компьютер видеозаписи вследствие несовершенства используемой при этом аппаратуры в них вносятся различные искажения, в том числе искажения в передаче цвета. Эти искажения проявляются в сдвиге цвета, который представляет собой появление постороннего оттенка на изображении. В ряде случаев, особенно в случае, когда в технологическую цепочку получения изображений включен фотопроцесс, сдвиги цвета в различных участках тонового диапазона могут носить противоположный характер. В большинстве практических случаев эти искажения в передаче цвета могут быть скорректированы или, по крайней мере, значительно ослаблены путем соответствующей обработки изображений. В некоторых случаях целью обработки изображений является сознательное искажение передачи цвета с тем, чтобы, выделив на изображении наиболее важные в смысловом отношении детали, придать ему наибольшую выразительность. И та, и другая задачи решаются путем компьютерной обработки изображений с использованием графических редакторов. В дальнейшем для большей конкретности изложения будем ориентироваться на компьютеры PC IBM, которые получили в нашей стране наибольшее распространение, и на популярные графические редакторы Photoshop и Corel PHOTO-PAINT, наиболее часто применяемые при работе на этих компьютерах.

Перед тем как начинать коррекцию сдвига цвета, необходимо сначала его обнаружить и оценить его величину. Многие сдвиги цвета легко обнаружить визуально. Обычно они хорошо заметны на ахроматических объектах при рассеянном освещении, а также на объектах, цвет которых хорошо известен, например, на изображениях лиц, травы, неба и т. д. Поэтому вначале необходимо исследовать именно эти области изображений. При исследовании нужно пользоваться инструментом "пипетка", который имеется в названных выше графических редакторах. Несколько сложнее обстоит дело в том случае, если изображения не содержат ахроматических объектов, но и в этом случае, опираясь на опыт, можно решить проблему. Дело в том, что сдвиги цвета хорошо заметны на ярких диффузных элементах изображения, например, на бежевом цвете, светло-синем и т. п. После того как такие исследования проведены и обнаружены сдвиги цвета, можно приступать к их нейтрализации.

Наиболее часто применяются два метода, позволяющих нейтрализовать сдвиги цвета. В тех случаях, когда сдвиг цвета наиболее сильно проявляется в одном каком-либо тоновом диапазоне, например, в области наибольшей яркости, для его коррекции целесообразно применять коррекцию с использованием инструмента, который имеется в графических редакторах Photoshop и Corel PHOTO-PAINT и называется **Curves** (Кривые), т. е. осуществлять нелинейное преобразование сигнала соответствующего цветового канала. При таком преобразовании имеет место перераспределение цветового контраста корректируемого цвета по динамическому диапазону, однако при этом уменьшение цветового контраста в одной части цветового диапазона сопровождается его усилением в другой части, и здесь необходима осторожность. Этот метод коррекции может быть особенно полезен при коррекции цветовых сдвигов, обусловленных погрешностями фотохимических процессов. Другой метод нейтрализации цветовых сдвигов применяется в том случае, когда сдвиг цвета одинаков по всему тоновому диапазону. В этом случае применяются инструменты Color Balance (Цветовой баланс) при работе в цветовой модели RGB и Selective Color (Выборочный цвет) при работе в цветовой модели СМҮК. В этом случае для нейтрализации цветового сдвига в изображении усиливается цвет, дополнительный к нейтрализуемому оттенку, например, для нейтрализации голубого оттенка необходимо усиливать красный цвет, для нейтрализации желтого оттенка следует усиливать синий цвет и, наконец, для нейтрализации пурпурного оттенка нужно усиливать зеленый цвет.

Кроме перечисленных инструментов в графических редакторах Photoshop и Corel PHOTO-PAINT имеется еще целый ряд инструментов, предназначенных для работы с цветом, а именно **Hue/Saturation** (Оттенок/насыщение), уменьшение насыщенности (**Desaturate**), **Invert** (Инвертировать). Первый из этих инструментов позволяет изменять насыщенность цвета, цветовой оттенок и яркость всего изображения, посредством второго цветное изображение преобразуется в полутоновое черно-белое, а третий преобразует цветное изображение в его цветной негатив, т. е. все цвета в изображении заменяет дополнительными.

Заканчивая краткое изложение принципов коррекции цветных изображений, необходимо отметить, что часто обработка изображений преследует совсем иные цели, в частности, цели так называемого улучшения изображений, при котором в изображения преднамеренно вносятся те или иные искажения, например, искажения цвета с тем, чтобы придать ему определенную выразительность, выделить те или иные представленные на нем предметы. Особенно часто это делается при подготовке к печати изображений, используемых в рекламе. Более подробные сведения по этим вопросам и вопросам, связанным с технологиями подготовки цветных цифровых изображений для печати, имеются в книге [2].

Глава 3



Зрительная система и восприятие изображений

3.1. Проблема оценки качества воспроизведения изображений

При съемке, передаче и воспроизведении изображений в них вносятся искажения и добавляются помехи. Искажения могут быть линейными (частотными), нелинейными, цветовыми и геометрическими. Частотные искажения подразделяются на амплитудно-частотные и фазо-частотные искажения. При амплитудно-частотных искажениях изменяются первоначальные амплитудные соотношения между различными частотными компонентами спектра. При фазо-частотных искажениях изменяются начальные фазы частотных компонентов спектра. Нелинейные искажения сигнала проявляются в том, что нарушаются первоначальные соотношения между мгновенными значениями сигнала. При цветовых искажениях нарушается правильная передача цвета. Наконец, при геометрических искажениях изменяется форма отображаемых объектов сцены. Все эти искажения по-разному воспринимаются зрителем. Амплитудно-частотные искажения изображений проявляются в виде искажений передаваемых световых и цветовых границ и следующих за ними участков изображения. Так, ослабление верхних пространственных частот спектра влечет за собой размытие резких световых и цветовых границ, а ослабление нижних пространственных частот спектра приводит к появлению на изображении так называемых тянучек. Если изображение ахроматическое, то тянучки представляют собой тянущиеся темные продолжения после светлых деталей изображения с резкими световыми границами, и наоборот, светлые продолжения после темных деталей. В случае цветных изображений к ним добавляются цветные тянучки, представляющие собой тянущиеся продолжения в дополнительном цвете после ярко окрашенных деталей с резкими цветовыми переходами. Фазо-частотные искажения в пространственном спектре изображений в зависимости от их вида могут приводить к оконтуриванию деталей изображения, появлению повторов и других нежелательных явлений. Следует отметить, что зрительная система человека крайне чувствительна к фазо-частотным искажениям.

Нелинейные искажения сигнала изображения приводят к ошибкам в передаче градаций яркости, а также к неправильной передаче цветовой гаммы. При передаче изображений также приходится заботиться об обеспечении малой величины цветовых и геометрических искажений.

Искажения и помехи снижают качество воспроизведения изображений и для их оценки необходимы количественные критерии.

Вероятно, одним из первых критериев, предложенных для оценки качества воспроизведения изображения звезды оптическим прибором, был *критерий Рэлея*. С помощью критерия Рэлея оценивался всего один параметр качества воспроизведения, а именно разрешение. Однако уже развитие фотографии показало, что оценивать качество воспроизведения изображения по одному, хотя и важному параметру разрешения, недостаточно. Для оценки качества воспроизведения фотографических изображений необходимо было принимать во внимание наличие зерна (шума) на изображении, ограничения величины воспроизводимого контраста, обусловленного наличием вуали, наличие геометрических искажений, обусловленных несовершенством оптики, и целого ряда других факторов. Появление цветной фотографии, кино и телевидения еще более усложнило проблему.

Простейшим критерием, который часто применяют для оценки качества воспроизведения ахроматических изображений, является среднеквадратичная оценка, нормированная к динамическому диапазону яркости, занимаемому изображением от [73]

$$\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{uck}(x, y)\right]^{2}}}{L_{c. Makc}}.$$

В этой формуле использованы следующие обозначения: $L_c(x, y)$ — яркость неискаженного (эталонного) изображения в точке с координатами (x, y), $L_{иск}(x, y)$ яркость изображения, содержащего искажения в этой же точке, $L_{c. макс}$ — максимальное значение яркости в неискаженном изображении, которое определяет динамический диапазон яркости. Черта над квадратными скобками является символом усреднения по всей площади изображения.

Несомненным достоинством этого критерия является простота его вычисления и измерения. Часто на практике применяется величина, обратная σ — так называемое отношение пикового сигнала к шуму $\psi = 1/\sigma$. В англоязычной литературе для этого критерия принято обозначение PSNR. Однако использование этого критерия, как показал опыт, ограничивается случаем сравнения величин искажений одного типа, например, искажений, обусловленных гауссовым шумом с одинаковыми спектральными интенсивностями. Попытка применить этот критерий для оценки ухудшения качества воспроизведения изображения при воздействии различных источников искажений, например, шума и размытия границ, приводит к неудаче. В этом случае обнаруживается, что при одной и той же величине σ размытие границ в большей степени снижает качество, чем шум (более подробно об этом *см. в разд. 3.10* и *3.12*).

Стремления разработать так называемый обобщенный критерий качества воспроизведения изображений до настоящего времени не увенчались успехом по двум причинам.

Во-первых, как выяснилось в процессе исследований, вид критерия сильно зависит от задачи, для решения которой используется изображение. Объясняется это тем, что влияние искажений и помех, вносимых в изображения, на конечный результат сильно зависит от вида решаемой задачи. Например, при аэрофотосъемках, предназначенных для картографирования местности, велика роль геометрических искажений, в то время как небольшие искажения в передаче цветовой гаммы играют меньшую роль. Иначе обстоит дело в системах вещательного телевидения, в которых от изображения требуется, чтобы искажения были бы незаметны для зрителя при стандартных условиях наблюдения.

Второй причиной, обусловившей отсутствие подходящих критериев оценки качества воспроизведения изображений, по крайней мере, для случаев, с которыми мы имеем дело при производстве репродукций, художественных альбомов, телевизионных программ и т. п., является сложность моделирования зрительной системы человека, которая в конечном итоге оценивает качество воспроизведения изображений.

Вследствие отсутствия решения этой проблемы в целом, инженеры были вынуждены для оценки качества воспроизведения изображений использовать так называемые частные критерии качества [80], а также метод экспертных оценок в баллах, о котором речь пойдет дальше.

При использовании частных критериев наиболее часто оцениваемыми являются следующие параметры:

- разрешающая способность, определяемая числом различимых штрихов миры на изображении по горизонтали и вертикали;
- количество воспроизводимых градаций яркости внутри яркостного динамического диапазона;
- контраст, под которым понимают отношение максимальной яркости изображения к минимальной;
- отношение сигнала к шуму, определяемое как отношение размаха сигнала от черного до белого к среднеквадратичному значению шума;
- геометрические искажения, характеризующие точность воспроизведения координат отдельных элементов исходного изображения;
- цветовоспроизведение, характеризующее степень отличия цветов в полученном изображении от цветов в исходном изображении, а также ряд других параметров, количество которых при использовании этого метода в телевидении достигает нескольких десятков.

Основным недостатком метода экспертных оценок является отсутствие формул, устанавливающих надежную математическую связь между величиной различных искажений и помех, с одной стороны, и величиной экспертной оценки (проставленного балла), с другой стороны. Стремление создать критерии оценки качества воспроизведения изображений, свободные от указанных недостатков, инициировало дальнейшее исследование зрительной системы и разработки на их основе функциональных моделей зрительной системы, которые позволили бы разработать такие критерии. В связи с этим рассмотрим строение зрительной системы и некоторые, важные для дальнейшего рассмотрения проблемы, ее характеристики.

3.2. Краткие сведения о строении зрительной системы

Оптика глаза

На рис. 3.1 показан разрез глаза. Наружная часть глаза, называемая склерой, представляет собой упругую оболочку белого цвета, которая в передней части переходит в прозрачную роговую оболочку. Под роговой оболочкой расположены радужная оболочка, хрусталик, а на противоположной стороне склеры — сетчатка. Внутренняя часть глаза наполнена прозрачным стекловидным веществом. Перевернутое изображение наблюдаемого объекта фокусируется на сетчатке хрусталиком, который представляет собой двояковыпуклую линзу. Фокусное расстояние хрусталика изменяется под воздействием глазных мышц, которые изменяют кривизну его поверхности таким образом, чтобы сфокусированное на сетчатке изображение было бы максимально резким. Как и для большинства одиночных линз, фокусное расстояние хрусталика зависит от длины волны света, синие лучи фокусируются на более близком расстоянии, чем зеленые или красные.

Радужная оболочка играет роль диафрагмы. Диаметр ее отверстия (диаметр зрачка) под воздействием другой группы мышц изменяется при изменении интенсивности света. При низкой освещенности диаметр равен 7,5 мм, при высокой освещенности — 1,8 мм. Когда зрачок сужается, увеличивается четкость изображения на сетчатке [95].



Рис. 3.1. Разрез глаза: 1 — склера; 2 — роговая оболочка; 3 — сосудистая оболочка; 4 — радужная оболочка; 5 — сетчатка; 6 — зрительный нерв; 7 — центральная ямка; 8 — хрусталик; 9 — слепое пятно

Сетчатка состоит из нескольких слоев, причем слои, обращенные к хрусталику, прозрачны и не препятствуют прохождению света к слою, образованному фоторецепторами двух видов: палочками и колбочками.

Колбочки, в сетчатке их около 6,5 млн, представляют собой светочувствительные клетки, которые входят в систему дневного (фотопического) зрения, способную различать цвета. Эта способность обусловлена тем, что сетчатка включает в себя три типа колбочек, различающихся между собой спектральной чувствительностью. Один тип колбочек имеет максимум спектральной чувствительности в области красных излучений, другой — в области зеленых, а третий — в области синих излучений. Благодаря тому, что в центральной части сетчатки (в центральной ямке) каждая колбочка передает возбуждение по отдельному волокну зрительного нерва в последующие отделы зрительной системы, сетчатка дневного зрения имеет высокое пространственное разрешение.

Палочки, в сетчатке их около 130 млн, входят в систему сумеречного (скотопического) зрения, которая не способна различать цвета и имеет низкое пространственное разрешение, но зато имеет высокую световую чувствительность. Высокая световая чувствительность палочковой сетчатки обусловлена тем, что сигналы от большой группы палочек (их насчитывается до 400) объединяются вместе.

В сетчатке, схема синаптических связей которой показана на рис. 3.2, осуществляется преобразование оптического изображения в так называемое *нейронное изображение*, которое представляет собой распределение возбуждений фоторецепторов, а также его первичная обработка. В результате воздействия света на фоторецепторы в них протекает процесс разложения светочувствительного вещества (зрительного пигмента), что приводит к возникновению на их выходах (пресинаптических окончаниях) электрических потенциалов. В живом глазу одновременно с разложением зрительного пигмента идет процесс его регенерации, причем для каждого уровня освещенности сетчатки устанавливается свое динамическое равновесие между этими процессами. Фоторецепторы соединены с горизонтальными и биполярными клетками сетчатки, образуя так называемые триады (см. рис. 3.2).

Согласно современным представлениям сигналы, которые поступают с фоторецепторов, возбуждают систему горизонтальных клеток. Горизонтальные клетки имеют сильную взаимную связь, так что возбуждение даже одной из них распространяется по всей системе. Система горизонтальных клеток осуществляет регуляцию передачи сигналов от фоторецепторов к биполярным клеткам, т. е. через систему горизонтальных клеток осуществляется обратная связь [21]. Благодаря наличию обратной связи происходит перекодировка сигналов, вследствие которой распределение освещенности на сетчатке преобразуется в распределение возбуждений биполярных клеток, которые пропорциональны распределениям локальных и временных контрастов.

Другими словами, величина возбуждения (выходной сигнал) оказывается пропорциональной дроби, где в числителе — разница между локальной освещенностью и средней освещенностью на сетчатке, а в знаменателе — средняя освещенность сетчатки [9, 37, 122]. Результат принимает положительное значение при включении света (или при переводе взора на светлый участок наблюдаемой сцены) и отрицательное значение при его выключении. Это преобразование сигнала триадой создает предпосылки для формирования сигналов в оп- и off-системах, и в частности, сигналов, возбуждающих ганглиозные клетки, показанные на рис. 3.2. Описанный процесс обеспечивает адаптацию зрительной системы к уровню освещенности наблюдаемой сцены или изображения. Благодаря адаптации динамический диапазон сигналов, представленных в виде возбуждений биполярных клеток, уменьшается до 2 логарифмических единиц. Это позволяет зрительной системе, в которой зрительный нерв способен проводить сигналы с небольшим динамическим диапазоном (не более 2 логарифмических единиц), эффективно работать в условиях реальных изменяющихся освещенностей, динамический диапазон которых составляет около 9 логарифмических единиц.



Рис. 3.2. Схема синаптических связей в сетчатке: R — палочка, C — колбочка, MB — карликовый биполяр, RB — палочковый биполяр, FB — плоский биполяр, H — горизонтальная клетка, A — амакриновая клетка, MG — карликовая ганглиозная клетка, DG — диффузная ганглиозная клетка

До настоящего времени остается неясным, объединены ли горизонтальные клетки сетчатки человека в три отдельные системы, каждая из которых связана только с одним типом колбочек "красных", "зеленых" и "синих", или же таких систем меньше. Первое предположение представляется более вероятным, поскольку оно объясняет ряд экспериментально наблюдаемых фактов, в частности, феномен константности цветового восприятия [95]. Однако пока были найдены только две системы горизонтальных клеток в сетчатке человека.

Биполярные клетки, в свою очередь, соединены с амакриновыми и ганглиозными клетками, образуя триады, подобные рассмотренным ранее. Имеющиеся экспериментальные данные позволяют предполагать, что амакриновые клетки, подобно горизонтальным клеткам, также образуют системы связанных между собой клеток, выполняя регуляцию передачи сигналов от биполярных клеток к ганглиозным.

Таким образом, в зрительной системе действуют два последовательно включенных механизма адаптации к освещенности наблюдаемой сцены. Первый, реализуемый триадами "фоторецептор — горизонтальные клетки — биполярная клетка", осуществляющий адаптацию к средней освещенности сетчатки, и второй, реализуемый триадами "биполярная клетка — амакриновые клетки — ганглиозная клетка". Первый механизм инерционен. Время, в течение которого он обеспечивает адаптацию, измеряется минутами, т. к. определяется постоянными времени регенерации зрительных пигментов. Второй механизм малоинерционный. Необходимость во втором механизме адаптации обусловлена тем, что в наблюдаемой сцене встречаются как затемненные участки, так и участки с высокой освещенностью. При наблюдении реальных сцен ось зрения скачками, с интервалом в десятые доли секунды, перемещается в пространстве, что приводит к мгновенным изменениям сигнала, иногда довольно большим. Малоинерционный механизм адаптации уменьшает динамический диапазон сигналов, возбуждающих ганглиозные клетки, до одной логарифмической единицы. Выполняемое им преобразование сводится к локальному центрированию возбуждающих сигналов относительно среднего значения в пределах поля ясного видения.

При передаче сигналов от биполярных клеток к ганглиозным клеткам происходит также их преобразование из аналоговой формы в импульсную форму. Начиная с ганглиозных клеток, по зрительному нерву передаются сигналы в виде импульсных последовательностей, частота следования импульсов в которых определяется интенсивностью передаваемых сигналов.

Место, в котором зрительный нерв, состоящий примерно из миллиона волокон, выходит из глаза, называется слепым пятном, т. к. в этом месте сетчатки нет фоторецепторов.

Зрительные нервы, выходя из глаз, идут к задней части головы, где они разветвляются на два тракта и перекрещиваются, так что дальше в каждом зрительном тракте содержатся волокна от обоих глаз. Оба пучка продолжаются до основания мозга, и каждый соединяется с наружным коленчатым телом. Было найдено, что в каждом глазу имеется центральная, приблизительно круглая зона, которая соединена с обоими полушариями мозга. По сторонам от этой зоны левая половина каждого глаза

(если разделить его по вертикали) соединена с одной стороной мозга, а правая с другой. В то время как объекты при центральном зрении представлены в обоих полушариях мозга, периферические объекты представлены в каком-либо одном полушарии. Наружное коленчатое тело сложной системой нервов соединено с 17-м полем зрительной коры. В зрительной коре происходит анализ зрительной информации.

Наиболее изученным отделом зрительной системы является сетчатка, свойствами которой определяются основные психофизические законы. Высшие отделы зрительной системы, локализованные в зрительной коре, изучены в меньшей степени.

3.3. Адаптация зрительной системы к освещенности и контрастная чувствительность

Как уже было отмечено в предыдущем разделе, взаимодействие рецепторов, системы горизонтальных клеток и биполярных клеток обеспечивает адаптацию зрительной системы к средней освещенности наблюдаемой сцены. При этом величина сигнала на выходе *i*-ой биполярной клетки оказывается пропорциональной дроби, где в числителе стоит разность между локальной освещенностью *I_i* (освещенностью *i*-

го рецептора) и средней освещенностью на сетчатке I, а в знаменателе — средняя освещенность сетчатки [9, 37]. Так, например, величина сигнала на выходе биполярной клетки, передающей возбуждение от колбочки, имеющей максимум чувствительности в области красных излучений, будет равна

$$S_{\kappa i} = C_{\kappa} \frac{I_i - \overline{I}}{\overline{I}}, \qquad (3.1)$$

где $C_{\rm k}$ — постоянный коэффициент, согласующий размерности, величина которого определяется спектральной чувствительностью фоторецептора и спектральным составом излучения. Аналогичным образом находятся значения сигналов на выходах биполярных клеток, соединенных с фоторецепторами, имеющими максимумы чувствительности в области зеленых и синих излучений.

Формула (3.1) отражает две важные особенности преобразования распределения освещенности на сетчатке I_i в распределение сигналов на выходе биполярных клеток $S_{\kappa i}$, осуществляемых триадой фоторецептор — горизонтальная клетка — биполярная клетка:

- увеличение освещенности изображения в п раз не приводит к изменению сигналов S_{кі}, т. к. при этом числитель и знаменатель в (3.1) возрастают в одно и то же число раз, что собственно и обеспечивает адаптацию;
- \Box сигналы $S_{\kappa i}$ являются линейными функциями освещенностей I_i / \overline{I} (рис. 3.3).

Благодаря этому выполняются основные законы колориметрии. В частности, восприятия цветового тона и насыщенности не зависят от яркости в широком диапазоне ее изменений, а воспринимаемая яркость стимула является взвешенной суммой цветовых компонентов. И, наконец, отметим, что поскольку в результате преобразования, описываемого (3.1), возникают как положительные, так и отрицательные значения сигналов $S_{\kappa i}$, то это, как уже указывалось, создает базу для двух систем: *on* и *off*, работающих в разные стороны от нулевого уровня.



Рис. 3.3. Зависимость $S_{\kappa i}/C_{\kappa}$ от I_i/\overline{I}

Примечание

Формула (3.1) представляет собой математическое описание закона преобразования распределения освещенности на сетчатке в распределение возбуждений (сигналов).

Рассмотрим классическую задачу определения порогового контраста в условиях полной адаптации, полагая изображение стимула в виде диска, состоящего из двух половинок с яркостью $L_s + \Delta L_s / 2$ и $L_s - \Delta L_s / 2$, размещенного на окружающем его поле, яркость которого L_a . Так как освещенность на сетчатке *I* прямо пропорциональна яркости соответствующего участка тестового изображения *L*

$$I = a_0 L , \qquad (3.2)$$

то для средней освещенности сетчатки будем иметь

$$L = a_0 \left(c_1 L_a + L_s \right), \tag{3.3}$$

где a_0 — постоянный коэффициент, согласующий размерности, c_1 — коэффициент, определяющий добавку, которую вносит яркость поля, окружающего стимул, в среднюю освещенность сетчатки. Вычисляя по формулам (3.1—3.3) значения сигналов на выходах биполярных клеток, возбуждаемых, например фоторецепторами, имеющими максимум чувствительности в области красных излучений, на которые проецируются правая и левая половинки стимула, и взяв разность этих значений, получим

$$\Delta S_{\rm K} = \frac{C_{\rm K}}{\left(c_1 L_a + L_s\right)} \Delta L_s \,. \tag{3.4}$$

В (3.4) множитель, стоящий при ΔL_s , можно рассматривать как коэффициент передачи, устанавливающий связь между яркостями элементов стимула и разностью сигналов $\Delta S_{\rm k}$ на выходах соответствующих биполярных клеток. Аналогичным образом находятся значения $\Delta S_{\rm s}$ и $\Delta S_{\rm c}$ для биполярных клеток, связанных с фоторецепторами, имеющих максимумы чувствительности в областях зеленого и синего излучений. Формулы, по которым вычисляются $\Delta S_{\rm s}$ и $\Delta S_{\rm c}$, отличаются от формулы (3.4) лишь коэффициентами пропорциональностей $C_{\rm s}$ и $C_{\rm c}$.

При наблюдении ахроматического изображения (в рассматриваемом случае стимула) в зрительной системе формируются сигналы ΔS , пропорциональные сумме сигналов $\Delta S_{\rm k}$, $\Delta S_{\rm 3}$ и $\Delta S_{\rm c}$

$$\Delta S = \Delta S_{\rm K} + \Delta S_{\rm 3} + \Delta S_{\rm c}$$

Выполняя очевидные преобразования, получим

$$\Delta S = \frac{C}{\left(c_1 L_a + L_s\right)} \Delta L_s \,, \tag{3.5}$$

где

$$C = C_{\rm K} + C_{\rm 3} + C_{\rm c}$$
.

Для того чтобы имело место пороговое обнаружение стимула, необходимо, чтобы разность сигналов ΔS , обусловленная перепадом яркости между его половинками ΔL_s , равнялась порогу, определяемому среднеквадратичным значением шума в зрительной системе σ , угловым размером стимула α и временем его предъявления τ :

$$\Delta S = D(\alpha, \tau) \sigma, \qquad (3.6)$$

где $D(\alpha, \tau)$ — функция, учитывающая влияние α и τ на величину порога вследствие пространственной и временной суммации. Шум в зрительной системе, и в частности, в ахроматическом канале, как известно, включает в себя два компонента. Первый компонент обусловлен квантовыми флуктуациями светового потока от стимула, а также флуктуациями, которые возникают в фоторецепторе при усилении сигнала. Средний квадрат этого компонента шума σ_F^2 прямо пропорционален средней яркости стимула L_s

$$\sigma_F^2 = gL_s \,, \tag{3.7}$$

где *g* — постоянный коэффициент.

Второй компонент шума обусловлен флуктуационными процессами в последующих нейронах, его средний квадрат σ_n^2 не зависит от яркости стимула. Поэтому средний квадрат шума в зрительной системе, пересчитанный к выходу биполяров, будет равен

$$\sigma^{2} = \sigma_{n}^{2} + \frac{C^{2}}{\left(c_{1}L_{a} + L_{s}\right)^{2}} gL_{s}.$$
(3.8)

Делая подстановку в (3.6) значений ΔS и σ из (3.5) и (3.8), а затем, решая получившееся уравнение относительно $\Delta L_s/L_s$, найдем

$$\frac{\Delta L_s}{L_s} = \delta_1(\alpha, \tau) \left(1 + \frac{c_1 L_a}{L_s} \right) \sqrt{1 + \frac{G}{\left(1 + c_1 L_a / L_s \right)^2 L_s}}, \qquad (3.9)$$

где обозначено $\delta_1(\alpha, \tau) = D(\alpha, \tau)\sigma_n/C$. Значения $\delta_1(\alpha, \tau)$, *G* и c_1 определяются из экспериментальных данных.

В случае, когда яркость поля, окружающего стимул, равна нулю, а средняя яркость стимула достаточно велика, для того чтобы можно было пренебречь дробью под квадратным корнем вследствие ее малости по сравнению с единицей, (3.9) переходит в закон Вебера — Фехнера

$$\Delta L_s/L_s=\delta_1(\alpha,\tau),$$

причем $\delta_1(\alpha, \tau)$ имеет смысл дифференциального порога, который зависит от углового размера α и времени предъявления τ стимула.

В случае, когда яркость поля, окружающего стимул, равна нулю, а средняя яркость стимула настолько мала, что под квадратным корнем можно пренебречь единицей ввиду ее малости по сравнению со вторым слагаемым, формула (3.9) переходит в флуктуационный закон

$$\Delta L_s / L_s = \delta_1 (\alpha, \tau) \sqrt{G / L_s}$$

На рис. 3.4 для рассмотренного случая приведены экспериментальные точки, взятые из [95], и теоретическая зависимость, рассчитанная по формуле (3.9). При расчете было принято $\delta_1(\alpha, \tau) = 0,02$, G = 0,286 кд/м², $c_1 = 0$.



Рис. 3.4. Зависимость $\Delta L_s / L_s$ от L_s



Рис. 3.5. Зависимость разностного порога ΔL_s от средней яркости стимула L_s при различных яркостях поля, окружающего стимул L_a , $\alpha = 1,5^\circ$: $L_a = 3426$ (1), 343 (2), 34 (3), 3,4 (4), 0 (5) кд/м², при расчетах принято: $\delta_1(\alpha, \tau) = 0.0115$, G = 0.46 кд/м², $c_1 = 0.057$

Обращает на себя внимание хорошее совпадение расчетных и экспериментальных данных в интервале изменения яркости, составляющем более семи логарифмических единиц.

На рис. 3.5 приведены экспериментальные точки, взятые из [144], а также семейство теоретических кривых, рассчитанных по формуле (3.9) для случая, когда яркость фона, окружающего стимул, не равна нулю. В данном случае расчетные зависимости также находятся в хорошем согласии с экспериментальными точками во всем диапазоне измерений.

Примечание

Формула (3.9) представляет один из законов адаптации, описывающий зависимость дифференциального порога от средней яркости стимула и яркости окружающего его поля, частными случаями которого являются законы Вебера — Фехнера и флуктуационный.

Полученные результаты играют определяющую роль при разработке дисплеев для отображения изображений сцен с большим динамическим диапазоном яркостей (*см. разд. 6.8 и 6.9*). Попытка решить эту проблему только путем создания дисплеев, динамический диапазон яркости которых составлял бы десятки и сотни тысяч, проблемы не решает. Дело в том, что следующим звеном, ограничивающим диапазон воспроизведения яркостей, является зрительная система, которая адаптируется к средней яркости, определяемой средней яркостью экрана и освещенностью помещения. Только при использовании очень больших экранов в полностью затем-

ненном помещении возможны условия, когда адаптация зрительной системы будет определяться локальными яркостями экрана на участках, рассматриваемых в данный момент зрителем. Аналогичная ситуация возможна при использовании виртуального шлема, однако в последнем случае не потребуются дисплеи с повышенным динамическим диапазоном воспроизводимых яркостей.

3.4. Разрешающая способность зрительной системы в пространстве (острота зрения)

Пространственная разрешающая способность зрительной системы определяется плотностью расположения фоторецепторов сетчатки, а она в высшей степени неравномерна. В центральной части сетчатки глаза, называемой центральной ямкой, фоторецепторы расположены наиболее плотно, образуя мозаику, напоминающую пчелиные соты. В этой области сетчатки расположены в основном колбочки, максимум спектральной чувствительности которых приходится на область красных или зеленых излучений. Поперечный размер этих колбочек составляет 0,6 угловых минуты. Малые размеры рецептивных полей, зоны возбуждения которых представлены одной колбочкой, при малых размерах колбочек и их плотной упаковке обеспечивают высокое пространственное разрешение в этой области сетчатки, которое составляет около одной угловой минуты. По мере удаления от центра сетчатки размеры рецептивных полей возрастают за счет того, что их зона возбуждения представляется уже не одной, а несколькими колбочками, что влечет за собой уменьшение пространственного разрешения. При еще большем удалении от центра сетчатка становится смешанной, состоящей из колбочек и палочек, образующих большие рецептивные поля. Периферия сетчатки представлена исключительно палочками, рецептивные поля которых состоят из нескольких десятков и даже сотен палочек.



Рис. 3.6. Зависимость остроты зрения от места раздражения на сетчатке [95]. По оси абсцисс отложено расстояние проекции изображения от середины центральной ямки сетчатки в градусах, по оси ординат — острота зрения в относительных единицах
Обычно разрешающую способность зрительной системы определяют в терминах остроты зрения. Под остротой зрения понимают отношение расстояния, с которого наблюдатель различает деталь объекта, например, разрыв в кольце Ландольта, к расстоянию, с которого эта деталь видна под углом в одну угловую минуту. На рис. 3.6 приведена зависимость остроты зрения от положения различаемой детали на сетчатке. Из рисунка видно, что только очень небольшая часть сетчатки характеризуется высокой остротой зрения (высоким разрешением), в то время как острота зрения остальных участков в десятки раз ниже. Измерения показывают, что лишь 20 угловых минут зрительного пространства глаза приходятся на область сетчатки с высоким разрешением (область фовеолы), тогда как остальные 100° попадают на область сетчатки с малым разрешением. Из этого следует, что при наблюдении сцен мы не способны их видеть одновременно целиком, а вынуждены рассматривать по частям, формируя их образ в зрительной памяти. Исследования, выполненные Ярбусом, в которых испытуемому предлагали для наблюдения изображения и при этом записывали движения его глаз, полностью подтверждают это. На рис. 3.7 приведены два изображения. Первое из них было предъявлено участвовавшему в опытах наблюдателю для рассматривания, а на втором показаны записанные траектории движения его глаз во время опыта, а также точки фиксации взора, которые в основном пришлись на области, где градиент изменения яркости велик. На рис. 3.7, а показаны проекции области сетчатки с высоким разрешением (фовеолы) в виде окружностей, центр которых совмещен с точками фиксации взора. Интересно отметить, что области фиксации взора располагаются на наблюдаемом изображении таким образом, что на область сетчатки с высоким разрешением проецируются наиболее информативные части наблюдаемого изображения — световые границы.



Рис. 3.7. Траектории движения глаз при восприятии изображений [95]: *a* — изображение, предъявленное испытуемому для наблюдения; *б* — запись движения глаза при рассматривании изображения; *в* — фрагменты контуров в зоне ясного видения. Точками отмечены фиксации взора на элементах изображения, тонкими линиями — траектории движения оси зрения при переводе взора в новую точку фиксации, окружностями — зоны ясного видения

В процессе рассматривания сцены или изображения ось зрения совершает скачкообразные движения — саккады. Длительность скачка, независимо от угла, на который перемещается ось зрения, составляет 50—60 мс, причем во время скачка происходит подавление зрительного восприятия.

В результате проведенных исследований было также показано, что в зависимости от решаемой зрительной задачи расположение точек фиксации взора на изображении может варьироваться в широких пределах, неизменным остается одно — точки фиксации всегда располагаются вблизи контуров на изображениях, обеспечивая тем самым высокое разрешение наиболее информативных частей наблюдаемых сцен и изображений.

Традиционно используемая в офтальмологии в качестве меры разрешающей способности зрительной системы острота зрения является достаточно грубой оценкой. Для более точной характеристики способности зрительной системы различать мелкие объекты применяют частотную передаточную функцию и частотно-контрастную характеристику. Для измерения и той, и другой характеристик применяют тестовые изображения в виде решеток, яркость в которых изменяется по синусоидальному закону

$$L = L_0 + L_1 \sin \omega_c x ,$$

где L_0 и L_1 — средняя яркость и амплитуда изменения яркости в решетке соответственно, ω_c — круговая пространственная частота решетки, а *x* — координата, измеряемая в направлении, перпендикулярном полосам решетки.

Частотная передаточная функция зрительной системы $K(f_x, f_y)$ показывает, как изменяется оцениваемый наблюдателем контраст синусоидальной решетки в надпороговых условиях наблюдения в зависимости от ее пространственной частоты. Для измерения частотной передаточной функции зрительной системы применяют метод согласования контраста тестовой решетки с контрастом эталонной решетки. Характерными особенностями частотной передаточной функции зрительной системы являются: наличие максимума в области около 15 пер/град, небольшой спад в области низких пространственных частот, обусловленный взаимодействием биполярных, амакриновых и ганглиозных клеток, вызывающим латеральное торможение, и спад в области высоких пространственных частот, вызванный конечными размерами колбочек и несовершенством оптики глаза. Наличие спада в области низких пространственных частот приводит к небольшому подчеркиванию границ на изображении. Этот эффект, в случае рассматривания градационного клина, называется полосками Маха. На рис. 3.8 представлена частотная передаточная функция для центральной ямки. При переходе к другим областям сетчатки обнаруживается сильный спад частотной передаточной функции в области верхних пространственных частот, причем тем больший, чем дальше от центра проецируется изображение тестовой решетки.



Рис. 3.8. Частотная передаточная функция зрительной системы



Рис. 3.9. Частотно-контрастная характеристика

Частотно-контрастная характеристика зрительной системы в отличие от частотной передаточной функции, напротив, измеряется в пороговых условиях наблюдения, когда обнаружение тестовой решетки ограничивается внутренним шумом зрительной системы. Характерной особенностью частотно-контрастной характеристики является наличие максимума в области пространственной частоты 4 периода на градус и резкий спад как в области низких пространственных частот, так и в области верхних пространственных частот. Наличие сильного спада частотно-контрастной характеристики в области низких пространственных частот обусловлено тем, что спектральная интенсивность внутреннего шума зрительной системы имеет резкий подъем в данной области пространственных частот, что было обнаружено как в психофизических, так и в электрофизиологических опытах. Сильный

спад частотно-контрастной характеристики в области верхних пространственных частот вызван также ростом спектральной интенсивности внутреннего шума зрительной системы в этой области, обусловленного пространственной дискретизацией изображения на сетчатке. На рис. 3.9 приведена частотно-контрастная характеристика для центральной ямки. Так же, как и частотная передаточная функция, частотно-контрастная характеристика зависит от того, на какой участок сетчатки проецируется тестовая решетка.

Знание приведенных выше характеристик необходимо для того, чтобы иметь возможность оценивать видность искажений на изображениях, обусловленных как помехами, так и искажениями сигнала изображения при его передаче или консервации. Этими характеристиками в конечном итоге определяется число строк, на которое разлагается изображение при его вводе в компьютер или при передаче по каналу связи в телевидении.

3.5. Инерционность зрения

Инерционность зрительной системы обусловлена двумя причинами: инерционностью фотохимических процессов, протекающих в фоторецепторах сетчатки, и инерционностью процессов распространения сигналов по ее нейронным сетям.

Первый вид инерционности проявляется при изменении средней освещенности наблюдаемой сцены, в результате которого нарушается динамическое равновесие между скоростями распада и регенерации зрительных пигментов в фоторецепторах. Для восстановления этого равновесия требуется от нескольких минут до нескольких десятков минут в зависимости от начального и конечного уровней освещенностей, при которых протекает адаптация.

Инерционность же, определяемая процессами распространения сигналов в нейронных сетях зрительной системы, проявляется на более коротких временных интервалах, не превышающих одной секунды.

Рассмотрим механизм возникновения этого вида инерционности. Известно, что рецептивные поля сетчатки зрительной системы человека включают в себя центральную зону возбуждения и периферическую зону торможения. Таким образом, результирующий сигнал от рецептивного поля сетчатки представляет собой разность между сигналом возбуждения от центральной зоны рецептивного поля и сигналом торможения от его периферической зоны. При постоянном во времени равномерном уровне освещенности сетчатки сигнал возбуждения от центра рецептивного поля и сигнал возбуждения от периферической зоны. При постоянном во времени равномерном уровне освещенности сетчатки сигнал возбуждения от центра рецептивного поля и сигнал торможения от периферии уравновешивают друг друга. Если же освещенность сетчатки изменяется во времени относительно своего среднего значения, например, по синусоидальному закону с частотой мельканий f, то изменения сигналов торможения относительно изменения сигнала возбуждения будут запаздывать. Причем чем с более дальней периферии будет приходить сигнал торможения, тем на большее время он будет запаздывать. Запаздывание это проявляется в том, что фаза изменения сигнала торможения отстает от фазы изменения сигнала

возбуждения, в результате чего они уже не будут уравновешивать друг друга. При этом по нейронной сети зрительной системы будет распространяться сигнал разности этих двух сигналов, т. е. сигнал мелькания. Рисунок 3.10 поясняет изложенное.



Рис. 3.10. Зависимость от времени t сигналов возбуждения, торможения и мелькания



Рис. 3.11. Зависимость $1/\Delta I$ от f

На низких частотах увеличение частоты мелькания f влечет за собой увеличение амплитуды сигнала мелькания или, что то же самое, уменьшение значения пороговой амплитуды мелькания ΔI . Однако так будет продолжаться лишь до тех пор, пока не начнет проявляться затухание сигналов вследствие разброса латентности распространения сигналов, а также вследствие инерционности рецепторов и проводящих нервных путей и рецепторов. В результате этого начиная с некоторой частоты будет иметь место увеличение пороговой амплитуды мелькания ΔI с ростом f и, наконец, при некотором значении частоты мелькания $f_{\rm kp}$, которое называется критической частотой мелькания, мелькания станут незаметными. На рис. 3.11 приведены экспериментальные и расчетные зависимости $1/\Delta I$ от f.

При частотах мелькания, превышающих критическую частоту, мелькания незаметны, и яркость наблюдаемого поля L(t) кажется неизменной и пропорциональной своему среднему значению

$$L_{\rm cp} = \frac{1}{T} \int_{0}^{T} L(t) dt , \qquad (3.10)$$

где T = 1/f — период мелькания. Формула (3.10) представляет собой закон Тальбота. Этот закон справедлив не только для случая, когда яркость L(t) изменяется по синусоидальному закону, но и для любого другого закона ее изменения. Так, если яркость наблюдаемого поля изменяется по импульсному закону, что, например, имеет место при наблюдении телевизионного экрана, то условие, при котором зрителю будет казаться, что яркость экрана неизменна, запишется следующим образом

$$LT_{\rm H} = {\rm const} , \qquad (3.11)$$

где L — яркость свечения экрана во время импульса, $T_{\rm u}$ — длительность, в течение которой светится экран. Выражение (3.11) представляет собой известный закон Блоха — Шарпантье, который выполняется при условии, что $T_{\rm u} \leq 60$ —100 мс. Критическая частота мельканий, как показали исследования, зависит как от средней освещенности сетчатки I или, что то же самое, от яркости наблюдаемого мелькающего поля, так и от его угловых размеров. При проектировании устройств воспроизведения изображений, например дисплеев, знание этой величины очень важно, т. к. на ее основании определяется частота смены кадров (*см. также разд. 11.1*). Так, при используемых яркостях экранов компьютерных дисплеев $f_{\rm kp} \approx 60$ Гц, поэтому частота кадров в них выбирается выше этого значения. Сле-

дует заметить, что инерционность центральной части сетчатки и ее периферической части существенно различны. Инерционность периферической части сетчатки значительно ниже, в чем легко убедиться, наблюдая экран дисплея на электроннолучевой трубке боковым зрением, в этом случае хорошо заметны его мелькания, которые совсем незаметны, если на него смотреть прямо.

3.6. Восприятие движения

Восприятие движения — одна из важнейших задач решаемых зрительной системой. В настоящее время в этой области имеется много неясного. В общем случае движение может быть разложено на два компонента: на радиальную составляющую (движение на наблюдателя или от него) и на тангенциальную составляющую. Почти все характеристики восприятия движения получены для тангенциальной составляющей. Установлено, что для фовеального зрения минимально различимая скорость движения объекта при наличии в поле зрения неподвижных ориентиров составляет 1—6 угловых минут в секунду, а при отсутствии ориентиров — 10—20 угловых минут в секунду. Максимальная скорость, при которой еще воспринимается движение объекта, лежит в пределах 12—32 град/с. Установлено также, что чувствительность к восприятию движения уменьшается от центральной ямки к периферии по линейному закону. Для периферического зрения при отклонении в 9° от центральной ямки минимальные пороги составляют 18 угловых минут в секунду при наличии неподвижных ориентиров в поле зрения, а при отсутствии неподвижных ориентиров величина минимальных порогов в 10—20 раз больше.

Для создания ощущения движения объекта необязательно, чтобы его проекция на сетчатку непрерывно смещалась. Это ощущение не нарушается и в том случае, если на сетчатку глаза проецировать ряд быстро сменяющих друг друга изображений объекта, на которых представлены последовательные фазы его движения. Именно это свойство зрения используется для создания иллюзии движения в кино и в телевидении. Для того чтобы движение в направлении, перпендикулярном оси наблюдения, воспринималось непрерывным, частота смены изображений с различными фазами движения должна быть достаточно большой, иначе движение будет казаться прерывистым. Чем больше яркость экрана, тем выше эта частота, однако при используемых яркостях экрана она не превышает 60—70 Гц.

Остановимся еще на таком важном явлении, как смаз движущегося изображения. Известно, что резкие световые границы движущегося объекта воспринимаются зрителем размытыми вследствие инерционности зрения. Однако если мы наблюдаем изображение движущегося объекта, границы которого уже размыты, и если это размытие превышает 10 угловых минут, то такое изображение воспринимается нами как более четкое. С таким случаем мы встречаемся при просмотре спортивных телевизионных программ, в которых встречаются сцены, содержащие быстрые движения. Результаты экспериментальных исследований этого феномена приведены в публикации [133], а результаты его теоретического исследования — в работе [57]. Этот феномен может быть использован при разработке методов сжатия движущихся изображений.

3.7. Восприятие цвета

Способность различать цвета обусловлена наличием в сетчатке трех типов колбочек, различающихся между собой спектральной чувствительностью. На рис. 3.12 приведены нормированные кривые спектральной чувствительности, заимствованные из [95]. Максимумы этих кривых по данным Уолда приходятся на 430, 540 и 575 нм.



Рис. 3.12. Нормированные кривые спектральной чувствительности колбочек

Обращает на себя внимание широкополосность и сильное взаимное перекрытие кривых спектральной чувствительности, особенно для колбочек, имеющих максимумы спектральной чувствительности в области красных и зеленых излучений. При проецировании изображений на сетчатку глаза в колбочках под действием света происходит распад зрительного пигмента, что приводит к появлению на их выходах электрических сигналов e_{κ} , e_{3} и e_{c} , величина которых определяется спектром излучения $F(\lambda)$ и спектральными характеристиками колбочек

$$e_{\kappa} = \int_{\lambda_{\text{MUH}}}^{\lambda_{\text{MAKC}}} \varepsilon_{\kappa}(\lambda) F(\lambda) d\lambda, \ e_{3} = \int_{\lambda_{\text{MUH}}}^{\lambda_{\text{MAKC}}} \varepsilon_{3}(\lambda) F(\lambda) d\lambda, \ e_{c} = \int_{\lambda_{\text{MUH}}}^{\lambda_{\text{MAKC}}} \varepsilon_{c}(\lambda) F(\lambda) d\lambda,$$

где e_{κ} , e_{3} и e_{c} — сигналы на выходах колбочек, имеющих максимумы спектральной чувствительности в областях красного, зеленого и синего излучений соответственно;¹ $\varepsilon_{\kappa}(\lambda)$, $\varepsilon_{3}(\lambda)$ и $\varepsilon_{c}(\lambda)$ — спектральные характеристики этих колбочек; λ — длина волны светового излучения; $\lambda_{\text{макс}}$ и $\lambda_{\text{мин}}$ — граничные значения длин волн, определяющих диапазон световых излучений.

Таким образом, первой операцией, которую выполняет зрительная система над изображением, является его цветоделение. Сигналы с выхода колбочек после предварительной обработки в сетчатке и перекодировки в ахроматические и две группы цветоразностных сигналов передаются в высшие отделы зрительной системы по зрительному нерву.

По-видимому, основной задачей, которая решается при обработке изображения сетчаткой глаза, является формирование передаваемых в высшие отделы зрительной системы сигналов, которые были бы не зависимыми не только от интенсивно-

¹ Строго говоря, e_{κ} , e_{3} и e_{c} представляют собой сигналы на выходе изолированных колбочек.

сти освещения наблюдаемой сцены, но и в значительной степени, от спектрального состава этого освещения. Независимость восприятия цвета от спектрального состава освещения называется константностью восприятия цвета. Исследованию механизма константности восприятия цвета посвящено большое количество работ, однако эта проблема до конца не решена.

В настоящее время известно, что в сетчатках ряда животных, и в частности рыб, горизонтальные клетки объединены в системы, связанные с теми или иными группами фоторецепторов, имеющих одинаковую спектральную чувствительность. Такая организация сетчатки обеспечивает раздельную регуляцию фоторецепторов с различной спектральной чувствительностью при изменении спектрального состава освещения.

В отношении организации горизонтальных клеток в сетчатках приматов и человека в настоящее время много неясного. Если, однако, допустить, что в сетчатке приматов и человека присутствуют 3 системы горизонтальных клеток, связанных с фоторецепторами, имеющими различную спектральную чувствительность, то модель обработки изображений зрительной системой может быть представлена следующим образом.

При проецировании изображения сцены на сетчатку глаза на выходах колбочек возникают электрические сигналы, которыми возбуждаются три системы горизонтальных клеток, каждая из которых связана с колбочками, имеющими одинаковые спектральные чувствительности. Возбуждение каждой из систем горизонтальных клеток пропорционально возбуждению колбочек сетчатки с одинаковой спектральной чувствительностью, усредненному по всем колбочкам данного типа. Через эти системы горизонтальных клеток осуществляется обратная связь, посредством которой регулируется передача сигналов от колбочек к биполярным клеткам. В результате сигналы на выходах биполярных клеток, связанных с колбочками, имеющими максимумы спектральной чувствительности в областях красного, зеленого и синего излучений, будут

$$s_{\rm K} = \frac{e_{\rm K} - \overline{e_{\rm K}}}{\overline{e_{\rm K}}}, \ s_{\rm 3} = \frac{e_{\rm 3} - \overline{e_{\rm 3}}}{\overline{e_{\rm 3}}}, \ s_{\rm c} = \frac{e_{\rm c} - \overline{e_{\rm c}}}{\overline{e_{\rm c}}},$$
 (3.12)

где $e_{\rm K}$, e_3 , $e_{\rm C}$ — средние значения сигналов от этих трех групп колбочек. Из формул (3.12) следует, что триады "колбочка — система горизонтальных клеток биполярная клетка" осуществляют такое преобразование сигналов, при котором распределения "яркостей" красного, зеленого и синего преобразуются в соответствующие распределения контрастов. Благодаря этому преобразованию достигается инвариантность к интенсивности освещенности наблюдаемой сцены, поскольку распределение контрастов не зависит от ее величины. Кроме того, это преобразование в значительной степени ослабляет влияние спектрального состава освещения сцены на сигналы $s_{\rm K}$, s_3 , $s_{\rm C}$.²

² Влияние спектрального состава освещения в рассматриваемой модели было бы сведено к нулю, если спектральные характеристики колбочек были бы бесконечно узкими, а число типов колбочек, различающихся спектральными характеристиками, и соответствующих им систем горизонтальных клеток бесконечно большим, так что весь спектр видимых излучений ими был бы перекрыт.

Дальнейшее преобразование сигналов в сетчатке заключается в перекодировании цветовой информации путем образования из сигналов $s_{\rm k}$, $s_{\rm 3}$, $s_{\rm c}$ ахроматических и двух групп цветоразностных сигналов в соответствии с равенствами

$$U_{a} = s_{\kappa} + s_{3} + s_{c}$$
, $U_{\kappa-3} = s_{\kappa} - s_{3}$, $U_{c-\kappa} = s_{c} - (s_{\kappa} + s_{3})$,

что после подстановки значений s_{κ} , s_{3} , s_{c} из формул (3.12) дает

$$U_{a} = \frac{e_{\kappa} - \overline{e_{\kappa}}}{\overline{e_{\kappa}}} + \frac{e_{3} - \overline{e_{3}}}{\overline{e_{3}}} + \frac{e_{c} - \overline{e_{c}}}{\overline{e_{c}}}, \qquad (3.13)$$

$$U_{\rm K-3} = \frac{e_{\rm K} - e_{\rm K}}{\overline{e_{\rm K}}} - \frac{e_{\rm 3} - e_{\rm 3}}{\overline{e_{\rm 3}}}, \qquad (3.14)$$

$$U_{\text{c-xc}} = \frac{e_{\text{c}} - \overline{e_{\text{c}}}}{\overline{e_{\text{c}}}} - \left(\frac{e_{\text{c}} - \overline{e_{\text{c}}}}{\overline{e_{\text{c}}}} + \frac{e_{3} - \overline{e_{3}}}{\overline{e_{3}}}\right).$$
(3.15)

В результате этого перекодирования по зрительному нерву передается информация уже не о распределении контрастов в "красных", "зеленых" и "синих" компонентах изображения сцены на сетчатке, а о распределении их алгебраических сумм. Благодаря этому обеспечивается дальнейшее снижение влияния спектрального состава освещения на восприятие цвета. Существенна при этом и близость спектральных характеристик "красной" и "зеленой" колбочек, благодаря чему изменение спектра освещения меньше влияет на разность сигналов. В литературе имеются сведения, что это перекодирование осуществляется вследствие взаимодействия биполярных, амакриновых и ганглиозных клеток, при этом ряд данных свидетельствует о том, что основную роль в передаче информации о цвете играет красно-зеленый компонент изображения и вторичную — синежелтый.

Рассмотренная модель предсказывает ряд иллюзий, связанных с восприятием цвета, и в том числе явление так называемого одновременного цветового контраста, который проявляется, например, в феномене окрашенной тени. Этот феномен можно наблюдать, если на белый экран, равномерно подсвеченный белым цветом, направить яркий пучок лучей красного цвета, на пути которого установить небольшой непрозрачный предмет, как показано на рис. 3.13 и ЦВ.2. Зрителю будет казаться, что тень, отбрасывае-



Рис. 3.13. К пояснению феномена окрашенной тени

мая на экран непрозрачным предметом, окрашена в зеленый цвет. Объяснение этому явлению дает формула (3.14). Из формулы следует, что зеленому цвету соответствуют отрицательные значения сигнала $U_{\rm k-3}$, красному — положительные. Так как локальный контраст, определяемый значением сигнала, измеряется относительно среднего значения, то недостаток красного в области тени проявляется в том, что первая дробь формулы (3.14) становится отрицательной, что для зрительной системы эквивалентно увеличению абсолютного значения второй дроби, которое имеет место при увеличении зеленого. Аналогичным образом объясняется явление последовательного цветового контраста.

3.8. Восприятие объема

Об удаленности наблюдаемого объекта зрительная система человека может судить по ряду эффектов, сопровождающих процесс наблюдения.

Первым и, по-видимому, главным эффектом, является эффект уменьшения угловых размеров наблюдаемого объекта по мере его удаления. Если нам известны истинные линейные размеры объекта, а также его угловые размеры, то не составляет труда определить его удаление от точки наблюдения. Примерами, иллюстрирующими это положение, может служить просмотр кинофильмов, телевизионных передач, фотографий и картин, написанных с соблюдением законов перспективы. При наблюдении этих изображений у нас не возникает проблем с оценкой расстояния объектов от точки наблюдения, хотя сами изображения являются двумерными. Большую роль при этих наблюдениях играет также априорная информация о наблюдаемых объектах, которой располагает наблюдатель. Так, например, при рассматривании фотографии, мы без труда определяем удаленность изображенного на ней даже незнакомого объекта, если нам известно, что он находится на земле, ориентируясь при этом лишь на удаленность его изображения вниз от линии горизонта. Расстояние до незнакомого нам объекта легко определяется и в том случае, когда он находится в окружении объектов, расстояние до которых нам известно, и известно, что все они находятся на одном уровне, например, на земле.

Вторым эффектом, который обеспечивает восприятие объема, является *стереоэф*фект. Существо стереоэффекта поясняется рис. 3.14.



Рис. 3.14. К пояснению стереоэффекта

На этом рисунке символом A обозначен точечный объект, который наблюдается с расстояния D. Глаза наблюдателя (правый и левый), расстояние между которыми (база) равно B, условно показаны в виде окружностей. Из рисунка следует, что когда наблюдатель смотрит в направлении объекта, проекции объекта на сетчатках глаз оказываются взаимно смещенными. Обусловлено это тем, что углы между направлениями на объект и нормалью к базе для каждого из глаз отличны от нуля и имеют противоположные знаки. В рассматриваемом случае $\alpha = -\beta$. Чем ближе к наблюдателью, тем больше будет взаимное несовпадение проекций объектов на сетчатках глаз, которое называется ducnapaнmностью. Благодаря наличию диспарантности воспринимается удаленность объекта от точки наблюдения.

Третьим эффектом, обеспечивающим восприятие объема наблюдаемой сцены, является *рельефность* составляющих ее объектов, обусловленная тем, что отражающие поверхности этих объектов в зависимости от их угловой ориентации поразному отражают падающий на них свет. Благодаря этому эффекту мы легко отличаем изображение цилиндрической колонны от изображения колонны, имеющей прямоугольное сечение. Как будет показано в *разд. 10.11*, для диффузно отражающих поверхностей по их трем изображениям, полученным при трех разных положениях источника освещения, можно получить их трехмерное изображение, т. е. составить полное представление об их протяженности в глубину³.

Восприятие объема наблюдаемой сцены обеспечивается также эффектом экранирования далеко расположенных объектов объектами, которые расположены ближе к наблюдателю.

Помимо перечисленных имеют также место и другие, более второстепенные эффекты, способствующие восприятию объема, как например, атмосферная дымка, удаленность от источников освещения и т. д.

3.9. Обнаружение и опознавание изображений объектов в пороговых условиях наблюдения

В пороговых условиях опознавания известных объектов лимитирующим фактором является шум. Это может быть внутренний шум зрительной системы, если наблюдаемое изображение имеет очень малый контраст, шум светового потока,

³ Более того, как это следует из формул, приведенных в этом разделе, имеющаяся априорная информация о наблюдаемом объекте может снизить это требование с трех источников света до двух источников света, а если еще априорно известны значения коэффициента отражения в каждой точке объекта, то и до одного источника света. Зрительная система человека, судя по всему, широко использует это обстоятельство, благодаря чему мы легко воспринимаем изображенные на фотографии трехмерные объекты как объемные и рельефные даже при наличии одного-единственного источника света. Существенно заметить также, что чувствительность нашего зрения к восприятию погрешности представления размеров наблюдаемого объекта по глубине много ниже чувствительности восприятия погрешности представления его размеров по двум другим координатам. Толерантность зрительной системы к погрешности, с которой воспринимается координата глубины объекта, делает распределение светотени в его изображении (в проекции его на сетчатки глаз), по-видимому, основным источником информации о координате глубины объектов.

если наблюдение ведется при низкой освещенности, шум, наложенный на изображение при его передаче по каналу связи, и т. д. В процессе обработки изображения зрительной системой имеет место пространственная фильтрация этого шума.

В 1958 г. нами была выдвинута и опубликована гипотеза, согласно которой при обнаружении и опознавании объекта на изображении с априорно известными параметрами, зашумленном аддитивным белым гауссовым шумом, зрительная система осуществляет его пространственную фильтрацию, которая является согласованной [42]. В последующих наших и зарубежных работах эта гипотеза получила экспериментальное подтверждение [53, 54, 108]. В соответствии с этой гипотезой опознаваемое изображение L(x, y) сравнивается с эталонными изображениями $L_c(x, y, i)$, составляющими алфавит, и идентифицируется с тем эталоном, для которого

$$\overline{\left[L(x,y)-L_{c}(x,y,j)\right]^{2}}=\min,$$

т. е. выполняется правило

$$\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left[L(x,y)-L_{c}(x,y,j)\right]^{2}dxdy < \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\left[L(x,y)-L_{c}(x,y,i)\right]^{2}dxdy,$$

представляющее алгоритм Зигерта — Котельникова. Здесь i и j — номера эталонов. При этом i = 1, 2, 3, ..., N, но $i \neq j$.

В том случае, когда часть параметров опознаваемого изображения априорно неизвестна, они находятся путем перебора. Например, если априорно неизвестно положение объекта на плоскости изображения, т. е. неизвестны его координаты x и y, то правило принятия решения принимает вид

$$\left\{ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[L(x, y) - L_{c}(x - \xi, y - \eta, j) \right]^{2} dx dy \right\}_{\min} < \left\{ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[L(x, y) - L_{c}(x - \xi, y - \eta, i) \right]^{2} dx dy \right\}_{\min},$$

где ξ и η — переменные, которые подбираются для каждого из интегралов таким образом, чтобы его значение было минимальным. Поскольку задачу обнаружения можно рассматривать как частный случай задачи опознавания, то все сказанное выше остается справедливым и для задачи обнаружения [48, 64].

3.10. Функциональные модели зрительной системы

При проектировании систем обработки и передачи изображений, а также оптимизации их параметров отсутствие критерия качества воспроизведения изображения (формализованного критерия видности искажений на изображениях), функционально связанного с электрооптическими характеристиками разрабатываемых систем, вызывает серьезные трудности.

В настоящее время, как уже было отмечено, для оценки качества воспроизведения изображений на экране монитора компьютера или на экране телевизора используется либо метод экспертных оценок, либо набор так называемых частных критериев, к числу которых относятся: разрешающая способность, отношение "сигнал/шум", число воспроизводимых градаций яркости, погрешности воспроизведения цвета и т. д. Значения частных критериев определяются параметрами системы и могут быть вычислены. Недостатком первого метода является отсутствие математических выражений, устанавливающих связь между оценкой и электрооптическими характеристиками аппаратуры, недостатком второго — невозможность сравнивать системы, которые различаются между собой по разным частным критериям. Так, например, если система А имеет большую разрешающую способность, но меньшее отношение сигнала к шуму, чем система В, то, не опираясь на опыт, невозможно решить, какая из двух сравниваемых систем воспроизводит более качественное изображение. Из приведенных рассуждений следует, что метод экспертных оценок не может быть использован при аналитическом решении оптимизационных задач, а метод частных критериев — при решении оптимизационных задач, в которых имеют место обменные операции между отдельными частными критериями качества, как например, в известной задаче апертурной коррекции при наличии шума.

Попытки создать обобщенный критерий качества воспроизведения изображений путем комбинирования частных критериев, а также другие эвристические подходы к решению этой задачи, как известно, пока не увенчались успехом (*см., например*, [6]).

Более обнадеживающие результаты при создании критериев качества воспроизведения изображений были получены путем разработки и применения для этих целей функциональных моделей зрительной системы человека, позволивших при оценке качества учитывать свойства зрения. В качестве примера на рис. 3.15 приведена одна из первых таких моделей, предложенная Ч. Холлом и Е. Холлом [131].



Рис. 3.15. Функциональная модель зрения, предложенная Ч. Холлом и Е. Холлом

Первое звено этой модели согласно определению авторов представляет собой фильтр нижних пространственных частот и описывает линейные преобразования изображения в оптической части глаз. Второе — поточечное логарифмическое преобразование яркости в величину, пропорциональную ее ощущению. И, наконец, третье звено модели — фильтр верхних пространственных частот — учитывает линейные преобразования при формировании так называемого нейронного изображения. При этом, как отмечают авторы модели, совместное действие фильтров нижних и верхних пространственных частот приводит к усилению в нейронном изображении спектральных составляющих средних пространственных частот. Несмотря на наивность представлений о характеристиках зрительной системы, положенных в основу этой модели, она позволяла получать достаточно надежные оценки видности флуктуационного шума на изображении путем вычисления среднего квадрата разности между зашумленным и не зашумленным компонентами нейронных изображений на ее выходе.

Серьезным недостатком рассмотренной модели, а также других аналогичных, функциональных моделей зрения, является то, что все они позволяют получать более или менее надежные оценки видности искажений в случае, когда присутствует только один вид искажений, например, флуктуационный шум.

В том случае, когда имеется два или большее количество одновременно действующих искажающих факторов, оценки, полученные с использованием этих моделей, сильно расходятся с оценками зрителей. Объясняется это тем, что степень влияния на ухудшение качества воспроизведения изображения сильно зависит от вида искажающего фактора. Так, например, при одной и той же величине среднего квадрата разности эталонного и искаженного изображения, искажения, обусловленные размытием световых границ, более заметны, чем искажения, обусловленные квазибелым гауссовым шумом.

Для устранения этого недостатка было предложено при вычислении среднего квадрата результирующей оценки σ_{Σ}^2 (среднего квадрата разности эталонного и искаженного изображений) различные искажающие факторы учитывать с различными весовыми коэффициентами, например,

$$\sigma_{\Sigma}^2 = \lambda \sigma_{\rm c}^2 + \sigma_{\rm III}^2 \,,$$

здесь σ_c^2 — средний квадрат компонента, обусловленного частотными искажениями, σ_m^2 — средний квадрат компонента, обусловленного флуктуационным шумом, λ — весовой коэффициент при компоненте σ_c^2 , при шумовом компоненте весовой коэффициент принят равным единице (более подробно этот подход рассмотрен в *разд. 5.6*). Однако это предложение не нашло сколько-нибудь широкого применения.

В дальнейшем был предложен ряд более детальных моделей зрения, сведения о которых читатель найдет в [125, 137], но и они позволяли получать надежные оценки только в случае, когда причиной искажений изображения являлся флуктуационный шум.

3.11. Обобщенная функциональная модель зрительной системы

Проведенный анализ причин несовершенства оценок видности на изображениях искажений, не связанных с флуктуационным шумом (речь идет об оценках, полученных путем использования предложенных функциональных моделей зрения), выявил общие для них недостатки, которые обусловлены тем, что ряд важных свойств зрения в этих моделях не учитывается. В результате этого анализа автором была предложена более совершенная обобщенная функциональная модель зрения человека [38, 138], свободная от этих недостатков, которая показана на рис. 3.16. При разработке обобщенной функциональной модели зрения в целях упрощения были приняты допущения, согласно которым считается, что сетчатка глаза обладает пространственной непрерывностью и инвариантностью. Принятие этих допущений позволило ввести в рассмотрение частотные характеристики отдельных звеньев зрительной системы. В модель также введены переключатели, которые могут занимать два положения. В первом положении переключателей в модель вводится (запоминается) эталонное неискаженное изображение, при втором положении переключателей в модель вводится оцениваемое изображение, которое в модели сравнивается с эталонным изображением, и измеряется степень их различия.



Рис. 3.16. Обобщенная функциональная модель зрения

Согласно рассматриваемой нами обобщенной функциональной модели зрения процесс наблюдения изображения и оценка качества его воспроизведения протекают следующим образом. Ось зрения, скачкообразно изменяя свое положение вследствие саккадических движений глаз, фиксируется вблизи контуров так, что в зоне ясного видения располагаются фрагменты контуров, подобные тем, что показаны на рис. 3.7. Поскольку одновременно все изображение в деталях зритель рассмотреть не может вследствие ограниченности зоны ясного видения, полное изображение наблюдаемой сцены формируется в его памяти последовательно во времени (на так называемом инвариантном экране детекторов в терминологии психологов). Так как искажения на наблюдаемом изображении, обусловленные помехами, потерей четкости и другими причинами, различимы только в зоне ясного видения, т. е. на фрагментах границ, попадающих в зону ясного видения, то и оценка их зрительной системой производится только на этих фрагментах.

Несовершенство оптики глазных сред при проецировании изображений на сетчатки глаз приводит к уменьшению его контраста, а также к ослаблению в нем верхних пространственных частот, которое в первом приближении может быть учтено посредством функции рассеяния точки $h_0(x, y)$ или связанной с ней через преобразование Фурье частотной передаточной функцией $K_0(\omega_x, \omega_y)$, где ω_x, ω_y — круговые пространственные частоты. Сетчатки глаз осуществляют цветоделение изображения с образование м трех цветоделенных составляющих $e_{\kappa}(x, y)$, $e_3(x, y)$, $e_c(x, y)$ (красной, зеленой и синей), представленных в сетчатках глаз в виде возбуждений рецепторов:

$$e_{\kappa}(x, y) = \int_{\lambda_{MHKC}}^{\lambda_{MHKC}} E_{\Omega}(x, y, \lambda) \varepsilon_{\kappa}(\lambda) d\lambda,$$
$$e_{3}(x, y) = \int_{\lambda_{MHH}}^{\lambda_{MHKC}} E_{\Omega}(x, y, \lambda) \varepsilon_{3}(\lambda) d\lambda,$$
$$e_{c}(x, y) = \int_{\lambda_{MHH}}^{\lambda_{MHKC}} E_{\Omega}(x, y, \lambda) \varepsilon_{c}(\lambda) d\lambda,$$

где $E_{\Omega}(x, y, \lambda)$ — распределение интенсивности света на сетчатке после фильтрации оптикой глаз исходного изображения $L(x, y, \lambda)$; $\varepsilon_{\kappa}(\lambda)$, $\varepsilon_{3}(\lambda)$, $\varepsilon_{c}(\lambda)$ — характеристики спектральной чувствительности "красных", "зеленых" и "синих" колбочек; λ — световая длина волны; $\lambda_{\text{мин}}$ и $\lambda_{\text{макс}}$ — границы светового диапазона длин волн. Далее в сетчатках глаз, как уже было рассмотрено в *разд. 3.7*, формируются яркостный (ахроматический) и два цветоразностных (красно-зеленый и синежелтый) компоненты изображений (сигналов):

$$U_{a}(x,y) = \frac{e_{\kappa}(x,y) - e_{\kappa}}{\overline{e_{\kappa}}} + \frac{e_{3}(x,y) - e_{3}}{\overline{e_{3}}} + \frac{e_{c}(x,y) - e_{c}}{\overline{e_{c}}}$$

$$U_{\mathrm{K-3}}(x,y) = \frac{e_{\mathrm{K}}(x,y) - \overline{e_{\mathrm{K}}}}{\overline{e_{\mathrm{K}}}} - \frac{e_{3}(x,y) - \overline{e_{3}}}{\overline{e_{3}}},$$
$$U_{\mathrm{c-w}}(x,y) = \frac{e_{\mathrm{c}}(x,y) - \overline{e_{\mathrm{c}}}}{\overline{e_{\mathrm{c}}}} - \left(\frac{e_{\mathrm{K}}(x,y) - \overline{e_{\mathrm{K}}}}{\overline{e_{\mathrm{K}}}} + \frac{e_{3}(x,y) - \overline{e_{3}}}{\overline{e_{3}}}\right)$$

Затем благодаря взаимодействию биполярных, амакриновых и ганглиозных клеток сетчатки эти компоненты подвергаются пространственной фильтрации. В результате этой фильтрации в спектре яркостного компонента за счет латерального торможения происходит небольшое ослабление низких пространственных частот, в результате которых возникает эффект полосок Маха, а спектры пространственных частот цветоразностных компонентов становятся приблизительно в три раза уже спектра яркостного компонента вследствие подавления их высокочастотных частей. Благодаря этому для зрителя останется незаметным, если и в самом наблюдаемом изображении эти части спектра будут отсутствовать. Факт сужения спектра цветоразностных компонентов при их обработке в сетчатке глаза позволил в совместимых системах цветного телевидения для передачи телевизионных цветоразностных сигналов использовать более узкие полосы частот, чем для передачи яркостного сигнала, а в цифровых системах телевидения передавать эти сигналы с более низкой частотой отсчетов без заметного влияния на качество воспроизведения изображений. Изображения, приведенные на рис. 3.17 и 3.18, а также на ЦВ.3 и ЦВ.4, иллюстрируют сказанное.

КИНО	КИНО	КИНО	КИНО	КИНО
КИНО	КИНО	KUHO	КИНО	КИНО a
КИНО	КИНО	КИНО	КИНО	КИНО
KIHO	КИНО	KIHO	КИНО	КИНО б
КИНО	КИНО	KNHO	KNHO	КИНО
KMHO	KINHO	MMHO	KHHO	KCHEO
КИНО	КИНО	КИНО	KNHO	КИНО
KNIHO		KNHO	KHHO	RHHO

Рис. 3.17. Здесь: *а* — исходное тестовое изображение; *б* — то же изображение, но с уменьшенной в три раза шириной спектров цветоразностных компонентов; *в* — то же изображение, но с уменьшенной в три раза шириной спектра яркостного компонента; *г* — то же изображение, но с уменьшенной в три раза шириной спектров яркостного и цветоразностных компонентов



а



в



б



Рис. 3.18. Здесь: *а* — исходное сюжетное изображение; *б* — то же изображение, но с уменьшенной в три раза шириной спектров цветоразностных компонентов; *в* — то же изображение, но с уменьшенной в три раза шириной спектра яркостного компонента; *г* — то же изображение, но с уменьшенной в три раза шириной спектров яркостного и цветоразностных компонентов

На рисунках хорошо видно, что сужение спектра цветоразностных компонентов практически не влияет на резкость границ, в то время как сужение спектра ахроматического компонента приводит к сильному падению резкости границ на воспроизводимых изображениях как тестовом, так и сюжетном. Свойства эквивалентных фильтров, описывающих обсуждаемые преобразования нейронных сигналов в сетчатке, могут быть учтены введением соответствующих функций рассеяния точки (профилей активности ганглиозных клеток в терминологии физиологов) $h_a(x, y)$, $h_{\kappa-3}(x, y)$, $h_{c-ж}(x, y)$. В результате перечисленных преобразований получаются три нейронных изображения наблюдаемого в данный момент фрагмента контура искаженного изображения:

$$B_{a}(x, y) = U_{a}(x, y) \otimes h_{a}(x, y),$$

$$B_{\kappa-3}(x, y) = U_{\kappa-3}(x, y) \otimes h_{\kappa-3}(x, y),$$

$$B_{c-m}(x, y) = U_{c-m}(x, y) \otimes h_{c-m}(x, y),$$

где ⊗— символ свертки.

Оценка качества воспроизведения изображения, как уже отмечалось, заключается в сравнивании оцениваемого изображения с неискаженным эталонным изображением, нейронные версии ахроматического и цветоразностных компонентов которого $B_{3,a}(x, y)$, $B_{3, K-3}(x, y)$, $B_{3, C-ж}(x, y)$ находятся к моменту сравнения в зрительной памяти. При сравнении вычисляются средние квадраты разностей

$$\varepsilon_{a}^{2} = \overline{\left[B_{a}(x, y) - B_{3, a}(x, y)\right]^{2}},$$

$$\varepsilon_{K-3}^{2} = \overline{\left[B_{K-3}(x, y) - B_{3, K-3}(x, y)\right]^{2}},$$

$$\varepsilon_{c-ж}^{2} = \overline{\left[B_{c-w}(x, y) - B_{3, c-w}(x, y)\right]^{2}},$$

которые комбинируются в результирующую оценку

$$\varepsilon_{\Sigma}^2 = \varepsilon_a^2 + \varepsilon_{\kappa-3}^2 + \varepsilon_{c-\pi}^2.$$

Обратим внимание, что оценка ε_{Σ}^2 вычисляется в зоне ясного видения, определяемой размерами центральной части сетчатки (фовеолы), равной примерно 20 угловым минутам. Эта оценка инвариантна к яркости наблюдаемого изображения, но зависит от расстояния, с которого оно рассматривается, что вполне естественно, поскольку с увеличением расстояния заметность ряда видов искажений уменьшается.

В качестве примера рассмотрим использование модели для оценки видности искажений, обусловленных аддитивным флуктуационным шумом на ахроматическом телевизионном изображении. Поскольку изображение ахроматическое, то компоненты $B_{\kappa-3}(x, y)$ и $B_{c-ж}(x, y)$ будут тождественно равны нулю, а следовательно,

$$\varepsilon_{\Sigma}^{2} = \overline{\left[B_{a}(x, y) - B_{\mathfrak{H}, a}(x, y)\right]^{2}}.$$

Из структуры модели следует, что нейронные изображения $B_{a}(x, y)$ и $B_{3, a}(x, y)$ связаны с соответствующими им оптическими изображениями через частотную

передаточную функцию $K_{\Sigma}(\omega_x, \omega_y)$, которая представляет собой произведение частотной передаточной функции оптической части глаза $K_o(\omega_x, \omega_y)$ на частотную передаточную функцию $K_a(\omega_x, \omega_y)$, описывающую фильтрацию ахроматического компонента, обусловленную взаимодействием биполярных, амакриновых и ганглиозных клеток сетчатки

$$K_{\Sigma}(\omega_{x},\omega_{y})=K_{0}(\omega_{x},\omega_{y})K_{a}(\omega_{x},\omega_{y}),$$

где

$$K_{a}(\omega_{x}, \omega_{y}) = \int_{-\infty}^{\infty} h_{a}(x, y) e^{-i(\omega_{x}x + \omega_{y}y)} dx dy.$$

Так как оптическое изображение, искаженное шумом L(x, y), отличается от эталонного $L_{c}(x, y)$ лишь шумовым компонентом $L_{u}(x, y)$, то и нейронные изображения $B_{a}(x, y)$ и $B_{3,a}(x, y)$ будут различаться между собой тоже только компонентом шума в месте их сравнения. Таким образом, оценка ε_{Σ}^{2} представляет собой не что иное, как средний квадрат флуктуационного шума в месте, где сравниваются искаженное и эталонное изображения. Зная спектральную интенсивность $S_{ul}(\omega_x, \omega_y)$ шума, наложенного на оптическое изображение, и частотную передаточную функцию, описывающую фильтрацию ахроматического изображения $K_{\Sigma}(\omega_x, \omega_y)$ в зрительной системе, не составляет труда вычислить оценку ε_{Σ}^{2} :

$$\varepsilon_{\Sigma}^{2} = \frac{k_{1}}{4\pi^{2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S_{\mathrm{III}}(\omega_{x}, \omega_{y}) \left| K_{\Sigma}(\omega_{x}, \omega_{y}) \right|^{2} d\omega_{x} d\omega_{y},$$

где k_1 — постоянный коэффициент. В аналоговой телевизионной системе спектральная интенсивность шума не зависит от пространственной частоты ω_y , т. е. имеет вид $S_{\rm m}(\omega_x)$, поэтому формула для ϵ_{Σ}^2 запишется как

$$\varepsilon_{\Sigma}^{2} = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{\text{III}}(\omega_{x}) A_{1}(\omega_{x}) d\omega_{x},$$

где

$$A_{1}(\omega_{x}) = \frac{k_{1}}{2\pi} \int_{-\infty}^{\infty} \left| K_{\Sigma}(\omega_{x}, \omega_{y}) \right|^{2} d\omega_{y}.$$

Выражая пространственную круговую частоту через "временную" частоту f в герцах, а спектральную интенсивность шума $S_{\rm m}(\omega_x)$ через энергетический спектр $F_{\rm m}(f)$, запишем окончательно

$$\varepsilon_{\Sigma}^{2} = \int_{0}^{\infty} F_{\rm III}(f) A_{0}(f) df , \qquad (3.16)$$

где $A_0(f) = A_1(\omega_x)b$ — весовая функция, а *b* — постоянный коэффициент, согласующий размерности.

Оценка видности шума на телевизионном изображении путем измерения или расчета его взвешенного значения в соответствии с (3.16) была предложена Шейдом как чисто эмпирический метод. Необходимость такой оценки была вызвана тем, что видность шума на телевизионном изображении сильно зависит от его спектрального состава. Так, например, видность шума с равномерным распределением энергии по спектру частот и шума, энергия которого сосредоточена в области высоких частот, возрастая пропорционально квадрату частоты (шум с треугольным спектром), в зависимости от условий наблюдения может различаться до 3 раз. На рис. 3.19 показана зависимость $A_0(f)$.

Достоинством рассмотренной модели по сравнению с ранее предложенными моделями является то, что с ее помощью можно получать надежные оценки заметности искажений на цветных изображениях, обусловленных как флуктуационным шумом, так и искажений, обусловленных размытием световых и цветовых границ вследствие амплитудно-частотных и фазочастотных искажений.



Рис. 3.19. Весовая функция

Вместе с тем эта модель, как и другие, о которых речь шла ранее, не позволяет оценивать видность (заметность) геометрических искажений, обусловленных, например, несовершенством оптики, искажений текстурного компонента изображений, искажений, обусловленных нелинейными искажениями сигнала изображения, и ряда других искажений.

Этот недостаток рассмотренной модели, как и других, предложенных ранее, обусловлен тем, что она представляет только те процессы, которые протекают в первичных отделах зрительной системы, не учитывая преобразования сигналов и извлечения из них информации о наблюдаемой сцене, имеющих место в ее высших отделах. Для того чтобы лучше понять эту проблему, рассмотрим особенности восприятия семантических изображений зрительной системой человека.

3.12. Особенности восприятия семантических изображений зрительной системой человека

Переходя к рассмотрению восприятия семантических изображений зрительной системой человека, обратим внимание на ее следующую особенность. При работе зрительная система использует не всю информацию, которая содержится в изображении наблюдаемой сцены, проецируемом на сетчатки глаз, а только незначительную, но важную ее часть, которая необходима для описания этой сцены. Проиллюстрируем это примером.

Так для представления десятиминутного цветного видеоролика при параметрах разложения 720×576 пикселов и 25 кадрах в секунду требуется цифровой поток около 249×10^6 бит/с. Вместе с тем согласно многочисленным исследованиям скорость зрительного восприятия не превышает 50—70 бит/с [22, 170]. Различие в приведенных значениях цифровых потоков составляет $3,5 \times 10^6 - 4,9 \times 10^6$ раз, но несмотря на это, мы можем достаточно подробно описать сюжет, представленный в видеоролике.

Многочисленные исследования показали, что наиболее важную информацию об изображениях объектов содержат их контуры. Другие компоненты изображения, такие как, например, градиентные или текстурные, играют меньшую роль. Представим изображение в виде композиции контурных, градиентных и фактурных компонентов.

 Под контурными компонентами (контурами) будем понимать границы на изображении, при пересечении которых значения яркости и/или цвета изменяются скачком. Контуры изображений объектов представляют собой проекции их оболочек, т. е. поверхностей, отделяющих объекты от окружающего пространства. Отсюда следует, что формы контуров определяются только формами оболочек самих объектов и ракурсами, под которым они рассматриваются, и не зависят ни от расположения, ни от интенсивности, ни от спектрального состава источников света, освещающих наблюдаемую сцену. Константность, а следовательно, и информативность контурных компонентов по этой причине велика. Этим можно объяснить тот хорошо известный факт, что зрители легко узнают изображаемые объекты по контурному рисунку. Сказанное иллюстрируется рис. 3.20.

Контурные компоненты изображений обладают двумя важными свойствами.

- Поскольку между оболочками реальных объектов, составляющих сцену, и окружающим их пространством за редкими исключениями имеются резкие границы, то их проекции, т. е. изображения объектов, также имеют четкие световые и цветовые границы (контуры), отделяющие их от фона, на котором они наблюдаются. Размытие этих границ в реальных изображениях, с которым приходится встречаться на практике, как правило, обусловлено несовершенством аппаратуры или ее плохой настройкой. Поэтому обработка изображений практически любых объектов, заключающаяся в замене размытых границ резкими границами, будет восприниматься зрительной системой как улучшение качества их воспроизведения.
- Как правило, оболочки реальных объектов, за исключением отдельных изломов, носят гладкий, не зазубренный характер. Из этого следует, что контуры в изображениях также в основном являются гладкими.

Контурные компоненты играют определяющую роль при опознавании изображений объектов, составляющих сцену, а также при определении их взаимного положения зрительной системой. Исследования зрительной коры высокоорганизованных животных показали, что в ней имеются так называемые сверхсложные рецептивные поля, посредством которых реализуются механизмы выделения элементов контуров, таких как углы, дуги и т. д. Эти механизмы иногда называют детекторами.

Важность контуров подтверждается исследованиями Ярбуса, который показал, что зрительная ось при наблюдении изображений фиксируется преимущественно в районе контуров.



Рис. 3.20. К пояснению важности контурных компонентов

- 2. Под градиентными компонентами понимают области изображения ограниченные контурами, каждая точка которых представляет значение сглаженной, плавно изменяющейся яркости и/или цвета. Константность этих компонентов ниже, чем константность контурных компонентов. Объясняется это тем, что распределение яркости и/или цвета в них зависит не только от коэффициентов отражения поверхностей объекта, являющихся свойствами самого объекта, но также от взаимного расположения наблюдаемого объекта и источников света, а также от спектрального состава источников света, которые могут сильно варьироваться. По этой причине при опознавании зрительной системой изображений объектов, составляющих сцену, градиентные компоненты играют меньшую роль.
- 3. Под фактурными компонентами будем понимать текстуры, из которых предварительно удалены и отнесены к контурным компонентам все световые и цветовые границы, если они в них имеются. Поясним сказанное примером текстуры, на которой изображена кирпичная кладка. В нашем определении эта текстура разделяется на контурный компонент, на котором изображены границы между кирпичами и фактурные компонент, представляющий неоднородность поверхности кирпичей. Фактурные компоненты носят шумовой характер и характеризуют шероховатость поверхностей объектов. Согласно гипотезе Юлеша зрительная система легко различает два шумовых рисунка только в том случае, если они отличаются между собой плотностями вероятности первого порядка или спектральными интенсивностями (энергетическими спектрами) при одинаковых плотностях вероятностей первого порядка. В противном случае для их различения требуется поэлементное сравнение рисунков [85].

Примечание

В первом приближении можно считать, что при определении шероховатости наблюдаемых объектов зрительная система использует спектральные интенсивности и, в меньшей степени, распределения плотностей вероятностей первого порядка фактурных компонентов их изображений [60, 141].

При наблюдении сцен и их изображений зрительная система человека по-разному реагирует на искажения контурных, градиентных и фактурных компонентов. При этом зрительная система "из опыта знает", какими могут и должны быть контурные, градиентные и фактурные компоненты изображения, представляющие ту или иную часть объекта. Если эти компоненты отличаются от тех, какими они должны быть, это воспринимается как искажение изображения.

Рассмотрим заметность на изображении искажений контурных, градиентных и фактурных компонентов.

- Поскольку контурные компоненты играют определяющую роль при восприятии изображений, то и искажения, заключающиеся:
 - в появлении новых или исчезновении имеющихся контуров на изображении при его преобразовании, например, в возникновении ложных контуров при квантовании изображения по яркости;



Рис. 3.21. Изображения: *а* — исходное; *б* — искаженное. Искаженное изображение получено из исходного путем его квантования по яркости на 16 уровней. Среднеквадратичная разность изображений (*a*) и (*б*), нормированная на величину динамического диапазона, занимаемого изображением,

составляет
$$\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{HCK}(x, y)\right]^{2}}}{255} = 0,0508$$





Рис. 3.22. Изображения: *а* — исходное; *б* — искаженное. Искаженное изображение получено из неискаженного путем внесения в один из его контурных компонентов локального геометрического искажения. Среднеквадратичная разность изображений (*a*) и (*б*), нормированная на величину динамического диапазона,

занимаемого изображением, составляет $\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{\mu c \kappa}(x, y)\right]^{2}}}{255} = 0,0505$



Рис. 3.23. Изображения: *а* — исходное; *б* — искаженное. Искаженное изображение было получено из неискаженного путем его фильтрации посредством низкочастотного фильтра, импульсная характеристика которого описывается гауссовой функцией. Среднеквадратичная разность изображений (*a*) и (*б*), нормированная на величину динамического диапазона, занимаемого изображением,

составляет
$$\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{HCK}(x, y)\right]^{2}}}{255} = 0,0503$$





Рис. 3.24. Изображения: *а* — исходное; *б* — искаженное. Искаженное изображение получено из исходного путем изменения его контраста. Среднеквадратичная разность изображений (*a*) и (*б*), нормированная на величину динамического диапазона,

занимаемого изображением, составляет $\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{HCK}(x, y)\right]^{2}}}{255} = 0,0507$



Рис. 3.25. Две различные реализации фактуры с идентичными статистическими характеристиками (белый гауссов шум). Среднеквадратичная разность изображений (*a*) и (*б*), нормированная на величину динамического диапазона,

составляет
$$\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{HCK}(x, y)\right]^{2}}}{255} = 0,2841$$



а



б

Рис. 3.26. Изображения: *а* — исходное; *б* — искаженное. Искаженное изображение было получено из неискаженного путем добавления к нему квазибелого гауссова шума. Среднеквадратичная разность изображений (*a*) и (*б*), нормированная на величину динамического диапазона, занимаемого изображением,

составляет
$$\sigma = \frac{\sqrt{\left[L_{c}(x, y) - L_{\mu c \kappa}(x, y)\right]^{2}}}{255} = 0,05$$

- в изменении формы контуров на изображении, например, в результате геометрических искажений;
- в изменении положения контуров на изображении при его преобразовании;
- в размытии контуров на изображении при его преобразовании и т. д.,

наиболее заметны зрителю.

На рис. 3.21—3.23 приведены некоторые примеры изображений, на которых искажены контурные компоненты. Искажения на рис. 3.21 заключаются в наличии ложных контуров, обусловленных квантованием изображения на недостаточно большое количество уровней. Ложные контуры особенно заметны на поверхностях, где яркость плавно изменяется — на плече и лице. На рис. 3.22 искажен контур полей шляпы. Искажения на рис. 3.23 обусловлены размытостью световых границ.

- Искажения градиентных компонентов изображений нелокального характера, которые могут быть обусловлены, например, нелинейностью световой характеристики при контрастировании изображения, воспринимаются зрителем как изменения условий освещения, а не как искажения, при этом небольшие искажения зритель не замечает. Рисунок 3.24 иллюстрирует это утверждение.
- Заметность искажений фактурных компонентов в сильной степени зависит от их вида. Так, например, замена одной реализации фактуры на другую с теми же статистическими характеристиками для зрителя остается практически незаметной, если только он не начинает поэлементно сравнивать изображения. Рисунок 3.25 подтверждает это утверждение. В то же время искажения, приводящие к изменению спектральной интенсивности (энергетического спектра) фактуры или к изменению ее плотности вероятностей первого порядка, зрителем будут замечены.

Появление гауссова шума на изображении воспринимается как наложение на него текстуры, и если ее в данном месте изображения не должно быть, то это будет воспринято как искажение. Рисунок 3.26 иллюстрирует этот случай.

Примечание

Сравнение изображений, показанных на рис. 3.21—3.26, искажения которых характеризуются практически одной и той же величиной среднеквадратичной нормированной разности σ , показывает, что заметность искажений приведенных изображений не одинакова и в сильной степени зависит от их характера. Другими словами, в данном случае σ является плохим критерием качества воспроизведения изображений.

Из изложенного непосредственно следует, что при воспроизведении изображения, для того чтобы оно было оценено зрительной системой как совершенное, необходимо с малой погрешностью воспроизводить положения и резкость световых границ, в меньшей степени это относится к точности воспроизведения фактурных компонентов и еще в меньшей степени к точности воспроизведения градиентных компонентов. Рассмотренный подход к структуре изображений подсказывает естественный путь для разработки методики оценки качества их воспроизведения. Особенностью этой методики должен быть учет различной "чувствительности" зрения к искажениям контурных, градиентных и фактурных компонентов, о чем было сказано ранее.

3.13. Методы оценки качества воспроизведения изображений

Как уже было сказано ранее, в настоящее время отсутствует критерий оценки качества воспроизведения изображений, посредством которого можно было бы оценивать результирующий уровень заметности искажений изображений при воздействии ряда искажающих факторов. Поэтому для определения качества воспроизведения изображения применяется *метод экспертных оценок*. Этот метод заключается в том, что наблюдателю вначале предъявляется эталонное изображение, качество которого принимается равным высшей оценке используемой шкалы, а затем изображение, которое он должен оценить. Метод экспертных оценок широко используется в телевидении. Он стандартизован для субъективных экспертиз в документах Международного консультативного комитета по радио (МККР). Несмотря на то, что применяются 5-балльные шкалы оценок, опыт показывает, что они не достаточно подробны. В связи с этим высказываются мнения о целесообразности использования 5-балльной шкалы со ступенями 1/2 и даже 1/4 балла.

В ходе экспертиз получают обычно большое число оценок, поэтому вопрос рационального представления результатов имеет важное значение. Современные методы обработки результатов экспертиз заключаются в том, что находится средняя оценка функции от искажений, а разброс результатов оценок наблюдателей выражается среднеквадратичным отклонением. Развитие этого метода в значительной степени было определено работами Проссера, Аллнатта и Левиса, а также работами других авторов.

При обработке результатов измерений, полученных при использовании 5-балльной шкалы, средняя оценка в баллах определяется следующим образом

$$g_{\rm cp} = \frac{1}{N} \sum_{i=1}^5 i n_i,$$

где N — полное количество оценок, n_i — количество оценок в i — баллов. Пятибалльные оценки g удобно преобразовать в нормализованные оценки p, занимающие интервал в пределах от нуля до единицы

$$p=(g-1)/4.$$

В этой методике оценки качества воспроизведения изображений используют такое понятие, как единица ухудшения качества, выражаемая в импах (от англ. *impairment* — ухудшение). Ухудшение I и оценка качества связаны соотношением

$$I = 1/p - 1$$
.

При изменении качества от нуля до единицы ухудшение изменяется от бесконечности до нуля. Авторы этой методики исходили из предположения, что ухудшения качества воспроизведения изображения I_i , обусловленные воздействием различных факторов, суммируются, при этом величина результирующего ухудшения I_{pes} , выраженная в импах, равна

$$I_{\text{pe3}} = \sum_{i=1}^{n} I_i ,$$

где n — число различных видов искажений, приводящих к ухудшениям качества воспроизведения. В соответствии с этим была разработана методика, устанавливающая аналитическую зависимость между величиной ухудшения I_i и величиной искажений, обусловленных воздействием какого-либо одного фактора, например, уровня шума, величины эхо-сигнала и т. п.

$$I_i = e^{-Q_i} ,$$

где *Q* — некоторая функция искажений. Зависимости ухудшений от величины искажений называются характеристиками ухудшений. Эти характеристики были измерены и опубликованы.

Дальнейшие исследования показали, что при изменении p в пределах от 0 до 1 между экспериментальными и расчетными данными имеет место расхождение, при этом среднеквадратичная ошибка расчетных данных составляет 0,083—0,13. В целях увеличения точности расчетов было предложено результирующее значение ухудшений определять путем квадратичного суммирования частных ухудшений

$$I_{\text{pes}}^2 = \sum_{i=1}^n I_i^2.$$

Переход к квадратичному суммированию ухудшений действительно позволил повысить точность расчетов в области высоких оценок, среднеквадратичная погрешность при этом уменьшилась до величины 0,028-0,033, однако в области низких значений p это привело к ее увеличению до величины 0,14-0,28, т. е. до четверти шкалы, что недопустимо.

В настоящее время в области создания оценок качества продолжаются интенсивные исследования. Наиболее перспективным направлением таких исследований, как нам представляется, является определение видности результирующих искажений на основе более глубокого учета свойств зрения в результате разработки и использования более совершенных функциональных моделей зрения.

3.14. Корреляционно-энергетический критерий качества изображения, кодированного по стандарту MPEG-2

Развитие и внедрение цифрового телевизионного вещания в рамках стандарта сжатия MPEG-2 привело к необходимости оценивать качество воспроизведения изображения, а следовательно, к необходимости разработки подходящего для этой цели критерия качества. В результате проведенных исследований был разработан так называемый *корреляционно-энергетический критерий* качества изображения [19, 20, 109, 112, 126, 132, 176].

Как известно, основным источником искажений изображений (артефактов) в цифровом телевидении при использовании стандарта сжатия МРЕС-2 является кодер, который вносит так называемый шум преобразования. Шум преобразования включает в себя более десяти различных компонентов. В результате проведенного анализа авторами работы были выделены, так называемые, представительные артефакты (компоненты шума преобразования), которые не являются частными случаями или комбинациями других компонентов шума преобразования. К этим представительным артефактам были отнесены: блочная структура, размытие изображения, шум базисных функций, вызванный квантованием коэффициентов дискретного косинусного преобразования (ДКП) и разрушение границ.⁴ Авторы исследования отмечают, что между представительными артефактами имеет место сильная корреляционная связь, и поэтому предлагают в дальнейшем производить оценку качества изображения по одному из них, а именно по заметности блочной структуры. Этот прием является одним из ключевых, поскольку благодаря его применению векторный критерий качества сводится к скалярному критерию. Выбор заметности блочной структуры в качестве используемого в дальнейшем артефакта при оценке качества изображения обусловлен тем, что этот вид помехи преобразования наиболее заметен.

Другим интересным предложением, использованным при разработке описываемого критерия качества, является выбор опорного изображения. Как известно, любой процесс измерения включает в себя сравнение измеряемой величины с эталоном. При оценке качества изображения оно обычно сравнивается с эталоном (неискаженным изображением) и в результате оценки определяется, насколько оно отличается от эталона, другими словами, как далеко от него отстоит. Необходимость иметь эталонное движущееся изображение в месте, где производится измерение, в ряде случаев составляет серьезную проблему, особенно если необходимо постоянно осуществлять текущий контроль телевизионной передачи в месте приема. Авторы описываемого критерия предложили оригинальное решение, заключающееся в том, что при оценке качества воспроизведения определяется расстояние оцениваемого изображения не от эталонного изображения, т. е. от наилучшего изображения, а определяется расстояние от наихудшего изображения, которое только возможно

⁴ О методе сжатия MPEG-2 и дискретном косинусном преобразовании см. в главе 13.

в рассматриваемом случае. В качестве наихудшего изображения было предложено использовать так называемое *D*-изображение, которое можно легко получить из принятого изображения. Для получения *D*-изображения достаточно в каждом блоке принятого изображения все коэффициенты ДКП кроме коэффициента, соответствующего средней составляющей, сделать равными нулю. Другими словами, предлагается оценивать, не насколько принятое изображение хуже наилучшего, а насколько оно лучше наихудшего.

Алгоритм оценки включает в себя следующие действия:

- 1. Вначале из принятого изображения изготавливается *D*-изображение путем обнуления коэффициентов ДКП, всех кроме одного.
- 2. Далее оба изображения и оцениваемое, и *D*-изображение преобразуются из цветового пространства RGB в равноконтрастное цветовое пространство LUV.
- 3. Затем для каждого из двух изображений изготавливаются контурные препараты, которые авторы называют картами градиентов

$$G_{ij} = \sqrt{\Delta X^2 + \Delta Y^2}$$

где

$$\Delta X = \sqrt{\left(L_{ij} - L_{ij-1}\right)^2 + \left(u_{ij} - u_{ij-1}\right)^2 + \left(v_{ij} - v_{ij-1}\right)^2},$$

$$\Delta Y = \sqrt{\left(L_{ij} - L_{i-1j}\right)^2 + \left(u_{ij} - u_{i-1j}\right)^2 + \left(v_{ij} - v_{i-1j}\right)^2}.$$

- 4. Далее выполняется сегментация, в результате которой каждый блок коэффициентов ДКП относится к одному из типов: либо к контурному компоненту, либо к градиентному компоненту, либо к фактурному компоненту. Для разделения изображения на сегменты используется гистограмма градиентов. Те блоки, для которых величина градиента невелика и не превышает порог, ограничивающий 75% площади гистограммы, относят к градиентному компоненту. Остальные блоки относят к контурному или фактурному компоненту в зависимости от типов соседних блоков.
- 5. После этого для каждого блока вычисляется корреляционно-энергетическое отношение

$$E_{k,l} = \frac{e_E}{e_E + e_I} r ,$$

где e_E — энергия контурного препарата, вычисленная на границах блока, e_I — энергия контурного препарата, вычисленная внутри блока, r — коэффициент взаимной корреляции между контурным препаратом блока, для которого выполняются вычисления и так называемого идеального блока, под которым понимается изображение его границ.

Роль коэффициента взаимной корреляции в написанном выражении заключается в том, что он представляет собой меру сходства обнаруженной прямоугольной структуры в оцениваемом изображении и прямоугольной структуры блока. Первая может представлять собой как полезный компонент, например, контур, имеющийся в передаваемом изображении, так и блочную структуру, вторая же — всегда только блочную структуру. Дробь оценивает заметность блочной структуры, которая маскируется деталями внутри блока.

6. Затем вычисляются усредненные корреляционно-энергетические отношения для каждого из сегментов путем усреднения ранее найденных корреляционноэнергетических отношений блоков $E_{k,l}$ для оцениваемого изображения $E^{(S)}$

и *D*-изображения $E_D^{(S)}$.

7. Следующим шагом алгоритма является вычисление корреляционно-энергетических критериев

$$PR^{(S)} = 1 - \frac{E^{(S)}}{E_D^{(S)}}$$

За оценку качества воспроизведения декодированных изображений принимается минимальное значение корреляционно-энергетического критерия в пределах десятиминутного интервала.

Из приведенных формул следует, что в случае эталонного изображения, для которого по определению $e_F = 0$, поскольку в нем блочная структура отсутствует, и, следовательно, $E_{k,l} = 0$, значение $PR^{(S)} = 1$, а для *D*-изображения $PR^{(S)} = 0$.

В работе [20] приводится также функциональная связь между предложенным критерием оценки качества воспроизведения изображений и экспертными оценками качества.

Глава 4



Оцифровка изображений

4.1. Представление изображений в памяти компьютера

Цифровая обработка изображений начинается с того, что вначале их вводят в память компьютера, используя для этой цели сканеры, цифровые камеры и видеокамеры, а также платы ввода видеоданных.

До ввода изображений в память компьютера они должны быть предварительно преобразованы из аналоговой формы в цифровую, т. е. оцифрованы. Так, например, в случае неподвижного ахроматического изображения непрерывное распределение яркости на его плоскости $L_c(x, y)$ будет преобразовано в матрицу чисел $||L_c(k, n)||$ с конечной разрядностью, подлежащую запоминанию на жестком диске (винчестере). Индексы k и n представляют номера строк и столбцов соответственно. При оцифровке в изображения неизбежно вносятся искажения, величина которых определяется плотностью и структурой расположения отсчетов на исходных изображениях, а также числом уровней квантования, определяемым разрядностью представления значений яркости. Результатом неправильного выбора этих параметров может быть как недопустимо большой уровень искажений, который сведет на нет все последующие работы, поскольку искажения, возникшие на этом этапе, в дальнейшем уже не смогут быть исправлены, так и чрезмерно большой объем файла, в котором записано изображение, вследствие неоправданного запаса при выборе исходных параметров оцифровки.

При выборе параметров оцифровки следует исходить из того, где и для каких целей в дальнейшем эти изображения будут использованы, поскольку это в значительной степени определит сам выбор. Так изображение, которое предназначено для альбома репродукций шедевров живописи, потребует высокого разрешения и точной передачи цвета, в то время как изображение, которое будет использовано в Интернете, скорее всего, будет представлено с меньшим разрешением и с палитрой в 256 цветов.

В этой главе мы подробно рассмотрим причины, приводящие к возникновению искажений на этапе оцифровки изображений, так называемых помех пространственной дискретизации изображений и шумов квантования, а также обсудим методы, позволяющие уменьшить эти искажения до допустимого значения.

4.2. Теорема Котельникова¹

При оцифровке непрерывное изображение вначале подвергается пространственной дискретизации, т. е. представляется в виде набора отсчетов (пикселов), поэтому обратимся к проблеме выбора плотности расположения отсчетов на исходном изображении. Рассмотрим для простоты случай пространственной дискретизации неподвижного ахроматического изображения. Будем считать, что пространственный спектр дискретизируемого изображения $L_{c0}(x, y)$ ограничен круговыми пространственными частотами ω_{xrp} , ω_{yrp} , т. е.

$$M_{c0}(\omega_{x},\omega_{y}) = \begin{cases} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L_{c0}(x,y) \exp\left[-i(\omega_{x}x + \omega_{y}y)\right] dx dy & \text{при} |\omega_{x}| < \omega_{xrp} \text{ и } |\omega_{y}| < \omega_{yrp}, \\ 0 & \text{при невыполнении этого условия,} \end{cases}$$

где $i = \sqrt{-1}$. В этом случае распределение яркости в дискретизируемом изображении определится следующим образом:

$$L_{\rm c0}(x,y) = \frac{1}{4\pi^2} \int_{-\omega_{\rm xrp}}^{\omega_{\rm xrp}} \int_{-\omega_{\rm yrp}}^{\omega_{\rm yrp}} M_{\rm c0}(\omega_x,\omega_y) \exp\left[i(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y.$$
(4.1)

Раскладывая спектр изображения $M_{c0}(\omega_x, \omega_y)$ в двойной ряд Фурье по переменным ω_x , ω_y на интервалах $-\omega_{xrp} < \omega_x < \omega_{xrp}$, $-\omega_{yrp} < \omega_y < \omega_{yrp}$, получаем

$$M_{c0}(\omega_x, \omega_y) = \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} D_{nk} \exp\left[i\left(\frac{2\pi n\omega_x}{2\omega_{xrp}} + \frac{2\pi k\omega_y}{2\omega_{yrp}}\right)\right],$$
(4.2)

где $2\omega_{xrp}$ и $2\omega_{yrp}$ — периоды разложения спектра изображения $M_{c0}(\omega_x, \omega_y)$, D_{nk} — коэффициенты разложения. На рис. 4.1 штриховкой показан разлагаемый в ряд Фурье спектр исходного изображения, а без штриховки — побочные спектры (так называемые реплики), обусловленные природой периодического ряда.



Рис. 4.1. К пояснению разложения спектра изображения в ряд Фурье

¹ Эта теорема известна также как теорема отсчетов.
Коэффициенты разложения определяются следующим образом:

$$D_{nk} = \frac{1}{2\omega_{xrp}} \int_{-\omega_{xrp}}^{\omega_{xrp}} \int_{-\omega_{yrp}}^{\omega_{yrp}} \int_{-\omega_{yrp}}^{\omega_{yrp}} M_{c0}(\omega_x, \omega_y) \exp\left[-i\left(\frac{2\pi n\omega_x}{2\omega_{xrp}} + \frac{2\pi k\omega_y}{2\omega_{yrp}}\right)\right] d\omega_x d\omega_y. \quad (4.3)$$

Сопоставляя формулы (4.1) и (4.3), видим, что

$$D_{nk} = \frac{\pi^2}{\omega_{xrp}\omega_{yrp}} L_{c0} \left(-\frac{\pi n}{\omega_{xrp}}, -\frac{\pi k}{\omega_{yrp}} \right).$$

Вводя обозначения $\frac{\pi}{\omega_{xrp}} = \Delta_x$, $\frac{\pi}{\omega_{yrp}} = \Delta_y$, будем иметь

$$D_{nk} = \frac{\pi^2}{\omega_{xrp}\omega_{yrp}} L_{c0} \left(-n\Delta_x, -k\Delta_y \right).$$
(4.4)

Здесь Δ_x , Δ_y представляют собой интервалы взятия отсчетов на изображении (интервалы пространственной дискретизации изображения) в направлении осей x и y. Обычно при дискретизации эти значения делают одинаковыми. Заменим в формуле (4.1) $M_{c0}(\omega_x, \omega_y)$ его представлением в виде ряда Фурье

$$L_{c0}(x, y) = \frac{1}{4\omega_{xrp}\omega_{yrp}} \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} L_{c0}(n\Delta_x, k\Delta_y) \times \int_{-\omega_{xrp}}^{\omega_{xrp}} \int_{-\omega_{yrp}}^{\omega_{yrp}} \exp\left[i\left(\omega_x(x-n\Delta_x)+\omega_y(y-k\Delta_y)\right)\right]d\omega_x d\omega_y.$$

Учитывая, что

$$\int_{-\omega_{xrp}}^{\omega_{xrp}} \exp\left[\mathbf{i}\omega_{x}\left(x-n\Delta_{x}\right)\right]d\omega_{x} = \frac{2\sin\omega_{xrp}\left(x-n\Delta_{x}\right)}{\left(x-n\Delta_{x}\right)}$$

$$\int_{-\omega_{yp}}^{\omega_{yp}} \exp\left[i\omega_{y}\left(y-k\Delta_{y}\right)\right]d\omega_{y} = \frac{2\sin\omega_{yp}\left(y-k\Delta_{y}\right)}{\left(y-k\Delta_{y}\right)},$$

И

запишем окончательно

$$L_{c0}(x,y) = \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} L_{c0}(n\Delta_x, k\Delta_y) \frac{\sin\omega_{xrp}(x - n\Delta_x)}{\omega_{xrp}(x - n\Delta_x)} \frac{\sin\omega_{yrp}(y - k\Delta_y)}{\omega_{yrp}(y - k\Delta_y)}.$$
(4.5)

Из этой формулы следует, что непрерывное изображение $L_{c0}(x, y)$ с ограниченным спектром круговых пространственных частот ω_x , ω_y полностью определяется значениями яркости, взятыми в точках дискретных отсчетов, расположенных в узлах прямоугольной решетки (рис. 4.2). Такая структура расположения отсчетов называется *ортогональной*.



Рис. 4.2. Ортогональная структура расположения точек дискретных отсчетов на изображении

Интервалы пространственной дискретизации Δ_x , Δ_y и граничные круговые пространственные частоты ω_{xrp} , ω_{yrp} , как уже было отмечено, связаны соотношениями $\Delta_x = \pi/\omega_{xrp}$, $\Delta_y = \pi/\omega_{yrp}$. Чем выше значения граничных, круговых, пространственных частот, тем меньшими должны быть значения интервалов пространственной дискретизации Δ_x , Δ_y для того, чтобы обеспечить правильное представление мелких деталей, которые содержит дискретизируемое изображение. Если известна площадь изображения s_{μ_3} , то при оговоренных выше условиях оно должно быть представлено $N = s_{\mu_3}/(\Delta_x \Delta_y)$ отсчетами, чтобы по ним можно было бы его точно восстановить. Выражение $\frac{\sin \omega_{xrp} (x - n\Delta_x)}{\omega_{xrp} (x - n\Delta_x)} \frac{\sin \omega_{yrp} (y - k\Delta_y)}{\omega_{yrp} (y - k\Delta_y)},$ называемое функцией отсчетов,

обладает следующими свойствами.

- 1. Во всех точках дискретных отсчетов, за исключением одной, координаты которой $x = n\Delta_x$, $y = k\Delta_y$, значения функции равны нулю. В точке же дискретного отсчета с координатами $x = n\Delta_x$, $y = k\Delta_y$ значение функции отсчетов равно единице.
- 2. Спектральная интенсивность функции отсчетов постоянна в интервале круговых пространственных частот $|\omega_x| < \omega_{xrp}$, $|\omega_y| < \omega_{yrp}$ и равна нулю за его пределами.

В более общем случае можно показать, что дискретные отсчеты не обязательно должны располагаться на равных расстояниях друг от друга. Более того, можно их передавать через один, но в каждой точке, где они передаются, кроме значения яркости передавать также значение ее первой производной или же вести передачу отсчетов через две точки, передавая при этом значения яркости, а также первую и вторую ее производные и т. д. Важно, чтобы число независимых данных на единицу площади было не меньше, чем $N_1 = \omega_{xrp} \omega_{yrp} / \pi^2$.

Аналогично доказывается теорема для случая, когда распределение яркости в изображении является функцией большего числа переменных, как это, например, имеет место в случае движущихся изображений.

Примечание

Значение теоремы отсчетов состоит в том, что она устанавливает связь между дискретными и аналоговыми изображениями.

В заключение заметим, что если при дискретизации изображений условие теоремы отсчетов $|\omega_x| < \omega_{xrp}$, $|\omega_y| < \omega_{yrp}$ не выполняется, а это, к сожалению, как правило, всегда имеет место, возникают своеобразные искажения, называемые *помехой про-странственной дискретизации* [49, 98, 103].

4.3. Помеха пространственной дискретизации

Переходя к изучению помехи пространственной дискретизации изображений, рассмотрим вначале случай, когда дискретизируемое изображение имеет ограниченный спектр пространственных частот и удовлетворяет условию теоремы Котельникова, т. е. $\Delta_x = \pi/\omega_{xrp}$, $\Delta_y = \pi/\omega_{yrp}$. Выделим на изображении *k*-ю строку, например строку, соответствующую k = 0, и рассмотрим значения отсчетов яркости вдоль этой строки $L_c(n\Delta_x, 0)$, для чего обратимся к рис. 4.3, *a*.





импульсной последовательности; $e - \phi$ ункции $L_{c0}(x, 0)$

Зависимость последовательности $L_{c0}(n\Delta_x, 0)$ от номера отсчета *n* можно рассматривать как результат амплитудной модуляции (перемножения) периодической последовательности бесконечно тонких импульсов, показанных на рис. 4.3, б, функ- $L_{c0}(x,0)$, представленной на рис. 4.3, *в*. Принимая во внимание, что цией периодическая последовательность бесконечно тонких импульсов имеет периодический дискретный спектр, условно показанный на рис. 4.4, а в виде отрезков одинаковой высоты, и принимая во внимание, что амплитудная модуляция последовательности бесконечно тонких импульсов функцией $L_{c0}(x,0)$ приводит к модуляции каждой из спектральных составляющих, приходим к заключению, что спектр изображения после его пространственной дискретизации имеет вид, показанный на рис. 4.4, б. Из рис. 4.4, б следует, что в результате амплитудной модуляции около каждого из спектральных компонентов периодической последовательности импульсов (отсчетов) возникли боковые полосы частот, представляющие собой основной спектр исходного изображения (на рис. 4.4, δ заштрихован), и побочные спектры.



Рис. 4.4. Спектры: a — немодулированной периодической импульсной последовательности; δ — периодической импульсной последовательности, модулированной функцией $L_{c0}(x,0)$, при условии, что частота дискретизации удовлетворяет условиям теоремы Котельникова

Аналогичная картина будет иметь место и в том случае, если мы будем рассматривать не одномерный, а двумерный спектр дискретизированного изображения $M_{cd}(\omega_x, \omega_y)$, который показан на рис. 4.5. В этом случае спектр дискретизированного изображения $M_{cd}(\omega_x, \omega_y)$ также включает в себя как компонент спектра исходного не дискретизированного изображения (на рисунке область, занимаемая этим компонентом спектра, заштрихована), так и компоненты побочных спектров (области, которые они занимают, показаны без штриховки). Из рисунка видно, что побочные спектры получаются путем смещения основного спектра по осям круговых пространственных частот ω_x , ω_y на величины $l\omega_{xd}$, $r\omega_{yd}$, где $\omega_{xd} = 2\pi/\Delta_x$, $\omega_{yd} = 2\pi/\Delta_y$ — частоты пространственной дискретизации по осям ω_x и ω_y , которые обычно берутся одинаковыми, l и r — целые числа. Положение частот дискретизации и их гармоник на рис. 4.5 показано кружками.

При этом

$$M_{\rm cd}\left(\omega_x,\omega_y\right) = \sum_{-\infty-\infty}^{\infty} M_{\rm c}\left(\omega_x - l\omega_{x\rm d}, \omega_y - r\omega_{y\rm d}\right).$$



Рис. 4.5. Двумерный спектр дискретизированного изображения, при условии, что частота пространственной дискретизации удовлетворяет условиям теоремы Котельникова



Рис. 4.6. Двумерный спектр дискретизированного изображения для случая, когда частота пространственной дискретизации не удовлетворяет условиям теоремы Котельникова

Из рис. 4.5 следует, что для того, чтобы из дискретизированного изображения восстановить исходное изображение, т. е. аналоговое, достаточно из спектра дискретизированного изображения выделить посредством фильтрации компонент исходного спектра. Эта операция может быть осуществлена путем свертки дискретизированного изображения с функцией отсчетов в соответствии с формулой (4.5).

Перейдем теперь к рассмотрению случая, когда дискретизируемое изображение не удовлетворяет условию теоремы отсчетов, т. е. когда $\Delta_x > \pi/\omega_{xrp}$, $\Delta_y > \pi/\omega_{yrp}$. Повторяя приведенные выше рассуждения, построим рисунок (рис. 4.6), на котором показаны области, занимаемые основным и побочными спектрами. Из рисунка видно, что теперь за счет того, что ω_{xrp} и ω_{yrp} возросли по сравнению с предыдущим случаем, возникло взаимное перекрытие основного и побочных спектров, что привело к возникновению так называемой помехи пространственной дискретизации. Поскольку спектр большинства реальных изображений не ограничен по частотам ω_x и ω_y , взаимное наложение спектров неизбежно, и после того как оно произошло, помеха пространственной дискретизации не может быть отфильтрована.

Проявление помехи пространственной дискретизации на изображениях определяется структурой самих изображений. Так на наклонных границах она проявляется в виде зазубрин. На участках, где много мелких деталей, но отсутствуют периодические структуры, помеха пространственной дискретизации проявляется в виде хаотически расположенных светлых и темных точек. Рисунок 4.7 демонстрирует сказанное.



а







Рис. 4.7. К пояснению проявления помехи пространственной дискретизации. При переходе от изображения к изображению слева направо шаг пространственной дискретизации каждый раз увеличивается в два раза





Если же изображение содержит периодические структуры, как, например, изображение, показанное на рис. 4.8, *a*, то помеха пространственной дискретизации проявляется в виде муарового рисунка, что хорошо видно на рис. 4.8, *б*.

Поскольку помеха пространственной дискретизации является мультипликативной помехой, то увеличение контраста изображения повлечет за собой увеличение также и ее контраста, отношение же "сигнал/помеха" при этом не изменится. Для того чтобы избежать возникновения этой помехи, спектр дискретизируемого изображения должен быть предварительно ограничен со стороны верхних пространственных

частот. Этого можно достичь, в частности, путем оптической фильтрации изображений. Осуществить такую фильтрацию можно, например, следующими двумя способами. Первый способ заключается в применении аподизированной оптики. Второй способ состоит в использовании когерентных оптических систем. При этом диапозитив фильтруемого изображения помещается в коллимированный поток когерентного света, который затем пропускается через линзу. В задней фокальной плоскости линзы, где распределение светового потока пропорционально двумерному спектру Фурье пространственного распределения прозрачности диапозитива, помещается диафрагма, которая отсекает световой поток в области верхних пространственных частот. Далее изображение фокусируется на светочувствительной поверхности датчика сигнала изображения. Профильтрованное таким образом изображение может быть продискретизировано без опасности возникновения помехи пространственной дискретизации. Однако описанные способы не удобны на практике — первый по причине существенного уменьшения светового потока аподизированной оптикой и, как следствие, снижения чувствительности преобразователя свет-сигнал, второй — по причине громоздкости.

Простым, но не лучшим методом ослабления помехи пространственной дискретизации на изображении, которым и пользуются на практике, является увеличение числа отсчетов на изображении при его оцифровке. Однако увеличение числа отсчетов влечет за собой пропорциональное увеличение размера файла, в котором изображение будет храниться. Все сказанное, с учетом специфики, можно также распространить на временную дискретизацию движущихся изображений, при которой возникает помеха временной дискретизации. В частности, эта помеха проявляется на изображении в виде хорошо всем знакомого эффекта обратного вращения спиц колеса при определенных скоростях его вращения.

В следующем разделе рассматривается разработанный нами метод, позволяющий сильно ослабить помеху пространственной дискретизации и при этом избежать увеличения числа отсчетов в изображении, а следовательно, и увеличения размера файла изображения [58, 61, 65].

4.4. Метод уменьшения помехи пространственной дискретизации

Приступая к рассмотрению метода уменьшения помехи пространственной дискретизации, будем полагать, что интервал пространственной дискретизации изображения одинаков в обоих направлениях, т. е. $\Delta = \Delta_x = \Delta_y$, а следовательно, одинакова и частота пространственной дискретизации в обоих направлениях $\omega_{\pi} = \omega_{\chi\pi} = \omega_{\chi\pi}$.

Как уже отмечалось, причиной возникновения помехи пространственной дискретизации на изображении является наложение на основной спектр побочных спектров. Так как спектры реальных изображений не ограничены по пространственным частотам, то избежать этого полностью невозможно, хотя можно в значительной степени ослабить последствия этих наложений путем увеличения частоты пространственной дискретизации ω_{d} . Рисунки 4.9, *а* и *б*, на которых представлены зависимости спектральной интенсивности дискретизированного изображения $S_{cd}(\omega_x, \omega_y)$ от круговой пространственной частоты ω_x , поясняют сказанное.



Рис. 4.9. Основной и побочные спектры изображения при двух различных частотах пространственной дискретизации

Это обстоятельство используется в описываемом методе уменьшения помехи пространственной дискретизации, включающем в себя три этапа. Вначале при оцифровке изображения выполняется его первичная (предварительная) дискретизация с круговой частотой пространственной дискретизации $\omega_{дn}$, в *n* раз превышающей основную частоту пространственной дискретизации ω_{dn} , т. е. $\omega_{dn} = n\omega_{dn}$. Это достигается за счет того, что интервал пространственной дискретизации на первом этапе берется в *n* раз меньше того, с которым изображение в дальнейшем будет записано в память. Благодаря такому выбору частоты пространственной дискретизации основной и побочные спектры дискретизированного изображения оказываются сильно смещенными друг относительно друга, так, как это показано на рис. 4.10, *a*, и поэтому их взаимное перекрытие оказывается пренебрежимо малым. На втором этапе предварительно дискретизированное изображение подвергается пространственной фильтрации путем его свертки с импульсной функцией типа sin *x*/*x*, благодаря чему без изменения остаются только те компоненты пространственного спектра изображения, которые удовлетворяют условию

$$|\omega_x - ln \omega_{\mu}| < \omega_{xrp} \wedge |\omega_y - rn \omega_{\mu}| < \omega_{yrp},$$



Рис. 4.10. Спектральные интенсивности дискретизированного изображения: *а* — после первичной дискретизации; *б* — после фильтрации с использованием импульсной функции вида sin *x*/*x*; *в* — то же после прореживания отсчетов

где l и r — целые числа, изменяющиеся от $-\infty$ до $+\infty$, все же прочие компоненты спектра отфильтровываются. На рис. 4.10, δ показана зависимость спектральной интенсивности изображения $S_{cd}(\omega_x, \omega_y)$ от круговой пространственной частоты ω_x после фильтрации. И, наконец, третьим, заключительным этапом реализации описываемого метода является передискретизация профильтрованного изображения, в результате которой частота пространственной дискретизации становится равной ω_{d} . Достигается это путем прореживания (децимации) отсчетов в n раз, как по горизонтали, так и по вертикали. В результате этой операции количество отсчетов на изображении уменьшается до величины, еще обеспечивающей выполнение теоремы отсчетов, а спектральная интенсивность дискретизированного изображения становится такой, как показано на рис. 4.10, *в*. Следует иметь в виду, что при этом линейные размеры воспроизводимого изображения также уменьшаются в n раз. Проведенные нами исследования показали высокую эффективность этого метода. Для его реализации достаточно предварительную частоту пространственной дискретизации увеличить всего лишь в два или три раза (*n* = 2...3) по отношению к основной.

На рис. 4.11 показано изображение, которое было оцифровано с использованием описанного метода при n = 2. Число отсчетов на этом изображении то же самое, что и на изображении, показанном на рис. 4.8, δ , однако помеха пространственной дискретизации на нем практически отсутствует.



Рис. 4.11. Изображение, при оцифровке которого был применен описанный здесь метод ослабления помехи пространственной дискретизации

Этот метод легко может быть распространен также на ослабление помехи временной дискретизации движущихся изображений. Принцип его реализации очевиден и пояснений не требует.

4.5. Структуры расположения отсчетов при дискретизации изображений

До сих пор при рассмотрении дискретизации изображений считалось, что спектр дискретизируемого изображения ограничен прямоугольником (в частном случае квадратом) со сторонами $2\omega_{xrp}$, $2\omega_{yrp}$. Из рис. 4.5 следует, что при выполнении условия теоремы отсчетов $|\omega_x| < \omega_{xrp} \wedge |\omega_y| < \omega_{yrp}$ в этом случае достигается наиболее плотная упаковка основного и побочного спектров на частотной плоскости. Однако, как следует из *разд. 1.7*, линии постоянной спектральной интенсивности большинства изображений скорее напоминают ромбы или квадраты, повернутые относительно своего центра на угол 45°. Поэтому, желая ограничить спектр реального изображения по пространственным частотам с внесением минимальных иска-

жений, его следует ограничивать по линиям постоянной спектральной интенсивности, в результате чего область, занимаемая таким спектром, приобретает форму квадрата, повернутого относительно своего центра на угол 45°, как это показано на рис. 4.12 штриховкой.

Полагая $\omega_{rp} = \omega_{xrp} = \omega_{yrp}$, что вполне допустимо, приходим к заключению, что для того, чтобы при ортогональной дискретизации изображения с таким спектром избежать наложения побочных спектров на основной компонент, необходимо частоту пространственной дискретизации выбирать из условия $\omega_{\pi} \ge 2\omega_{rp}$. При этом спектр

дискретизированного изображения будет иметь вид, приведенный на рис. 4.13. На этом рисунке темной штриховкой показана область, занимаемая основным компонентом спектра, а светлой штриховкой — области, занимаемые компонентами побочных спектров. Кроме того, на этом рисунке кружками показаны положения частоты пространственной дискретизации и ее гармоник. Как следует из рисунка, упаковка основного и побочного спектров на частотной плоскости при этом далека от совершенства — половина частотной плоскости (не заштрихованные области) оказывается не занятой спектральными компонентами.



Рис. 4.12. Штриховкой показана область, занимаемая спектром типичного изображения



Рис. 4.13. Спектр дискретизированного типичного изображения при ортогональной структуре дискретизации

Достичь более плотной упаковки основного и побочных спектров в рассматриваемом случае можно, если применить другую структуру расположения отсчетов на плоскости изображения. Поясним это следующим образом. Повернем изображение, спектр которого занимает область, показанную на рис. 4.12 штриховкой, на 45°. Не трудно видеть, что область, занимаемая спектром повернутого изображения, будет иметь форму квадрата со стороной $\sqrt{2}\omega_{\rm rp}$, при этом ортогональная структура расположения отсчетов с интервалом $\Delta = 2\pi/\omega_{\rm g}$, где $\omega_{\rm g} \ge \sqrt{2}\omega_{\rm rp}$ — частота пространственной дискретизации, обеспечит наиболее плотную упаковку основного

и побочных спектров на частотной плоскости, подобно тому, как это показано на рис. 4.5. Обратим внимание на то, что интервал пространственной дискретизации в рассматриваемом случае увеличился в $\sqrt{2}$ раз, что вызвано тем, что ширина и высота области, занимаемой спектром изображения, уменьшились в $\sqrt{2}$ благодаря его повороту. Если теперь после выполнения пространственной дискретизации сделать обратный поворот координатных осей, то повернется на 45° не только дискретизированное изображение, приняв свое нормальное положение, но и его спектр, который будет иметь вид, показанный на рис. 4.14, при этом структура расположения отсчетов на изображении окажется такой, как показано на рис. 4.15.



Рис. 4.14. Спектр дискретизированного типичного изображения при шахматной структуре дискретизации

Рис. 4.15. Шахматная структура дискретизации

Эту структуру расположения отсчетов принято называть *шахматной*. Применение шахматной структуры расположения отсчетов позволяет в два раза сократить число отсчетов на изображениях с рассмотренным спектром, а значит, и длину файлов, в которые они записываются, без сколько-нибудь заметного снижения их качества. Обусловлено это тем, что и в направлении строк, и в направлении столбцов шаг пространственной дискретизации при шахматной структуре расположения отсчетов увеличивается в $\sqrt{2}$ раз [98].

Из приведенного рассмотрения следует, что структура расположения отсчетов на дискретизируемом изображении, при которой достигается наиболее плотная упаковка основного и побочных спектров, а следовательно, и минимальный размер файла, в который оно записывается, полностью определяется видом его пространственного спектра. В настоящее время кроме ортогональной и шахматной структур расположения отсчетов, известны также *треугольная*, которая является оптимальной при анизотропном спектре изображения, и диагональная [98].

4.6. Интерполяция при воспроизведении изображений

Как уже было сказано в *разд. 4.1*, цифровое изображение представляет собой матрицу чисел, каждое из которых определяет значение яркости в точке взятия отсчета. Если бы такое изображение, без каких-либо дополнительных операций над ним, мы воспроизвели на плоскости, то получили бы картинку, похожую на изображение, показанное на рис. 4.16, *а*. То есть в этом случае мы увидели бы отдельные яркие точки на черном фоне, нечто вроде созвездия, глядя на которое трудно догадаться, какой "объект" оно изображает.



Рис. 4.16. К пояснению необходимости интерполяции при воспроизведении цифровых изображений

На самом деле при воспроизведении цифрового изображения на дисплее компьютера или на экране цифрового телевизора этого не происходит благодаря интерполяции, которая при этом имеет место.

В том случае, если используется дисплей с жидкокристаллической панелью, которая состоит из множества прямоугольных ячеек (пикселов), сигнал каждого отсчета яркости, включая соответствующую ячейку, засвечивает сразу весь пиксел, т. е. площадку прямоугольной формы, определяемую размерами жидкокристаллической ячейки. При этом имеет место так называемая интерполяция нулевого порядка, которая подробно рассмотрена в *разд.* 7.8. Даже такой простой вид интерполяции, как интерполяция нулевого порядка, исправляет ситуацию и делает изображение легко читаемым, как, например, это показано на рис. 4.16, *б*.

Аналогичное явление имеет место и когда применяется дисплей с электроннолучевой трубкой (кинескопом). В этом случае благодаря тому, что диаметр развертывающего луча электронно-лучевой трубки имеет конечные размеры, каждый отсчет, представляющий собой точку на плоскости изображения, воспроизводится пятнышком конечных размеров и результат получается аналогичным рассмотренному выше.

Обычно при выводе изображения на экран дисплея мы не задумываемся об этом, поскольку конечные размеры пиксела жидкокристаллической панели или конечные

размеры диаметра электронного луча дисплея, в котором применена электроннолучевая трубка, автоматически обеспечивают необходимую интерполяцию.

Интерполирующие свойства дисплеев могут играть не только положительную роль, как в рассмотренных выше случаях, но и отрицательную, ограничивая разрешение всей системы, поэтому количество строк и пикселов в строке выводимого на экран изображения должны согласовываться с разрешением экрана дисплея. При отсутствии такого согласования возможно возникновение таких нежелательных артефактов, как потеря четкости изображения или же появление на изображениях муаров (шума пространственной дискретизации).

4.7. Квантование изображений по яркости

Любая величина, подлежащая цифровой обработке, в том числе яркость пиксела, должна быть представлена в виде двоичного кода. Обрабатываемые изображения представляются в виде матрицы целых чисел $L_{\rm ckb}$, описывающих значения яркости в точках взятия отсчетов:

$$L_{\text{CKB}} = \frac{L_{\text{CMAKC}}}{m-1} \Big(a_{n_0} 2^{n_0 - 1} + a_{n_0 - 1} 2^{n_0 - 2} + \dots + a_1 2^0 \Big),$$

где $L_{\text{смакс}}$ — максимальное значение яркости в изображении; $a_k = 1$ или 0; k = 1, 2, ..., n_0 ; m — число уровней квантования, $m = 2^{n_0}$; n_0 — число разрядов двоичного кода на один отсчет (пиксел) изображения.

Преобразование аналоговой величины L_c, представляющей значение яркости в точке отсчета, которая может принимать любые значение в интервале $0 \div L_{\text{смакс}}$, в квантованную величину $L_{\text{скв}}$, принимающую только дискретные значения, называется квантованием. Квантование является нелинейным преобразованием сигнала. Отметим, что результат любого измерения, представленный числом с ограниченным количеством разрядов, можно рассматривать как квантованный. Таким образом, все величины, с которыми мы имеем дело в жизни и в инженерной практике, являются квантованными значениями. Главной характеристикой, описывающей работу квантующего устройства, является его амплитудная характеристика. Амплитудная характеристика квантующего устройства представляет собой зависимость значения выходного сигнала от его входного значения. На рис. 4.17, а в качестве примера приведена амплитудная характеристика квантующего устройства с равномерной шкалой квантования. На практике применяют также неравномерные шкалы квантования. В отличие от равномерной шкалы квантования, при неравномерной шкале квантования пороговые уровни и уровни квантования располагаются друг относительно друга на неодинаковом расстоянии. В качестве примера на рис. 4.17, б приведена амплитудная характеристика квантующего устройства с неравномерной шкалой квантования. Квантующие устройства с неравномерной шкалой квантования сложнее в своей реализации, но в ряде случаев они обеспечивают лучшие результаты, чем квантующие устройства, у которых шкала квантования равномерная.



Рис. 4.17. Амплитудные характеристики квантующих устройств: *а* — с равномерной шкалой квантования; *б* — с неравномерной шкалой квантования

При квантовании в исходный сигнал, например сигнал, определяющий яркость изображения в точке отсчета L_c , вносится ошибка, называемая *шумом квантования*. Величина этой ошибки равна

$$L_{\rm IIIKB} = L_{\rm CKB} - L_{\rm C} \, .$$

Одной из важнейших характеристик шума квантования является его средний квадрат (дисперсия)

$$\sigma_{\rm IIIKB}^2 = \overline{L_{\rm IIIKB}^2} \, .$$

Средний квадрат шума квантования $\sigma^2_{\text{шкв}}$ зависит:

 \square от среднего квадрата квантуемого сигнала $\overline{L_c^2}$;

- □ от числа уровней, на которое квантуется сигнал, т. е. от $m = 2^{n_0}$, где n_0 число двоичных разрядов кода, которым представляется квантованное значение сигнала L_c ;
- 🗖 от взаимного расположения пороговых уровней и уровней квантования;

□ от плотности вероятности квантуемого сигнала *W*.

В том случае, если шкала квантования задана, средний квадрат шума квантования можно определить следующим образом:

$$\sigma_{\text{шкв}}^2 = \overline{L_c^2} \sigma_0^2 \left(n_0, W \right), \tag{4.6}$$

где $n_0 = \log_2 m$, $\sigma_0^2(n_0, W)$ — средний квадрат шума квантования сигнала на m уровней при условии, что средний квадрат самого квантуемого сигнала равен единице, а распределение его по яркости описывается плотностью вероятности W. Другими словами, *средний квадрат шума квантования прямо пропорционален среднему квадрату квантуемого сигнала*. Это положение носит принципиально важный характер, и к нему мы будем в дальнейшем обращаться.

Если квантуемый сигнал распределен равномерно в интервале значений 0— $L_{\text{смакс}}$, оптимальной, т. е. обеспечивающей минимальный уровень шума квантования, будет равномерная шкала. При этом шаг квантования $\delta_{\text{кв}}$, представляющий собой разность двух смежных уровней квантования $L_{\text{квk}}$ и $L_{\text{квk}-1}$ с номерами k и k-1, будет постоянным $\delta_{\text{кв}} = \text{const}$ и равным $\delta_{\text{кв}} = L_{\text{смакс}}/(m-1)$.

При оцифровке изображений используется равномерная шкала квантования, несмотря на то, что в данном случае она не является оптимальной, поскольку закон распределения яркости в изображении не является равномерным. Обусловлено это тем, что при равномерной шкале после кодирования мы получаем код, который представляет номер уровня квантования в виде двоичного числа удобного для дальнейшего использования. В этом случае, если число уровней квантования $m \ge 8$, то, как и при равномерном распределении квантуемого сигнала, можно считать, что средний квадрат шума квантования равен $\sigma_{\rm kB}^2 = \delta_{\rm kB}^2/12$. Подставляя в эту формулу значение $\delta_{\rm kB}$, имеем

$$\sigma_{\rm KB}^2 = \frac{L_{\rm CMAKC}^2}{\left(m-1\right)^2 12}$$

откуда выражение для отношения сигнал-шум $\Psi = L_{c_{MAKC}}/\sigma_{\kappa_B}$ может быть записано в виде $\Psi = (m-1)\sqrt{12}$. Плотность вероятности шума квантования в рассматриваемом случае описывается выражением

$$W(L_{\rm III}) = \begin{cases} \frac{1}{\delta_{\rm KB}} & \text{при } -\frac{\delta_{\rm KB}}{2} \le L_{\rm III} \le \frac{\delta_{\rm KB}}{2}, \\ 0 & \text{при невыполнении этого условия.} \end{cases}$$

Ширина спектра шума квантования, как показывают расчеты и экспериментальные исследования, определяется числом уровней квантования и спектром квантуемого сигнала, и во много раз превышает последний.

Желая уменьшить уровень шума квантования, используют неравномерные шкалы, распределение пороговых уровней θ_{nk} и уровней квантования v_{kBk} в которых согласовано с плотностью вероятности W(v) квантуемого сигнала v. Примером использования неравномерных шкал может служить квантование сигнала ошибки предсказания в дифференциальной кодово-импульсной модуляции. Такие квантующие устройства называются оптимальными квантования оптимальных квантователей определяется следующим образом

$$\theta_{nk} = \frac{v_{KBk} + v_{KBk-1}}{2}, \ v_{KBk} = \int_{\theta_{nk}}^{\theta_{nk+1}} vW(v)dv / \int_{\theta_{nk}}^{\theta_{nk+1}} W(v)dv$$

В табл. 4.1 приведены значения пороговых уровней и уровней квантования, полученные в результате решения этих уравнений для случая, когда средний квадрат квантуемого сигнала равен единице [49].

	Значение уровня для следующего вида распределения квантуемого сигнала						
n ₀	равномерное		нормальное		Лапласа		
	$\Theta_{\Pi k}$	$v_{{}_{\mathrm{KB}k}}$	$\Theta_{\Pi k}$	$v_{{}_{\mathrm{KB}k}}$	$\Theta_{\Pi k}$	$\nu_{{}_{\mathrm{KB}k}}$	
	-1,7320	-0,8660	_∞	-0,7979	_∞	-0,7071	
1	0,0000	0,8660	0,0000	0,7979	0,0000	0,7071	
	1,7320		~		~		
2	-1,7320	-1,2990	_∞	-1,5104	∞	-1,8340	
	-0,8660	-0,4330	-0,9816	-0,4528	-1,1269	-0,4198	
	0,0000	0,4330	0,0000	0,4528	0,0000	0,4198	
	0,8660	1,2990	0,9816	1,5104	1,1269	1,8340	
	1,7320		∞		∞		
3	-1,7320	-1,5155	_∞	-2,1519	∞	-3,0867	
	-1,2990	-1,0825	-1,7479	-1,3439	-2,3796	-1,6725	
	-0,8660	-0,6495	-1,0500	-0,7560	-1,2527	-0,8330	
	-0,4330	-0,2165	-0,5005	-0,2451	-0,5332	-0,2334	
	0,0000	0,2165	0,0000	0,2451	0,0000	0,2334	
	0,4330	0,6495	0,5005	0,7560	0,5332	0,8330	
	0,8660	1,0825	1,0500	1,3439	1,2527	1,6725	
	1,2990	1,5155	1,7479	2,1519	2,3796	3,0867	
	1,7320		∞		∞		
4	-1,7320	-1,6237	_∞	-2,7326	_∞	-4,4311	
	-1,5155	-1,4072	-2,4008	-2,0690	-3,7240	-3,0169	
	-1,2990	-1,1907	-1,8435	-1,6181	-2,5971	-2,1773	
	-1,0825	-0,9742	-1,4371	-1,2562	-1,8776	-1,5778	
	-0,8660	-0,7577	-1,0993	-0,9423	-1,3444	-1,1110	
	-0,6495	-0,5412	-0,7995	-0,6568	-0,9198	-0,7287	
	-0,4330	-0,3247	-0,5224	-0,3880	-0,5667	-0,4048	
	-0,2165	-0,1082	-0,2582	-0,1284	-0,2664	-0,1240	
	0,0000	0,1082	0,0000	0,1284	0,0000	0,1240	
	0,2165	0,3247	0,2582	0,3880	0,2664	0,4048	
	0,4330	0,5412	0,5224	0,6568	0,5667	0,7287	
	0,6495	0,7577	0,7995	0,9423	0,9198	1,1110	
	0,8660	0,9742	1,0993	1,2562	1,3444	1,5778	
	1,0825	1,1907	1,4371	1,6180	1,8776	2,1773	
	1,2990	1,4072	1,8435	2,0690	2,5971	3,0169	
	1,5155	1,6237	2,4008	2,7326	3,7240	4,4311	
	1,7320		∞		∞		

Таблица 4.1. Значения пороговых уровней и уровней квантования для оптимального квантователя

Средний квадрат шума квантования в оптимальном среднеквадратичном квантователе при квантовании сигнала, дисперсия которого составляет единицу, равен

$$\sigma_0^2(n_0, W) = 1 - \sum_{k=0}^{m-1} (v_{\text{KB}k})^2 \int_{\theta_{nk}}^{\theta_{nk+1}} W(v) dv.$$

В табл. 4.2 приведены значения $\sigma_0^2(n_0, W)$, рассчитанные по этой формуле для трех плотностей распределения вероятностей.

Распределение	$\sigma_0^2(n_0)$ при различных распределениях квантуемого сигнала и различных n_0					
	1	2	3	4		
Равномерное	0,25	$6,25 \times 10^{-2}$	$1,56 \times 10^{-2}$	3,90×10 ⁻³		
Нормальное	3,63×10 ⁻¹	$1,16 \times 10^{-1}$	$3,54 \times 10^{-2}$	9,89×10 ⁻³		
Лапласа	0,5	$1,76 \times 10^{-1}$	$5,45 \times 10^{-2}$	$1,56 \times 10^{-2}$		

Таблица 4.2. Значения $\sigma_0^2ig(n_0,Wig)$ для оптимального квантователя

4.8. Ложные контуры и методы их ослабления

При квантовании изображений по яркости, как уже было отмечено, в них вносятся искажения, которые при недостаточном количестве уровней квантования проявляются на изображениях в виде так называемых ложных контуров. Ложные контуры особенно заметны на участках изображений с плавным изменением яркости и менее заметны на участках с высокой детальностью. Так на участках с плавными изменениями яркости они различимы и при квантовании на 32 уровня, тогда как на участках с высокой детальностью, например, на изображении ветвей кустарника на фоне снега, они почти незаметны и при 4 уровнях квантования. На рис. 4.18 в качестве примера показано исходное изображение и это же изображение, проквантованное на 2, 4 и 8 уровней, соответственно. Для того чтобы ложные контуры были бы гарантированно незаметными, в случае полутонового нецветного изображения, оно должно быть проквантовано не менее, чем на 128 уровней. Используемый в настоящее время стандарт предусматривает квантование таких изображений на 256 уровней, т. е. для представления яркости каждого пиксела отводится 8 дво-ичных единиц кода (один байт).

По этим же причинам при квантовании цветных изображений, включающих в себя три компонента RGB (красный, зеленый и синий), каждый из компонентов квантуется на 256 уровней, при этом на представление одного пиксела затрачивается 3 байта. Эта мода называется RGB, или True Color.



Рис. 4.18. К пояснению заметности на изображениях ложных контуров: *а* — изображение проквантовано на 256 уровней; *б* — изображение проквантовано на 8 уровней, *в* — изображение проквантовано на 4 уровня; *г* — изображение проквантовано на 2 уровня

г

в

Заметность ложных контуров на изображении, проквантованном на малое число уровней, можно ослабить, если перед квантованием к отсчетам изображения добавить псевдослучайную последовательность с малой дисперсией (псевдошум). В этом случае на участках изображения с плавным изменением яркости пересечение

квантуемым сигналом пороговых уровней за счет добавленной псевдослучайной последовательности будет происходить в случайных местах, благодаря чему размываются границы ложных контуров, делая их менее заметными. Аналогичный результат получается и при добавлении к квантуемому изображению специальной периодической последовательности (Pattern). Описанный метод используется в графических редакторах, например, в редакторе Photoshop, при переходе от моды True Color к моде с индексированными цветами. Недостатком этого метода является небольшое увеличение уровня шума на изображении.

4.9. Моды представления изображений

В настоящее время наиболее распространенными модами представления изображений являются следующие.

- Бинарная (Bitmap). Эта мода применяется для представления чертежей и других двухградационных изображений. При использовании этой моды на представление одного пиксела изображения затрачивается 1 бит, что позволяет воспроизводить только два уровня яркости: черный и белый. Достоинством этой моды является малая затрата двоичных единиц кода на представление изображений.
- □ Полутоновая (Grayscale). Эта мода применяется для представления полутоновых черно-белых, т. е. не цветных, изображений, например, фотографий. При использовании этой моды на представление одного пиксела изображения обычно затрачивается 8 бит, благодаря чему оказывается возможным воспроизводить на изображении 256 градаций яркости в интервале от 0 до 255. Нулевая градация соответствует уровню черного, а 255-я уровню белого. Однако предусмотрен вариант этой моды, при котором на представление одного пиксела затрачивается 16 бит.
- Палитровая (Indexed Color). Эта мода применяется для представления цветных изображений, цветовая палитра которых ограничена 256 цветами, поскольку на представление одного пиксела изображения затрачивается только 8 бит. В настоящее время применяется в Интернете. Особенностью этой моды является то, что цвета в ней проиндексированы, т. е. каждому цвету присвоен свой номер (индекс). При этом каждому пикселу изображения, в зависимости от его цвета, присваивается значение индекса. Файл, в который сохраняется индексированное изображение, включает в себя также палитру цветов, представляющую собой массив чисел, где для каждого значения индекса приведены значения R, G и В компонентов цвета. Поскольку реальные цветные изображения, как правило, содержат большее количество цветов, чем 256, то перед представлением в этой моде их цветовое содержание приводится к палитре из 256 цветов путем специальной обработки.
- □ Полноцветная (RGB Color, True Color). Эта мода в настоящее время широко применяется для представления цветных изображений. При использовании этой моды на представление одного пиксела изображения обычно затрачивается 24 единицы двоичного кода или, что то же самое, 3 байта. При этом 1 байт за-

трачивается на представление R-компонента, один на представление G-компонента и один на представление B-компонента. Не трудно сосчитать, что при этом обеспечивается воспроизведение более 16 млн цветов (лучше сказать, цветовых комбинаций). Благодаря этому обеспечивается хорошее представление цвета в цветном изображении. Однако предусмотрен вариант этой моды, при котором на представление одного пиксела затрачивается 48 бит.

Кроме перечисленных мод используются моды для представления изображения в других цветовых моделях, например, следующие.

□ CMY Color. При использовании этой моды на представление одного пиксела используется 3 байта. Переход в эту моду, которая представляет изображение в цветах, дополнительных к цветам R, G и B, осуществляется по формулам

C = 255 - R, M = 255 - G, Y = 255 - B,

где C, M, Y, R, G и B — интенсивности соответственно голубого, пурпурного, желтого, красного, зеленого и синего компонентов, представленных номерами квантовых уровней при максимальном значении 255. В силу ряда технологических проблем, возникающих при печати изображений на бумажный носитель, эта мода обычно самостоятельно не применяется.

СМҮК Color. Эта мода применяется для печати цветных изображений на бумажный носитель. При использовании этой моды на представление одного пиксела изображения обычно затрачиваются 4 байта. Три байта затрачиваются на представление цветовых компонентов С, М, Y и один байт — на представление так называемого компонента ключевого черного цвета К, о котором было сказано в *разд. 2.7*. Возможен вариант этой моды, при котором на представление одного пиксела затрачивается 8 байт.

4.10. Контраст, детальность, разрешение принтера и частота пространственной дискретизации

Рассмотрим теперь такие понятия, как разрешение и контраст, которые в значительной мере определяют качество воспроизведения изображения. Контраст изображения представляет отношение максимальной яркости в изображении (яркости в белом) к минимальной яркости в изображении (яркости в черном). Контраст воспроизводимого изображения определяется как контрастом исходного изображения, так и ограничениями со стороны воспроизводящего устройства, например, жидкокристаллической панели дисплея, кинескопа телевизора, или со стороны принтера.

Что касается разрешения, то здесь все не так просто. Впервые с этой проблемой встретились в оптике, где для оценки качества оптических устройств был введен критерий разрешения Рэлея. В соответствии с этим критерием при определении разрешающей способности исходили из того, что две точки считались разрешаемыми, если первый минимум дифракционной картины от первой точки совпадал

с первым максимумом дифракционной картины от второй точки. Однако в связи с развитием фотографии стало ясно, что разрешение зависит, и в очень сильной степени, также от уровня шума (в случае фотографии — зерна) на изображении. При оцифровке же даже при очень низком уровне шума, вносимого в изображение устройством, используемым для его ввода, в изображения вносятся дополнительные помехи, обусловленные, главным образом, его пространственной дискретизацией. Из этого следует, что при определении разрешения помимо амплитудночастотной характеристики устройства ввода необходимо учитывать указанные выше помехи. Однако если эти помехи свести к пренебрежимо малому уровню путем использования метода, описанного в *разд. 4.4*, то в случае небольших амплитудночастотных искажений устройства ввода его разрешение будет однозначно определяться числом отсчетов (пикселов) на единицу длины.

Качество воспроизведения изображения в сильной степени зависит как от числа уровней квантования, так и от разрешения, обеспечиваемого всем процессом обработки, который включает в себя ввод и оцифровку, собственно обработку и вывод. В случае вывода оцифрованных изображений на экран жидкокристаллической панели или кинескопа число воспроизводимых уровней квантования и обеспечиваемое ими разрешение можно считать в первом приближении величинами взаимно независимыми, если только жидкокристаллическая панель и кинескоп не вносят заметных апертурных искажений. Совсем иначе обстоит дело при выводе изображений на печать. Как известно, при печати тот или иной уровень серого обеспечивается путем заполнения пиксела (полутоновой ячейки) пятнами, формирующими точки того или иного размера. Рисунок 4.19 поясняет сказанное.



Рис. 4.19. К пояснению принципа растровой печати полутоновых ахроматических изображений

Чем больше пятен формируют точку, тем в большей степени она заполняет площадь пиксела черным, тем более темным он будет казаться зрителю на изображении, поскольку глаз вследствие ограниченной остроты зрения отдельных точек не видит, а воспринимает только среднюю яркость (зачерненность) пиксела. Количество уровней квантования m, которое можно воспроизвести таким способом, определяется отношением площади пиксела $s_{пик}$ к площади одного пятна $s_{пят}$ плюс единица, т. е. $m = s_{пик}/s_{пят} + 1$. На рис. 4.19 приведен пример для случая, когда максимальное число уровней квантования воспроизводимых печатающим устройством составляет 256. При выводе изображений на печать необходимо учитывать два важных параметра: число пикселов на единицу длины $n_{пик}$ и число пятен на единицу длины $n_{пят}$. Первый характеризует плотность отсчетов на изображении, второй — число пятен, которое печатающее устройство может воспроизвести на единицу длины. Количество воспроизводимых уровней квантования при печати можно определить также следующим образом

$$m = (n_{\text{пят}} / n_{\text{пик}})^2 + 1$$
.

Это соотношение всегда следует иметь в виду, работая с изображениями. До сих пор речь шла о выводе на печать черно-белых полутоновых изображений. В случае вывода на печать цветных изображений ситуация усугубляется. Как известно, при трехцветной печати цветных изображений используется четыре краски: голубая, желтая, пурпурная и черная. В соответствии с этим каждая точка в пикселе составляется из четырех субточек: голубой, желтой, пурпурной и черной. При этом площадь такой составной точки увеличивается в четыре раза, а линейный размер — в два раза. Этим объясняется тот факт, что струйные принтеры в режиме цветной печати обеспечивают в два раза меньшую (линейную) плотность печати.

В заключение следует еще упомянуть об одной новой технологии печати — стохастическом или частотно-модулированном растрировании. В отличие от описанного способа печати, при котором используются регулярные структуры — точки переменных размеров, расположенные периодически, в технологиях частотномодулированного растрирования точки фиксированного размера помещаются нерегулярным образом. Однако частотно-модулированное растрирование — сравнительно молодая технология, и поэтому о ее достоинствах и недостатках пока имеется мало достоверных сведений. Отмечается, что в этой технологии наблюдается тенденция к увеличению размера точки при печати [2, 3].

4.11. Сканеры

Для ввода неподвижных изображений в память компьютера используют сканеры, а также цифровые фотоаппараты и цифровые видеокамеры. Наилучшими характеристиками, определяющими качество введенного изображения, обладают сканеры с вращающимся барабаном.

Барабанные сканеры применяют в тех случаях, когда требуется обеспечить очень высокое качество изображений, например, при издании книг и альбомов по искусству. Однако до самого последнего времени такие сканеры имели большие размеры, требовали высокой квалификации обслуживающего персонала, а стоимость их составляла сотни тысяч долларов. Иметь такие сканеры могли позволить себе только фирмы, специализирующиеся на сканировании изображений. Однако в последние годы ситуация существенно изменилась, были разработаны небольшие по размерам сканеры с вращающимся барабаном, которые могут быть размещены на рабочем столе. Эти устройства соединяются с компьютерами Macintosh или IBM-совместимыми, работающими в среде Windows. Сканеры этого типа имеют практически те же характеристики, что и их старшие собратья, но стоят много дешевле.

Принцип действия барабанного сканера заключается в следующем. Изображение фиксируется на прозрачном барабане, который вращается с высокой скоростью (300—1350 оборотов в минуту), а на расстоянии в несколько миллиметров от него находится модуль сканирующего датчика. Внутри этого модуля находится мощный источник (галогенный или ксеноновый), свет которого фокусируется в виде малого пятна на поверхности вращающегося изображения. Свет, отраженный от поверхности изображения, попадает в модуль, проходит через малую диафрагму, определяющую размер пиксела, а затем через цветоделительные зеркала, в которых он разделяется на три световых потока: красный, зеленый и синий. Эти световые потоки проецируются на фотоэлектронные умножители (ФЭУ). Сигналы, снимаемые с выходов ФЭУ, усиливаются и подаются на аналогово-цифровые преобразователи (АЦП), где они преобразуются в цифровой код. В случае сканирования прозрачного оригинала модуль датчика освещает его изнутри барабана, так что свет от него проходит сквозь оригинал и собирается снаружи него.

Барабанные сканеры, как правило, производят выборку от 10 до 16 бит на каждый цвет, благодаря чему обеспечивается большой динамический диапазон, что позволяет сохранять детали изображения даже в затененных зонах. Применение ярких источников света в модуле датчика, а также ФЭУ обеспечивает высокое отношение сигнала к шуму, что необходимо для обеспечения большого динамического диапазона.

Разрешение барабанных сканеров определяется отношением диаметра барабана к диаметру диафрагмы и для современных моделей лежит в пределах от 2500 до 8000 пикселов на дюйм.

В качестве примера приведем параметры двух типичных настольных барабанных сканеров. Настольный барабанный сканер DT-S1015 фирмы Screen USA имеет разрешение 2500 пикселов на дюйм при 10-битовой разрядности считывания на каждый цвет. Область отображения для этого сканера составляет 5,8×5,9 дюймов. Другим примером является настольный барабанный сканер ColorGetter III Pro фирмы Optronics, разрешение которого составляет 8000 пикселов на дюйм с динамическим диапазоном 4,0 и областью отображения 11×15 дюймов. Этот сканер по своим параметрам приближается к барабанным сканерам высокого класса.

Планшетные сканеры являются наиболее распространенным типом устройств, применяемых для ввода изображений, что обусловлено их невысокой стоимостью и малыми размерами. В планшетных сканерах в качестве датчика сигнала изображения применяется прибор с зарядовой связью (ПЗС). В отличие от барабанных сканеров, в планшетных сканерах сканируемое изображение помещается на плоское стекло, отделяющее его от сканирующей головки, которая перемещается относительно неподвижного изображения. В качестве источника света в планшетных сканерах используют или флуоресцентные источники света с холодным катодом, или вольфрамовые галогенные лампы, что уменьшает выделение тепла, а следовательно, позволяет сканирующую головку поместить ближе к изображению. Разре-

шающая способность этих сканеров определяется количеством элементов в линейке ПЗС. При сканировании сигнал с выхода ПЗС усиливается и подается на вход АЦП, где он преобразуется в цифровой код. Динамический диапазон таких сканеров определяется двумя факторами: разрядностью преобразования, т. е. числом битов, которыми представляется отсчет в каждом цвете, и отношением сигнала к шуму. Эти параметры должны быть согласованы между собой. Одно только увеличение разрядности представления сигнала при низком отношении сигнала к шуму не приводит автоматически к увеличению динамического диапазона. Типичное значение разрядности для планшетных сканеров, выпускаемых промышленностью в настоящее время, составляет 10—12 бит на каждый цвет, а типичное разрешение — 600—1800 пикселов на дюйм.

В качестве примера приведем параметры двух типичных планшетных сканеров. Планшетный сканер Epson 1200С имеет разрешение 600×1200 пикселов на дюйм при 10-битовой разрядности считывания на каждый цвет. Сканер предназначен для работы с IBM-совместимыми компьютерами и Macintosh. Применение дополнительного адаптера диапозитивов дает возможность с помощью этого сканера сканировать прозрачные оригиналы. В качестве другого примера назовем планшетный сканер PowerLook фирмы Umax. Сканер обеспечивает разрешение 600×1200 пикселов на дюйм при 10-битовой глубине каждого цвета и может оцифровывать как отражающие, так и прозрачные оригиналы.

В сканерах, как в барабанных, так и в планшетных, выполняется предварительная обычно несложная цифровая обработка изображений.

4.12. Цифровые фото- и видеокамеры

В настоящее время на рынке имеется большое количество различных моделей цифровых фото- и видеокамер, изображения с которых вводятся в компьютер. В этих приборах в качестве преобразователей изображений в сигнал применяются матрицы на ПЗС. Изображение, спроецированное на матрицу, подвергается пространственной дискретизации и цветоделению. В зависимости от класса камеры возможны два случая.

В фото- и видеокамерах высокого класса оптический поток предварительно разделяется на красный, зеленый и синий компоненты посредством специальной цветоделительной системы, включающей в себя систему призм и дихроичных зеркал. После этого каждая цветовая составляющая светового потока проецируется на свою матрицу ПЗС, на выходе которой получается соответствующий сигнал. В качестве примера видеокамер такого класса можно назвать видеокамеру SDR-5150GC-5 фирмы Panasonic.

В недорогих цифровых видеокамерах и в цифровых фотокамерах применяются матрицы ПЗС с фильтрами Байера. Особенность этих камер состоит в том, что для получения цветного изображения в них используется одна-единственная матрица, которая генерирует все три цветовых компонента изображения. Для этого перед

матрицей ПЗС устанавливается набор цветофильтров. Компоненты R, G и B для каждого пиксела формируются путем использования сигналов, получаемых от:

🗖 одной ячейки, расположенной под цветофильтром красного цвета;

🗖 одной ячейки, расположенной под цветофильтром синего цвета;

🗖 двух ячеек, расположенных под цветофильтрами зеленого цвета.

При формировании красного, зеленого и синего компонентов сигналов каждого пиксела применяют алгоритмы интерполяции.

Получаемые таким образом изображения записываются на сменные носители данных большой емкости или на карточки флэш-памяти. Примером такого фотоаппарата является камера Digital IXUSII фирмы Canon, которая может записывать изображения с разрешением до 2048×1536 пикселов. Записанные изображения хранятся на быстродействующей сменной плате памяти в формате JPEG.

Существенным недостатком цифровых фотокамер является их недостаточно большой динамический диапазон, который заметно меньше, чем у пленочных камер. Обусловлено это особенностями работы матрицы. При увеличении света заряд, накапливаемый в каждой ячейке матрицы, также линейно возрастает, наконец, при некотором значении светового потока ячейка переполняется, и он начинает перетекать в соседние ячейки, что приводит к появлению на изображении белого пятна. Этим определяется верхняя граница динамического диапазона матрицы. При уменьшении светового потока рано или поздно достигается нижняя граница динамического диапазона, определяемая значением сигнала, при котором он становится сравнимым с шумом, генерируемым матрицей. Разрядность кода, представляющего интенсивность накопленных в ячейках сигналов, и динамический диапазон согласуются.

В настоящее время ведутся исследования, направленные на увеличение динамического диапазона матриц. Одним из перспективных направлений является разработка матриц, у которых сигнал каждого компонента пиксела генерируется двумя ячейками. Одна ячейка в такой матрице имеет большие размеры, а следовательно, более чувствительна к свету, но зато в ней раньше наступает насыщение, в то время как вторая ячейка имеет малые размеры, менее чувствительна, и в ней насыщение наступает позже. Результирующий сигнал получается путем комбинирования сигналов от ячеек этих двух типов. При этом ячейки больших размеров ответственны за передачу темных полутонов в изображении, а ячейки меньших размеров за передачу светлых полутонов. В настоящее время фотоаппараты, использующие такие матрицы, выпускаются промышленностью (примером может служить фотоаппарат Fuji S5), но они очень дороги.

Глава 5



Линейная фильтрация изображений

5.1. Цели применения линейной фильтрации изображений

Линейная фильтрация изображений — это одна из наиболее часто применяемых процедур обработки изображений, в том числе в современных графических редакторах. Так, например, графический редактор Photoshop предоставляет пользователю в меню 10 вариантов линейной фильтрации, включая как различные варианты сглаживания резких световых границ изображения, так и методы их подчеркивания. Широкое применение линейной фильтрации при обработке изображений обусловлено тем, что она позволяет в значительной мере скорректировать целый ряд искажений, возникающих при получении, передаче и при воспроизведении изображений. К этим искажениям, в первую очередь, относятся апертурные искажения, обусловленные несовершенством устройств, посредством которых формируются изображения (передающих телевизионных камер, сканеров, цифровых фотоаппаратов), а также ошибками, допущенными при съемке (неточностью фокусировки объектива, смазом изображения вследствие недостаточно короткой экспозиции при съемке движущихся объектов и т. д.).

Линейная фильтрация применяется для ослабления заметности растровой структуры на изображениях, получаемых в результате сканирования типографских репродукций. Как уже отмечалось в *главе 4*, линейная фильтрация применяется также для уменьшения помех, обусловленных наложением побочных спектров, возникающих при пространственной дискретизации изображений. Это касается и рассмотренного там же метода ослабления помех пространственной дискретизации и просто фильтрации, приводящей к ограничению пространственного спектра изображения при его вводе в компьютер. Кроме того, применение линейной фильтрации позволяет уменьшить ошибки интерполяции, т. е. заметность растровой структуры при воспроизведении изображений, например, при их печати. Посредством линейной фильтрации в известных пределах удается уменьшить заметность флуктуационного шума, а также других дефектов на воспроизводимых изображениях.

Следует также назвать такие области применения линейной фильтрации, как редактирование изображений, при котором достигается их большая выразительность.

Сюда, например, относится подчеркивание границ и сглаживание отдельных областей путем применения фильтров с различными характеристиками. Поэтому данному виду обработки изображений в отечественной и зарубежной литературе уделяется большое внимание (см., например, [23, 50, 102, 103]).

Переходя к рассмотрению перечисленных выше вопросов, рассмотрим вначале методы реализации линейной фильтрации, которая может быть реализована как в пространственной, так и в частотной области.

5.2. Метод фильтрации цифровых изображений путем их свертки с импульсной характеристикой

Фильтрация изображения $L_{c}(x, y)$ методом свертки с импульсной характеристикой h(x, y) является одним из распространенных методов фильтрации.¹ В случае непрерывного ахроматического изображения эта операция может быть записана следующим образом:

$$L_{c\Omega}(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L_{c}(\xi,\eta) h(x-\xi, y-\eta) d\xi d\eta, \qquad (5.1)$$

где $L_{c\Omega}(x, y)$ — распределение яркости в изображении после фильтрации, ξ , η — переменные интегрирования. При реализации этого метода фильтрации цифровым способом исходное изображение, изображение после фильтрации, а также импульсная характеристика представляются в виде массивов чисел, элементы которых обозначим соответственно через $L_c(k, n)$, $L_{c\Omega}(k, n)$ и h(k, n), а номера строк и столбцов — через k и n. При этом яркость пикселов профильтрованного изображения вычисляется следующим образом:

$$L_{c\Omega}(k,n) = \sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{c}(k+k',n+n')h(k',n'), \qquad (5.2)$$

где K и N — протяженность двумерной импульсной характеристики в обоих направлениях. Величины K и N выбираются нечетными для того, чтобы избежать смещения профильтрованного изображения относительно исходного.

Процесс фильтрации можно пояснить следующим образом. При фильтрации изображение сканируется окном (импульсной характеристикой), размеры которого,

¹ В настоящее время в разных областях техники используют различную терминологию для обозначения одной и той же функции. Так в оптике используют термин "функция рассеяния точки", в телевидении — "распределение прозрачности в апертуре", в теории электрических цепей — "импульсная функция", "импульсная характеристика", в теории обработки изображений — "окно" или "маска". В дальнейшем в зависимости от контекста мы будем применять ту или иную терминологию.

как уже было сказано, составляют $K \times N$ пикселов. Каждый отсчет окна представляет собой весовой коэффициент (значение импульсной характеристики), на который умножается пиксел изображения, покрываемый этим отсчетом окна. При этом интенсивность пиксела профильтрованного изображения, координаты которого совпадают с координатами центра окна, находятся путем суммирования всех произведений. Рисунок 5.1 поясняет изложенное.

На рис. 5.2 в целях дополнительного поясняющего примера показаны фрагмент исходного изображения, т. е. до фильтрации (слева), импульсная функция (в центре) и фрагмент изображения после фильтрации (справа) без учета краевого эффекта.



Рис. 5.1. К пояснению метода свертки



Рис. 5.2. Пример, поясняющий фильтрацию методом свертки

При разработке цифрового фильтра импульсная характеристика h(k, n) находится следующим образом.

Вначале, исходя из поставленных требований, находится частотная передаточная функция аналогового фильтра $K(\omega_x, \omega_y)$. Затем путем применения к ней двумер-

ного интегрального преобразования Фурье отыскивается соответствующая ей импульсная характеристика h(x, y):

$$h(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K(\omega_x, \omega_y) \exp\left[i(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y.$$
(5.3)

Найденную таким образом импульсную характеристику необходимо преобразовать в дискретную форму посредством ее пространственной дискретизации, при этом шаг пространственной дискретизации должен быть тем же самым, что и шаг пространственной дискретизации фильтруемого изображения.

Следующей операцией, которую нужно выполнить над дискретизированной импульсной характеристикой, является ее усечение, т. е. ограничение ее размеров по строкам и столбцам до разумных пределов. Дело в том, что частотным передаточным функциям, ограниченным в пространстве частот граничными частотами ω_{xrp} ,

 ω_{yrp} , соответствуют импульсные характеристики, не ограниченные в пространстве координат (x, y). Сказанное поясняет рис. 5.3, на котором показаны частотная передаточная функция фильтра нижних частот и соответствующая ей импульсная характеристика.



Рис. 5.3. Пример частотной передаточной функции фильтра нижних частот (*a*) и соответствующей ей импульсной характеристики (*б*)

Последней, заключительной операцией является нормировка усеченной импульсной характеристики, в результате которой сумма ее отсчетов должна стать равной единице, т. е.

$$\sum_{k=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n=-\frac{N-1}{2}}^{\frac{N-1}{2}} h(k,n) = 1.$$
(5.4)

Благодаря нормировке импульсной характеристики после ее усечения обеспечивается правильное воспроизведение средней яркости в профильтрованном изображении, которое иначе было бы нарушено вследствие операции усечения.

Обращаясь к проблеме усечения импульсной характеристики, отметим, что чем больше ее протяженность, тем больший объем вычислений необходимо выполнить при реализации цифровой фильтрации рассматриваемым методом. Кроме того, при этом на большей части изображения будет проявляться краевой эффект. Простое усечение импульсной характеристики посредством ее умножения на функцию окна W(k,n), удовлетворяющую условию

$$W(k,n) = \begin{cases} 1 & \text{при}|k|, |n| \le \frac{N-1}{2}, \\ 0 & \text{при невыполнении этого условия} \end{cases}$$

приводит к появлению нежелательной "волнистости" частотной передаточной функции, а также к ее расширению в частотной области. Для достижения компромисса между протяженностью импульсной характеристики в пространстве изображения и частотной передаточной функции в пространстве частот был разработан ряд окон специальной формы, среди которых наиболее известными являются: треугольное окно Бартлетта, окно Блэкмана, окно Ханна, окно Кайзера, а также окно Хэмминга, удовлетворяющее условию

$$W(n) = \begin{cases} 0,54+0,46\cos\frac{2\pi n}{N-1} & \text{при } |n| \le \frac{N-1}{2}, \\ 0 & \text{при невыполнении этого условия.} \end{cases}$$

Важной особенностью этих окон является то, что при приближении к границе усечения величина W(k,n) плавно убывает, благодаря чему ослабляются эффекты "волнистости" и расширения частотной передаточной функции.

После нахождения импульсной характеристики h(k,n) необходимо ее исследовать на разделимость в отношении переменных k и n. Если окажется, что она разделима, т. е. если

$$h(k,n) = h(k)h(n),$$

где h(k), h(n) — одномерные импульсные характеристики, то выражение (5.2) следует преобразовать к виду

$$L_{c\Omega}(k,n) = \sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} h(k') \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{c}(k+k',n+n')h(n').$$
(5.5)

Вычисление значений $L_{cO}(k, n)$ по формуле (5.5) позволяет существенно сократить количество необходимых математических операций по сравнению с количеством математических операций при использовании формулы (5.2). Так, например, если при вычислении $L_{cO}(k, n)$ по формуле (5.2) для определения значения одного отсчета профильтрованного изображения требуется выполнить (К-1) N операций умножения и (K-1)(N-1) операций сложения, то в случае вычисления $L_{cO}(k, n)$ по формуле (5.5) количество необходимых операций умножения сокращается до величины K + N, а количество операций сложения сокращается до величины K + N - 2. Если принять K = 7, N = 7, что совсем немного для типичных задач фильтрации, то даже и в этом случае выигрыш в количестве необходимых вычислительных затрат, обеспечиваемый за счет использования свойства разделимости импульсной характеристики, составит 3,5 раза в отношении операций умножения и 3 раза в отношении операций сложения. На самом деле, выигрыши при использовании свойства разделимости импульсной характеристики значительно больше. Следует отметить, что ряд импульсных характеристик, с которыми часто приходится иметь дело на практике, являются разделимыми. К ним относятся: импульсная характеристика, описываемая гауссовым законом, импульсная характеристика, имеющая постоянное значение внутри прямоугольного окна, и некоторые другие.

Далее следует обратить внимание еще на два существенных обстоятельства, о которых важно помнить при разработке фильтра.

Во-первых, необходимо устанавливать ограничители на величину яркости профильтрованного изображения перед его представлением восьмиразрядным кодом, предотвращающие ее выход за пределы принятого динамического диапазона. Появление таких значений яркости возможно, если на переходной характеристике фильтра имеются выбросы, обусловленные, например, резким спадом частотной передаточной функции. В этом случае отсутствие ограничителей будет приводить к переполнению разрядной сетки, что приведет к появлению на светлых участках профильтрованного изображения черных точек и пятен, а на темных — соответственно белых точек и пятен. Применение ограничителей динамического диапазона сигнала со стороны белого и со стороны черного позволяет избежать этих артефактов, хотя и вносит в профильтрованное изображение так называемый шум ограничения.

Во-вторых, ограничение разрядной сетки приводит к возникновению своеобразных искажений — шума. Дело в том, что алгоритм фильтрации методом свертки включает в себя операцию умножения, в результате которого количество разрядов кода, которым представляются отсчеты профильтрованного изображения, оказывается

равным сумме количеств разрядов кода, используемых для представления исходного изображения и импульсной характеристики. Поскольку и исходное изображение, и импульсная характеристика обычно представляются восьмиразрядными числами, то в результате фильтрации получается изображение, для точного представления интенсивностей которого требуется шестнадцатиразрядный код. Для того чтобы перейти к прежнему виду записи профильтрованного изображения, необходимо привести его в восьмиразрядному представлению, т. е. произвести округление, из-за чего возникает шум округления.

Изложенное полностью справедливо и применительно к фильтрации цветного изображения с той разницей, что фильтрации должны быть подвергнуты все три его компонента.

5.3. Метод фильтрации цифровых изображений в спектральной области

Альтернативой методу свертки изображения с импульсной характеристикой является фильтрация изображения в спектральной области. Сущность метода заключается в том, что вначале массив отсчетов, которым представлено исходное изображение $L_c(k,n)$, в рассматриваемом случае ахроматическое изображение, преобразуется по какому-либо базису в массив спектральных коэффициентов $M_c(u, v)$, а затем каждый из спектральных коэффициентов скалярно умножается на соответствующий отсчет дискретной частотной передаточной функции K(u, v):

$$M_{\mathrm{c}\Omega}(u,v) = M_{\mathrm{c}}(u,v) K(u,v),$$

где $M_{c\Omega}(u,v)$ — спектральные коэффициенты профильтрованного изображения, u, v — индексы, определяющие положение спектральных коэффициентов, а также отсчетов частотной передаточной функции в столбце и строке соответствующих массивов. На заключительном этапе этого метода фильтрации найденный массив спектральных коэффициентов преобразуется в массив отсчетов профильтрованного изображения $L_{c\Omega}(k, n)$.

Описанный метод полностью аналогичен фильтрации аналогового изображения $L_{\rm c}(x, y)$ в частотной (спектральной) области. Напомним, что в этом случае для исходного изображения путем двумерного интегрального преобразования Фурье находится его спектр $M_{\rm c}(\omega_x, \omega_y)$, который затем умножается на частотную передаточную функцию $K(\omega_x, \omega_y)$, а полученный таким образом спектр профильтрованного изображения $M_{\rm c\Omega}(\omega_x, \omega_y)$ путем обратного двумерного интегрального преобразования фурье преобразуется в профильтрованное изображение $L_{\rm c\Omega}(x, y)$. Этот метод фильтрации, в частности, может быть реализован оптическим способом.



Рис. 5.4. Зависимость амплитуды спектральных коэффициентов от индекса *и* для ДКП и ДПФ

При фильтрации изображения в спектральной области следует особенно внимательно подходить к определению дискретной частотной передаточной функции фильтра K(u, v). Обычно частотная передаточная функция цифрового фильтра определяется на основе предварительно найденной частотной передаточной функции аналогового фильтра $K(\omega_x, \omega_y)$. Однако сразу же следует предостеречь от соблазна получить частотную передаточную функцию K(u, v) путем простой дискретизации функции $K(\omega_x, \omega_y)$. Дело в том, что вид K(u, v), как и вид $M_c(u, v)$, определяется базисом, который используется при вычислении спектральных коэффициентов цифрового изображения. На рис. 5.4 показаны дискретные амплитудные спектры $|M_c(u, v)|$ одного и того же изображения, но полученные путем его преобразования по двум различным базисам.

В одном случае при определении спектральных коэффициентов $|M_c(u, v)|$ было использовано дискретное косинусное преобразование (ДКП), а в другом — дискретное преобразование Фурье (ДПФ). Из рисунка видно, что для двух различных базисов дискретные спектры сильно различаются между собой. Поэтому определение K(u, v) необходимо осуществлять следующим образом. Вначале посредством двумерного интегрального преобразования Фурье на основе известной аналоговой частотной передаточной функции $K(\omega_x, \omega_y)$ находится соответствующая ей импульсная характеристика h(x, y) (см. формулу (5.3)), которая подвергается про-
странственной дискретизации с тем же интервалом, что и фильтруемое изображение. После этого выполняется нормировка дискретной импульсной характеристики с тем, чтобы сумма всех ее отсчетов равнялась бы единице. И только затем вычисляется дискретная частотная передаточная функция K(u,v) путем применения к h(k,n) двумерного ортогонального преобразования с использованием тех же базисных функций, что и при вычислении спектральных коэффициентов.

При фильтрации цветного изображения все перечисленные выше операции выполняются над тремя его компонентами.

В заключение этого раздела сравним два метода линейной фильтрации изображений и дадим рекомендации относительно их применения.

В случае, если импульсная характеристика фильтра имеет большую протяженность и соответственно частотная передаточная функция — малую протяженность, метод фильтрации изображения в частотной области с учетом вычислительных затрат на прямое и обратное ортогональные преобразования может оказаться предпочтительнее вследствие меньших вычислительных затрат при его выполнении. Однако в большинстве практических случаев метод свертки фильтруемого изображения с импульсной характеристикой требует меньших вычислительных затрат и поэтому оказывается лучшим.

5.4. Апертурные искажения изображений

Апертурные искажения являются линейными искажениями. Наиболее общим способом описания этих искажений является задание импульсной характеристики (импульсной функции, функции рассеяния точки, распределения прозрачности в апертуре, маски) системы, вносящей эти искажения. Обычно функция рассеяния точки не постоянна на всей плоскости изображения, а изменяется с изменением координат x и y. В центре изображения кружок рассеяния, как правило, меньше, а по краям изображения больше. Поскольку эти изменения происходят плавно, то оказывается возможным все изображение разбить на так называемые изопланарные области, где функцию рассеяния точки можно считать неизменной. Для каждой изопланарной области можно ввести в рассмотрение частотную передаточную функцию, связанную с функцией рассеяния точки парой двумерных преобразований Фурье. Так как вид этих частотных передаточных функций определяется природой возникновения рассматриваемых апертурных искажений, то рассмотрим их для каждого случая отдельно.

Апертурные искажения, обусловленные неточной фокусировкой оптической системы

Рассмотрим случай, когда сфокусированное изображение перемещается из фокальной плоскости. Для простоты пренебрежем влиянием дифракции и аберрациями

оптической системы. Функция рассеяния точки при этом будет иметь вид кружка рассеяния с постоянным распределением яркости радиусом R, т. е.

$$h(r,\theta) = \begin{cases} 1/\pi R^2 & \text{при } r \le R, \\ 0 & \text{при } r > R, \end{cases}$$
(5.6)

где *г* — расстояние от центра, θ — полярный угол. В данном случае частотная передаточная функция, как известно [77], может быть найдена следующим образом:

$$K(\omega_r, \varphi) = \int_{0}^{2\pi} \int_{0}^{\infty} h(r, \theta) J_0(\omega_r r) r dr d\theta, \qquad (5.7)$$

где $J_0(.)$ — функция Бесселя 1-го рода нулевого порядка, φ — полярный угол в частотном пространстве, ω_r — круговая пространственная частота в полярной системе координат, связанная с круговыми пространственными частотами в декартовой системе координат соотношениями

$$\omega_x = \omega_r \cos \varphi, \ \omega_y = \omega_r \sin \varphi.$$
 (5.8)

Подставляя (5.6) в (5.7) и используя хорошо известное для функций Бесселя соотношение

$$\frac{d}{dz}\left[z^nJ_n(z)\right]=z^nJ_{n-1}(z)\,,$$

получим после интегрирования и подстановки пределов

$$K(\omega_r, \varphi) = \frac{2J_1(\omega_r R)}{\omega_r R},$$
(5.9)

где $J_1(.)$ — функция Бесселя 1-го рода первого порядка. Применяя (5.8) к (5.9), получим

$$K(\omega_x, \omega_y) = \frac{2J_1\left(R\sqrt{\omega_x^2 + \omega_y^2}\right)}{R\sqrt{\omega_x^2 + \omega_y^2}}.$$
(5.10)

На рис. 5.5 показана частотная передаточная функция, рассчитанная по формуле (5.10). Особенностью этой функции является то, что на определенных пространственных частотах она изменяет свой знак, проходя через нуль, что обусловлено скачкообразным изменением фазочастотной характеристики на угол, равный π . Первый раз переход функции через нуль, так называемый первый предел разрешения, соответствует круговой пространственной частоте $\omega_r = 3,83/R$. Если с помощью рассматриваемой дефокусированной оптической системы спроецировать на экран миру, состоящую из группы сходящихся полос, то на круговой пространственной частоте штрихов, равной 3,83/R, мы будем иметь нулевой контраст. На больших

частотах, где частотная передаточная функция принимает отрицательные значения, мы вновь будем наблюдать штрихи миры, однако с "отрицательным контрастом", т. е. в спроецированном изображении черным штрихам миры будут соответствовать белые и наоборот. Этот эффект, называемый в оптике ложным разрешением, будет повторяться при каждом переходе частотной передаточной функции в область отрицательных значений. Рисунок 5.6 демонстрирует это. Наличие нулей у частотной передаточной функции означает невозможность полной коррекции рассматриваемых искажений, т. к. на частотах, где частотная передаточная функция принимает нулевые значения, информация об этих частотных составляющих полностью теряется.





Рис. 5.6. Эффект ложного разрешения: *а* — неискаженное изображение миры; *б* — изображение миры, полученное в результате дефокусировки оптической системы

Апертурные искажения изображения, обусловленные смазом вследствие недостаточно короткой экспозиции при съемке движущегося объекта

В качестве примера рассмотрим фотографирование движущегося объекта, при этом будем считать, что его изображение идеально сфокусировано в плоскости светочувствительной поверхности, скорость перемещения составляет v м/с в направлении координатной оси x, а экспозиция равна $T_{3 \kappa cn}$ с. Если фотографируемым объектом является точка, например, звезда при фотографировании звездного неба без компенсации его перемещения, обусловленного вращением Земли, то поскольку в разные моменты времени она занимает на проекции изображения различные положения, на фотографии останется ее след в виде отрезка линии протяженностью Δx . Этот след может рассматриваться как функция рассеяния точки. В данном случае искажения, обусловленные движением, носят одномерный характер и могут быть описаны одномерной функцией рассеяния точки. Полагая, что движение происходит в направлении оси x, запишем

$$h(x) = \begin{cases} 1/\Delta x & \text{при} |x| \le \frac{\Delta x}{2}, \\ 0 & \text{при} |x| > \frac{\Delta x}{2}, \end{cases}$$
(5.11)

где

$$\Delta x = v T_{\rm 3KCII} \,. \tag{5.12}$$

Искажения, возникающие при фотографировании более сложных объектов, могут быть найдены путем свертки их неискаженных изображений с функцией рассеяния точки, описывающей результат смаза, обусловленный движением.

Определим частотную передаточную функцию, описывающую искажения, обусловленные недостаточно короткой экспозицией при фотографировании движущегося объекта. Для этого подвергнем интегральному преобразованию Фурье найденную функцию рассеяния точки (5.11)

$$K(\omega_x) = \frac{1}{\Delta x} \int_{-\Delta x/2}^{\Delta x/2} \exp(-\mathbf{i}\omega_x x) dx = \frac{\sin\left(\frac{\omega_x \Delta x}{2}\right)}{\left(\frac{\omega_x \Delta x}{2}\right)}$$

и с учетом (5.12) будем иметь

$$K(\omega_{x}) = \frac{\sin\left(\frac{\omega_{x}vT_{3\text{KC\Pi}}}{2}\right)}{\left(\frac{\omega_{x}vT_{3\text{KC\Pi}}}{2}\right)}.$$
(5.13)

На рис. 5.7 приведена частотная передаточная функция, рассчитанная по формуле (5.13). Как и в предыдущем случае, частотная передаточная функция с увеличением пространственной частоты изменяет свой знак, проходя через нуль на частотах $\omega_x = 2\pi n/vT_{_{3KC\Pi}}$, где n — целое число. Как уже было отмечено, изменение знака частотной передаточной функции означает скачкообразное изменение фазочастотной характеристики на угол, равный π . Наличие нулей означает невозможность полной коррекции рассматриваемых искажений.



Рис. 5.7. Частотная передаточная функция, описывающая искажения, возникающие вследствие смаза



Рис. 5.8. Здесь а — неискаженное изображение; б — изображение со смазом

На рис. 5.8, a в качестве примера показано изображение легкового автомобиля, полученное при достаточно короткой экспозиции, обеспечившей отсутствие заметных искажений, обусловленных смазом, а на рис. 5.8, δ представлено искаженное изображение этого же автомобиля, полученное при продолжительной экспозиции, приведшей к появлению заметного смаза.

Все вышеизложенное справедливо не только для фотографических устройств, но и для всех устройств, посредством которых производится съемка изображений.

Апертурные искажения изображения, обусловленные несовершенством преобразователей изображения в сигнал

Обратимся теперь к случаю, когда источником апертурных искажений является датчик сигнала изображения. Из опыта работы с такими датчиками, например, с датчиком сигнала изображения на основе прибора с зарядовой связью, известно, что их функции рассеяния точки (в телевидении в этом случае чаще используют термин "распределение прозрачности в апертуре") хорошо аппроксимируются гауссовой функцией

$$h(x, y) = \frac{1}{\pi r_e^2} \exp\left(-\frac{x^2 + y^2}{r_e^2}\right),$$
 (5.14)

где r_e — условный радиус кружка рассеяния, т. е. расстояние от его центра, на котором h(x, y) уменьшается в $e \approx 2,72$ раза. На рис. 5.9 показана функция рассеяния точки, построенная в соответствии с (5.14).



Рис. 5.9. Типичная функция рассеяния точки преобразователя изображения в сигнал

Найдем частотную передаточную функцию, соответствующую функции рассеяния точки, которая задана формулой (5.14):

$$K(\omega_x, \omega_y) = \frac{1}{\pi r_e^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left[-\frac{x^2 + y^2}{r_e^2} - \mathbf{i}(\omega_x x + \omega_y y)\right] dxdy.$$

Учитывая, что переменные в показателе степени разделяются, запишем:

$$K(\omega_x, \omega_y) = \frac{1}{\sqrt{\pi}r_e} \int_{-\infty}^{\infty} \exp\left[-\left(\frac{x^2}{r_e^2} + \mathbf{i}\omega_x x\right)\right] dx \frac{1}{\sqrt{\pi}r_e} \int_{-\infty}^{\infty} \exp\left[-\left(\frac{y^2}{r_e^2} + \mathbf{i}\omega_y y\right)\right] dy.$$

Преобразуя выражение, стоящее в показателе степени подынтегральной функции первого интеграла, к виду

$$\frac{x^2}{r_e^2} + \mathbf{i}\omega_x x = \left(\frac{x}{r_e} + \frac{\mathbf{i}r_e\omega_x}{2}\right)^2 + \frac{r_e^2\omega_x^2}{4}$$

и выполнив аналогичные преобразования для второго интеграла, будем иметь

$$K(\omega_x, \omega_y) = \frac{1}{\sqrt{\pi}r_e} \exp\left[-\left(\frac{r_e^2 \omega_x^2}{4}\right)\right] \int_{-\infty}^{\infty} \exp\left[-\left(\frac{x}{r_e} + \frac{\mathbf{i}r_e \omega_x}{2}\right)^2\right] dx \times \frac{1}{\sqrt{\pi}r_e} \exp\left[-\left(\frac{r_e^2 \omega_y^2}{4}\right)\right] \int_{-\infty}^{\infty} \exp\left[-\left(\frac{y}{r_e} + \frac{\mathbf{i}r_e \omega_y}{2}\right)^2\right] dy,$$

что после несложных преобразований и интегрирования дает

$$K(\omega_x, \omega_y) = \exp\left[-\frac{r_e^2(\omega_x^2 + \omega_y^2)}{4}\right].$$
(5.15)

На рис. 5.10 приведена частотная передаточная функция, построенная в соответствии с формулой (5.15). Особенностью этой функции, отличающей ее от ранее найденных частотных передаточных функций, является то, что при всех значениях пространственных частот она остается положительной, плавно стремясь к нулю при неограниченном их возрастании. Это указывает на отсутствие фазовых искажений при преобразовании изображения в сигнал рассматриваемыми датчиками. Вносимые в сигнал искажения проявляются в размытии резких световых границ на изображении, которое в дальнейшем будет воспроизведено на основе этого сигнала, а также в уменьшении контраста мелких деталей. Что же касается эффекта ложного разрешения, то он в данном случае отсутствует. Рисунок 5.11 иллюстрирует рассмотренный случай.



Рис. 5.10. Типичная частотная передаточная функция преобразователя изображения в сигнал



б — изображение миры, полученное из искаженного сигнала

Апертурные искажения изображения, обусловленные несовершенством воспроизводящих устройств

В настоящее время в качестве воспроизводящих устройств применяют жидкокристаллические панели, реже кинескопы и плазменные панели. Что касается жидкокристаллических и плазменных панелей, то они вносят малый, по сравнению с остальными звеньями системы, уровень апертурных искажений. Воспроизводящие устройства на основе кинескопов вносят больший уровень апертурных искажений. Их частотная передаточная функция может быть аппроксимирована гауссовой функцией.

Искажения, обусловленные турбулентностью атмосферы

Как известно, при рассматривании удаленных объектов через толстый слой атмосферы наблюдается дрожание их изображений, а если угловые размеры объектов малы, как это имеет место при рассматривании звездного неба, то наблюдается не только их дрожание, но и мерцание. Объясняется это тем, что турбулентно движущиеся потоки воздуха неоднородны по плотности вследствие температурных различий и поэтому проходящие через них лучи света, преломляясь, отклоняются на различные углы. Другими словами, турбулентность атмосферы приводит как к флуктуациям по яркости, так и к геометрическим искажениям. Однако если произвести фотографирование удаленных объектов при достаточно продолжительной экспозиции, то дрожащие и мерцающие детали фотографируемого изображения, накладываясь друг на друга, создадут картину размытого изображения объекта, на которое наложена световая вуаль. Размытие изображения на фотографии может быть смоделировано линейным фильтром, частотная передаточная функция которого имеет вид

$$K_T\left(\omega_x,\omega_y\right) = \exp\left[-\left(\frac{\omega_x^2 + \omega_y^2}{\Omega_T^2}\right)^{5/6}\right],$$

где Ω_T — так называемая приведенная частота — параметр, определяющий скорость, с которой уменьшается значение $K_T(\omega_x, \omega_y)$ при возрастании пространственных частот ω_x и ω_y . Чем меньше значение Ω_T , тем быстрее происходит спад $K_T(\omega_x, \omega_y)$ и тем сильнее будет размыто изображение. Значение Ω_T определяется величиной турбулентности атмосферы [23]. Обратим внимание на тот факт, что частотная передаточная функция, посредством которой моделируются пространственно-частотные искажения, вносимые турбулентной атмосферой, близка к гауссовой функции.

При нахождении результирующей частотной передаточной функции всей системы в целом необходимо учитывать все источники апертурных искажений. Это можно сделать двумя путями: либо путем свертки исходного изображения с результирующей функцией рассеяния точки, которая получается в результате сверток функций рассеяния точки, учитывающих соответствующие источники апертурных искажений, либо путем перемножения спектра исходного изображения на результирующую частотную передаточную функцию, которая находится путем перемножений частотных передаточных функций, учитывающих соответствующие источники апертурных искажений.

5.5. Фильтрация изображений, искаженных гауссовым белым шумом

Простейшим методом ослабления аддитивного гауссова белого шума на изображениях является его фильтрация посредством низкочастотного линейного фильтра с прямоугольной частотной передаточной функцией. Поскольку спектральная интенсивность изображения на высоких пространственных частотах уменьшается, как правило, пропорционально квадрату пространственных частот, а спектральная интенсивность белого шума остается постоянной, то в результате такой фильтрации энергия шума ослабляется в большей степени, чем энергия изображения, но при этом, к сожалению, уменьшается четкость границ на корректируемом изображении.

Несколько лучшие результаты можно получить при использовании фильтра Винера, частотная передаточная функция которого в случае белого шума имеет вид:

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = \frac{S_{\text{c}}\left(\omega_{x},\omega_{y}\right)}{S_{\text{c}}\left(\omega_{x},\omega_{y}\right) + S_{\text{III}}},$$

где $S_{c}(\omega_{x}, \omega_{y})$ — спектральная интенсивность корректируемого изображения, S_{iii} — спектральная интенсивность белого шума, которая не зависит от пространственных частот [12, 23, 50].

Другой подход к реставрации изображений, искаженных шумом, заключается в применении адаптивной фильтрации. Рассматриваемый адаптивный фильтр также является низкочастотным фильтром, а следовательно, фильтром, сглаживающим не только шум, но также яркостные и цветовые границы на изображении. При выполнении этого вида фильтрации вначале анализируется окрестность каждого пиксела, для которого делается оценка яркости. Эта окрестность представляет собой окно размером $K \times N$ пикселов. В результате этого анализа находится среднее значение яркости $\overline{L_c}$ внутри окна

$$\overline{L_{\rm c}} = \frac{1}{KN} \sum_{-K/2}^{K/2} \sum_{-N/2}^{N/2} L_{\rm c}(k,n),$$

а также средний квадрат отклонения от среднего значения яркости σ_{Σ} в этом окне

$$\sigma_{\Sigma}^{2} = \frac{1}{KN} \sum_{-K/2}^{K/2} \sum_{-N/2}^{N/2} \left[L_{c}(k,n) - \overline{L_{c}} \right]^{2},$$

после чего производится сама оценка (фильтрация). Обратим внимание на то, что σ_{Σ} включает в себя как компонент, обусловленный шумом $\sigma_{\rm m}$, так и компонент, обусловленный изменением яркости в незашумленном изображении σ_L , в свою очередь обусловленную его текстурой и контурами. Поскольку эти компоненты взаимно не коррелированы, то

$$\sigma_{\Sigma}^2 = \sigma_{\rm III}^2 + \sigma_L^2 \,.$$

Адаптивная фильтрация выполняется таким образом, что в тех местах изображения, для которых дисперсия яркости велика σ_{Σ}^2 , фильтр выполняет слабое сгла-

живание, поскольку значительная доля отклонения яркости от среднего значения в этих местах изображения обусловлена наличием световых границ или текстурой, которые необходимо сохранить. В тех же местах изображения, для которых она мала, фильтр осуществляет сглаживание в большей степени, поскольку отклонение яркости от среднего в этих местах изображения обусловлены шумом, который необходимо ослабить. При этом значения яркости пикселов в профильтрованном изображении находятся по формуле

$$\hat{L}_{c}(k,n) = \overline{L_{c}} + \frac{\sigma_{\Sigma}^{2} - \sigma_{uu}^{2}}{\sigma_{\Sigma}^{2}} \Big[L_{c}(k,n) - \overline{L_{c}} \Big].$$

Из формулы следует, что при отсутствии шума, когда $\sigma_{\rm m}^2 = 0$, результат фильтрации (оценка) совпадает с исходным значением яркости фильтруемого изображения, как того и следовало ожидать.

В другом крайнем случае, когда дисперсия шума велика, множитель перед квадратными скобками в формуле становится очень малым и оценка яркости пиксела $\hat{L}_{c}(k,n)$ приближается к среднему значению яркости $\overline{L_{c}}$ внутри окна. В результате такого подхода, как правило, результат фильтрации получается лучше, чем при неадаптивной фильтрации. Рассмотренный адаптивный фильтр является более избирательным, чем сопоставимый ему линейный фильтр, который сохраняет световые границы и другие высокочастотные части изображения в той же степени, что и адаптивный фильтр, однако адаптивная фильтрация требует больших вычислительных затрат, чем неадаптивная линейная фильтрация. Отметим, что рассмотренная адаптивная фильтрация лучше всего работает в том случае, когда шум является белым гауссовым.

На рис. 5.12 для сравнения приведены: исходное изображение, зашумленное изображение, результат фильтрации зашумленного изображения посредством низкочастотного фильтра с прямоугольной частотной передаточной функцией и результат фильтрации зашумленного изображения посредством адаптивного фильтра. Отношение размаха сигнала от черного до белого к среднеквадратичному значению шума в зашумленном изображении составляет 5. При фильтрации изображения низкочастотным фильтром ширина спектра изображения была сокращена в 4 раза по обеим пространственным частотам. При сравнении изображений, представленных на рис. 5.12, в и г, видим, что применение адаптивной фильтрации при том же ослаблении шума на изображении, что и в случае фильтрации простым низкочастотным фильтром, удается получить немного более четкие световые границы, чем в предыдущем случае. Однако если сравнивать изображения, приведенные на рис. 5.12, a и c, т. е. исходное незашумленное изображение и изображение, полученное после фильтрации адаптивным фильтром, то не трудно заметить, что в последнем случае резкость световых границ ниже, чем в первом случае.



Рис. 5.12. Здесь: *а* — исходное изображение; *б* — изображение, к которому добавлен аддитивный гауссов белый шум; *в* — результат фильтрации зашумленного изображения низкочастотным фильтром с прямоугольной частотной передаточной функцией; *г* — результат фильтрации зашумленного изображения адаптивным фильтром

5.6. Коррекция апертурных искажений в изображениях при наличии шума

Линейная фильтрация часто применяется для реставрации изображений, которые среди прочих искажений содержат также апертурные искажения. Коррекция апертурных искажений изображений в отсутствие шума, по крайней мере, теоретически, не приводит к каким-либо затруднениям. Если частотная передаточная функция искажающей системы, описывающая апертурные искажения, $K_{\mu c \kappa} (\omega_r, \omega_v)$

известна, то для полной коррекции достаточно профильтровать искаженное изображение, используя для этой цели линейный корректирующий фильтр, частотная передаточная функция которого должна быть обратной частотной передаточной функции искажающей системы, т. е.

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = 1/K_{\text{иск}}\left(\omega_{x},\omega_{y}\right).$$

Фильтр с такой частотной передаточной функцией и осуществляемая им фильтрация называются *инверсными*. При инверсной фильтрации мы встречаемся с двумя проблемами.

Первая проблема заключается в том, что частотная передаточная функция искажающей системы $K_{uck}(\omega_x, \omega_y)$ может принимать нулевые или малые значения в областях пространственных частот, где спектральная интенсивность изображения еще достаточно велика (см., например, рис. 5.7). То есть при формировании корректирующей частотной передаточной функции мы встречаемся с проблемой деления на ноль. Выходом из этой ситуации может служить применение функции окна $W(\omega_x, \omega_y)$, которая бы в этих областях принимала нулевые значения, а вне этих областей равнялась бы единице. Умножая результат инверсии на эту функцию, мы тем самым решаем проблему, т. е.

$$K_{\mathrm{kop}}\left(\omega_{x},\omega_{y}\right)=W\left(\omega_{x},\omega_{y}\right)/K_{\mathrm{HCK}}\left(\omega_{x},\omega_{y}\right).$$

Другими словами, мы формируем частотную передаточную функцию корректирующего фильтра только для тех областей пространственных частот, в которых она отлична от нуля. По-существу, функция окна представляет собой частотную передаточную функцию режекторного фильтра.

Вторая проблема обусловлена присутствием шума на корректируемом изображении. В результате при коррекции апертурных искажений изображения пространственно-частотные компоненты, которые в результате искажений были ослаблены, должны быть усилены корректирующим фильтром, а вместе с ними будут усилены и соответствующие частотные составляющие шума. При такой коррекции вместо улучшения изображение может стать даже хуже некорректированного вследствие увеличения зашумленности. Поэтому при выборе частотной передаточной функции, обеспечивающей наилучшую реставрацию, необходимо учитывать не только апертурные искажения изображения, вносимые искажающей системой, но также необходимо принимать во внимание спектральный состав изображения, а также спектральную интенсивность шума. При этом следует найти такой компромиссный вид частотной передаточной функции корректирующего фильтра, при котором заметность суммарных искажений, обусловленных остаточными апертурными искажениями и шумом, была бы для зрителя минимальна.

Задача должна быть сформулирована таким образом: для обеспечения наилучшей реставрации искаженного изображения посредством линейной фильтрации, требуется найти частотную передаточную функцию корректирующего фильтра $K_{\text{кор}}(\omega_x, \omega_y)$, при которой достигается наилучшее приближение скорректированного изображения $\hat{L}_{c}(x, y)$ к исходному неискаженному изображению $L_{c}(x, y)$ в рамках принятого критерия качества. Скорректированное изображение $\hat{L}_{c}(x, y)$ принято называть оценкой искаженного изображения L(x, y).

Переходя к решению поставленной задачи, примем в качестве допущения, что функция рассеяния точки искажающей системы $h_{\text{иск}}(x, y)$ не зависит от положения на плоскости изображения. Другими словами, будем считать, что искажающая система обладает свойством инвариантности. Кроме того, будем считать, что на корректируемое изображение наложен аддитивный гауссов шум, спектральная интенсивность которого $S_{\text{ш}}(x, y)$ также одинакова для всех точек изображения. В том случае, если условие инвариантности не выполняется, то, как уже отмечалось, корректируемое изображение может быть разбито на изопланарные области, в пределах которых это условие выполняется.

На рис. 5.13 приведена функциональная схема системы, включающая в себя искажающий фильтр, источник аддитивного гауссова шума и корректирующий фильтр. На этом рисунке использованы следующие обозначения: $L_c(x, y)$ — исходное неискаженное изображение; $M_c(\omega_x, \omega_y)$ — спектр исходного изображения; $L_{c\Omega}(x, y)$ — изображение, отличающееся от исходного наличием апертурных искажений; $M_{c\Omega}(\omega_x, \omega_y)$ — спектр изображения с апертурными искажениями; $L_{iii}(x, y)$ — шумовое изображение, т. е. функция, описывающая флуктуации яркости на плоскости изображения, обусловленные гауссовым шумом; L(x, y) — искаженное изображение, которое включает в себя как апертурные искажения, так и флуктуационный шум

$$L(x, y) = L_{c\Omega}(x, y) + L_{III}(x, y), \qquad (5.16)$$

 $\hat{L}_{c}(x, y)$ — наилучшая оценка искаженного изображения; $\hat{M}_{c}(\omega_{x}, \omega_{y})$ — спектр наилучшей оценки искаженного изображения; $h_{\mu c \kappa}(x, y)$ — функция рассеяния

точки искажающей системы, которая связана с ее частотной передаточной функцией $K_{\text{иск}}(\omega_x, \omega_y)$ интегральным преобразованием Фурье

$$h_{\mu c\kappa}(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K_{\mu c\kappa}(\omega_x, \omega_y) \exp\left[\mathbf{i}(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y;$$

 $h_{\text{кор}}(x, y)$ — функция рассеяния точки корректирующего фильтра, связанная с его частотной передаточной функцией $K_{\text{кор}}(\omega_x, \omega_y)$ интегральным преобразованием Фурье

$$h_{\text{kop}}(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} K_{\text{kop}}(\omega_x, \omega_y) \exp\left[\mathbf{i}(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y$$

Все это справедливо при условии, что значения $L_{c\Omega}(x, y)$ и L(x, y) остаются положительными для всех значений x и y. Как правило, это условие выполняется при работе с реальными изображениями. В отдельных точках, где $L_{c\Omega}(x, y)$ принимает отрицательные значения, эти значения необходимо заменять нулями, т. е. провести ограничения снизу.



Рис. 5.13. Функциональная схема системы

Можно было бы попытаться, как это иногда делают, в качестве простейшего критерия качества использовать средний квадрат ошибки ε_1^2 , представляющий расхождение между оценкой $\hat{L}_c(x, y)$ и исходным изображением $L_c(x, y)$, т. е.

$$\varepsilon_1^2 = \overline{\left[\hat{L}_c(x, y) - L_c(x, y)\right]^2},$$

который включает в себя два компонента $\varepsilon_1^2 = \varepsilon_c^2 + \varepsilon_m^2$, где ε_c^2 — средний квадрат ошибки, обусловленной не полностью скорректированными апертурными искаже-

ниями, $\epsilon_{\rm m}^2$ — средний квадрат ошибки, обусловленной не полностью подавленным шумом.

Однако эксперимент показывает, что эти компоненты не равнозначны для зрителя, оценивающего качество реставрированного изображения. При одной и той же величине обоих компонентов компонент ε_c^2 сильнее портит изображение, чем компонент ε_m^2 , т. е. имеет больший вес. Поэтому для оценки качества, точнее, степени испорченности изображения, было предложено использовать модифицированную оценку ε_2^2 :

$$\varepsilon_2^2 = \lambda \varepsilon_c^2 + \varepsilon_{\rm III}^2, \qquad (5.17)$$

где λ — весовой множитель, учитывающий большую заметность неточности воспроизведения изображения вследствие остаточных апертурных искажений, чем заметность неточности его воспроизведения, обусловленной остаточным шумом. Приведенный критерий ε_2^2 можно получить, исходя из функциональной модели зрения, описанной *главе 3*. В зависимости от условий наблюдения значение весового множителя λ лежит в пределах $4 \le \lambda \le 6$.

Основываясь на принятом критерии качества, найдем такую частотную передаточную функцию линейного корректирующего фильтра, которая обеспечит минимальное значение ϵ_2^2 .

Фильтрация корректируемого изображения может быть реализована как методом его свертки с функцией рассеяния точки корректирующего фильтра, так и в спектральной области. В целях наглядности изложения рассмотрим второй метод. Переходя к поиску частотной передаточной функции линейного корректирующего фильтра $K_{\text{кор}}(\omega_x, \omega_y)$, найдем вначале значения компонентов ε_c^2 и ε_m^2 .

Значение ϵ_c^2 может быть найдено следующим образом:

$$\varepsilon_{\rm c}^2 = \frac{1}{s_{\rm H3}} \left\{ \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left| K_{\rm kop} \left(\omega_x, \omega_y \right) K_{\rm HCK} \left(\omega_x, \omega_y \right) M_{\rm c} \left(\omega_x, \omega_y \right) - M_{\rm c} \left(\omega_x, \omega_y \right) \right|^2 d\omega_x d\omega_y \right\}.$$

В этой формуле выражение, стоящее в фигурных скобках, имеет смысл энергии разности двух изображений, одно из которых — исходное, а другое — получающееся после коррекции в условиях отсутствия шума.

Для простоты дальнейшего изложения будем считать, что $K_{\text{иск}}(\omega_x, \omega_y)$ и $K_{\text{кор}}(\omega_x, \omega_y)K_{\text{иск}}(\omega_x, \omega_y)$ представляют собой действительные, положительные функции, что соответствует отсутствию в изображении до и после его коррекции фазочастотных искажений. Сделанное допущение не является принципиальным, но

оно заметно упрощает математические выводы. С учетом этого допущения запишем

$$\varepsilon_{\rm c}^{2} = \frac{1}{4\pi^{2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{\left|M_{\rm c}\left(\omega_{x},\omega_{y}\right)\right|^{2}}{s_{_{\rm H3}}} \left[K_{\rm kop}\left(\omega_{x},\omega_{y}\right)K_{_{\rm HCK}}\left(\omega_{x},\omega_{y}\right)-1\right]^{2} d\omega_{x} d\omega_{y},$$

а поскольку, согласно определению (1.11),

$$\frac{\left|M_{c}\left(\omega_{x},\omega_{y}\right)\right|^{2}}{s_{_{H3}}}=S_{c}\left(\omega_{x},\omega_{y}\right),$$

где $S_{c}(\omega_{x},\omega_{y})$ — спектральная интенсивность изображения, будем иметь

$$\varepsilon_{\rm c}^2 = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S_{\rm c} \left(\omega_x, \omega_y \right) \left[K_{\rm kop} \left(\omega_x, \omega_y \right) K_{\rm HCK} \left(\omega_x, \omega_y \right) - 1 \right]^2 d\omega_x d\omega_y \right].$$

Значение среднего квадрата шумового компонента на выходе корректирующего фильтра, как не трудно видеть, будет равно

Подставляя значения ϵ_c^2 и ϵ_{III}^2 в (5.17), получим

$$\epsilon_{c}^{2} = \frac{1}{4\pi^{2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[K_{\text{kop}}^{2} \left(\omega_{x}, \omega_{y} \right) S_{\text{III}} \left(\omega_{x}, \omega_{y} \right) + \lambda K_{\text{kop}}^{2} \left(\omega_{x}, \omega_{y} \right) K_{\text{Hck}}^{2} \left(\omega_{x}, \omega_{y} \right) S_{c} \left(\omega_{x}, \omega_{y} \right) - (5.18) - 2\lambda K_{\text{kop}} \left(\omega_{x}, \omega_{y} \right) K_{\text{Hck}} \left(\omega_{x}, \omega_{y} \right) S_{c} \left(\omega_{x}, \omega_{y} \right) + \lambda S_{c} \left(\omega_{x}, \omega_{y} \right) \right] dx dy.$$

Так как подынтегральное выражение положительно, то для нахождения частотной передаточной функции корректирующего фильтра $K_{\text{кор}}(\omega_x, \omega_y)$, обеспечивающей минимум ε_2^2 , нужно минимизировать значение самой подынтегральной функции на всех пространственных частотах ω_x, ω_y . Для этого необходимо частотную передаточную функцию корректирующего фильтра $K_{\text{кор}}(\omega_x, \omega_y)$ выбрать таким образом, чтобы на всех пространственных частотах первая производная от подынтегральной

функции по $K_{\text{кор}}(\omega_x, \omega_y)$ обращалась бы в ноль, а четная производная наименьшего порядка, которая на этих частотах не обращается в ноль, была бы положительной. В случаях, представляющих интерес для практики, второе условие, как правило, выполняется, поэтому определение частотной передаточной функции корректирующего фильтра сделаем следующим образом. Вначале продифференцируем подынтегральную функцию выражения (5.18) по $K_{\text{кор}}(\omega_x, \omega_y)$, а затем производную приравняем нулю и решим получившееся уравнение относительно $K_{\text{кор}}(\omega_x, \omega_y)$. Выполняя дифференцирование и приравнивая результат нулю, находим

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right)S_{\text{III}}\left(\omega_{x},\omega_{y}\right)+ +\lambda K_{\text{kop}}\left(\omega_{x},\omega_{y}\right)K_{\text{uck}}^{2}\left(\omega_{x},\omega_{y}\right)S_{\text{c}}\left(\omega_{x},\omega_{y}\right) - -\lambda K_{\text{uck}}\left(\omega_{x},\omega_{y}\right)S_{\text{c}}\left(\omega_{x},\omega_{y}\right) = 0.$$

$$(5.19)$$

Решим это уравнение относительно $K_{\text{кор}}(\omega_x, \omega_y)$:

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = \frac{K_{\text{HCK}}\left(\omega_{x},\omega_{y}\right)S_{\text{c}}\left(\omega_{x},\omega_{y}\right)}{K_{\text{HCK}}^{2}\left(\omega_{x},\omega_{y}\right)S_{\text{c}}\left(\omega_{x},\omega_{y}\right) + S_{\text{III}}\left(\omega_{x},\omega_{y}\right)/\lambda}.$$
(5.20)

Фильтр, имеющий такую частотную передаточную функцию, называют фильтром Бакуса — Гильберта [12, 78]. В предельном случае, при $S_{\rm III}(\omega_x, \omega_y) \rightarrow 0$, т. е. в случае отсутствия шума, фильтр Бакуса — Гильберта переходит в инверсный фильтр, для которого

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = 1/K_{\text{HCK}}\left(\omega_{x},\omega_{y}\right).$$
(5.21)

В другом частном случае, когда $\lambda = 1$, фильтр Бакуса — Гильберта переходит в *фильтр Винера*, при этом имеем

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = \frac{K_{\text{HCK}}\left(\omega_{x},\omega_{y}\right)S_{\text{c}}\left(\omega_{x},\omega_{y}\right)}{K_{\text{HCK}}^{2}\left(\omega_{x},\omega_{y}\right)S_{\text{c}}\left(\omega_{x},\omega_{y}\right) + S_{\text{m}}\left(\omega_{x},\omega_{y}\right)}.$$
(5.22)

Этот второй случай соответствует выделению из шума одномерного сигнала, который можно представить себе как изображение, состоящее всего из одной строки. При наблюдении такого изображения зрительная система лишена возможности усреднять шум вдоль контуров, что и обуславливает равенство λ единице.

В общем случае в зависимости от того, является ли частотная передаточная функция искажающей системы действительной или комплексной, частотная передаточная функция корректирующего фильтра может быть как действительной, так и комплексной, поскольку фазочастотные искажения, если они вносятся искажающей системой, должны быть скорректированы корректирующим фильтром.

Обычно частотная передаточная функция корректирующего фильтра является неразделимой функцией в отношении переменных ω_x , ω_y , даже если все остальные функции в формуле (5.20) являются разделяющимися.

Выражение (5.20) допускает простую физическую трактовку, которая заключается в следующем. Частотная передаточная функция корректирующего фильтра в рассматриваемом случае выбирается таким образом, чтобы на пространственных частотах, где спектральные компоненты сигнальной составляющей велики, а шумовой — малы, она имела бы подъемы, а на пространственных частотах, где шумовые компоненты велики, а сигнальные — малы, она имела бы спады.

Все изложенное в полной мере справедливо также и при коррекции апертурных искажений телевизионного сигнала.

Практика показывает, что при коррекции апертурных искажений цветных изображений в присутствии шума изображения целесообразно преобразовывать к виду, при котором они будут состоять из яркостной и хроматических составляющих, а затем корректировать только их яркостный компонент. Это обусловлено тем, что зрительная система не использует высокочастотные хроматические компоненты (*см. разд. 3.11*).

5.7. Принципы обработки изображений, основанные на учете особенностей их восприятия зрительной системой

Недостатком традиционных методов коррекции изображений является то, что при их реализации в той или иной степени учитываются только характеристики изображений и помех, но, как правило, не принимаются во внимание особенности восприятия изображений зрительной системой человека, что делает эти методы малоэффективными. Проиллюстрируем сказанное рядом примеров.

Так в целях ослабления квазибелого гауссова шума на изображениях в простейшем случае иногда используют *низкочастотный фильтр*, амплитудно-частотная характеристика которого описывается гауссовой зависимостью. При этом методе коррекции используется различие в спектральных составах шума и изображений, спектральная интенсивность которых с ростом пространственных частот в отличие от спектральной интенсивности шума довольно быстро убывает. Ослабляя при фильтрации верхние пространственные частоты спектра, добиваются существенного ослабления среднеквадратичного значения шума при сравнительно незначительном ослаблении энергии изображений. Однако опыт показывает, что при этом одновременно с ослаблением шума имеет место размытие световых и цветовых границ, сильно портящее изображения. Обусловлено это тем, что в данном методе не учтена роль верхних пространственных частот при зрительном восприятии световых и цветовых и цветовых и цветовых границ.

Другим примером, иллюстрирующим неэффективность подобных методов, является применение фильтра Винера для фильтрации изображений, к которым добавлен аддитивный гауссов шум с произвольной спектральной интенсивностью $S_{\rm m}(x, y)$, а также адаптивного фильтра. В случае фильтра Винера в большей степени используется различие в спектральных интенсивностях изображения и шума, чем в первом случае. Особенностью фильтра Винера является то, что его частотнопередаточная функция $K_{\text{кор}}(\omega_x, \omega_y)$ выбирается с учетом спектральных интенсивностей конкретного фильтруемого изображения и шума

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = \frac{S_{\text{c}}\left(\omega_{x},\omega_{y}\right)}{S_{\text{c}}\left(\omega_{x},\omega_{y}\right) + S_{\text{III}}\left(\omega_{x},\omega_{y}\right)},$$
(5.23)

где $S_{c}(x, y)$ — спектральная характеристика конкретного фильтруемого изображения, ω_{x} и ω_{y} — пространственные частоты. Получаемые в данном случае результаты фильтрации оказываются несколько лучше, чем в предыдущем примере, но по-прежнему световые и цветовые границы на изображении размываются, хотя и в меньшей степени.

Попытка найти компромисс между уменьшением заметности шума и увеличением заметности размытия границ была реализована в фильтре Бакуса — Гильберта, который рассмотрен в *разд. 5.6.* Однако это лишь немного улучшило результаты коррекции, обеспечиваемые методами, приведенными в двух предыдущих примерах. Во всех этих случаях имеет место попиксельное приближение обработанного изображения к неискаженному образцу вне зависимости от важности для зрительной системы той или иной структуры, образуемой пикселами. Можно привести и другие примеры.

В последнее время формируется принципиально новый подход к обработке изображений, который базируется не только на учете характеристик изображений и помех, но также на максимально возможном учете особенностей восприятия изображений зрительной системой человека [46, 47, 120]. Изображение при этом рассматривается как композиция контурных, градиентных и фактурных компонентов. Конечной целью обработки при этом является правильное, "с точки зрения" зрительной системы, воспроизведение этих компонентов. При этом не важно, насколько точно воспроизводятся отдельные пикселы по отношению к эталонному изображению, особенно это относится к фактурному компоненту. Этот подход включает в себя два направления.

В методе, относящемся к первому направлению, на первом этапе обработки, исходя из известной или измеренной искажающей частотно-передаточной функции отображающей системы, выполняют первичную частотную коррекцию изображения, посредством которой восстанавливается крутизна световых и цветовых границ, но при этом, как правило, увеличивается зашумленность изображения. На втором этапе осуществляют адаптивную анизотропную фильтрацию полученного промежуточного препарата, при которой ослабляется шум, но резкость границ не нарушается, благодаря тому, что протяженная импульсная характеристика фильтра, имея ширину в один пиксел, всегда ориентирована вдоль границы [71]. Третий этап заключается в регенерации фактуры, о чем будет сказано далее.

Метод, относящийся ко второму направлению решения задач такого рода, рассмотрен в *разд. 6.10*.

5.8. Ослабление шума на изображениях методом адаптивной анизотропной фильтрации

Рассмотрим этот метод на примере фильтрации черно-белого полутонового изображения [71]. Важной особенностью этого метода, благодаря которой достигается ослабление шума на изображении при минимальном искажении самого изображения, является то, что сглаживание осуществляют в направлении, для которого градиент изменения яркости минимален, т. е. в направлении световых границ и линий [71, 156]. С этой целью, ширину импульсной характеристики сглаживающего фильтра выбирают равной одному пикселу, в то время как ее протяженность берут достаточно большой. Вследствие этого количество отсчетов, формирующих импульсную характеристику, также оказывается большим, что обеспечивает значительный фильтрующий эффект. Поскольку при реализации этого метода фильтрации импульсная характеристика всегда располагается в направлении минимального изменения яркости на изображении, то ее протяженность может быть выбрана достаточно большой без опасения размытия световых границ, т. к. она не будет их пересекать, а будет располагаться параллельно им. На рис. 5.14 в качестве примера показаны две такие импульсные характеристики. Черным цветом выделен центральный элемент импульсной характеристики. Достоинством импульсных характеристик, представляющих собой восьмисвязные структуры, подобных показанной на рис. 5.14, а, является то, что они позволяют лучше отслеживать световые границы на изображении. Применение же четырехсвязных импульсных характеристик, подобных приведенной на рис. 5.14, б, позволяет существенно сократить их набор по сравнению с восьмисвязными характеристиками, а значит, количество необходимых вычислительных операций, которые требуется выполнить на этапе адаптации (понятия четрехсвязности и восьмисвязности подробно рассмотрены в разд. 8.4). Однако применение таких характеристик приводит к более грубой их адаптации под световые границы и, как следствие, влечет за собой некоторые искажения фильтруемого изображения.



Рис. 5.14. Примеры форм импульсных характеристик



Рис. 5.15. Поясняющий пример: *a* — исходное тестовое изображение; *б* — это же изображение, но после его фильтрации с использованием восьмисвязных импульсных характеристик; *в* — это же изображение, но после фильтрации с использованием четырехсвязных импульсных характеристик

В качестве примера, демонстрирующего эти искажения, на рис. 5.15 приведены: исходное тестовое изображение, это же изображение, но после его фильтрации с использованием восьмисвязных импульсных характеристик, а также изображение, полученное после фильтрации исходного изображения с использованием четырехсвязных импульсных характеристик.

Из рисунка видно, что применение импульсных характеристик, являющихся четырехсвязными структурами, приводит практически к полному исчезновению на изображениях наклонных линий толщиной в один пиксел, в то время как использование импульсных характеристик, являющихся восьмисвязными структурами, к таким искажениям не приводит.

При реализации описываемого метода набор импульсных характеристик всевозможных форм заданной в диалоге протяженности генерируется перед началом фильтрации. В процессе фильтрации изображение, как обычно, сканируется строка за строкой. При этом для каждого нового положения центрального элемента импульсной характеристики на изображении отыскивается ее оптимальная форма. Делается это путем выбора подходящего экземпляра из множества ее сгенерированных экземпляров [71]. Критерием выбора оптимального экземпляра является минимум дисперсии σ_{ux}^2 яркости пикселов, перекрытых импульсной характеристикой, т. е. из всего множества импульсных характеристик выбирается та характеристика, для которой эта дисперсия минимальна. Расчет дисперсии для случая, когда все отсчеты импульсной характеристики одинаковы и равны 1/N, выполняется по формуле



где

$$\overline{L_{\rm c}} = \frac{1}{N} \sum_{k=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{\rm c}(k),$$

 $\overline{L_{\rm c}}$ — среднее значение яркости пикселов фильтруемого изображения, перекрываемых импульсной характеристикой; $L_{\rm c}(k)$ — яркость пиксела фильтруемого изображения, перекрываемого отсчетом импульсной характеристики, номер которого k; N — количество отсчетов в импульсной характеристике. После этого вычисляется яркость центрального пиксела путем усреднения яркости пикселов, оказавшихся в пределах выбранной импульсной характеристики.

В результате адаптивной анизотропной фильтрации контурные компоненты изображения сохраняются (в случае применения восьмисвязных структур), в то время как флуктуации, обусловленные шумом вследствие большой протяженности импульсной характеристики, ослабляются. Значения отсчетов импульсной характеристики могут быть выбраны по различным законам, но наибольшее ослабление белого шума достигается, если ее отсчеты одинаковы и равны 1/N. При этом в случае аддитивного гауссова белого шума отношение сигнала к шуму вследствие рассматриваемой фильтрации увеличивается не в \sqrt{N} раз, как можно было бы ожидать, а несколько меньше. Исследования показали, что этот эффект обусловлен тем, что алгоритм и при фильтрации участков изображения, где световые границы отсутствуют, каждый раз выбирает такой экземпляр из сгенерированного набора импульсных функций, при котором она покрывает пикселы, яркость которых максимально близка между собой. Так, например, при протяженности импульсной функции в 5 отсчетов увеличение отношения сигнала к шуму составляет не $\sqrt{5} = 2,24$, а только 1,58 раз. Для увеличения степени ослабления шума необходимо увеличивать протяженность импульсной характеристики N. Однако, как справедливо отмечено в [71], увеличение протяженности импульсной характеристики приводит к резкому возрастанию генерируемого набора импульсных характеристик, различающихся между собой по форме, и, как следствие, к резкому возрастанию времени обработки изображения. Так при протяженности импульсной характеристики N = 7 пикселов набор импульсных характеристик состоит из 197 различных экземпляров типа, показанного на рис. 5.14, δ , а при N = 9 пикселам количество экземпляров импульсных характеристик этого типа, различающихся между собой по форме, становится равным 1113.

На рис. 5.16, *а* показано исходное изображение, на рис. 5.16, *б* приведено это же изображение с добавленным к нему аддитивным гауссовым белым шумом. При этом отношение максимального перепада яркости от черного до белого в изображении к среднеквадратичному значению шума составило 20. На рис. 5.16, *г* приведено это же изображение после его фильтрации описанным выше методом при четырехсвязной импульсной характеристике, имеющей размер 1×9 пикселов, все отсчеты которой имеют одинаковую величину.



в





г

Рис. 5.16. Изображения, поясняющие эффективность метода [71]: *а* — исходное изображение; *б* — зашумленное; *в* — профильтрованное с применением импульсной характеристики квадратной формы размером 3×3 отсчета; *г* — профильтрованное в соответствии с данным методом при размере импульсной характеристики 1×9 отсчетов Для сравнения на рис. 5.16, e показано это же изображение, но после его фильтрации с использованием импульсной характеристики квадратной формы, размером 3×3 пиксела, все отсчеты которой также имеют одинаковую величину. Из сравнения рис. 5.16, e и r видно, что при фильтрации описанным методом световые границы на изображении размываются в меньшей степени, чем при фильтрации с использованием импульсной характеристики квадратной формы с тем же количеством отсчетов.

Реализация этого метода в случае цветных изображений сводится к тому, что каждый цветовой компонент должен быть профильтрован по отдельности.

Основным недостатком рассмотренного метода является необходимость в выполнении большого количества вычислительных операций на этапе адаптации импульсной характеристики, особенно это касается импульсных характеристик с восьмисвязной структурой.

Другим недостатком этого метода является ослабление фактурного компонента изображения. В дальнейшем данный метод был усовершенствован, благодаря чему отмеченный недостаток удалось ослабить [173].

В последнее десятилетие интенсивно исследуется методы ослабления аддитивных помех на изображении с использованием для этой цели вейвлет-преобразований, однако рассмотрение этих методов выходит за рамки данной книги. Читатель, желающий ознакомиться с этим направлением работ, найдет их изложение в публикациях [146, 158, 159, 161].

5.9. Фильтрация изображений в ограниченной пространственной области

В ряде случаев обработки изображений возникает необходимость в их фильтрации в пределах некоторой области, ограниченной произвольным контуром. Здесь возможны два случая.

В первом случае, когда контур задается пользователем и, как правило, имеет произвольный вид, фильтрация выполняется в два этапа. Вначале исходное изображение подвергается целиком фильтрации. После этого из профильтрованной версии исходного изображения вырезается область, ограниченная заданным контуром, а затем эта область аппликируется в исходное непрофильтрованное изображение. Рисунок 5.17 поясняет работу описанного алгоритма. Посредством этого алгоритма одно из лиц на фотографии воспроизводится резким, в то время как остальные лица фотографии воспроизведены размытыми. Этот алгоритм использован в графическом редакторе Photoshop.

Со вторым случаем мы встречаемся при решении ряда специальных задач, например, при фильтрации области изображения, ограниченной световой и/или цветовой границей, с целью ее разделения на градиентный и текстурный (фактурный) компоненты. В этой ситуации алгоритм, описанный выше, не применим, поскольку в областях изображения, которые прилегают к световым/цветовым границам, при фильтрации, например, методом свертки, будут захватываться области изображения, лежащие за пределами этих границ, а, следовательно, результаты фильтрации будут искажаться. Для того чтобы этого не происходило, импульсную характеристику фильтра необходимо адаптировать к фильтруемой области изображения. Поясним сказанное примером. Предположим, что импульсная характеристика представлена массивом чисел

$$h(k,n) = \begin{vmatrix} 0,04 & 0,04 & 0,04 & 0,04 & 0,04 \\ 0,04 & 0,04 & 0,04 & 0,04 & 0,04 \\ 0,04 & 0,04 & 0,04 & 0,04 & 0,04 \\ 0,04 & 0,04 & 0,04 & 0,04 & 0,04 \\ 0,04 & 0,04 & 0,04 & 0,04 & 0,04 \end{vmatrix}.$$



б



Рис. 5.17. Иллюстрация к фильтрации изображений в ограниченной пространственной области: *а* — исходное изображение; *б* — область, вырезанная из профильтрованной версии исходного изображения; *в* — исходное изображение после аппликации в него профильтрованной (размытой) части



Рис. 5.18. Иллюстрация адаптации импульсной характеристики фильтра к фильтруемой области изображения

При фильтрации области изображения, удаленной от границ, картина будет иметь вид, показанный на рис. 5.18, *а*. На этом рисунке малыми квадратами показаны пикселы изображения. Жирной линией обведена область, занимаемая на изображении импульсной характеристикой. Крестиком обозначен центр импульсной характеристики.

На рис. 5.18, б показан случай, когда импульсная характеристика начинает перекрывать область изображения, яркость которой сильно отличается от яркости фильтруемой области изображения, что недопустимо. Для того чтобы этого избежать, необходимо импульсную характеристику адаптировать к границе фильтруемой части изображения, переместив часть ее отсчетов, например, так, как это показано на рисунке, стрелками. В результате такой адаптации импульсная характеристика принимает вид, показанный на рис. 5.18, *в*. Заметим, что оптимальный вид импульсной характеристики, который она приобретает в результате адаптации, зависит от конкретной решаемой задачи. Близкие результаты могут быть получены при использовании метода билатеральной фильтрации, который рассматривается в *разд. 6.9*.

5.10. Подчеркивание световых границ на изображениях

Опыт показывает, что фотографические и телевизионные изображения с немного подчеркнутыми границами субъективно воспринимаются как изображения более высокого качества, чем фотометрически точные репродукции [50, 84]. Более того, мелкие объекты на таких изображениях обнаруживаются и опознаются быстрее и легче. Объяснить этот эффект, по-видимому, можно тем, что при наблюдении изображений в условиях хорошей освещенности в первичных отделах зрительной системы человека также происходит подчеркивание границ в результате воздействия тормозных зон рецептивных полей сетчатки. Поэтому, подчеркивая границы на изображении, мы как бы помогаем зрительной системе, поскольку главную роль при опознавании семантических изображений играют их границы и контуры.



Рис. 5.19. Зависимости $L_{c}(x, y)$, grad $L_{c}(x, y)$ и $\Delta(x, y)$ от *x* применительно к случаю вертикальной границы на изображении

В ахроматическом изображении под границей понимается линия, разделяющая две области на изображении $L_c(x, y)$, на которой происходит скачок яркости. В случае цветного изображения граница может быть образована как скачком яркости, так и изменением цветового тона (в меньшей степени, насыщенности). Как и ранее, рассмотрение проведем на примере ахроматических изображений. Формально граница может быть определена как геометрическое место точек, где абсолютное значение градиента функции достигает своего максимального значения, а оператор Лапласа —

$$\Delta(x, y) = \frac{\partial^2 L_{\rm c}(x, y)}{\partial x^2} + \frac{\partial^2 L_{\rm c}(x, y)}{\partial y^2} = 0,$$

при условии, что он расположен между двумя своими экстремумами, что соответствует точке перегиба функции $L_c(x, y)$. Рисунок 5.19, на котором показаны зависимости $L_c(x, y)$, grad $L_c(x, y)$ и $\Delta(x, y)$ применительно к случаю вертикальной границы на изображении, поясняет изложенное.

В настоящее время для подчеркивания границ разработан ряд методов как линейных, так и нелинейных. Ограничиваясь в этой главе линейными методами, рассмотрим вначале подчеркивание границ методом нерезкого маскирования, после чего рассмотрим другие методы подчеркивания границ, применяемые в настоящее время.

5.11. Метод нерезкого маскирования

При использовании метода нерезкого маскирования подчеркивание границ на изображении достигается за счет того, что из исходного изображения $L_{c}(x, y)$ вычитается его оператор Лапласа $\Delta(x, y)$, умноженный на весовой коэффициент *b*.

При этом распределение яркости в изображении с подчеркнутыми границами вычисляется по формуле

$$L_{\rm cr}\left(x,\,y\right) = L_{\rm c}\left(x,\,y\right) - b\Delta\left(x,\,y\right). \tag{5.24}$$

Изменением величины весового коэффициента *b* можно влиять на степень подчеркивания границ. На рис. 5.20 показаны зависимость яркости в исходном изображении вертикальной световой границы от координаты *x*, зависимость значения оператора Лапласа от *x* и зависимость яркости от *x* в изображении с подчеркнутыми границами. Из рисунка видно, что на графике зависимости $L_{\rm cr}(x, y)$ вследствие подчеркивания границ возникли два выброса, а сама зависимость в точке перегиба стала круче, благодаря чему создается впечатление повышенной резкости границы.



Рис. 5.20. Зависимость $L_{c}(x, y)$, $\Delta(x, y)$ и $L_{cr}(x, y)$

от координаты х применительно к случаю вертикальной границы на изображении

При использовании метода нерезкого маскирования величина весового коэффициента подбирается экспериментально, при этом следует иметь в виду, что выбор слишком большого значения *b* приводит к появлению на изображениях неприятных окантовок, а также к возрастанию уровня шума.

Рассмотрим операцию подчеркивания границ с позиций фильтрации изображения в частотной области, для этого выразим исходное изображение $L_{c}(x, y)$ через его

CHEKTP
$$M_{c}(\omega_{x},\omega_{y})$$
:

$$L_{c}(x,y) = \frac{1}{4\pi^{2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{c}(\omega_{x},\omega_{y}) \exp\left[i(\omega_{x}x+\omega_{y}y)\right] d\omega_{x}d\omega_{y}.$$
(5.25)

Для того чтобы написать выражение для оператора Лапласа, найдем вторые производные от $L_{c}(x, y)$ путем двойного дифференцирования выражения (5.25) по x и y

$$\frac{\partial^2 L_{\rm c}(x,y)}{\partial x^2} = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (-1)\omega_x^2 M_{\rm c}(\omega_x,\omega_y) \exp\left[\mathbf{i}(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y,$$
$$\frac{\partial^2 L_{\rm c}(x,y)}{\partial y^2} = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (-1)\omega_y^2 M_{\rm c}(\omega_x,\omega_y) \exp\left[\mathbf{i}(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y,$$

при этом

$$\Delta(x,y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (-1) \left(\omega_x^2 + \omega_y^2\right) M_c\left(\omega_x, \omega_y\right) \exp\left[\mathbf{i}\left(\omega_x x + \omega_y y\right)\right] d\omega_x d\omega_y .$$
(5.26)

Подставляя выражения для $L_{c}(x, y)$ и $\Delta(x, y)$ из (5.25) и (5.26) в формулу (5.24), получим

$$L_{\rm cr}(x,y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{\rm c}(\omega_x,\omega_y) \Big[1 + b(\omega_x^2 + \omega_y^2) \Big] \exp\Big[\mathbf{i} (\omega_x x + \omega_y y) \Big] d\omega_x d\omega_y . \quad (5.27)$$

Сопоставляя формулы (5.25) и (5.27), видим, что исходное изображение и изображение с подчеркнутыми границами различаются своими спектрами. Спектр исходного изображения $M_c(\omega_x, \omega_y)$ и спектр изображения с подчеркнутыми границами $M_{cr}(\omega_x, \omega_y) = M_c(\omega_x, \omega_y) \Big[1 + b (\omega_x^2 + \omega_y^2) \Big]$ различаются между собой множителем $\Big[1 + b (\omega_x^2 + \omega_y^2) \Big]$, который можно рассматривать как частотную передаточную функцию линейного фильтра, подчеркивающего границы.

Впервые метод нерезкого маскирования был применен в фотографии, а затем довольно успешно использовался при создании специальных кинофильмов, предназначенных для передачи по сети телевизионного вещания. В последнем случае благодаря подчеркиванию границ создавалась иллюзия большей четкости изображений на экранах телевизоров. В процессе изготовления изображения с подчеркнутыми границами фотографическим способом при его печати на негатив накладывается так называемая маска, которая представляет собой нерезкую, позитивную версию изготавливаемого изображения. В настоящее время используются два других способа реализации метода нерезкого маскирования. В первом из них изображение с подчеркнутыми границами получается следующим образом:

$$L_{\rm cr}(x,y) = cL_{\rm c}(x,y) - (c-1)L_{\rm c\Omega}(x,y), \qquad (5.28)$$

где $L_{c\Omega}(x, y)$ — нерезкая версия исходного изображения, а c — весовой коэффициент. Обратим внимание на тот факт, что в выражении (5.28) при всех значениях весового коэффициента обеспечивается правильное воспроизведение средней яркости изображения.

Нерезкая версия исходного изображения $L_{c\Omega}(x, y)$ в данном случае получается путем свертки исходного изображения с гауссовой импульсной характеристикой, что эквивалентно умножению его спектра $M_c(\omega_x, \omega_y)$ на частотную передаточную функцию

$$K(\omega_x, \omega_y) = \exp\left[-\left(\omega_x^2 + \omega_y^2\right)/\Omega^2\right], \qquad (5.29)$$

т. е.

$$M_{c\Omega}(\omega_x, \omega_y) = M_c(\omega_x, \omega_y) K(\omega_x, \omega_y), \qquad (5.30)$$

где Ω — так называемая приведенная частота.

Выражая в формуле (5.28) $L_{c}(x, y)$ и $L_{c\Omega}(x, y)$ через их спектры и учитывая (5.30), найдем после несложных преобразований

$$L_{\rm cr}(x,y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(M_{\rm c} \left(\omega_x, \omega_y \right) \left[c - (c-1) K \left(\omega_x, \omega_y \right) \right] \right) \times \exp \left[i \left(\omega_x x + \omega_y y \right) \right] \right) d\omega_x d\omega_y.$$
(5.31)

Далее, подставляя $K(\omega_x, \omega_y)$ из (5.29) в (5.31), разлагая $K(\omega_x, \omega_y)$ в ряд Тейлора и выбирая значение Ω достаточно большим для того, чтобы можно было ограничиться первыми двумя членами ряда, будем иметь

$$L_{\rm cr}(x,y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left[M_{\rm c} \left(\omega_x, \omega_y \right) \left[1 + \frac{c-1}{\Omega^2} \left(\omega_x^2 + \omega_y^2 \right) \right] \times \exp \left[\mathbf{i} \left(\omega_x x + \omega_y y \right) \right] \right] d\omega_x d\omega_y.$$
(5.32)

Сопоставление (5.27) и (5.32) показывает, что оба выражения идентичны, если принять, что $b = (c-1)/\Omega^2$.

В случае цифровой обработки изображения методом нерезкого маскирования фильтрация выполняется цифровым способом.

В другом способе подчеркивание границ методом нерезкого маскирования достигается непосредственно путем свертки исходного изображения $L_{c}(k,n)$ с импульсной характеристикой вида

$$h(k,n) = \begin{vmatrix} 0 & -b & 0 \\ -b & 1+4b & -b \\ 0 & -b & 0 \end{vmatrix}.$$

При этом изображение с подчеркнутыми границами вычисляется по формуле

$$L_{\rm cr}(k,n) = \sum_{k'=-1}^{1} \sum_{n'=-1}^{1} L_{\rm c}(k+k',n+n')h(k',n').$$

При подчеркивании границ в цветном изображении операция подчеркивания применяется ко всем трем его компонентам.

Рассмотренный метод используется в современных графических редакторах, например, в редакторе Photoshop. Однако, работая в этом графическом редакторе, следует иметь в виду, что в нем применена модифицированная версия фильтрации методом нерезкого маскирования. Подробнее эта модификация описана в *разд. 5.15*.

5.12. Другие линейные методы подчеркивания границ на изображениях

Другим методом, который близок методу нерезкого маскирования, является метод свертки исходного изображения с корреляционной маской, элементы которой определяются коэффициентами корреляции отсчетов на изображении [84]. Поскольку реальные стационарные изображения могут быть промоделированы двумерным процессом Маркова первого порядка, то для них корреляционная маска (импульсная характеристика) может быть представлена следующим образом:

$$h(k,n) = \begin{vmatrix} \rho_n \rho_k & -\rho_n \left(1 + \rho_k^2\right) & \rho_n \rho_k \\ -\rho_k \left(1 + \rho_n^2\right) & \left(1 + \rho_n^2\right) \left(1 + \rho_k^2\right) & -\rho_k \left(1 + \rho_n^2\right) \\ \rho_n \rho_k & -\rho_n \left(1 + \rho_k^2\right) & \rho_n \rho_k \end{vmatrix},$$

где ρ_k , ρ_n — коэффициенты корреляции между соседними по строке и столбцу отсчетами на изображении.

В частном случае, когда $\rho_k = \rho_n = 0$, т. е. в случае отсутствия корреляции между соседними отсчетами изображения, в результате свертки исходного изображения с маской подчеркивания границ не происходит.

В другом частном случае, при $\rho_k = \rho_n = 1$, подчеркивание границ достигает максимума, а корреляционная маска переходит в маску Лапласа

$$h(k,n) = \begin{vmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{vmatrix}.$$

Замечательным свойством метода нерезкого маскирования является то, что степень подчеркивания границ при его использовании не зависит от их ориентации. Объясняется это тем, что в данном методе подчеркивание границ достигается путем вычитания из исходного изображения оператора Лапласа, который представляет собой скалярную величину.

Однако в ряде практических задач, связанных с обнаружением и выделением границ, возникает необходимость подчеркивания границ с заранее заданной ориентацией. Достигается это посредством суммирования исходного изображения $L_{\rm c}(x, y)$ с его первой производной, взятой по заданному направлению z.

$$L_{\rm cr}(x,y) = L_{\rm c}(x,y) + b \frac{\partial L_{\rm c}(x,y)}{\partial z},$$

где $\frac{\partial L_{c}(x,y)}{\partial z}$ — производная от $L_{c}(x,y)$ по направлению z, составляющему угол

α с горизонтальной осью x, по определению равная

$$\frac{\partial L_{c}(x, y)}{\partial z} = \frac{\partial L_{c}(x, y)}{\partial x} \cos \alpha + \frac{\partial L_{c}(x, y)}{\partial y} \sin \alpha,$$

b — весовой коэффициент, определяющий степень подчеркивания границ [84]. Поскольку величина и знак производной зависят от направления, по которому производится дифференцирование, то и степень подчеркивания границ будет зависеть от этого направления.

В случае цифровой реализации этого метода вычисление первой производной по заданному направлению заменяется вычислением дискретной разности. При вычислении дискретной разности в направлении горизонтальной оси *x* имеем

$$\Delta L_{c}(k,n) = \left[L_{c}(k,n+1) - L_{c}(k,n)\right],$$

и соответственно при вычислении дискретной разности в направлении вертикальной оси у

$$\Delta L_{c}(k,n) = \left[L_{c}(k+1,n) - L_{c}(k,n)\right].$$

Аналогичным образом осуществляется вычисление дискретных разностей по другим направлениям. Двумерное дискретное дифференцирование удобно выполнять путем свертки исходного изображения с курсовыми градиентными масками (импульсными характеристиками), которые приведены далее:

...

...

север	h(k,n) =	$\begin{vmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{vmatrix};$
северо-восток	h(k,n) =	$\begin{vmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{vmatrix};$
восток	h(k,n) =	$\begin{vmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{vmatrix};$
юго-восток	h(k,n) =	$\begin{vmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{vmatrix};$
ЮГ	h(k,n) =	$\begin{vmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{vmatrix};$
юго-запад	h(k,n) =	$\begin{vmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{vmatrix};$
запад	h(k,n) =	$\begin{vmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{vmatrix};$
северо-запад	h(k,n) =	$\begin{vmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{vmatrix}.$

Использованные названия "курсов" масок указывают на направление склона перепада яркости, при котором маска дает максимальный отклик. Так, например, градиентная маска "Север" дает максимальный, положительный отклик при положительном изменении яркости в вертикальном направлении снизу вверх. Обратим внимание на то, что сумма элементов маски равна нулю, поэтому на участках изображения, где яркость не изменяется, значение дискретной разности равно нулю.



Рис. 5.21. Подчеркивание границ разными методами: а — исходное изображение;
 б — изображение с подчеркнутыми границами методом нерезкого маскирования;
 в — изображение с подчеркнутыми границами методом суммирования с первой производной, полученной при использовании градиентной маски "Восток"

На рис. 5.21 для сравнения приведены исходное изображение и две его версии, полученные в результате подчеркивания границ методом нерезкого маскирования и методом суммирования исходного изображения с его первой пространственной производной (дискретной разностью). Из рассмотрения этих изображений видно, что в первом случае усиление контраста на границах не зависит от их ориентации, во втором случае вертикальные границы максимально усилены, а горизонтальные границы остались такими же, как и на исходном изображении.

Рассмотренные выше линейные методы подчеркивания границ применяются как самостоятельно, так и как составная часть более сложных нелинейных методов, рассматриваемых в *главе* 6, посредством которых полутоновые изображения преобразуются в контурные.

В заключение этого раздела отметим, что эффективное применение методов подчеркивания границ возможно лишь в том случае, когда шум на изображении мал. При заметном уровне шума подчеркивание границ приводит к его значительному усилению, что сильно ухудшает изображение.

5.13. Ослабление дефектов изображения путем размытия

Реальные изображения, например негативы старых фотографий, часто содержат такие дефекты, как пылинки, царапины, а также другие мелкие неоднородности, которые при подготовке изображений к печати должны быть устранены. Поскольку энергия таких неоднородностей распределена в широком интервале пространственных частот, а энергия изображения — в сравнительно более узком интервале, в основном в области низких пространственных частот, эффективным средством их сглаживания может быть низкочастотная пространственная фильтрация, приводящая к их размытию. В соответствии с методами, рассмотренными в *разд. 5.2*, эта фильтрация может быть выполнена путем свертки исходного (реставрируемого) изображения со сглаживающей импульсной характеристикой h(k, n). Далее приведены три сглаживающие импульсные характеристики:

$$h(k,n) = \frac{1}{9} \begin{vmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix}, \ h(k,n) = \frac{1}{10} \begin{vmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{vmatrix}, \ h(k,n) = \frac{1}{16} \begin{vmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{vmatrix}.$$

Приведенные импульсные характеристики имеют две особенности. Во-первых, все элементы их массивов положительны. Во-вторых, элементы массивов нормированы таким образом, что их сумма равна единице, благодаря чему обеспечивается правильное воспроизведение средней яркости изображения.

При сглаживании дефектов на изображениях мы встречаемся с двумя противоречивыми требованиями. С одной стороны, необходимо ослабить на изображении дефекты в виде пылинок, царапин и т. п., а с другой стороны, надо сохранить резкость границ. Одним из способов разрешения этого противоречия является предварительное выделение областей, в пределах которых необходимо провести ослабление этих дефектов, а затем их сглаживание в пределах выделенных областей. Такая возможность имеется во многих графических редакторах и, в частности, в редакторе Photoshop.

5.14. Пределы увеличения разрешения

Проблема количественной оценки разрешающей способности изображающих систем возникла довольно давно, вероятно, вместе с созданием первых оптических приборов.

Как известно, вследствие дифракции световых лучей, исходящих от наблюдаемого объекта, в оптической системе любая его точка изображается в виде диска, окруженного дифракционными кольцами, при этом радиус этого диска зависит от размеров линз (зеркал), примененных в системе, и от длины световой волны. При использовании такой оптической системы, например, телескопа, для наблюдения или фотографирования двух близко расположенных звезд, их изображения в виде дисков, окруженных дифракционными кольцами, взаимно перекрываются, и если они расположены друг к другу достаточно близко, то изображения сливаются в один светящийся объект.

Одним из первых критериев, предложенных для оценки разрешающей способности зрительных труб, был критерий Рэлея. Согласно этому критерию две близко расположенные точки считаются разрешаемыми на изображении, если угловое расстояние ϕ между ними таково, что первый минимум дифракционной картины от первой точки совпадет с главным максимумом дифракционной картины от второй точки.
Величина, обратная этому, обычно малому, угловому расстоянию

$$\frac{1}{\varphi} \approx \frac{R}{0,61\lambda}$$

называется разрешающей способностью объектива трубы. В этой формуле использованы следующие обозначения: R — радиус линзы, λ — длина световой волны.

Так как полоса пространственных частот, пропускаемых оптической системой, прямо пропорциональна радиусу линзы, то из критерия Рэлея непосредственно следует, что разрешающая способность оптической системы определяется этой полосой пропускания.

Развитие фотографии и радиоэлектроники, в частности, телевидения, а также связанные с ним исследования проблем разрешения привели к выводу, что разрешающая способность изображающей системы, кроме того, в сильной степени зависит от уровня шума, присутствующего в изображении.

Поясним сказанное на примере наблюдения зашумленной фотографии двух близко расположенных объектов, показанных на рис. 5.22.



Рис. 5.22. К пояснению разрешения при наличии аддитивного шума

Будем считать, что объекты разрешаются, если вероятности правильного обнаружения каждого из них, а также вероятность правильного обнаружения разделяющего их интервала превышает некоторое заданное пороговое значение $p_{\rm пор}$. В противном случае будем считать, что объекты не разрешаются. В зависимости от площади объектов и разделяющего их интервала, а также в зависимости от создаваемого объектом перепада яркости ΔL_c над яркостью фона L_{ϕ} это пороговое значних уровнях шума. Чем больше площадь объектов и разделяющего их интервала и чем больше перепад яркости ΔL_c , тем при большем уровне шума объекты перестают разрешаться. При аддитивном белом гауссовом шуме эта зависимость имеет вид

$$\frac{\Delta L_{\rm c}a}{\sqrt{S_{\rm III}}} = C \,, \tag{5.33}$$

где $S_{\rm m}$ — спектральная интенсивность белого гауссова шума, C — постоянный коэффициент, зависящий от принятого значения $p_{\rm пор}$. Эта зависимость обусловлена

тем, что при наблюдении крупных объектов белый шум в большей степени усредняется зрительной системой по поверхности объектов и разделяющего их интервала, чем при наблюдении мелких объектов. Решая уравнение (5.33) относительно 1/a, найдем зависимость разрешающей способности изображающей системы от спектральной интенсивности шума и перепада яркости ΔL_c :

$$1/a = \Delta L_{\rm c} / \left(C \sqrt{S_{\rm III}} \right). \tag{5.34}$$

Подробное изложение этой проблемы имеется в [49].

Из изложенного следует, что разрешающая способность изображающей системы ограничивается не только полосой пространственных частот, которую система способна передавать, но также и уровнем шума, наложенного на изображение.

Однако возможны случаи, когда разрешение изображающей системы может превышать пределы, определяемые полосой передаваемых пространственных частот. Это так называемый феномен сверхразрешения, который мы рассмотрим далее.

Известно, что при наблюдении объектов с применением оптических систем апертура объектива (или зеркала), определяемая его радиусом, ограничивает ширину спектра пространственных частот, пропускаемых системой, величиной ω_{rp} . При

этом область пространственных частот ω_x , ω_y , передаваемых оптической системой, определяется неравенством

$$\sqrt{\omega_x^2 + \omega_y^2} \le \omega_{\rm rp}$$

Рисунок 5.23 поясняет изложенное.



Рис. 5.23. К определению зоны прозрачности оптической системы

Ограничение полосы пропускаемых частот, как уже было отмечено, приводит к размытию каждой из точек объекта на изображении и возникновению вокруг них дифракционных колец.

Возникает вопрос: возможна ли экстраполяция спектра изображения за пределы зоны прозрачности? Если мы ничего не знаем относительно изображения наблюдаемого объекта, то никакая осмысленная экстраполяция спектра невозможна, т. к. можно предложить неограниченное количество его различных продолжений в зоне непрозрачности.

Однако ситуация в корне меняется, если нам априорно известно, что изображение наблюдаемого объекта, или объектов, имеет, например, ограниченную пространственную протяженность. Это условие всегда выполняется, поскольку любой оптический прибор имеет ограниченное поле зрения.

Для простоты изложения рассмотрим вначале одномерную задачу, полагая, что яркость изображения, воспроизводимого системой, зависит только от одной пространственной координаты x. В этом случае условие ограниченной пространственной протяженности изображения величиной X может быть записано как

$$L_{\mathrm{c}}(0) = 0$$
 при $|x| \ge X$.

Вычисляя спектр пространственно ограниченного изображения, найдем

$$M_{\rm c}(\omega_x) = \int_{-X}^{X} L_{\rm c}(\omega_x) \exp(-\mathbf{i}\omega_x x) d\omega_x, \qquad (5.35)$$

при этом *n*-я производная от $M_{c}(\omega_{x})$ по ω_{x} может быть выражена как

$$M_{\rm c}^{(n)}(\omega_x) = (-\mathbf{i})^n \int_{-X}^X x^n L_{\rm c}(\omega_x) \exp(-\mathbf{i}\omega_x x) d\omega_x.$$
 (5.36)

Для любого реального изображения интегралы в выражении (5.36) являются единственными и вследствие конечных пределов интегрирования конечными. Отсюда следует, что для любого реального изображения могут быть найдены все производные от его спектра, а это значит, что спектр $M_c(\omega_x)$ — функция аналитическая. Но аналитическая функция $M_c(\omega_x)$ может быть однозначно определена для всех значений ω_x , если известны ее значения на конечном интервале частот, или если известны значения всех ее производных на одной частоте. Сделать это можно, например, с помощью ряда Маклорена

$$M_{\rm c}(\omega_x) = M_{\rm c}(0) + \frac{\omega_x}{1!} M_{\rm c}'(0) + \frac{\omega_x^2}{2!} M_{\rm c}''(0) + \cdots$$

В случае реального двумерного изображения ряд Маклорена для спектра изображения $L_{c}(x, y)$ может быть записан в виде

$$M_{c}(\omega_{x}, y) = M_{c}(0, 0) + \frac{1}{1!} \left\{ \frac{\partial M_{c}(0, 0)}{\partial \omega_{x}} \omega_{x} + \frac{\partial M_{c}(0, 0)}{\partial \omega_{y}} \omega_{y} \right\} + \frac{1}{2!} \left\{ \frac{\partial^{2} M_{c}(0, 0)}{\partial \omega_{x}^{2}} \omega_{x}^{2} + 2 \frac{\partial^{2} M_{c}(0, 0)}{\partial \omega_{x} \partial \omega_{y}} \omega_{x} \omega_{y} + \frac{\partial^{2} M_{c}(0, 0)}{\partial \omega_{y}^{2}} \omega_{y}^{2} \right\} + \cdots$$

В случае, если объектив в зоне прозрачности вносит частотные искажения, то перед использованием рассматриваемого метода они должны быть скорректированы путем применения, например, инверсной фильтрации.

Таким образом, даже использование такой, казалось бы, незначительной информации относительно ограниченной пространственной протяженности изображения позволяет однозначно экстраполировать пространственный спектр изображения за пределы зоны прозрачности и тем самым обеспечить "сверхразрешение".

К сожалению, для того чтобы практически реализовать рассмотренный метод, необходимо, чтобы уровень шума в изображении был бы ничтожно мал. Объясняется это тем, что наличие шума приводит к погрешностям при вычислении производных. Величина этих погрешностей тем больше, чем выше порядок производных, поэтому при заметном уровне шума результаты экстраполяции сводятся на нет. Вследствие этого для успешной экстраполяции спектра и, как следствие, практической реализации "сверхразрешения" необходимо предварительно уменьшать уровень шума в изображении, например, путем накопления сигнала, как это делается в астрономии или радиоастрономии.

Метод накопления сигнала реализуют двумя способами.

При реализации первого способа исходят из того, что сигнал, накапливаемый преобразователем "свет/сигнал" от наблюдаемого объекта, например от звезды, с течением времени наблюдения возрастает прямо пропорционально этому времени T, поскольку средняя яркость объекта с течением времени остается неизменной. Среднеквадратичное же значение накопленного белого шума с течением времени возрастает прямо пропорционально только корню квадратному из T, т. к. флуктуации шума во времени некоррелированы. Благодаря этому отношение сигнала к шуму ψ при увеличении времени наблюдения будет возрастать пропорционально корню квадратному от T, т. е.

$$\psi \sim \sqrt{T}$$

При реализации второго способа результирующее изображение получают путем суммирования ряда изображений, полученных при одинаковых экспозициях. При этом в силу некоррелированности шума на суммируемых изображениях его среднеквадратичное значение будет возрастать прямо пропорционально корню квад-

ратному из числа суммируемых изображений N, поэтому отношение сигнала к шуму будет возрастать прямо пропорционально, N, т. е.

$$\psi \sim \sqrt{N}$$
.

Обычно эти два способа реализуют последовательно.

В настоящее время развит ряд других методов, в которых путем использования априорного знания об обрабатываемом изображении удается заметно увеличить разрешение. К таким методам относится, например, метод Шелла и Биро, который позволил практически реализовать "сверхразрешение" при наблюдении звезды альфа Лебедя [78].

5.15. Линейная фильтрация изображений в графических редакторах

Большинство графических редакторов, таких как Photoshop, Corel PHOTO-PAINT и им подобных, предназначено для использования в настольных издательских системах. Поэтому методы фильтрации, используемые в этих редакторах, предназначены в основном для подготовки изображений к печати. Вместе с тем эти редакторы используются также при обработке изображений, предназначенных для применения в мультимедийных приложениях, Интернете, а также в ряде других областей. В этих редакторах предусмотрены две группы вариантов линейной фильтрации изображений: одна группа — это размытие, при котором резкие световые границы делаются более размытыми (смягчаются), и другая группа — это увеличение резкости, достигаемое за счет подчеркивания границ. Каждая из этих групп включает в себя ряд вариантов. Так, например, в графическом редакторе Photoshop в меню **Blur** (Размытие) предлагаются следующие варианты этой операции.

- □ Average (Среднее). Вычисляет среднюю яркость изображения и заполняет его ровным фоном этой яркости.
- Blur (Размытие). Фильтрация осуществляется путем свертки обрабатываемого изображения с импульсной характеристикой, протяженность которой составляет 3×3 отсчета. Этот вариант фильтрации применяется для ослабления заметности шума на изображении, а также для смягчения цветовых переходов на резких границах.
- □ Blur More (Размытие +). Выполняет ту же операцию, что и предыдущий вариант, но в 3—4 раза сильнее.

□ Gaussian Blur (Размытие по Гауссу). Этот вид фильтрации осуществляется путем свертки обрабатываемого изображения с импульсной характеристикой, которая описывается гауссовой функцией. Величина размытия изображения может регулироваться путем установки соответствующего значения условного радиуса кружка рассеяния r_e. Чем больше радиус, тем больше размытие. В частности, этот вид фильтрации применяют для создания эффекта дымки при редактировании изображений в соответствии с творческими задачами автора.

- □ Lens Blur (Размытие оптикой). Имитирует размытие изображения, обусловленное плохой фокусировкой оптики.
- Motion Blur (Размытие в движении). Данный вид фильтрации осуществляется путем свертки обрабатываемого изображения с импульсной характеристикой, имеющей протяженность только в одном, заданном в диалоге, направлении. Благодаря такому виду импульсной характеристики размытие изображения происходит только в одном заданном направлении, что создает эффект смаза, получающийся при съемке движущегося изображения. Величина смаза регулируется в диалоге.
- **Radial Blur** (Радиальное размытие) Этот вид фильтрации в зависимости от установок в диалоговой панели позволяет имитировать размытие изображения, которое получается при его вращении или при увеличении лупой. В первом случае эффект достигается за счет того, что импульсная характеристика, с которой осуществляется свертка изображения, имеет вид отрезка дуги окружности, причем угловой размер этой дуги остается постоянным, а следовательно, чем дальше от центра происходит фильтрация, тем размытие получается больше. Во втором случае импульсная характеристика имеет вид отрезка радиуса, соответственно ориентированного для каждого направления, при этом протяженность импульсной характеристики возрастает с увеличением расстояния от центра. Таким образом, для каждой точки исходного изображения, подвергаемого операции свертки, протяженность импульсной характеристики зависит от расстояния между этой точкой и центром преобразования.

Величина размытия и положение центра, относительно которого производятся преобразования, задаются в диалоговом режиме. На рис. 5.24 приведены изображения, полученные в результате применения описанных вариантов фильтрации.

□ Smart Blur ("Умное" размытие). Один из видов размытия.



Рис. 5.24. Примеры эффектов, получаемых при фильтрации изображения с использованием различных видов импульсных характеристик:
а — исходное изображение; б — результат фильтрации с применением импульсной характеристики в виде отрезка дуги; в — результат фильтрации с применением импульсной характеристики в виде отрезка радиуса

Рассмотренные варианты фильтрации применяются для создания всевозможных эффектов при подготовке изображений к печати или для мультимедийных приложений.

Другая группа вариантов фильтрации графического редактора Photoshop, включает в себя следующие типы:

- Sharpen (Резкость). В данном случае фильтрация осуществляется путем свертки обрабатываемого изображения с импульсной характеристикой, имеющей отрицательные боковые лепестки, протяженность которой составляет 3×3 отсчета (пиксела). Этот вид фильтрации применяется для усиления резкости границ в изображении.
- □ Sharpen More (Резкость +). Выполняет ту же операцию, что и предыдущий вариант, но в большей степени подчеркивает границы.
- □ Sharpen Edges (Резкость на краях). С двух сторон границы создается выброс, составляющий 25% от перепада яркости на границе.
- Unsharp Mask (Нерезкое маскирование). Этот вид фильтрации реализуется методом нерезкого маскирования. В отличие от классического метода нерезкого маскирования в варианте, который использован в редакторе Photoshop, добавлены еще две регулировки. Кроме установки степени подчеркивания границ имеется еще возможность устанавливать радиус импульсной характеристики, определяющий величину размытия нерезкой версии изображения, вычитаемого из исходного изображения, для получения изображения с подчеркнутыми границами, а также порог, которым ограничивается снизу нерезкая версия перед вычитанием. Отметим, что пороговая операция, введенная в алгоритм фильтрации, делает эту обработку нелинейной.

Аналогичные инструменты с несколько отличающимися названиями также имеются в графическом редакторе Corel PHOTO-PAINT.

Дополнительную информацию по методам реализации линейной фильтрации изображений можно найти в [5, 12, 103].

Глава 6



Нелинейная обработка изображений

6.1. Искажения воспроизведения градаций яркости

Динамический диапазон изменения яркости в изображаемых сценах обычно существенно больше динамического диапазона, который может быть воспроизведен в изображениях. Поэтому при съемке изображений в них неизбежно вносятся искажения, которые выражаются в ограничении динамического диапазона как в области темных полутонов, так и в области светлых полутонов. Поскольку динамический диапазон зрительной системы человека, адаптированной к яркости наблюдаемой сцены, также ограничен, то при правильном выборе воспроизводимого интервала яркости на изображении эти искажения в большинстве случаев (но не всегда) оказываются незаметными, а следовательно, не снижают качества изображения.

Кроме отмеченных искажений изображающие системы и их отдельные звенья, такие как: фотографические системы, воспроизводящие устройства и некоторые другие, вследствие нелинейности световых характеристик вносят искажения, которые приводят к перераспределению градаций яркости внутри воспроизводимого на изображении динамического диапазона яркости. Искажения этого типа по-разному проявляются на черно-белых и на цветных изображениях.

На черно-белых изображениях перераспределение градаций яркости внутри воспроизводимого интервала яркостей приводит к сдвигу тональности изображения. Этот сдвиг будет в область светлого в том случае, если световая характеристика носит насыщающийся характер, при этом изображение в целом становится более светлым. В случае же, если крутизна световой характеристики возрастает с увеличением яркости, сдвиг будет в область темного, в результате чего в целом изображение становится более темным. Результатом этого сдвига является потеря на изображении малоконтрастных деталей, в первом случае в области светлой, во втором случае в области темной части динамического диапазона. Кроме того, в зависимости от характера изменения тональности изображение воспринимается или как более контрастное, или как более вялое. Эта особенность восприятия эффективно используется на практике. В системах черно-белого телевидения, в черно-белом кино и в черно-белой фотографии с этой целью в световые характеристики, влияющие на распределение яркости в изображении, преднамеренно вводится небольшая нелинейность.

Перераспределение градаций интенсивности в цветных изображениях проявляется иначе. Даже небольшая нелинейность световых характеристик приводит к неправильному воспроизведению как насыщенности, так и цветового тона изображаемых объектов. В этом случае желтые цвета могут воспроизводиться как коричневые, телесные цвета превратиться в бордово-красные и т. п. Поэтому световые характеристики изображающих систем, предназначенных для передачи и воспроизведения цветных изображений, стремятся сделать линейными путем коррекции составляющих их звеньев.

Рассмотрим особенности световых характеристик типичных звеньев изображающих систем.

Фотоматериалы

В фотографии аналогом световой характеристики, устанавливающей связь между яркостью элемента фотографируемой сцены и яркостью этого элемента на изображении, является сенситометрическая характеристика, которая устанавливает связь между экспозицией *Et* данного элемента и его оптической плотностью *D*

$$D = f(Et), \tag{6.1}$$

где E — освещенность элемента на поверхности фотоматериала, t — время экспозиции. При этом связь между оптической плотностью и коэффициентом пропускания проявленного фотоматериала τ определяется соотношением

$$D = -\lg \tau$$
.

Для реальных фотоматериалов изменение экспозиции от 0 до ∞ приводит к изменению оптической плотности в ограниченных пределах от $D_{\text{мин}}$ до $D_{\text{макс}}$.

Это следует из того, что даже при нулевой экспозиции коэффициент пропускания проявленного фотоматериала вследствие наличия вуали оказывается меньше единицы.

Фотографический процесс, как правило, включает в себя два этапа. На первом этапе при фотографировании сцены получается негатив, с которого на втором этапе изготавливается позитивное изображение. Поскольку в конечном итоге нас интересует позитивное изображение, то имеет смысл рассматривать зависимость оптической плотности элемента позитивного изображения D_{no3} , а еще лучше — зависимость коэффициента пропускания позитивного изображения τ_{no3} от его экспозиции. В общем случае эта зависимость нелинейная, более того, на ограниченном интервале экспозиций она может быть аппроксимирована степенной функцией

$$\tau_{\Pi O 3} = \beta (Et)^{\gamma},$$

где β и γ — коэффициенты, зависящие от типа фотоматериала и режима фотопроцесса. Для сохранения правильной цветопередачи при получении цветных изображений значение у должно быть равно единице.

В черно-белой фотографии и черно-белых кинофильмах для создания иллюзии повышенной контрастности ("сочности") изображений значения у выбирают равными 1,3—1,4.

Преобразователи изображения в видеосигнал

В современных преобразователях изображения в видеосигнал обычно используют датчики на основе приборов с зарядовой связью (видеокамеры, планшетные сканеры), а также плумбиконы (видеокамеры), фотоэлектронные умножители (барабанные сканеры). Световые характеристики всех этих приборов в пределах рабочего диапазона линейны. Единственным видом нелинейных искажений, которые вносят эти приборы, являются искажения, обусловленные ограниченностью их динамического диапазона. Отсюда следует, что для того чтобы избежать на изображениях, получаемых посредством этих приборов, нелинейных искажений, которые впоследствии не смогут быть скорректированы, необходимо в самом начале согласовать динамический диапазон яркостей изображения с динамическим диапазоном изображаемой сцены или сканируемого изображения. Независимо от названия конкретных регулировок, используемых в том или ином приборе, фактически это достигается путем правильного выбора экспозиции.

Воспроизводящие устройства

Наиболее распространенными воспроизводящими устройствами, преобразующими видеосигнал в изображение, в настоящее время являются жидкокристаллические панели, а также кинескопы. Световая характеристика (характеристики, в случае масочных кинескопов, предназначенных для воспроизведения цветных изображений), представляющая собой зависимость яркости свечения экрана L_c от напряжения e_m , приложенного к модулятору трубки, нелинейна и может быть аппроксимирована степенной функцией

$$L_{\rm c} = \beta_{\rm \kappa} \left(e_m + e_0 \right)^{\gamma_{\rm K}} , \label{eq:Lc}$$

где β_{κ} — коэффициент, согласующий размерности; e_0 — напряжение запирания; γ_{κ} — показатель степени, который в зависимости от типа трубки лежит в пределах 2,5—3,5. Световые характеристики жидкокристаллических панелей носят более сложный *S*-образный характер.

В современных телевизионных системах и компьютерных мониторах единственным источником нелинейных искажений, вызывающих перераспределение градаций яркости в рабочем диапазоне, являются воспроизводящие устройства. Для коррекции этих искажений применяют так называемые *гамма-корректоры* — устройства, амплитудная характеристика которых имеет вид, обратный виду амплитудной харак-

теристики воспроизводящего устройства. В случае применения кинескопов амплитудная характеристика гамма-корректора имеет вид

$$e_c = \beta_{\gamma} e_{\text{c.bx}}^{1/\gamma_{\text{K}}}$$
,

где β_{γ} — коэффициент, согласующий размерности; $e_{c.вx}$ — значение сигнала на входе гамма-корректора; e_c — значение сигнала на выходе гамма-корректора. Гамма-корректор вносит в сигнал изображения предыскажения, которые затем корректируются нелинейной световой (амплитудной) характеристикой кинескопа. Световая характеристика черно-белого кинескопа, амплитудная характеристика гамма-корректора, а также результирующая характеристика последовательно включенных гамма-корректора и кинескопа

$$L = \beta_{\kappa} \left(\beta_{\gamma} e_{c.BX}^{1/\gamma_{\kappa}} \right)^{\gamma_{\kappa}} = \beta_{\kappa} \beta_{\gamma}^{\gamma_{\kappa}} e_{c.BX}$$

приведены на рис. 6.1.



Рис. 6.1. Световые характеристики кинескопа, гамма-корректора и результирующая

Для воспроизведения цветного изображения на экране масочного кинескопа необходимо применять 3 гамма-корректора, по одному для каждого канала цветности. Аналогичным образом решается проблема коррекции нелинейных искажений, обусловленных нелинейностью световых характеристик жидкокристаллических панелей.

6.2. Коррекция искажений воспроизведения градаций яркости

В предыдущем разделе были рассмотрены искажения, вызванные нелинейностью амплитудных характеристик звеньев системы, предназначенной для съемки, передачи и воспроизведения изображений. В ряде случаев, как, например, в случае кинескопа, амплитудные (световые) характеристики могут быть аппроксимированы

степенными функциями, в других случаях, например, в случае жидкокристаллической панели, они могут носить более сложный характер.

Для коррекции световых характеристик системы (ее калибровки) необходимо вначале произвести измерения характеристик для каждого из звеньев, составляющих систему, а затем найти амплитудную характеристику гамма-корректора. Поясним это на примере коррекции сквозной световой характеристики системы, предназначенной для передачи черно-белых изображений, функциональная схема которой приведена на рис. 6.2. В целях общности рассмотрения будем считать, что световые характеристики и датчика сигнала изображения, и воспроизводящего устройства могут быть нелинейными. Для измерения световых характеристик системы необходимо использовать тестовое изображение, которое бы включало в себя градационный клин, измеритель величины сигнала изображения (например, калиброванный осциллограф) и микрофотометр для измерения яркости выбранной ступеньки градационного клина на экране воспроизводящего устройства.



Рис. 6.2. Функциональная схема системы

Измерения начинают со световой характеристики датчика сигнала изображения. При ее измерении для каждой ступеньки градационного клина, яркость которой обозначим через $L_{\text{н.n.}}$, где n — номер ступеньки, находят соответствующую ей величину сигнала e_n и данные записывают в табл. 6.1, которая называется *таблицей соответствия*. Затем при измерении световой характеристики воспроизводящего устройства устанавливают такие значения сигнала на его входе $e_{\text{к.n.}}$, при которых яркость свечения его экрана $L_{\text{р.n.}}$ будет равна или прямо пропорциональна значениям яркости соответствующих ступенек градационного клина $L_{\text{н.n.}}$. После заполнения таблицы соответствия требуемая амплитудная характеристика гаммакорректора определится путем прочтения ее 2-й и 3-й строк

$$e_{\mathbf{K},n} = f\left(e_n\right).$$

В случаях, когда не требуется высокая степень точности корректировки сквозной световой характеристики, можно использовать упрощенный метод. В этом методе используется замечательное свойство зрения, заключающееся в том, что разностный порог определяется яркостью адаптации (*см. разд. 3.3*). Благодаря этому свойству зрения при наблюдении градационного клина на фоне большого поля, имеющего яркость, равную яркости самой яркой его ступеньки, пороговая разность будет одинакова и для светлых, и для темных участков, если геометрические размеры клина невелики. В этом случае критерием, по которому можно судить о линейности световой характеристики, является одинаковая различимость и светлых, и темных ступенек.

Таблица 6.1. Таблица соответствия

1	$L_{{}_{\mathrm{H}.n}}$			
2	e _n			
3	e _{k.n}			
4	L _{p.n}			

Рассмотренный способ удобно использовать для корректировки, например, световых характеристик кинескопа дисплея. Для этого необходимо изготовить и записать в память компьютера тестовое изображение в виде белого поля размером во весь экран и яркостью, соответствующей 255 уровню квантования, с аппликированным в него узким градационным клином из 32 ступенек, отличающихся друг от друга по яркости на один и тот же интервал.

Упрощенные способы корректировки световой характеристики кинескопа (корректировки гамма), основанные на использовании свойств зрения, в настоящее время используются в компьютерных приложениях. Примером тому может служить утилита Gamma, поставляемая вместе с графическим редактором Photoshop.

6.3. Видоизменение гистограмм

При рассмотрении в предыдущем подразделе метода нахождения амплитудной характеристики гамма-корректора мы исходили из того, что световые (амплитудные) характеристики звеньев корректируемой системы, которые вносят нелинейные искажения, заранее известны. Однако возможны случаи, когда единственное, чем мы располагаем, — это набор изображений, подлежащих коррекции, о которых априорно известно, что градации яркости в этих изображениях воспроизведены неправильно. На первый взгляд, задача отыскания амплитудной характеристики гамма-корректора при такой ее постановке может показаться неразрешимой, т. к. неизвестна световая характеристика системы, которую нужно скорректировать. Тем не менее ситуация не столь безнадежна, если нам известны плотности вероятности (гистограммы) распределения яркости в неискаженных и искаженных изображениях данного класса. Первые, как правило, известны, а вторые могут быть измерены на изображениях, которые нужно исправить. В этом случае принцип нахождения амплитудной характеристики гамма-корректора основан на том, что плотности вероятности неискаженного и искаженного изображений связаны известным соотношением

$$W_{1}(L_{c1}) = W\left[\phi^{(-1)}(L_{c1})\right] \frac{d\phi^{(-1)}(L_{c1})}{dL_{c1}}, \qquad (6.2)$$

где $W(L_c)$ — плотность вероятности распределения яркости L_c в неискаженном изображении; $W_1(L_{c1})$ — плотность вероятности распределения яркости L_{c1} в ис-

каженном изображении; $L_c = \varphi^{(-1)}(L_{c1})$ — функция, обратная световой характеристике системы, вносящей нелинейные искажения, которую обозначим $L_{c1} = \varphi(L_c)$. По своему смыслу функция $L_c = \varphi^{(-1)}(L_{c1})$ представляет собой искомую амплитудную характеристику гамма-корректора.

Однако аналитическое решение этой задачи в большинстве случаев либо затруднительно, либо вообще невозможно. Поэтому для нахождения амплитудной характеристики гамма-корректора воспользуемся графическим способом ее решения. С этой целью обратимся к рис. 6.3, на котором приведены плотности вероятности распределения яркости в неискаженных $W(L_c)$ и искаженных $W_1(L_{c1})$ изображениях данного класса. Разделим площадь S_{Σ} , заключенную под кривой $W(L_c)$, показанной на рис. 6.3, *а*, вертикальными отрезками на *K* равных частей так, чтобы каждая из них была равна s = 1/K. Это следует из того, что по определению

$$S_{\Sigma} = \int_{0}^{L_{\text{смакс}}} W(L_{\text{c}}) dL_{\text{c}} = 1.$$

Далее разделим площадь под кривой $W_1(L_{c1})$ также на *K* равных частей, каждая из которых равнялась бы также s = 1/K. После этого пронумеруем от нуля до *K* абсциссы вертикальных отрезков на рис. 6.3, *a* и *б*. Не трудно видеть, что значения L_{ci} и L_{c1i} связаны соотношениями

$$L_{cli} = \varphi(L_{ci})$$
 и $L_{ci} = \varphi^{(-1)}(L_{cli}),$



Рис. 6.3. Плотности вероятности распределения яркости в неискаженном и искаженном изображениях

т. е. представляют собой световую характеристику корректируемой системы и соответственно амплитудную характеристику гамма-корректора. Этот вывод следует из очевидных равенств

$$\int_{L_{ci}}^{L_{ci+1}} W(L_{c}) dL_{c} = \int_{L_{c1i}}^{L_{c1i+1}} W(L_{c1}) dL_{c1} = 1/K.$$

Практическая реализация рассмотренного метода нахождения амплитудной характеристики гамма-корректора отличается от описанной выше реализации лишь тем, что вместо плотностей вероятностей в ней используются соответствующие им гистограммы распределения, которые предварительно должны быть измерены. Описанный метод нахождения амплитудной характеристики гамма-корректора применяют также в задачах, когда требуется гистограмму неискаженного изображения привести к заранее выбранному виду в целях его улучшения. Такая необходимость возникает, например, в случаях, когда требуется улучшить различимость деталей на темных участках изображения. Эндрюс, Холл, а также другие исследователи [23, 84] улучшали изображения путем выравнивания гистограмм распределения, т. е. они добивались равномерного распределения яркости в обработанных изображениях. Фреем [84] был исследован метод, при котором распределения яркости в обработанных изображениях принимали экспоненциальную или гиперболическую форму. Кетчам [84] модернизировал этот метод, применяя "локальные", подсчитанные по частям изображения, гистограммы. Вместе с тем, как отмечает Прэтт [84], степень улучшения восприятия обработанных таким образом изображений сильно зависит от их характера.

6.4. Методы согласования динамического диапазона изображений с динамическим диапазоном дисплея

Типичным недостатком ряда технических изображений является их малый контраст. Примерами таких изображений являются: изображения отпечатков пальцев, рентгеновские изображения, в некоторых случаях, изображения, получаемые с космических аппаратов и т. п. Причинами низкого контраста таких изображений могут быть: природа фотографируемых объектов, неудовлетворительные условия освещения, ошибки, допущенные при определении экспозиции, высокий уровень шума, а также ряд других. Контраст таких изображений во многих случаях может быть повышен фотографическими методами. Примером использования фотографических методов для этой цели может служить повышение контраста первых изображений обратной стороны Луны, полученных в Советском Союзе в 1959 г., поскольку принятые с борта автоматической межпланетной станции "Луна-3" изображения имели чрезвычайно малый контраст [10].

Однако фотографические методы изменения контраста имеют ряд весьма существенных недостатков, главными из которых являются: большие временные за-

траты на фотопроцесс и невозможность реализовать произвольно заданные характеристики передачи уровней. Развитие вычислительной техники позволило разработать методы обработки изображений, свободные от этих недостатков. В целях повышения контраста используют коррекцию изображений путем применения цифрового гамма-корректора, реализованного программным способом. Исходными данными, которые необходимы для нахождения его амплитудной характеристики, являются минимальное $L_{\rm с.мин}$ и максимальное $L_{\rm с.макс}$ значения яркости в изображении, контраст которого необходимо повысить. При этом возможны два подхода к решению поставленной задачи.

В первом случае динамический диапазон обрабатываемого изображения увеличивается ("растягивается") до динамического диапазона, который способна воспроизвести система, но без внесения ограничений по яркости в само изображение. Если обрабатываемое изображение является полутоновым черно-белым и предполагается, что после обработки оно должно будет выводиться на экран монитора компьютера, то его динамический диапазон после увеличения должен быть заключен в пределах от нулевого до 255-го уровня квантования, который соответствует максимальной яркости на экране дисплея $L_{д.макс}$. При этом значения яркости во входном и выходном изображениях связаны соотношением

$$L_{\rm c.BMX} = aL_{\rm c.BX} + b ,$$

где значения коэффициентов а и b находятся по формулам

$$a = \frac{L_{\text{д.макс}}}{L_{\text{с.макс}} - L_{\text{с.мин}}}, \ b = -\frac{L_{\text{д.макс}}L_{\text{с.мин}}}{L_{\text{с.макс}} - L_{\text{с.мин}}}.$$

На рис. 6.4 приведена характеристика, обеспечивающая это преобразование.



Рис. 6.4. Амплитудная характеристика гамма-корректора для случая, когда в обрабатываемое изображение ограничения по яркости не вносятся. Через *L*_{с.вх} и *L*_{с.вых} обозначены значения яркости в исходном и обработанном изображениях



Рис. 6.5. Здесь: *а* — исходное изображение;

- б изображение после согласования его динамического диапазона яркостей с динамическим диапазоном яркостей, воспроизводимых дисплеем;
 - *в* изображение, динамический диапазон яркостей которого ограничен в результате преобразования; *г* — гистограмма изображения (*a*);
 - *д* гистограмма изображения (б); *е* гистограмма изображения (*в*)

В качестве поясняющего примера на рис. 6.5, *а* показано исходное изображение, а на рис. 6.5, *г* приведена его гистограмма, из которой видно, что это изображение занимает только часть динамического диапазона яркостей, воспроизводимых дисплеем. Вследствие этого изображение выглядит вялым и невыразительным.

На рис. 6.5, *б* показано это же изображение, но после того как динамический диапазон его яркостей был согласован с динамическим диапазоном яркостей, воспроизводимых дисплеем посредством гамма-корректора. На рис. 6.5, *д* приведена его гистограмма, из которой видно, что теперь весь динамический диапазон яркостей, воспроизводимых дисплеем, используется. Сопоставление исходного изображения, имеющего малый контраст, с изображением, полученным после его обработки, указывает на его явное улучшение.

В последнем случае амплитудная характеристика гамма-корректора выбирается таким образом, что создается ограничение динамического диапазона и со стороны малых значений яркости, и со стороны больших значений яркости обрабатываемого изображения, благодаря чему обработанное изображение выглядит более контрастным. Рисунок 6.6 поясняет изложенное. В некоторых случаях этот способ обеспечивает более высокое субъективное качество изображения, особенно если изображение содержит небольшое число элементов, яркость которых выходит за уровни ограничения.

На рис. 6.5, *в* показано изображение, при обработке которого допущено небольшое ограничение динамического диапазона яркости и со стороны больших значений, и со стороны малых значений, что подтверждается гистограммой, имеющей пики и в области белого, и в области черного (см. рис. 6.5, *e*). Сопоставление этого изображения с изображением, приведенным на рис. 6.5, *б*, показывает, что в этом случае имеется небольшое улучшение.



Рис. 6.6. Амплитудная характеристика гамма-корректора для случая, когда в обрабатываемое изображение вносятся ограничения по яркости. Через *L*_{с.вых} и *L*_{с.вых} обозначены значения яркости в исходном и обработанном изображениях Другим примером согласования динамического диапазона изображения и воспроизводящего устройства является отображение на экране дисплея изображения энергетического спектра Фурье (спектральной интенсивности), который с увеличением частоты убывает обратно пропорционально квадрату частоты. Как известно, его изображение характеризуется чрезвычайно большим динамическим диапазоном и поэтому при воспроизведении значительная часть (высокочастотная) спектральной интенсивности вследствие ограничения по яркости не воспроизводится. Простым выходом из сложившейся ситуации является предварительное преобразование сигнала яркости гамма-корректором с амплитудной характеристикой вида

$$L_{\rm BMX} = \beta \log (L_{\rm BX} + 1),$$

где $L_{\rm bx}$ — сигнал на входе гамма-корректора, представленный номером квантового уровня; $L_{\rm выx}$ — сигнал на выходе гамма-корректора, представленный номером квантового уровня; β — постоянный коэффициент. При таком виде преобразования диапазон малых значений, соответствующий представлению интенсивности высоких частот, растягивается, а диапазон больших значений сигнала, соответствующий низким частотам, сжимается, в результате чего обеспечивается воспроизведение деталей спектра как в области высоких частот, так и в области низких частот. Сказанное демонстрируется рис. 6.7.



Рис. 6.7. Иллюстрация использования логарифмического преобразования изображения спектральной интенсивности: *a* — изображение; *б* — спектральная интенсивность этого изображения при ее непосредственном отображении; *в* — спектральная интенсивность этого же изображения, но после ее логарифмического преобразования

Применение цифровых методов обработки предоставляет и другие возможности, которые используют при детальном изучении изображений, выведенных на экран монитора компьютера.

Одной из таких возможностей является режим просмотра изображения с использованием так называемой "контрастирующей лупы". Сущность этого режима состоит

в том, что программным способом создается гамма-корректор, амплитудная характеристика которого выделяет из исходного изображения небольшую часть его динамического диапазона и растягивает его на весь динамический диапазон воспроизводящего устройства, увеличивая таким образом контраст деталей, занимающих выделенную часть диапазона. Ширина выделяемой части динамического диапазона и его положение на шкале яркостей устанавливаются в диалоговом режиме. В качестве примера на рис. 6.8 приведена амплитудная характеристика "контрастирующей лупы".



Рис. 6.8. Амплитудная характеристика "контрастирующей лупы". Через *L*_{с.вх} и *L*_{с.вых} обозначены значения яркости в исходном и контрастированном изображениях

Подобно тому как обычная лупа позволяет просматривать по очереди мелкие детали на изображении, увеличивая их до необходимого размера, "контрастирующая лупа" позволяет по очереди просматривать малоконтрастные детали, расположенные в различных участках динамического светового диапазона, которые иначе вследствие малого контраста оставались бы для наблюдателя плохо различимыми. Рисунок 6.9 демонстрирует сказанное. На рис. 6.9, *а* приведено изображение участка Луны, на котором имеются слабо различимые детали поверхности, часть из которых расположена в темных участках изображения, а часть — в светло-серых. При воспроизведении это изображение не подвергалось гамма-коррекции, т. е. формально можно считать, что амплитудная характеристика гамма-корректора была линейной, как показано на рис. 6.9, *г*.

На рис. 6.9, δ показано это же изображение, но перед воспроизведением оно было обработано в гамма-корректоре, характеристика которого показана на рис. 6.9, ∂ . Благодаря тому, что в результате обработки контраст деталей, расположенных в области светло-серого, был существенно увеличен ценой потери деталей в черном и белом, детали в области светло-серого стали лучше различимы, что видно из сопоставления изображений, показанных на рис. 6.9, *a* и δ . Аналогичным образом путем обработки исходного изображения гамма-корректором, амплитудная

характеристика которого показана на рис. 6.9, *e*, получено изображение, приведенное на рис. 6.9, *в*. На этом изображении существенно увеличен контраст деталей, расположенных в темной части изображения, что улучшило их различимость. В этом не трудно убедиться путем сравнения изображений, показанных на рис. 6.9, *a* и *в*. Как и в предыдущем случае, данное улучшение достигнуто за счет потери деталей, расположенных в других участках динамического светового диапазона.



Рис. 6.9. Демонстрация результатов применения "контрастирующей лупы": *а* — исходное изображение; *б* — изображение, полученное при использовании "контрастирующей лупы", амплитудная характеристика которой (*д*); *в* — изображение, полученное при использовании "контрастирующей лупы", амплитудная характеристика которой (*e*); *г* — линейная амплитудная характеристика

Другой возможностью, которую предоставляет цифровой метод обработки, является *обращение контраста*, преобразующее негативное изображение в позитивное. Амплитудная характеристика гамма-корректора, реализующего обращение контраста, черно-белого полутонового изображения показана на рис. 6.10. В случае обращения контраста цветного изображения аналогичные амплитудные характеристики применяют для обращения контраста каждого из трех компонентов: R, G и B. Обращение контраста используют для просмотра аэрофотографий, а также других изображений, получаемых фотографическим способом. Удобство этого вида обработки состоит в экономии времени, потому что при его применении отпадает необходимость изготовления позитивных изображений с негативов фотографическим способом.



Рис. 6.10. Амплитудная характеристика гамма-корректора, осуществляющего обращение контраста



Рис. 6.11. К пояснению обращения контраста: *a* — негативное изображение; *б* — изображение, полученное путем обращения контраста изображения (*a*)

Кроме того, поскольку динамический диапазон негативной пленки, как правило, выше, чем диапазон позитива, это позволяет на негативном изображении после обращения его контраста обнаруживать детали, контраст которых недостаточен для их обнаружения на позитивном изображении, полученном путем фотографической печати. (Заметим, что широкое использование цифровых фотокамер не исключило применения пленочных фотокамер, которые по некоторым параметрам, например, по динамическому диапазону, для специалистов являются предпочтительными.) Использование этого вида преобразования удобно также при фотопечати с цветной негативной пленки, поскольку при этом можно сразу же произвести разбраковку негативов и отобрать для печати только удачные снимки. Можно привести еще ряд примеров применения этого вида преобразования. На рис. 6.11 показаны негатив изображения и результат его обработки методом обращения контраста. Описанные в этом разделе методы обработки изображений реализованы практически во всех основных графических редакторах.

6.5. Проблема разрежения шкалы квантования при коррекции световых характеристик

Как уже было отмечено, особенностью цифровых изображений является то, что они проквантованы по яркости. При этом яркость пикселов, на которые они разлагаются, может принимать не любые, а лишь строго определенные значения, которые называются *уровнями квантования*. Обычно число уровней яркости, на которое квантуются изображения, т. е. число разрешенных градаций яркости, в случае черно-белых изображений составляет 256, а в случае цветных — по 256 на каждый из 3-х основных цветов, при этом расстояние между соседними уровнями квантования на выходе АЦП одинаково. Если проквантованные по яркости изображения преобразуются аналоговым устройством с нелинейной световой характеристикой, например, кинескопом или жидкокристаллической панелью, расстояния между соседними уровнями квантования в изображении изменяются, т. е. происходит перераспределение градаций яркости внутри динамического диапазона изображения, число же уровней при этом сохраняется.

Совсем иначе обстоит дело, если такое преобразование осуществляется цифровым устройством, например, цифровым гамма-корректором, разрядность представления яркости у которого одинакова на входе и на выходе. В этом случае наряду с изменением расстояния между градациями яркости на его выходе, представляемыми соседними уровнями квантования на его входе, часть градаций яркости в выходном сигнале теряется. Это объясняется тем, что вследствие сближения градаций яркости в части динамического диапазона в один и тот же квантовый интервал выходного сигнала попадает несколько градаций яркости, соответствуюцих различным уровням квантования входного сигнала. Рисунок 6.12, на котором приведена амплитудная характеристика цифрового гамма-корректора, поясняет изложенное. Уменьшение количества используемых уровней квантования в выходном сигнале приводит к разрежению шкалы воспроизводимых градаций яркости на изображении и может вызвать появление на изображении ложных контуров. Для того чтобы этого не произошло, необходимо входной сигнал квантовать на большее число уровней, т. е. использовать для представления входного сигнала код большей разрядности. Это обстоятельство учтено в современных профессиональных цифровых камерах.



Рис. 6.12. Амплитудная характеристика цифрового гамма-корректора, здесь $k_{\text{вх}}$ и $k_{\text{вых}}$ — номера квантовых уровней на входе и выходе гамма-корректора соответственно

Реализация цифрового гамма-корректора при компьютерной обработке изображения крайне проста. С этой целью создается специальный одномерный массив чисел, в который записывается амплитудная характеристика гамма-корректора. При этом номера уровней квантования входного сигнала рассматриваются как номера (индексы) элементов массива, а числа, записанные в массив, — как скорректированные значения сигнала.

Исходя из этого, оценим максимальное расстояние между воспроизводимыми на экране монитора смежными уровнями квантования, считая, что для коррекции его световой характеристики применен цифровой гамма-корректор, на выходе которого включен цифроаналоговый преобразователь. Для простоты в качестве примера рассмотрим монитор на основе кинескопа. Аппроксимируя его световую характеристику степенной функцией, представим ее как зависимость яркости свечения экрана кинескопа L_c от величины сигнала e_c , вызывающего это свечение, т. е.

$$L_{\rm c} = \beta_{\rm \kappa} e_{\rm c}^{\gamma} \,,$$

а не как обычно принято от величины напряжения на модуляторе кинескопа $e_m = e_c - e_0$,

$$L_{\rm c} = \beta_{\rm \kappa} \left(e_m + e_0 \right)^{\gamma},$$

где e_0 — напряжение запирания, β_{κ} — коэффициент, согласующий размерности, γ — показатель степени, который для современных кинескопов лежит в пределах 2,5 $\leq \gamma \leq 3,5$.

Для обеспечения полной коррекции световой характеристики кинескопа, амплитудная характеристика гамма-корректора должна иметь вид

$$e_{\rm c} = \beta_{\gamma} e_{\rm c.BX}^{1/\gamma_{\rm K}},$$

где $e_{c.bx}$ — сигнал на входе гамма-корректора, β_{γ} — коэффициент, согласующий размерности. Поскольку по условию задачи для коррекции градационных искажений используется цифровой гамма-корректор, у которого для представления сигнала на его входе и выходе используется одно и то же количество разрядов двоичного кода, соотношение между номером уровня квантования сигнала на входе k и соответствующим ему номером уровня квантования на выходе i имеет следующий вид

$$i = \operatorname{Round}\left[\left(m - 1 \right) \left(\frac{k}{m - 1} \right)^{1/\gamma_{\kappa}} \right],$$

где *m* — число уровней квантования сигнала, поступающего на вход гаммакорректора.

Как уже было отмечено, вследствие сближения градаций яркости на выходе гаммакорректора в части динамического диапазона (в данном случае в области белого) некоторые из них пропадают, что после преобразования сигнала световой характеристикой кинескопа приведет к разрежению шкалы квантования в области белого. Для того чтобы оценить это разрежение, найдем номер уровня квантования k на входе гамма-корректора, которому соответствует уровень квантования на его выходе i = m - 2, т. е. уровень, ближайший к максимальному. Принимая для простоты вычислений $\gamma = 3$, что близко к действительности, будем иметь

$$m-2 = \operatorname{Round}\left[\left(m-1\right)\left(\frac{k}{m-1}\right)^{1/3}\right],$$
откуда $k \approx \frac{\left(m-2\right)^3}{\left(m-1\right)^2}.$

Представляя правую часть этого выражения в виде степенного ряда и ограничиваясь первыми двумя членами разложения, получим $k \approx (m-1)-3$. Из этого следует, что расстояние между максимальным уровнем яркости и ближайшим к нему воспроизводимым уровнем увеличивается в три раза и составляет 1/85 от динамического диапазона при 256 уровнях квантования. Вызовет ли это появление ложных контуров на изображении?

Если учесть, что минимальное количество уровней квантования, при котором на изображении ложные контуры еще не видны, равно 64, при этом расстояние между ними составляет 1/64 динамического диапазона, то можно ожидать, что в рассмотренном ранее случае этого не произойдет. Опыт работы с реальными изображениями подтверждает этот вывод, однако имеющийся запас не велик.

6.6. Тоновая коррекция

Целью тоновой коррекции изображений является придание им большей выразительности при воспроизведении, а также их улучшение, особенно если при съемке они были неправильно экспонированы.

В полиграфии принято весь тоновый (динамический) диапазон, который занимают изображения, разделять на семь поддиапазонов.

- 1. Область белого, составляющая 2% от всего динамического диапазона. Область белого это область, которая при воспроизведении не содержит видимых деталей.
- Область наибольшей яркости, которая занимает часть динамического диапазона в интервале от 2 до 18%. В отличие от области белого детали, попавшие в область наибольшей яркости на изображениях, воспроизводятся.
- 3. Область четвертных тонов занимает интервал от 18 до 35% на шкале динамического диапазона.
- 4. Область средних тонов занимает интервал от 35 до 65% на шкале динамического диапазона.
- 5. Область трехчетвертных тонов занимает интервал от 65 до 80% на шкале динамического диапазона.
- 6. Область тени занимает часть динамического диапазона в пределах от 80 до 98%.
- 7. Область черного занимает интервал от 98 до 100% на шкале динамического диапазона. Эта область воспроизводится сплошным черным цветом.

Реальные изображения, как правило, не используют весь динамический диапазон яркости, при этом изображения можно условно разделить на три группы.

- 1. Яркие изображения. В этих изображениях основные детали расположены в областях наибольшей яркости и четвертных тонов. Примером такого изображения является фотография белого медведя на фоне снега.
- 2. Сбалансированные изображения, в которых наиболее интересные детали занимают либо область средних яркостей, либо распределены между яркими и темными тонами. Этот вид изображений наиболее распространен.
- Темные изображения. В темных изображениях наиболее интересные детали содержатся в темных областях. Примером такого изображения является фотография города в сумерках.

В ряде случаев характер изображения определяется не его содержанием, а неправильным выбором экспозиции при съемке. Так при недостаточной экспозиции изображение будет темным, а при чрезмерно большой экспозиции — светлым.

При воспроизведении изображения, не зависимо от того, выводим ли мы его на печать или на экран монитора компьютера, необходимо согласовать динамический диапазон яркостей в изображении с динамическим диапазоном воспроизводящего устройства. Процедура согласования динамических диапазонов состоит в определении на изображении областей наибольшей яркости и наиболее темных областей, которые еще должны содержать детали, и согласования их с соответствующими областями воспроизводящего устройства. Отметим, что не всегда легко определить точку наибольшей яркости и точку черного, особенно на изображениях, перегруженных деталями. Поэтому при компьютерной обработке изображений для этого следует использовать инструмент **Eyedropper** (Пипетка).

После согласования динамических диапазонов изображения и воспроизводящего устройства следует перейти к настройке амплитудной характеристики, определяющей вид нелинейного преобразования. В графических редакторах этот инструмент называется Curves (Кривые). Изменяя вид этой характеристики по своему усмотрению, можно растягивать одни участки динамического диапазона изображений за счет сокращения других участков, добиваясь при этом увеличения контраста наиболее важных в смысловом отношении деталей на обрабатываемом изображении. Обратим внимание на то, что нелинейные преобразования, приводящие к перераспределению градаций яркости в изображении, не воспринимаются зрительной системой как искажения, что позволяет добиваться путем применения тоновой коррекции большей выразительности изображений, расставляя желательные акценты и усиливая тем самым их смысловое содержание. (Подробнее об этих видах обработки см. в [2, 3].) Однако в случае, когда требуется воспроизвести изображение сцены с большим динамическим диапазоном яркостей, такими простыми методами обойтись не удается. В этом случае используют более сложные методы обработки, к рассмотрению которых мы перейдем.

6.7. Фотографирование сцен с большим динамическим диапазоном яркостей

Обратимся теперь к проблеме, связанной с фотографированием сцен, имеющих большой динамический диапазон [163]. Динамический диапазон реальных сцен может достигать 100 000 и более, в то время как динамический диапазон большинства цифровых фотокамер и цифровых телевизионных камер существенно меньше, что обусловлено ограниченностью динамического диапазона используемых в них матриц, который, как правило, не превышает 1000. Типичное значение динамического диапазона этих устройств обычно составляет 255, что соответствует глубине цвета 8 бит на пиксел. В более дорогих моделях цифровых фотокамер (зеркальных) глубина цвета в формате JPEG составляет 8, а в формате RAW^1 12 бит на пиксел, что после обработки специальными компьютерными программами приводится к 16 битам на пиксел. Несколько лучше обстоит дело с динамическим диапазоном пленочных фотокамер, у которых динамический диапазон негатива составляет около 10 000. Однако при печати позитивного изображения по данным [163] он снижается до 100. Следствием этого является то, что при фотографировании высококонтрастных сцен часть деталей на изображении неизбежно оказывается потерянной. Так, например, при фотографировании комнаты и окна в этой комнате в зависимости от выбранной экспозиции возможны два варианта. В первом случае на снимке будут хорошо проработаны детали предметов, расположенных в комнате, но сильно пересвеченное окно, при этом детали предметов, расположенных за окном, будут потеряны вследствие передержки. Во втором случае на снимке будут хорошо проработаны детали предметов, расположенных за окном, будут потеряны детали предметов, расположенных будут хорошо проработаны детали предметов, в силу недодержки. Рисунок 6.13 поясняет приведенный пример.





Рис. 6.13. Изображения, полученные: а — при передержке; б — при недодержке

Для разрешения этой проблемы был разработан специальный метод, который называется *брекетингом* экспозиции (Exposure Bracketing). Существо метода состоит в том, что при фотографировании в режиме брекетинга одновременно делаются три кадра — с "нормальной экспозицией", с недодержкой и с передержкой. При этом степени недодержки и передержки (эксповилка) могут выбираться, исходя из конкретных условий. Например, при эксповилке в одну ступень камера делает "нормальный", недодержанный в два раза и передержанный в два раза кадры (увеличение экспозиции на одну ступень означает увеличение освещенности в два раза,

¹ RAW — сырой, необработанный (англ.). RAW-файл содержит информацию, полученную прямо с цифровой матрицы, без какой-либо обработки фотокамерой, и кроме того информацию о параметрах съемки — EXIF-данные (диафрагма, экспозиция и т. д.). Файл RAW требует последующей обработки на компьютере посредством специальных программ, так называемых RAW-конвертеров.

на две ступени — в четыре раза). Такой набор из трех кадров чаще всего используется при создании файлов с большим динамическим диапазоном (High Dynamic Range, HDR) файлов, хотя возможны и другие варианты. Глубина цвета HDR-файлов — 32 бита на пиксел.

Рассмотрим подробнее формирование шкалы квантования при использовании этого метода. При этом методе из недодержанного кадра берется информация о светлых тонах изображения фотографируемой сцены, а из передержанного кадра информация о темных тонах. Поскольку динамический диапазон камеры в области светлых тонов ограничен "насыщением" матрицы, а в области темных — шумами, то необходимо при выборе экспозиций передержанного и недодержанного кадров принимать это во внимание. Рисунок 6.14, на котором показаны амплитудные характеристики матрицы для трех различных экспозиций, поясняет сказанное.



Рис. 6.14. Амплитудные характеристики матрицы для трех различных экспозиций

В нашем примере наиболее короткая экспозиция соответствует наиболее яркой детали фотографируемой сцены, которая равна L_3 (линия 3). Наиболее продолжительная экспозиция выбирается таким образом, чтобы минимальный квантовый уровень примерно равнялся среднеквадратичному уровню шума (линия 1). Обозначая отношения яркостей L_1 , L_2 и L_3 через $L_2/L_1 = k_1$ и $L_3/L_1 = k_2$, запишем правило формирования результирующей амплитудной характеристики синтезированного изображения, которое записывается в файл.

$$m_{\text{pes}} = \begin{cases} m_1, & \text{если } L \le L_1, \\ m_2 \times k_1, & \text{если } L_1 < L \le L_2, \\ m_3 \times k_2, & \text{если } L_2 < L \le L_3. \end{cases}$$



Рис. 6.15. Шкала квантования синтезированного изображения

Особенностью синтезированного таким методом изображения будет то, что в области малых значений яркости оно будет проквантовано с малым шагом, в области средних значений яркости с шагом в k_1 раз большим, а в области больших значений яркости с вагом в k_2 раз.

Рисунок 6.15 поясняет изложенное.

Несмотря на то, что получающаяся при этом шкала квантования не является равномерной, она оказывается неплохо согласованной со свойствами зрительной системы.

6.8. Методы отображения сцен с большим динамическим диапазоном яркостей на дисплеях с ограниченным динамическим диапазоном

Рассмотренный выше метод брекетинга экспозиции позволяет создавать HDRфайлы, в которых сохраняются изображения с большим динамическим диапазоном, т. е. файлы изображений, в которых сохраняются малоконтрастные детали объектов, расположенных на краях динамического диапазона яркостей. Однако при попытке воспроизвести эти изображения на современных устройствах отображения эти малоконтрастные детали все равно будут утрачены вследствие ограничений динамического диапазона этими устройствами (жидкокристаллическими панелями, кинескопами, плазменными панелями и т. д.).

По сообщениям средств массовой информации (СМИ) в настоящее время ряд ведущих фирм проводят разработки отображающих устройств, в частности жидкокристаллических панелей, способных воспроизводить изображения, имеющие большой динамический диапазон без его ограничения. Однако даже появление устройств отображения с большим динамическим диапазоном не решает проблему. Дело в том, что следующим "звеном", где происходит ограничение динамического диапазона, является сама зрительная система. Поясним сказанное. Известно, что при наблюдении натурных сцен с большим динамическим диапазоном яркости зритель хорошо различает имеющиеся в них объекты как в области высоких, так и в области низких освещенностей. Так, например, наблюдая сцену в саду при ярком солнечном освещении, мы одинаково хорошо видим и предметы, расположенные под яркими лучами солнца, и предметы, находящиеся в тени деревьев, правда, не одновременно, а после небольшого времени адаптации. Объясняется это тем, что наша зрительная система обладает свойством локальной адаптации к величине освещенности. При перемещении оси зрения с ярко освещенной области рассматриваемой сцены к области сцены, которая слабо освещена, происходит переадаптация зрительной системы вследствие того, что средняя освещенность центрального участка сетчатки при этом изменяется. Этого не происходит или же происходит в слабой степени при наблюдении изображения на экране отображающего устройства, экран которого, как правило, имеет сравнительно небольшие размеры. В условиях наблюдения изображений на таких экранах зрительная система по-прежнему адаптируется к средней освещенности сетчатки глаз, однако эта средняя освещенность в силу относительно небольших размеров изображения определяется уже не локальными его участками, имеющими различную яркость, а средней яркостью экрана, а также уровнем освещенности помещения, в котором происходит наблюдение. Поэтому при переводе взгляда с одного участка изображения на другой адаптация зрительной системы почти не изменяется. Исключение может составлять случай наблюдения изображений на очень больших экранах в затемненных помещениях, что является скорее исключением, чем правилом.

Указанная причина заставила искать другие способы, которые бы позволили отображать изображения сцен с большим динамическим диапазоном яркости без потерь деталей, расположенных в области темных и светлых полутонов.

Простейшим методом, посредством которого можно хотя бы частично решить эту проблему, является ү-коррекция. При выполнении ү-коррекции области динамического диапазона, в которых необходимо предотвратить потерю деталей, растягиваются, при этом контраст деталей, расположенных в этих областях, увеличивается. Однако достигается это за счет сжатия других областей динамического диапазона, что влечет за собой уменьшение контраста расположенных в них деталей, поскольку контраст всего изображения при таком преобразовании остается неизменным. При использовании этого метода после процедуры ү-коррекции перед отображением изображения на дисплее необходимо преобразовать HDR-файл в стандартный LDR-файл (Low Dynamic Range) с низким динамическим диапазоном.

Другим методом решения рассматриваемой проблемы является взвешенное суммирование недодержанного, нормально экспонированного и передержанного изображений с весами 1/3. При этом яркость изображения, полученного этим способом, будет представлена следующим образом:

$$L_{\text{pe3}} = A \begin{cases} m_1 + m_2 + m_3, & \text{если } L \le L_1, \\ m_{\text{макс}} + m_2 + m_3, & \text{если } L_1 < L \le L_2, \\ m_{\text{макс}} + m_{\text{макс}} + m_3, & \text{если } L_2 < L \le L_3. \end{cases}$$

В этой формуле использованы следующие обозначения: A — коэффициент пропорциональности; $m_{\text{макс}}$ — максимальное значение уровня квантования, определяемое числом двоичных разрядов кода *n* соотношением $m_{\text{макс}} = 2^n - 1$; m_1 , m_2 , m_3 — номера уровней квантования в недодержанном, нормально экспонированном и передержанном изображениях, соответствующих яркости *L* детали фотографируемой сцены. Через L_1 , L_2 , L_3 по-прежнему обозначены яркости на натуре, при которых наступает насыщение матриц при фотографировании недодержанного, нормально экспонированного и передержанного изображений. Учитывая, что яркость пиксела в синтезированном изображении $L_{\text{peз}}$ и номер уровня квантования $m_{\text{peз}}$, соответствующего этой яркости, связаны линейной зависимостью, не трудно видеть, что

$$m_{\text{pe3}} = \frac{1}{3} \begin{cases} m_1 + m_2 + m_3, & \text{если } L \le L_1, \\ m_{\text{макс}} + m_2 + m_3, & \text{если } L_1 < L \le L_2, \\ m_{\text{макс}} + m_{\text{макс}} + m_3, & \text{если } L_2 < L \le L_3. \end{cases}$$

На рис. 6.16, *а* показаны амплитудные характеристики матрицы для трех различных экспозиций, а на рис. 6.16, δ приведена зависимость m_{pes} от *L*.



Рис. 6.16. К пояснению формирования синтезированной амплитудной характеристики

На рис. 6.17 и ЦВ.5 приведены три исходных изображения — недодержанное, нормально экспонированное и передержанное (рис. 6.17, a—e) — и результирующее изображение (рис. 6.17, r), полученное описанным способом.

При использовании описанного метода как количество исходных (недодержанных и передержанных изображений), так и размеры эксповилок могут быть иными, чем в рассмотренном случае, и должны определяться экспериментально для каждого конкретного случая.





Рис. 6.17. Изображения: *а* — недодержанное; *б* — нормально экспонированное; *в* — передержанное; *г* — полученное описанным ранее методом. Изображения (*a*), (*б*), (*в*) взяты из [163]

В настоящее время появилась серия публикаций, посвященных так называемому методу увеличения локальных контрастов изображения [107, 123, 149, 164, 172, 174]. Идея метода заключается в том, что вначале поочередно для каждого пиксела изображения находится его локальный контраст, который определяется как

$$K(k,n) = \frac{L(k,n,) - L(k,n)}{\overline{L(k,n)}},$$

где L(k,n) — исходная яркость обрабатываемого пиксела; k — номер строки; n — номер столбца; $\overline{L(k,n)}$ — средняя яркость в области изображения, окружающей пиксел с координатами (k,n), которая может быть задана различными способами. В простейшем случае эта область задается в виде квадратного окна размером $N \times N$, где N — нечетное число. В более совершенных алгоритмах ее размеры и форма могут быть адаптивными. Затем осуществляется усиление локального контраста изображения до величины $K_{\text{вых}}(k,n)$,

$$K_{\text{вых}}(k,n) = K(k,n) A[L(x,y)],$$

причем величина усиления A[L(x, y)] зависит от распределения яркости в области, окружающей пиксел. Заключительным этапом метода является определение яркости пиксела в обработанном изображении, в котором локальные контрасты усилены при сохранении максимального контраста всего изображения.

$$L_{\text{Bbix}}(k,n) = K_{\text{Bbix}}(k,n)\overline{L(x,y)} + \overline{L(x,y)}.$$

Несмотря на кажущуюся простоту метода, его реализация связана с трудностью определения зависимости усиления A[L(x, y)] от распределения яркости в окне, окружающем корректируемый пиксел. Неудачное определение этой зависимости приводит к появлению в скорректированном изображении ореолов около протяженных границ крупных деталей. Другой, сравнительно недавно разработанный метод, основан на использовании билатеральной фильтрации [123, 172], применение которой позволяет при сжатии динамического диапазона обрабатываемого изображения сохранять его детали, создающие на изображении небольшие перепады яркости. Подробно этот метод описан в следующем разделе.

6.9. Билатеральная фильтрация изображений

Билатеральная фильтрация позволяет устранять из изображения малоконтрастные детали небольших размеров, сохраняя, в то же время, четкими световые границы, созданные большими перепадами яркости. Достигается это благодаря тому, что в процессе свертки фильтруемого изображения с импульсной характеристикой фильтра ее весовые коэффициенты все время корректируются путем умножения на так называемую стоп-функцию, значение отсчетов которой определяется яркостным рельефом изображения. При фильтрации участков изображения, где изменения яркости невелики, значения отсчетов стоп-функции близки к единице, и она практически не изменяет весовые коэффициенты импульсной характеристики. Однако при приближении центрального отсчета импульсной характеристики к световой границе, обусловленной большим перепадом яркости, значения отсчетов стопфункции, оказавшихся в области значений яркости, сильно отличающихся от яркости центрального пиксела, становятся малыми и, умножаясь на отсчеты импульсной характеристики, они сильно уменьшают эти отсчеты. В целях пояснения на рис. 6.18 показана скорректированная импульсная характеристика в момент, когда она приблизилась к световой границе с большим перепадом яркости.



Рис. 6.18. К пояснению корректирующего действия стоп-функции

Влияние световой границы проявилось в отсечении правой части импульсной характеристики, что предотвратило размытие световой границы в процессе фильтрации изображения. Математически билатеральная фильтрация описывается следующим образом:

$$L_{c\Omega}(k,n) = \frac{1}{K(k,n)} \times \\ \times \sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{c}(k+k',n+n')h(k',n')g[L_{c}(k+k',n+n')-L_{c}(k,n)],$$

где *K* и *N* — протяженность двумерной импульсной характеристики h(k', n')в обоих направлениях; $g[L_c(k+k', n+n') - L_c(k, n)]$ — стоп-функция; *k* и *n* — номера строки и столбца центров импульсной характеристики и стоп-функции; *k'* и *n'* — текущие значения номеров строк и столбцов соответственно; K(k, n) — нормирующий множитель, который вычисляется по формуле

$$K(k,n) = \sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} h(k',n') g \Big[L_{c}(k+k',n+n') - L_{c}(k,n) \Big].$$

Выражение

$$h_{\Sigma}(k, k', n, n') = \frac{h(k', n') g[L_{c}(k+k', n+n') - L_{c}(k, n)]}{K(k, n)}$$

в формуле, по которой вычисляется $L_{c\Omega}(k,n)$, можно рассматривать как скорректированную импульсную характеристику. Как уже было сказано, при билатеральной фильтрации из изображения устраняются детали небольшого размера, создающие небольшие перепады яркости, в то время как резкость границ, создающих большие перепады яркости, сохраняется. На практике импульсную характеристику обычно выбирают гауссовой

$$h(k,n) = k_0^{-1} \exp\left[-\frac{(k^2 + n^2)}{r_e^2}\right],$$

где r_e — условный радиус импульсной характеристики, k_0 — нормирующий множитель, равный

$$k_0 = \sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} h(k',n').$$

Такой выбор импульсной характеристики позволяет избежать фазочастотных искажений. Что касается стоп-функции, то здесь выбор шире. В [123] рекомендуется использовать для стоп-функции также гауссовую функцию,

$$g = \exp\left\{-\frac{\left[L_{c}\left(k+k',n+n'\right)-L_{c}\left(k,n\right)\right]^{2}}{\Delta L^{2}}\right\},\$$

где ΔL — параметр, определяющий "чувствительность" стоп-функции к величине перепада яркости световой границы ΔL . Однако в ряде случаев, как показали выполненные нами экспериментальные исследования, лучшие результаты получаются при использовании для этой цели бинарной функции, а именно

$$g\left[L_{c}(k+k',n+n')-L_{c}(k,n)\right] = \begin{cases} 0 & \text{при}\left|L_{c}(k+k',n+n')-L_{c}(k,n)\right| \ge P, \\ 1 & \text{при невыполнении этого условия,} \end{cases}$$

где P — пороговое значение перепада яркости, при превышении которого стопфункция изменяет свое значение с 1 на 0. На рис. 6.19 и ЦВ.6 в целях иллюстрации действия билатерального фильтра приведены два изображения: a — исходное изображение, δ — изображение, полученное в результате билатеральной фильтрации с использованием бинарной стоп-функции при значении порога P = 46.



Рис. 6.19. Иллюстрация действия билатерального фильтра: *а* — исходное изображение; *б* — изображение, полученное в результате билатеральной фильтрации

Сопоставление изображений показывает, что выполнение билатеральной фильтрации позволило полностью сохранить четкость границ объектов, зато малоконтрастные детали оказались устранены. Достигнутый эффект может быть успешно использован в целом ряде прикладных задач как один из необходимых этапов алгоритмов обработки изображений. Взяв разность исходного и обработанного
изображений, мы тем самым получим извлеченные малоконтрастные детали. Усилив контраст этих деталей и прибавив результат к профильтрованному изображению, мы получим новое изображение с повышенным контрастом мелких деталей. Проиллюстрируем это на примере следующего алгоритма.

Применение билатеральной фильтрации для улучшения воспроизведения небольших малоконтрастных деталей на изображениях сцен с большим динамическим диапазоном на носителях, у которых динамический диапазон сравнительно невелик

Основная идея, положенная в основу этого алгоритма [123], заключается в том, что вначале обрабатываемое изображение $L_c(k,n)$ подвергается билатеральной фильтрации, в результате чего получается изображение $L_{c\Omega}(k,n)$, из которого исключены детали небольшого размера, создающие на изображении небольшие перепады яркости. Затем путем вычитания изображения, полученного в результате билатеральной фильтрации, из исходного изображения находится "изображение" деталей, которые необходимо сохранить:

$$L_{\mathrm{fl}}(k,n) = L_{\mathrm{c}}(k,n) - L_{\mathrm{c}\Omega}(k,n).$$

Далее уменьшается динамический диапазон изображения $L_{c\Omega}(k,n)$ до величины динамического диапазона дисплея или другого носителя путем умножения на множитель, много меньший единицы $K \ll 1$. На заключительном этапе к компоненту $KL_{c\Omega}(k,n)$ добавляются сохраненные детали:

$$L_{\text{pe3}}(k,n) = K L_{\text{c}\Omega}(k,n) + L_{\mu}(k,n).$$

На рис. 6.20 для сравнения приведены два изображения интерьера с большим динамическим диапазоном яркости. Изображение, показанное на рис. 6.20, *a*, получено путем простого сжатия его динамического диапазона до величины, определяемой шкалой квантования в 255 уровней. На этом изображении в области темных полутонов невозможно различить какие-либо детали вследствие того, что при сжатии динамического диапазона исходного изображения сжатию были подвергнуты и перепады яркости, создаваемые деталями. Изображение на рис. 6.20, *б* получено описанным выше методом. На этом изображении хорошо различимы детали в области темных полутонов (предметы на стене вокруг дверного проема) при практически полном отсутствии ореолов на всем изображении благодаря тому, что при сжатии динамического диапазона детали были сохранены.

Как показали дальнейшие исследования, билатеральная фильтрация может быть с успехом применена и для ряда других целей.





Рис. 6.20. К иллюстрации результата обработки изображения методом, в котором применена билатеральная фильтрация; *а* — исходное изображение; *б* — изображение, полученное методом, в котором применена билатеральная фильтрация

Применение билатеральной фильтрации для ослабления заметности ложных контуров на проквантованном изображении

На основе билатеральной фильтрации можно реализовать алгоритм устранения ложных контуров из изображения, возникающих при его оцифровке. Поясним сказанное примером.

Предположим, что в нашем распоряжении имеется изображение, яркость каждого пиксела которого представлена четырьмя разрядами двоичного кода, т. е. изображение, проквантованное на 16 уровней. На таком изображении хорошо различимы ложные контуры. Для того чтобы их устранить, следует сначала преобразовать это изображение в изображение с большим количеством разрядов кода, например в восьмиразрядное, путем дописывания к коду каждого пиксела 4-х дополнительных разрядов, заполненных нулями. При этом расстояния между квантовыми уровнями, соответствующими этим ложным контурам в новой шкале квантования, составят 16 квантовых уровней. После этого преобразованное изображение подвергается билатеральной фильтрации, причем стоп-функция, корректирующая значения коэффициентов импульсной функции, должна быть выбрана таким образом, чтобы

отклонения яркости, величина которых составляет 16 уровней квантования и менее, подавлялась бы. На рис. 6.21, a в качестве примера приведено изображение, проквантованное на 16 уровней после его преобразования к восьмиразрядному коду, на котором хорошо различимы ложные контуры, а на рис. 6.21, δ — это же изображение, но после его обработки описанным алгоритмом.



Рис. 6.21. Иллюстрация ослабления заметности ложных контуров на изображении: a — изображение, проквантованное на 16 уровней после его преобразования к восьмиразрядному коду; δ — это же изображение, но после его обработки описанным алгоритмом (P = 17, $r_e = 4$). Изображение (a) заимствовано из [84]

Рассмотрение рис. 6.21, *б* показывает, что описанный алгоритм действительно полностью подавляет на изображении ложные контуры, улучшая его вид, но одновременно с этим на изображении подавляются также световые границы, обусловленные перепадом яркости, величина которых равна 16 квантовым уровням. Это особенно хорошо заметно в изображении соломенной шляпы, структура которой стала гладкой.

Применение билатеральной фильтрации для ослабления заметности гауссова шума на изображении

Билатеральная фильтрация также может быть использована для ослабления небольшого по величине аддитивного гауссова шума на изображении.

В этом случае, применяя бинарную стоп-функцию и задавая небольшое значение порога *P*, можно получить некоторое ослабление шума, поскольку малые выбросы

шума, не превышающие значения порога, будут сглажены. Рисунок 6.22 демонстрирует это. При увеличении порога пропадают малоконтрастные детали, но увеличивается ослабление шума, при уменьшении порога ослабляется эффект от фильтрации, но начинают воспроизводиться малоконтрастные детали.



Рис. 6.22. Иллюстрация эффекта сглаживания гауссова шума на изображении: *а* — исходное изображение; *б* — сглаженное изображение (P = 17, $r_e = 4$)

Применение билатеральной фильтрации для выделения текстур из изображений

Билатеральная фильтрация может быть применена для выделения текстур из изображений, которые обычно представляют собой небольшие, часто шумоподобные, отклонения яркости от среднего значения. В этом случае текстуры могут быть найдены как разность между изображением $L_c(k,n)$ и его версией $L_{c\Omega}(k,n)$, полученной в результате билатеральной фильтрации:

$$L_{\mathrm{T}}(k,n) = L_{\mathrm{c}}(k,n) - L_{\mathrm{c}\Omega}(k,n).$$

При выделении текстуры пользователю необходимо в каждом отдельном случае решать, какие изменения яркости следует отнести к текстуре, а какие к контурному компоненту изображения [47]. При выделении текстуры необходимо установить значение порога P, определив тем самым значение максимального отклонения яркости, которое еще может быть отнесено к текстуре, и условный радиус импульсной характеристики r_e , посредством которого определяется, детали какого размера будут сглажены при билатеральной фильтрации.



Рис. 6.23. Пример выделения текстуры из изображения: a — изображение; δ — текстура. При выделении текстуры были использованы следующие параметры билатерального фильтра: P = 8 квантовых уровней, $r_e = 8$. Контраст текстуры, показанной на рисунке, увеличен

На рис. 6.23 в качестве примера приведено исходное изображение и выделенная из него текстура, помещенная на серый фон, яркость которого составляет 127 квантовых уровней. Поскольку контраст текстуры невелик, то для лучшей различимости ее контраст на рисунке увеличен.

Билатеральную фильтрацию целесообразно также использовать для предварительной обработки изображений перед получением из них контурных препаратов одним из известных методов. Это позволяет уменьшить в полученном препарате количество разрывов контуров, а также точек и штрихов в его фоновой части.

6.10. Методы коррекции апертурных искажений изображений, основанные на использовании их семантической структуры

В качестве примера рассмотрим частный случай, когда апертурные искажения обусловлены несовершенством преобразователя изображения в сигнал, а его частотная передаточная функция $K_{\mu c \kappa}(\omega_x, \omega_y)$ достаточно хорошо может быть аппроксимирована гауссовой функцией

$$K_{\mu c\kappa}\left(\omega_{x},\omega_{y}\right) = \exp\left[-\frac{\left(\omega_{x}^{2}+\omega_{y}^{2}\right)}{\Omega^{2}}\right],$$

где Ω — приведенная частота. Для коррекции апертурных искажений применяют апертурные корректоры, частотная передаточная функция которых в условиях ма-

лого уровня шума выбирается равной $K_{\text{кор}}(\omega_x, \omega_y) = 1/K_{\text{иск}}(\omega_x, \omega_y)$, т. е. выполняют по существу инверсную фильтрацию искаженного изображения

$$K_{\text{kop}}\left(\omega_{x},\omega_{y}\right) = \exp\left[\frac{\left(\omega_{x}^{2}+\omega_{y}^{2}\right)}{\Omega^{2}}\right],$$

при этом спектр корректируемого изображения $M_{c\Omega}(\omega_x, \omega_y)$ и спектр скорректированного изображения $M_c(\omega_x, \omega_y)$ оказываются связанными соотношением

$$M_{c}(\omega_{x},\omega_{y}) = M_{c\Omega}(\omega_{x},\omega_{y})K_{\kappa o p}(\omega_{x},\omega_{y})$$

На практике частотная характеристика апертурного корректора $K_{\text{кор}}(\omega_x, \omega_y)$ формируется упрощенным способом, при котором экспоненциальную функцию, описывающую частотную передаточную функцию апертурного корректора, представляют в виде двух первых членов степенного ряда

$$M_{c}\left(\omega_{x},\omega_{y}\right) = M_{c\Omega}\left(\omega_{x},\omega_{y}\right) + \frac{\left(\omega_{x}^{2}+\omega_{y}^{2}\right)}{\Omega^{2}}M_{c\Omega}\left(\omega_{x},\omega_{y}\right).$$
(6.3)

Применяя двумерное преобразование Фурье к левой и правой частям этого равенства, найдем

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{c}(\omega_{x}, \omega_{y}) \exp\left[\mathbf{i}(\omega_{x}x + \omega_{y}y)\right] d\omega_{x}d\omega_{y} =$$

$$= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{c\Omega}(\omega_{x}, \omega_{y}) \exp\left[\mathbf{i}(\omega_{x}x + \omega_{y}y)\right] d\omega_{x}d\omega_{y} +$$

$$+ \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \frac{(\omega_{x}^{2} + \omega_{y}^{2})}{\Omega^{2}} M_{c\Omega}(\omega_{x}, \omega_{y}) \exp\left[\mathbf{i}(\omega_{x}x + \omega_{y}y)\right] d\omega_{x}d\omega_{y},$$

где $\mathbf{i} = \sqrt{-1}$. Учитывая, что

$$L_{c\Omega}(x,y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{c\Omega}(\omega_x,\omega_y) \exp\left[i(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y, \qquad (6.4)$$

$$L_{\rm c}(x,y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} M_{\rm c}(\omega_x,\omega_y) \exp\left[i(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y , \qquad (6.5)$$

где $L_{c\Omega}(x, y)$ и $L_{c}(x, y)$ — распределение яркости по координатам x и y в корректируемом и скорректированном изображениях, а также что

$$\frac{\partial^2 L_{c\Omega}(x,y)}{\partial x^2} = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (-1)\omega_x^2 M_{c\Omega}(\omega_x,\omega_y) \exp\left[\mathbf{i}(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y ,$$

$$\frac{\partial^2 L_{c\Omega}(x,y)}{\partial y^2} = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (-1)\omega_y^2 M_{c\Omega}(\omega_x,\omega_y) \exp\left[\mathbf{i}(\omega_x x + \omega_y y)\right] d\omega_x d\omega_y ,$$

в чем не трудно убедиться путем двойного дифференцирования выражений (6.4) и (6.5) по *x* и *y*, найдем

$$L_{c}(x, y) = L_{c\Omega}(x, y) - \frac{1}{\Omega^{2}} \left[\frac{\partial^{2} L_{c\Omega}(x, y)}{\partial x^{2}} + \frac{\partial^{2} L_{c\Omega}(x, y)}{\partial y^{2}} \right],$$

а поскольку

$$\Delta(x, y) = \left[\frac{\partial^2 L_{c\Omega}(x, y)}{\partial x^2} + \frac{\partial^2 L_{c\Omega}(x, y)}{\partial y^2}\right]$$

представляет собой оператор Лапласа, получим окончательно

$$L_{c}(x, y) = L_{c\Omega}(x, y) - \frac{1}{\Omega^{2}}\Delta(x, y).$$

Найденное выражение описывает *метод нерезкого маскирования* (см. разд. 5.11). Другими словами, апертурную коррекцию обычно выполняют посредством применения метода нерезкого маскирования. В цифровых системах координаты x и y заменяют номерами строк и столбцов, а само нерезкое маскирование выполняют путем свертки корректируемого изображения $L_1(k,n)$ с импульсной функцией h(k,n)

$$h(k,n) = \begin{vmatrix} 0 & -b & 0 \\ -b & 1+4b & -b \\ 0 & -b & 0 \end{vmatrix}$$

с использованием выражения

$$L_{c}(k,n) = \sum_{k'=-1}^{1} \sum_{n'=-1}^{1} L_{c\Omega}(k+k', n+n')h(k', n'),$$

где *b* — коэффициент, определяющий глубину коррекции. Чем больше размыто корректируемое изображение, тем большей величины должна выбираться величина этого коэффициента.

Рассмотренный метод коррекции обладает двумя недостатками.

- Уже при сравнительно небольшом уровне аддитивного гауссового шума коррекция апертурных искажений приводит к возрастанию зашумленности скорректированного изображения. Вследствие этого коррекцию вынуждены делать неполной, для того чтобы избежать снижения качества воспроизведения изображения вследствие возрастания его зашумленности.
- При сколько-нибудь значительном уровне апертурных искажений попытка их полностью скорректировать приводит к появлению на скорректированном изображении окантовок. Обусловлено это тем, что нерезкое маскирование не обеспечивает полной коррекции апертурных искажений вследствие того, что требуемая частотная передаточная функция апертурного корректора и частотная передаточная функция алгоритма, реализующего нерезкое маскирование, не идентичны, как это видно из рис. 6.24. Эта неидентичность вызвана приближением (6.3), принятым при расчете частотной передаточной функции апертурного корректора.



Рис. 6.24. График, иллюстрирующий несовпадение требуемой частотной передаточной функции алгоритма, реализующего нерезкое маскирование

На рис. 6.25, *а* показана осциллограмма резкой неискаженной световой границы, на рис. 6.25, *б* осциллограмма этой же границы при наличии апертурных искажений, а на рис. 6.25, *в* осциллограмма скорректированной границы рассмотренным выше методом.

На осциллограмме рис. 6.25, в хорошо видно, что вблизи резких световых границ появляются выбросы, подобные тем, которые имеют место при резком ограничении спектра. На рис. 6.26, а приведено изображение прямоугольника с резкими

световыми границами, на рис. 6.26, *б* показано это же изображение, но с размытыми световыми границами вследствие апертурных искажений, а на рис. 6.26, *в* приведено скорректированное изображение, на котором видны окантовки, обусловленные выбросами вблизи границ.



Рис. 6.25. Осциллограммы: *a* — сигнала от неискаженной вертикальной световой границы; *б* — сигнала от той же границы, но при наличии апертурных искажений; *в* — сигнала после коррекции рассмотренным методом



Рис. 6.26. Изображения: a — исходное, без искажений; δ — с апертурными искажениями; b — скорректированное

С аналогичной проблемой мы встречаемся и при коррекции апертурных искажений, обусловленных неточной фокусировкой или недостатками оптики.

Указанных недостатков можно избежать, если при коррекции искаженные контурные компоненты заменять регенерированными, т. е. неразмытыми. В этом случае вначале на корректируемом изображении находятся контурные, градиентные и фактурные компоненты, которые заменяются их регенерированными версиями. Для правильной регенерации величины скачков интенсивности на контурных компонентах (особенно это касается регенерации тонких линий) необходимо располагать информацией о степени размытости изображения (об импульсной характеристике эквивалентного размывающего фильтра), которая обычно легко определяется. При выделении контурных компонентов необходимо также опытным путем определить пороговые значения скачков интенсивности, при превышении которых пиксел следует считать принадлежащим контурному компоненту. Не останавливаясь подробно на алгоритме коррекции размытых границ изображений, который очевиден, отметим, что такой подход к коррекции в ряде случаев реализован в коммерческих приложениях. Хорошей иллюстрацией этому может служить приложение PhotoZoom Pro 2.1.4, предназначенное для изменения размера изображений. Но это приложение можно также использовать и для коррекции размытых границ изображения без его увеличения, что демонстрируется рис. 6.27.



Рис. 6.27. Изображения: a — исходное, без искажений; δ — с апертурными искажениями; s — скорректированное

В данном случае при выделении контурного компонента использовался S-Spline XL. Априорное знание относительно резкости и гладкости контурных компонентов, о котором было сказано ранее, может быть использовано также при увеличении изображений.

6.11. Гомоморфная фильтрация изображений

Мультипликативные помехи, воздействуя на изображения, искажают их и одновременно увеличивают динамический диапазон яркостей, необходимый для их представления. Это, в свою очередь, создает проблемы при воспроизведении таких изображений, поскольку динамический диапазон воспроизводящих устройств, таких, как мониторы, принтеры и др., не превышает 100—255. Возникают проблемы и при их передаче по каналам связи.

Эффективным методом реставрации изображений, искаженных мультипликативными помехами, в ряде случаев может быть *гомоморфная фильтрация* [23, 84, 168].

Рассмотрим алгоритм гомоморфной фильтрации на примере реставрации ахроматического изображения для случая, когда мультипликативная помеха обусловлена неравномерностью освещения по полю, хотя он может быть применен и для фильтрации цветных изображений. Согласно модели, приведенной в *разд.* 1.5, распределение яркости в изображении можно рассматривать как результат перемножения функции $r_c(x, y)$, представляющей распределение коэффициентов отражения в передаваемой сцене, на функцию E(x, y), которая представляет распределение освещенности

$$L(x, y) = k E(x, y) r_{\rm c}(x, y), \qquad (6.6)$$

где k — коэффициент, согласующий размерности. Если освещенность по полю постоянна, т. е. если E(x, y) = E = const, то распределение яркости в изображении $L_{c}(x, y)$ с точностью до постоянного множителя kE повторяет распределение коэффициентов отражения

$$L_{\rm c}(x, y) = k E r_{\rm c}(x, y).$$

Будем считать такое изображение неискаженным.

Динамический диапазон изменения яркости в неискаженных изображениях определяется динамическим диапазоном изменения коэффициентов отражения и обычно не превышает 100. В большинстве же реальных случаев формирования изображений освещенность различных участков снимаемой сцены изменяется, причем в больших пределах, что приводит к увеличению динамического диапазона яркостей формируемых изображений. Поэтому яркость черных букв при ярком солнечном освещении оказывается больше яркости белой бумаги, на которой они напечатаны, в тени.

Цель гомоморфной фильтрации состоит в ослаблении мультипликативной помехи путем отыскания так называемой оценки изображения $\hat{L}_{c}(x, y)$. Достигается это следующим образом. Вначале искаженное изображение, представленное распределением яркости на плоскости L(x, y), логарифмируется, при этом мультипликативная помеха, как не трудно видеть, преобразуется в аддитивную

$$\ln\left[L(x, y)\right] = \ln(k) + \ln\left[E(x, y)\right] + \ln\left[r_{c}(x, y)\right].$$

Так как изменение освещенности с изменением координат x, y, как правило, происходит медленно, основная энергия пространственного спектра функции E(x, y)оказывается сосредоточенной в области низких пространственных частот. Для функции же $r_c(x, y)$, представляющей распределение коэффициентов отражения в передаваемой сцене, напротив, характерно наличие резких изменений на световых границах, что делает ее пространственный спектр широкополосным. Это отличие в пространственных спектрах функций $r_c(x, y)$ и E(x, y), имеющее место также и в логарифмах этих функций, используется для ослабления помехи путем линейной фильтрации функции ln [L(x, y)].



Рис. 6.28. Алгоритм гомоморфной фильтрации



Рис. 6.29. Гомоморфная фильтрация изображения: *а* — исходное (искаженное) изображение; *б* — это же изображение после гомоморфной фильтрации

Заключительным шагом алгоритма гомоморфной фильтрации является антилогарифмирование, в результате которого находится оценка неискаженного изображения $\hat{L}_{c}(x, y)$. На рис. 6.28 приведен алгоритм гомоморфной фильтрации.

Обычно в качестве иллюстрации эффективности рассмотренного метода приводят изображения, показанные на рис. 6.29 (см., например, [23, 168]).

На этих изображениях видно, как ранее неразличимые детали, находящиеся внутри затемненного помещения, становятся хорошо различимыми в результате гомоморфной фильтрации. Однако этот метод далеко не всегда дает такие результаты. В тех случаях, когда изображение содержит ряд небольших областей, яркость которых сильно отличается от яркости остальной его части, на изображении в результате гомоморфной фильтрации возникают ореолы. В этих случаях лучшие результаты дают методы, в которых применена билатеральная фильтрация.

6.12. Ослабление искажений фотокопии, обусловленных неравномерностью освещенности поверхности фотографируемого оригинала

В ряде случаев при изготовлении копий изображений путем их фотографирования или сканирования на них возникают характерные искажения, обусловленные неравномерной освещенностью оригиналов. Особенно часто с этим приходится встречаться при сканировании текстов. Эти искажения можно существенно ослабить путем гомоморфной фильтрации искаженных изображений. Однако, как показывает практика, существенно лучшие результаты можно получить посредством метода коррекции, заключающегося в нормировке искаженного изображения на изображение, представляющее собой неравномерное распределение освещенности по поверхности копируемого оригинала [36].

Существо этого метода поясним на примере коррекции изображения, показанного на рис. 6.30. Исходя из модели изображения, представленной формулой (6.6), приближенно определим распределение освещенности $\tilde{E}(x, y)$ в копируемом оригинале путем его низкочастотной фильтрации, т. е.

$$\tilde{E}(x, y) = L(x, y) \otimes h(x, y),$$

где \otimes — символ свертки, h(x, y) — сглаживающая импульсная характеристика.

В книге рассмотрен широкий круг вопросов начиная от ввода изображений в память компьютера и вывода либо на экран монитора в случае мультимедиа и цифрового видео, либо на печать в случае политрафии, а также вопросам оцифровки изображений, вопросам сжатия, линейной и нелинейной обработкам. Важной отличительной особенностью изложения является то, что

оно проводится применительно не к абстрактным двумерным массивам чисед, над которыми совершаются некоторые математические преобразования, как это часто практикуется при изложении подобных вопросов, а применительно к реальным семантическим изображениям, характеризуемым вполне определенными свойствами. При этом сама обработка преследует цель улучшить качество изображений в рамках того или иного критерия качества, полученного на основе модели зрения. Такой подход потребовал включения в книгу глав, посвященных описанию характеристик семантических изображений, свойств зрительной системы, а также изложения основных положений колориметрии. При изложении материала показаны имеющиеся ограничения в достижении результатов обработки, обусловленные самой природой используемых процессов. Книга ориентирована на инженеров, работающих в области обработки изображений, аспирантов и студентов старших курсов. В монографию вошли материалы исследований, поддержанные грантом РФФИ 98-06-80001.

Рис. 6.30. Изображение с искажениями,

обусловленными неравномерной освещенностью оригинала при его копировании



Рис. 6.31. Изображение, представляющее приближенное распределение освещенности оригинала при его копировании

Параметры этой импульсной характеристики (ее протяженность) следует экспериментально подобрать таким образом, чтобы отфильтровать все детали самого изображения, в нашем примере буквы текста. На рис. 6.31 показан результат фильтрации изображения, приведенного на рис. 6.30.

Заключительной операцией обработки является нормировка яркости обрабатываемого изображения L(x, y), т. е. корректируемого изображения, на найденное значение освещенности $\tilde{E}(x, y)$ (на профильтрованное изображение) с последующим приведением его к желаемому динамическому диапазону яркостей путем умножения на новое значение коэффициента, согласующего размерности k_1 , величина которого подбирается экспериментально. При этом скорректированное изображение $\hat{L}_c(x, y)$ находится по формуле

$$\hat{L}_{c}(x, y) = k_{1} \frac{L(x, y)}{\tilde{E}(x, y)}.$$

На рис. 6.32 приведено скорректированное изображение, полученное описанным методом. Рассмотрение этих рисунков показывает, что выравнивание имеет место, но естественно не полное, хотя погрешности практически незаметны. Достичь идеальной компенсации не удается, поскольку найденное приближенное распределение освещенности $\tilde{E}(x, y)$, показанное на рис. 6.31, в своем составе содержит остатки исходного (текста). В книге рассмотрен широкий круг вопросов начиная от ввода изображений в память компьютера и вывода либо на экран монитора в случае мультимедиа и цифрового видео, либо на печать в случае полиграфии, а также вопросам оцифровки изображений, вопросам сжатия, линейной и нелинейной обработкам.

Важной отличительной особенностью изложения является то, что оно проводится применительно не к абстрактным двумерным массивам чисед над которыми совершаются некоторые математические преобразования, как это часто практикуется при изложении подобных вопросов, а применительно к реальным семантическим изображениям, характеризуемым вполне определенными свойствами. При этом сама обработка преследует цель улучшить качество изображений в рамках того или иного критерия качества, полученного на основе модели зрения. Такой подход потребовал включения в книгу глав, посвященных описанию характеристик семантических изображений, свойств зрительной системы, а также изложения основных положений колориметрии. При изложении материала показаны имеющиеся ограничения в достижении результатов обработки, обусловленные самой природой используемых процессов.

Книга ориентирована на инженеров, работающих в области обработки изображений, аспирантов и студентов старших курсов.

В монографию вошли материалы исследований, поддержанные грантом РФФИ 98-06-80001.









в

Рис. 6.33. Изображения: *а* — исходное, неискаженное, *б* — искаженное в результате неравномерного распределения освещенности; *в* — скорректированное

В случае обработки сюжетного изображения, структура которого имеет нерегулярный характер, остаточные явления становятся более заметными, что иллюстрируется рис. 6.33, *a—в*. Из приведенных рисунков следует, что рассмотренный метод может быть использован только для коррекции изображений, носящих более или менее регулярный характер. Примером может служить коррекция изображений отсканированных страниц журнала, на которых имеют место искажения рассматриваемого типа, перед их распознаванием программой FineReader и подобных ей.

6.13. Медианная и ранговая фильтрация изображений

Медианная фильтрация была разработана Тьюки [23, 73, 84]. Реализуется медианная фильтрация путем сканирования изображения окном, перекрывающим нечетное число его пикселов N, с последующим ранжированием значений яркости последовательности пикселов, оказавшихся в данный момент в окне, и записи в массив профильтрованного изображения значения, равного медиане этих пикселов. Медианой последовательности L_1 , L_2 , ..., L_N является тот ее элемент, для которого (N-1)/2 элементов последовательности меньше или равны ему по величине, и (N-1)/2 элементов больше или равны ему по величине. Поясним это на численном примере. Пусть в окно медианного фильтра попала последовательность пикселов изображения, яркость которых определяется числами: 95, 122, 121, 90, 99, 200, 100, 111, 120. В рассматриваемом примере медианой этой дискретной последовательности будет число 111, поскольку после ее ранжирования оно займет в ней центральное положение: 90, 95, 99, 100, 111, 120, 121, 122, 200.

Медианная фильтрация не гарантирует того, что в результате ее применения на изображении будут подавлены именно помехи, а полезные изменения яркости сохранятся. Так, в рассмотренном примере значение 200 подавляется, не зависимо от того, обусловлено ли оно выбросом помехи или же является значением яркости пиксела в исходном изображении.

На рис. 6.34 приведены примеры фильтрации трех последовательностей дискретных отсчетов яркости изображения одномерным медианным и одномерным линейным фильтрами. В обоих случаях использовались пятиэлементные окна. При этом все отсчеты импульсной характеристики линейного фильтра были одинаковыми и равными 1/5.

Из рассмотрения диаграмм, показанных на рис. 6.34, следует, что медианный фильтр в отличие от линейного фильтра не искажает передачу ступенчатой последовательности отсчетов и в то же время эффективно подавляет импульсные последовательности, если число импульсов в последовательности не превышает половину ширины окна. Благодаря этому свойству медианного фильтра можно ожидать, что при медианной фильтрации изображений, искаженных импульсными помехами, воспроизведение имеющихся в них контуров и резких световых границ будет мало искажаться, при эффективном подавлении импульсных помех.



Рис. 6.34. К сравнению результатов фильтрации медианным и линейным фильтрами



Рис. 6.35. Примеры двумерной медианной фильтрации с применением крестообразного и квадратного окон

До сих пор мы рассматривали примеры, в которых фильтрация импульсных последовательностей осуществлялась одномерным медианным фильтром, т. е. фильтром, окно которого охватывало последовательность пикселов, расположенных вдоль прямой линии. Однако лучшие результаты при фильтрации изображений можно получить при использовании двумерных фильтров, т. е. фильтров, окна которых имеют протяженность в двух направлениях. На рис. 6.35 приведены примеры двумерной медианной фильтрации изображения квадрата фильтрами, окна которых имели прямоугольную и крестообразную формы.

Размеры окон составляли соответственно 3×3 и 5×5 отсчетов. Из рассмотрения рис. 6.35 следует, что медианный фильтр, имеющий прямоугольное окно, сглаживает углы квадрата на профильтрованном изображении, в то время как медианный фильтр, имеющий крестообразное окно, оставляет их без изменения.

Известно, что при определении медианы последовательности отсчетов f(i), где i — номер отсчета, выполняются следующие соотношения:

$$\operatorname{med}[kf(i)] = k \operatorname{med}[f(i)],$$
$$\operatorname{med}[k + f(i)] = k + \operatorname{med}[f(i)].$$

Вместе с тем, медиана суммы двух произвольных последовательностей f(i) и g(i) не равна сумме их медиан, т. е.

$$\operatorname{med}\left[f(i) + g(i)\right] \neq \operatorname{med}\left[f(i)\right] + \operatorname{med}\left[g(i)\right].$$

Так как медианная фильтрация является эвристическим методом обработки изображений, то не представляется возможным разработать методику синтеза "оптимального медианного фильтра", т. е. определения размеров и формы окна, обеспечивающих наилучшие результаты в рамках какого-либо критерия качества при заданных характеристиках изображений и помех. Поэтому параметры медианного фильтра (размеры и форма окна) должны определяться опытным путем.

Применяют различные стратегии решения этой задачи. Одна из них состоит в том, что сначала берут окно крестообразной формы наименьших размеров, т. е. 3×3 и осуществляют пробную фильтрацию изображения. Если при этом искажения изображения, обусловленные фильтрацией, незначительны, а уровень остаточной помехи на изображении велик, размер окна увеличивают до 5×5 и снова выполняют пробную фильтрацию. Поступая таким образом, находят приемлемые размеры окна медианного фильтра.

Другая стратегия заключается в реализации каскадной медианой фильтрации изображения с применением окон с фиксированными или изменяемыми размерами. При этом те области изображения, которые остаются без изменения после однократной обработки медианным фильтром, не изменятся и после его повторной обработки. Те же области изображения, в которых протяженность импульсной последовательности менее половины размеров окна, будут подвергаться изменениям после каждого цикла обработки.





Рис. 6.36. Медианная фильтрация изображения:
 a — изображение с импульсной помехой;
 δ — изображение после медианной фильтрации;
 в — изображение после линейной фильтрации

В качестве примера на рис. 6.36, *а* и *б* показаны исходное изображение, искаженное импульсной помехой, и это же изображение после его медианой фильтрации фильтром с крестообразным окном размерами 5×5.

Из рассмотрения рис. 6.36, δ следует, что применение медианной фильтрации позволило почти полностью устранить на изображении импульсную помеху, при этом практически не исказив само изображение. Для сравнения на рис. 6.36, *в* приведено изображение, полученное из изображения 6.36, *a* путем его линейной фильтрации с использованием прямоугольной импульсной характеристики с размером окна 5×5 пикселов. Сопоставление рисунков 6.36, δ и *в* показывает, что в последнем случае имеет место сильное размытие границ на изображении и неполное удаление помехи.

Распространение процедуры медианной фильтрации на случай обработки цветных изображений очевидно и пояснений не требует.

Следует отметить, что медианный фильтр может быть не только одномерным или двумерным, но также и трехмерным, используя в качестве третьего измерения время.



Рис. 6.37. Крестообразное окно трехмерного медианного фильтра

На рис. 6.37 показано крестообразное окно трехмерного медианного фильтра. Использование трехмерной медианной фильтрации расширяет ее возможности, т. к. позволяет подавлять импульсные помехи, воздействующие на телевизионное изображение.

Алгоритм *ранговой фильтрации* во многом подобен алгоритму медианной фильтрации и отличается от нее лишь тем, что в массив профильтрованного изображения записывается значение яркости отсчета, положение которого в ранжированной последовательности r предварительно задается пользователем в диалоге. Таким образом, медианная фильтрация является частным случаем ранговой фильтрации для случая, когда r = N/2+1. Наличие дополнительной степени свободы при выборе параметров рангового фильтра в некоторых случаях позволяет достичь несколько лучших результатов.

6.14. Адаптивная медианная фильтрация изображений

Недостатком медианной и ранговой фильтрации является то, что она эффективна только при сравнительно небольших вероятностях импульсной помехи, порядка $p_n \leq 0, 2$. Для устранения этого недостатка был разработан алгоритм адаптивной медианной фильтрации, который предназначен для ослабления более интенсивной биполярной импульсной помехи, вероятность появления импульсов которой превышает приведенное выше значение [23]. Кроме того, этот алгоритм обладает тем преимуществом, что он в меньшей степени искажает детали изображения, не поврежденные импульсным шумом. Особенностью адаптивного алгоритма является то, что в отличие от обычного медианного фильтра он при определенных условиях увеличивает размер окна, перекрывающего нечетное число пикселов, которым сканируется фильтруемое изображение. Рассмотрим его подробнее. При реализации

алгоритма измеряются следующие значения интенсивностей пикселов, оказавшихся в пределах окна, которое, как и ранее, может иметь любую форму (прямоугольную, крестообразную и т. д.):

□ максимальное значение интенсивности *L*_{макс};

- □ минимальное значение интенсивности (яркости) *L*_{мин};
- \square значение интенсивности пиксела, занимающего центральное положение в окне L_c ;
- □ медиана последовательности пикселов, оказавшихся в окне *L*_{мел};
- □ максимальный допустимый размер фильтрующего окна *S*_{макс}, который в диалоге задается числом пикселов.

Алгоритм адаптивной медианной фильтрации включает в себя две ветви: I и II. Задача, которую выполняет первая ветвь, — это определение, является ли медиана L_{мел} результатом воздействия импульса помехи (положительного или отрицательного) на изображение, или нет. В том случае если выполняется условие $L_{\text{мин}} < L_{\text{мел}} < L_{\text{макс}}$, то считается, что найденное значение $L_{\text{мел}}$ не является результатом воздействия импульса помехи на изображение, и тогда совершается переход к выполнению второй ветви алгоритма. При выполнении второй ветви алгоритма проверяется, является ли значение интенсивности пиксела, занимающего центральное положение в окне, L_c результатом воздействия импульса помехи (положительного или отрицательного) на изображение, или нет. В том случае если выполняется условие $L_{\text{мин}} < L_{\text{с}} < L_{\text{макс}}$, то считается, что значение $L_{\text{с}}$ не является результатом воздействия импульса помехи на изображение, и в качестве результата фильтрации принимается значение $L_{\rm c}$, а не значение медианы $L_{\rm мел}$. Благодаря этому минимизируются искажения, неизбежно возникающие при фильтрации изображения. В том случае, если это неравенство не выполняется, т. е. либо $L_{c} = L_{makc}$, либо $L_{\rm c} = L_{\rm MMH}$, считается, что это результат воздействия импульса помехи на изображение, и в качестве результата фильтрации принимается значение L_{мел}, которое, как следует из результата работы первой ветви алгоритма, не является следствием воздействия импульса помехи.

Продолжая изложение работы алгоритма, рассмотрим случай, когда при выполнении первой ветви алгоритма условие $L_{\text{мин}} < L_{\text{мед}} < L_{\text{макс}}$ оказывается нарушенным, т. е. случай, когда медиана считается результатом воздействия импульса помехи на изображение. В этом случае согласно алгоритму увеличиваются размеры фильтрующего окна и повторяются вычисления первой ветви алгоритма. Так будет продолжаться до тех пор, пока либо не будет найдена медиана, не считающаяся результатом воздействия импульса помехи, либо размеры окна не достигнут максимально разрешенного размера $S_{\text{макс}}$. В последнем случае в качестве результата фильтрации принимается значение $L_{\rm c}$.

Далее приводится схема алгоритма.

в

Ветвь I: $A1 = L_{Med} - L_{MuH};$ $A2 = L_{Med} - L_{Makc};$ если A1 > 0 и A2 < 0, то перейти к ветви II; иначе увеличить размер окна; если размер окна меньше либо равен S_{Makc} , повторить ветвь I; иначе принять в качестве результата L_c .

Ветвь II: $B1 = L_c - L_{MUH}$; $B2 = L_c - L_{MAKC}$; если B1 > 0 и B2 < 0, принять в качестве результата L_c ; иначе принять в качестве результата L_{MER} .



На рис. 6.38 в качестве иллюстрации приведены изображения, иллюстрирующие работу рассмотренного алгоритма для случая, когда величина импульса помехи составляла 255 уровней квантования, т. е. равнялась динамическому диапазону изображения. При этом воздействие импульса помехи на изображение, в зависимости от его полярности, делало соответствующий пиксел либо белым, либо черным. Из рассмотрения рисунка следует, что применение приведенного алгоритма адаптивной медианной фильтрации практически полностью устраняет заметность биполярной импульсной помехи на изображении, в то время как обычная медианная фильтрации не справляется с этой задачей.

Проведенное нами более детальное исследование этого алгоритма показало, что, к сожалению, он эффективен лишь в случае, когда импульсы помехи занимают весь динамический диапазон изображения. Если же величина импульсов помехи невелика, то его эффективность резко снижается. Рисунок 6.39 иллюстрирует этот случай.



Рис. 6.39. Медианная фильтрация изображения: а — исходное изображение; б — изображение с биполярной импульсной помехой при вероятности импульса p_n = 0,3 и величине 100 уровней квантования; в — изображение после адаптивной медианной фильтрации;

г — изображение после обычной медианной фильтрации

Из рассмотрения рисунка следует, что в этом случае алгоритм адаптивной медианной фильтрации уступает по своей эффективности алгоритму обычной (неадаптивной) медианной фильтрации.

6.15. Некоторые другие виды нелинейной фильтрации изображений

В литературе при обсуждении проблемы усреднения помех на изображении рассматривается ряд фильтров, предназначенных для этой цели [23]. Приведем характерные особенности некоторых из них.

Среднегеометрический фильтр

При использовании этого вида фильтрации фильтруемое изображение сканируется прямоугольным окном, высота которого задается количеством строк K, а ширина — количеством столбцов N. Значение яркости пиксела $L_{c\Omega}(k,n)$, расположенного в k-й строке и n-м столбце профильтрованного изображения, определяется выражением

$$L_{c\Omega}(k,n) = \begin{bmatrix} \frac{K-1}{2} & \frac{N-1}{2} \\ \prod_{k'=-\frac{K-1}{2}}^{2} & \prod_{n'=-\frac{N-1}{2}}^{2} L_{c}(k-k',n-n') \end{bmatrix}^{\frac{1}{KN}},$$

где $L_{c}(k,n)$ — распределение яркости в исходном изображении. Обратим внимание на то, что значение яркости пиксела профильтрованного изображения определяется через корень степени $\frac{1}{KN}$ из произведения значений яркости пикселов, оказавшихся в пределах сканирующего окна. В [23] отмечается, что фильтрация этого типа приводит к сглаживанию шума, сравнимому с тем, которое имеет место при линейной фильтрации изображений с использованием импульсной функции, имеющей равномерное распределение и размеры KN, но при этом теряется меньше деталей изображения.

Среднегармонический фильтр

Этот вид фильтрации отличается от предыдущего видом выражения, посредством которого вычисляются значения яркости пикселов в профильтрованном изображении:

$$L_{c\Omega}(k,n) = \frac{KN}{\sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} \frac{1}{L_{c}(k-k',n-n')}}$$

Считается, что фильтрация этого типа обеспечивает эффективное подавление униполярной помехи, вызывающей появление на изображении белых точек, но не обеспечивает подавление униполярной помехи, вызывающей появление на изображении черных точек. Отмечается, что эта фильтрация также эффективно ослабляет помехи других типов, таких, как, например, гауссов шум.

Контргармонический фильтр

При этом виде фильтрации изображение также сканируется окном прямоугольной формы, значения яркости пикселов в профильтрованном изображении определяются в соответствии с выражением

$$L_{c\Omega}(k,n) = \frac{\left[\sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{c}(k-k',n-n')\right]^{Q+1}}{\left[\sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{c}(k-k',n-n')\right]^{Q}},$$

где Q является порядком фильтра. В [23] отмечается, что фильтрация этого типа эффективно подавляет импульсную помеху. При положительных значениях Qфильтр подавляет отрицательные импульсы помехи, а при отрицательных значениях Q фильтр подавляет положительные импульсы помехи. Однако если помеха биполярная, то одновременно фильтр не может подавить и положительные, и отрицательные выбросы помехи. Обратим внимание на то, что при Q = 0 фильтрация этого типа обращается в обычную линейную фильтрацию с использованием импульсной функции, имеющей равномерное распределение и размеры KN, а при Q = -1 — в среднегармоническую фильтрацию. Примеры фильтрации изображений рассмотренными фильтрами читатель найдет в [23].

6.16. Методы фильтрации гауссова шума, основанные на учете особенностей восприятия семантических изображений зрительной системой

Принципиальной особенностью рассматриваемых в этом разделе методов является то, что при их реализации стремятся с минимальной погрешностью воспроизвести детали изображения, наиболее важные для правильного восприятия зрительной системой объектов, составляющих изображаемую сцену, допуская при этом значительные погрешности при воспроизведении второстепенных, с точки зрения зри-

тельной системы, деталей. Благодаря этому изображение воспринимается как менее искаженное, чем в случае ранее применяемых традиционных методов фильтрации, хотя при этом средний квадрат разности между неискаженным (эталонным) и восстановленным изображениями может быть большим, чем в первом случае. Проиллюстрируем сказанное двумя методами на примерах обработки черно-белых полутоновых изображений, хотя эти методы с таким же успехом могут быть применены для фильтрации цветных изображений.

Метод, базирующийся на использовании кусочно-гладкой модели изображения

Этот метод [120] предназначен для оценки и устранения шума из изображения в автоматическом режиме, он базируется на использовании кусочно-гладкой модели изображения (piecewise smooth image model) [171]. Алгоритм этого метода включает в себя следующие этапы.

- 1. Вначале зашумленное изображение L(x, y) подвергается сегментации, при этом из множества известных методов сегментации авторы применяют так называемый *K*-метод, как это описано в [179]¹. В результате проведенной сегментации изображение оказывается разделенным на сегменты (области) Ω_i . При этом каждый сегмент представляется средним значением цвета и некоторой пространственной протяженностью. Пространственная протяженность задается таким образом, чтобы форма сегмента имела бы тенденцию к выпуклой форме и чтобы все сегменты имели бы примерно одинаковые размеры.
- Следующая операция, авторы ее называют посегментной аффинной реконструкцией (per-segment affine reconstruction), заключается в том, что каждый сегмент подвергается аффинному преобразованию, в результате которого для каждого сегмента находится функция L_{AF}(x, y), определяющая распределение яркости в его пределах, для которой

$$\sigma^{2} = \overline{\left[L(x, y) - L_{AF}(x, y)\right]^{2}}$$

минимально. В цитируемой работе эта функция называется аффинной реконструкцией сегмента. Далее принимается, что разность между зашумленным изображением и его аффинной реконструкцией

$$\Delta L(x, y) = L(x, y) - L_{AF}(x, y),$$

состоит из двух составляющих: составляющей текстуры $L_{\rm r}(x, y)$, и составляющей шума $L_{\rm m}(x, y)$:

$$\Delta L(x, y) = L_{\mathrm{T}}(x, y) + L_{\mathrm{III}}(x, y).$$

¹ Подробнее о *К*-методе *см. в разд. 9.12*.

Таким образом, исходное, зашумленное изображение рассматривается как сумма трех составляющих

$$L(x, y) = L_{AF}(x, y) + L_{T}(x, y) + L_{III}(x, y),$$

при этом составляющими, представляющими незашумленное изображение $L_{c}(x, y)$, т. е. сигнальными, являются первые две.

$$L_{c}(x, y) = L_{AF}(x, y) + L_{T}(x, y).$$

Далее в цитируемой работе принимается, что:

- аффинная реконструкция сегмента не является случайным процессом;
- текстура и шум являются случайными взаимно некоррелированными процессами, ковариационные матрицы которых — К_т и К_ш соответственно;
- сигнальный и шумовой компоненты взаимно независимы.
- 3. Авторы цитируемой работы отмечают, что если просто использовать аффинные реконструкции сегментов для реконструкции всего изображения в целом, то в нем возникнут ложные контуры и, кроме того, реальные границы станут более резкими. Во избежание этого производится оценка размытости границ в исходном, зашумленном изображении следующим образом. Вычисляются ряд размытых версий $L_{AF\Omega}(x, y, r)$ аффинной реконструкции $L_{AF}(x, y)$ путем ее свертки с импульсной характеристикой

$$h(x, y) = \frac{1}{\pi r^2} \exp\left(-\frac{x^2 + y^2}{r^2}\right),$$

где r — параметр, определяющий степень размытия. Чем больше r, тем больше размытие. Затем каждая граница C_{ij} между сегментами Ω_i и Ω_j расширяется в пять раз как в направлении области Ω_i , так и в направлении области Ω_j для того, чтобы получить маску Γ_{ij} . После этого находятся значения средних квадратов разностей исходного изображения L(x, y) и его размытых версий $L_{AF\Omega}(x, y, r)$ для каждого значения параметра r в пределах маски, т. е.

$$\sigma^{2}(r) = \left[L(x, y) - L_{AF\Omega}(x, y, r)\right]^{2}.$$

За значение параметра, характеризующего степень размытия в исходном изображении, принимается то, обозначим его $r_{\text{опт}}$, которому соответствует минимум среднего квадрата $\overline{\left[L(x, y) - L_{AF\Omega}(x, y, r)\right]^2}$, вычисленного в пределах маски Γ_{ij} . После этого производится замена неразмытых границ в пределах, определяемых маской Γ_{ij} , на размытые границы, взятые из аффинной реконструкции $L_{AF\Omega}(x, y, r)$, полученной при найденном параметре размытия $r_{\text{опт}}$.







в



Рис. 6.40. К пояснению метода устранения гауссова шума
 с зашумленного изображения: *a* — исходное не зашумленное изображение;
 δ — изображение, на которое наложен 10-процентный шум (пиковое отношение сигнала к шуму равно 10);
 в — это же изображение после обработки описанным методом;
 г — разность между изображениями (*a*) и (*b*)

- Далее, применяя Байесовский подход к решению задачи, находятся апостериорные оценки ковариационных матриц шума К_{аш} и текстуры К_{ат} соответственно.
- 5. Заключительным этапом алгоритма является реконструкция обрабатываемого изображения. Для этой цели авторы используют: исходное, зашумленное изображение, его аффинную реконструкцию, полученную при найденном параметре размытия r_{опт}, а также апостериорные оценки матриц шума и текстуры. Подробное описание метода см. в [120].

Для иллюстрации описанного метода мы намеренно взяли изображения, полученные самими авторами метода, и вычислили разность между исходным и восстановленным изображениями. На рис. 6.40 приведены эти изображения. Поскольку представленные изображения взяты из Интернета, то искажения, обусловленные процессом печати на бумаге, в них отсутствуют. Рассмотрение приведенных изображений приводит к двум выводам:

- во-первых, описанный метод действительно является эффективным в отношении подавления шума;
- □ во-вторых, как и следовало ожидать, между незашумленным оригиналом и восстановленным после обработки изображением имеются существенные расхождения, которые представлены на рис. 6.40, г. При внимательном рассмотрении рис. 6.40, а и в заметно несоответствие текстуры, например, на каске рабочего, а также в ряде других мест. Однако это не бросается в глаза. Причины этого были рассмотрены в разд. 3.12.

Метод, базирующийся на регенерации контурных, градиентных и фактурных компонентов

В этом методе [39, 40] обработка изображения включает в себя три этапа.

- 1. На первом этапе выполняется анализ корректируемого изображения, в результате которого производится обнаружение контурных, градиентных и фактурных компонентов и оценивается среднеквадратичное значение шума (распределение энергии шума по спектру частот обычно известно).
- 2. На втором этапе осуществляется регенерация контурных, градиентных и фактурных компонентов.
- 3. На третьем этапе искаженные контурные, градиентные и фактурные компоненты заменяются их регенерированными версиями.

Рассмотрим, как выполняются обнаружение и регенерация контурных компонентов изображений.

Выделение контурных компонентов может быть реализовано одним из методов, описанных в *разд. 9.4*. Вместе с тем для уменьшения количества разрывов в контурах, а также количества точек и штрихов, обусловленных шумом, рекомендуется

следующая процедура. Вначале изображение подвергается свертке с импульсной характеристикой типа

$$h(k,n) = \begin{vmatrix} 0 & -b & 0 \\ -b & 1+4b & -b \\ 0 & -b & 0 \end{vmatrix}$$
 или
$$\begin{vmatrix} -b & -b & -b \\ -b & 1+8b & -b \\ -b & -b & -b \end{vmatrix},$$

где параметр b подбирается экспериментально. При этом на изображении оказываются подчеркнутыми световые границы. Затем осуществляется анизотропная адаптивная фильтрация полученного препарата в целях ослабления шума путем его свертки с импульсной характеристикой, которая представляет собой протяженное окно шириной в один пиксел (см. разд. 5.8). Импульсная характеристика все время должна быть ориентирована вдоль контура, причем на участках изгиба контура форма окна должна адаптироваться к форме контура и тоже изгибаться. Благодаря этой фильтрации ослабляется шум. Протяженность окна (импульсной характеристики) определяется экспериментально. Чем более протяженным будет окно, тем в большей степени будут ослаблены флуктуации интенсивности, в том числе пикселов, расположенных вдоль контуров. Однако с увеличением протяженности импульсной характеристики резко возрастает количество вычислительных операций, требующихся для реализации этого алгоритма. Заключительным этапом регенерации контурных компонентов является выделение контуров методом Канни. При этом необходимо экспериментально определить, какие самые малые перепады яркости следует учитывать при выделении контуров, т. е. экспериментально определить пороги.

Регенерация градиентных компонентов изображения заключается в низкочастотной фильтрации его участков, ограниченных контурными компонентами, благодаря чему происходит дальнейшее ослабление шума. Импульсная характеристика низкочастотного фильтра должна адаптироваться к границам контурных компонентов, как это показано на рис. 6.41. Для этой цели хорошо также использовать билатеральную фильтрацию.







Рис. 6.41. К пояснению адаптации импульсной характеристики

Регенерированные градиентные компоненты в дальнейшем замещают в обрабатываемом изображении градиентные компоненты, которые искажены шумом.

Заметим, что полученное таким образом обрабатываемое изображение содержит сильно ослабленные фактурные компоненты, которые необходимо после их регенерации восстановить.

На заключительном этапе регенерируют фактурные компоненты изображения. Так как зрительная система при идентификации фактур реагирует на них так же, как на шумовые рисунки, не различая конкретных реализаций, то можно *зашумленные* реализации фактурных компонентов (шумовые рисунки) заменить шумовыми рисунками, спектральные интенсивности и плотности вероятности первого порядка которых были бы такими же, как у *незашумленных* фактурных компонентов. Для этого необходимо определить спектральные интенсивности и плотности и плотности вероятности вероятности первого порядка незашумленных фактурных компонентов путем вычитания из зашумленного изображения регенерированных градиентных компонентов в областях, занятых градиентными компонентами. Естественно, что выделенные таким образом фактурные компоненты также искажены шумом. Обозначая спектральные интенсивности зашумленных фактурных компонентов через $S_{\phi ш n}(\omega_x, \omega_y)$, где ω_x и ω_y — круговые пространственные частоты, n — номер фактурного компонента, и исходя из известной спектральной интенсивности шума $S_{\rm ш}(\omega_x, \omega_y)$, находим

$$S_{\phi n}\left(\omega_{x},\omega_{y}\right)=S_{\phi \amalg n}\left(\omega_{x},\omega_{y}\right)-S_{\amalg}\left(\omega_{x},\omega_{y}\right).$$

В качестве регенерированных фактурных компонентов используют шумовые реализации с найденными спектральными интенсивностями. Что касается плотностей вероятности первого порядка, то, как правило, их можно принять гауссовыми.

Для иллюстрации работы описанного метода фильтрации зашумленных изображений был выполнен следующий эксперимент. В качестве исходного изображения было использовано изображение, показанное на рис. 6.42, *а*. На это изображение был наложен аддитивный квазибелый гауссов шум. Отношение перепада сигнала от черного до белого к среднеквадратичному значению шума составляло 5. На рис. 6.42, *б* показано это изображение с наложенным на него шумом.

Далее была проведена описанная выше обработка. Параметры обработки были следующими.

- □ При фильтрации контурных компонентов использовалась протяженная импульсная характеристика размером 1×5 пикселов. При этом к контурным компонентам относились компоненты, образуемые перепадами интенсивности в 16 и более квантовых уровней.
- □ При фильтрации градиентных компонентов применялась прямоугольная импульсная характеристика размером 3×3 пиксела, при этом фильтрация осуществлялась последовательно два раза.
- 🗖 Фактурный компонент был получен описанным выше способом.





а

б

Рис. 6.42. Здесь: а — исходное изображение; б — зашумленное изображение





Рис. 6.43. Сравнение фильтрации, выполненной двумя различными методами: *а* — фильтрация методом, базирующимся на регенерации контурных, градиентных и фактурных компонентов; *б* — адаптивная фильтрация На рис. 6.43, *а* показано изображение, полученное в результате фильтрации зашумленного изображения описанным методом.

Сопоставление рис. 6.43, a и 6.42, δ показывает, что при использовании описанной методики имеет место сильное ослабление гауссова шума, но при этом резкость границ на изображении сохраняется, в то время как использование адаптивной фильтрации (рис. 6.43, δ) при сравнительно небольшом ослаблении шума приводит к заметному размытию световых границ. Кроме того, при адаптивной фильтрации структура остаточного шума становится более крупной, вследствие того, что в его спектре ослабляются высокочастотные компоненты, что, как известно, увеличивает при прочих равных условиях его заметность.

Обратим внимание, что представление изображений в виде композиции контурных, градиентных и фактурных компонентов может быть также эффективно использовано в целях их сжатия, поскольку оно позволяет ряд данных представлять с меньшей точностью без заметного снижения качества их воспроизведения.

6.17. Нелинейные преобразования изображений в области, ограниченной произвольным контуром

При нелинейной обработке изображений в ограниченной области, определенной пользователем, применяют, так называемую, *маску*. Маска, наложенная на обрабатываемое изображение, защищает те его части, которые должны остаться неизменными, и делает доступными те его части, которые должны быть подвергнуты обработке. Маска является силуэтным изображением области, доступной для фильтрации. На рис. 6.44, *а* приведено изображение, на котором контуром показана область, подлежащая обработке. На этом же рисунке в центре (рис. 6.44, *б*) показана маска, соответствующая этой области, а справа (рис. 6.44, *в*) исходное изображение с наложенной на него маской. Для большей наглядности область непрозрачности маски сделана полупрозрачной для того, чтобы лучше было видно ее положение на рисунке.



Рис. 6.44. Нелинейные преобразования изображений в области, ограниченной произвольным контуром: *a* — исходное изображение; *б* — маска; *в* — маска, наложенная на изображение

В простейшем случае точечного преобразования, когда изображение сканируется "окном" размером в один пиксел, все отсчеты, попавшие в область прозрачности маски, подвергаются обработке, в то время как отсчеты изображения, оказавшиеся в области непрозрачности, просто копируются из массива исходного изображения в массив обработанного изображения. Примером такой обработки может служить изменение средней яркости изображения в выделенной области, изменение контраста, цветового тона и т. д.

Иначе обстоит дело, когда размеры сканирующего окна больше, чем один пиксел, например, при ранговой фильтрации. Здесь возможны следующие три случая:

- 🗖 все отсчеты сканирующего окна оказываются в области прозрачности маски;
- □ все отсчеты сканирующего окна оказываются в области, где маска непрозрачна;
- часть отсчетов сканирующего окна оказывается в области, где маска прозрачна, а часть отсчетов сканирующего окна — в области, где маска не прозрачна.

В первом случае определение значений отсчетов (интенсивности пикселов), попавших в окно, не составляет труда и выполняется так, как если бы обрабатывалось все изображение.

Во втором случае значения отсчетов обработанного изображения просто копируются из исходного изображения, поскольку по определению эта область изображения защищена маской от каких-либо изменений.

В третьем случае при определении значений отсчетов, попавших в сканирующее окно, следует учитывать только те отсчеты, которые оказались в области прозрачности маски.

6.18. Методы нелинейной обработки изображений в графических редакторах

Современные графические редакторы, такие как Photoshop, Corel PHOTO-PAINT и им подобные, ориентированы, главным образом, на применение в настольных издательских системах. В соответствии с этим применяемые в них методы обработки предназначены для подготовки изображений к печати. Эти редакторы применяются также для создания изображений, предназначенных для использования в мультимедийных приложениях, Интернете, а также в ряде других областей.

Основным видом нелинейной обработки, который представлен инструментарием этих редакторов, является *тоновая коррекция изображений*. Посредством тоновой коррекции изображений могут быть решены два вида задач. Первый вид связан с коррекцией нелинейных искажений изображений, которые обусловлены технологическими процессами их изготовления. С этим случаем мы встречаемся, когда требуется воспроизводить изображения, максимально приближенные к оригиналу. Примером этого могут служить воспроизведение фотометрически точных копий документов, изображений, иллюстрирующих результаты некоторых научных исследований и т. п. Другой вид задач связан с согласованием динамического диапазона яркостей в изображениях, подготавливаемых для печати с динамическим диапазоном, который может быть обеспечен самим технологическим процессом печати. При решении задач второго вида, обычно динамический диапазон исходного изображения оказывается существенно больше того, который может быть воспроизведен. Кроме того, при подготовке изображений для печати, мультимедийных приложений, презентаций и т. д. стремятся "улучшить" редактируемое изображение в соответствии с творческими задачами автора.

Ограничиваясь рассмотрением нелинейной обработки черно-белых полутоновых изображений, рассмотрим технологию этого процесса.

В том случае, если целью обработки является коррекция нелинейных искажений, обусловленных технологическими процессами изготовления изображений, необходимо предусмотреть, чтобы вместе с корректируемым изображением при его сканировании был бы отсканирован градационный клин с равномерным шагом изменения яркости, который используется для измерения возникающих искажений. На рис. 6.45 приведено такое изображение с впечатанным в него градационным клином.



Рис. 6.45. Изображение с аппликированным градационным клином

Измеряя посредством инструмента **Eyedropper** (Пипетка) значения яркостей на ступеньках градационного клина во введенном изображении, не трудно определить амплитудную характеристику сканера с тем, чтобы в дальнейшем ее учесть и скорректировать. Далее после коррекции нелинейных искажений, вносимых сканером, нужно изображение отпечатать. Отпечатанное изображение снова сканируется и снова посредством инструмента **Eyedropper** (Пипетка) измеряются значения яркостей на ступеньках клина, а затем по данным этих измерений определяется амплитудная (световая) характеристика устройства вывода изображения на печать. На основании найденных амплитудных характеристик устройств ввода и вывода изображений рассчитывается результирующая корректирующая характеристика. Коррекция изображений осуществляется посредством инструмента **Curves** (Кривые) путем соответствующей настройки кривой в окне.

В том случае, если целью обработки является согласование динамического диапазона яркостей в изображении, подготавливаемом для печати с динамическим диапазоном, который может быть обеспечен самим технологическим процессом печати, применяется другой метод нелинейной обработки. Для этого вначале производится анализ изображения на предмет распределения значений яркостей по динамическому диапазону. Этот анализ выполняется с использованием инструментов графического редактора **Histogram** (Гистограмма) и **Eyedropper** (Пипетка). Используя инструмент Histogram, легко определить диапазон яркостей изображения, общую тональность изображения, а также выяснить, имеется ли достаточный запас для коррекции изображения, т. е. не приведет ли коррекция к такому перераспределению используемых уровней квантования, при котором на изображении появятся ложные контуры. Затем посредством инструмента Eyedropper (Пипетка) определяются области высокой яркости и наиболее темные области изображения, которые должны воспроизводиться с сохранением подробностей, а также области средних тонов, детали которых не должны быть утеряны. Используя полученную информацию, с помощью инструментов Levels (Уровни) и Curves (Кривые) устанавливают уровни светлого и темного, а также подбирают форму кривой (корректирующей амплитудной характеристики) инструмента Curves. Принцип выбора корректирующей кривой заключается в том, что на интервалах динамического диапазона, содержащих детали, которые не должны быть потеряны, крутизна корректирующей кривой должна быть увеличена за счет ее уменьшения на участках динамического диапазона, не содержащих важных деталей. Помимо перечисленных инструментов коррекции графический редактор Photoshop включает в себя такой инструмент, как Brightness/Contrast (Яркость/Контраст), позволяющий раздельно регулировать яркость и контраст обрабатываемых изображений, а также инструмент Auto Levels (Автоуровни), который автоматически выполняет описанную выше операцию установки уровней светлого и темного. Аналогичные инструменты с несколько отличающимися названиями имеются также и в графическом редакторе Corel PHOTO-PAINT. Кроме того, в графическом редакторе Corel PHOTO-PAINT имеется инструмент, посредством которого можно выделять на изображении контуры, создавая контурный препарат. Более подробные сведения по рассмотренным вопросам нелинейной обработки черно-белых полутоновых цифровых изображений для печати можно найти в книге [2].
Глава 7



Геометрические преобразования изображений

7.1. Геометрические искажения изображений

Причиной геометрических искажений изображений являются как ошибки, допущенные при съемке объектов, так и несовершенство используемой при этом аппаратуры.

Типичными искажениями первого типа, которые приходится в дальнейшем корректировать, являются: искажения, обусловленные неправильной ориентацией камеры относительно снимаемого объекта, например, поворот и т. д. На рис. 7.1 в качестве примера приведено изображение с сильными геометрическими искажениями, которые обусловлены неправильным расположением камеры относительно снимаемого объекта, в данном случае картины.



Рис. 7.1. Изображение с сильными геометрическими искажениями



Рис. 7.2. Геометрические искажения, обусловленные дисторсией: *а* — подушкообразные искажения; *б* — бочкообразные искажения

Типичным видом геометрических искажений, обусловленных несовершенством оптики камеры, является дисторсия. Дисторсия проявляется в нарушении геометрического подобия между предметом и его изображением. В результате дисторсии изображение прямоугольной сетки приобретает бочкообразную (отрицательная дисторсия) или подушкообразную (положительная дисторсия) форму, как это показано на рис. 7.2, *а* и *б*. В сложных объективах характер дисторсии может иметь смешанный, более сложный вид. Дисторсия обусловлена изменением увеличения объектива при изменении угла, под которым объектив видит снимаемый предмет. Наиболее сильно дисторсия проявляется в широкоугольных объективах при съемке близко расположенных объектов.

Имеются задачи, в которых к величине геометрических искажений изображения предъявляются жесткие требования. К этим задачам относится обработка изображений карт и чертежей. Обычно дисторсия составляет значительную часть всех геометрических искажений изображения.

Дисторсия также имеет место при сканировании изображений планшетным сканером. При проецировании изображения на ПЗС-линейку наличие дисторсии проявляется в искажении положения вертикальных линий сетки. Неискаженность горизонтальных линий сетки обеспечивается равномерностью механического привода с шаговым двигателем.

В ряде случаев геометрические искажения сознательно вносятся в изображения с целью создания спецэффектов, переходов между смежными клипами, при реализации варпинга и морфинга и т. д.

7.2. Аффинные преобразования изображений

Простейшими и вместе с тем наиболее часто применяемыми геометрическими преобразованиями являются: сдвиг изображения, поворот изображения относительно начала координат и изменение его масштаба. Рассмотрим эти преобразования на примере ахроматических изображений, поскольку аналогичные преобразования цветных изображений выполняются таким же образом.

Сдвиг изображения

При сдвиге изображения к координатам x и y каждой точки исходного изображения $L_{c}(x, y)$ прибавляются величины, на которые оно должно быть сдвинуто, т. е. Δx и Δy . После сдвига распределение яркости в сдвинутом изображении будет описываться следующим образом: $L_{c}(x + \Delta x, y + \Delta y)$.

В случае цифрового ахроматического изображения распределение яркости в нем после сдвига может быть записано в виде $L_c(k + k_0, n + n_0)$. Здесь k и n — номера строк и столбцов массива изображения, а k_0 и n_0 — величины сдвигов по строкам и по столбцам. Обычно операция сдвига цифрового изображения осуществляется путем переадресации. В силу своей простоты эта операция выполняется наиболее быстро.

Поворот изображения относительно начала координат

При повороте изображения для каждой его точки вычисляются новые координаты *x* и *y* в соответствии с выражениями:

$$x = x' \cos \varphi - y' \sin \varphi,$$

$$y = x' \sin \varphi + y' \cos \varphi,$$

где x' и y' — старые координаты точки изображения, φ — угол поворота относительно начала координат. В случае цифрового изображения вычисляются новые номера строки и столбца k и n, определяющие положение каждого пиксела на изображении по формулам:

$$k = n' \sin \varphi + k' \cos \varphi,$$

$$n = n' \cos \varphi - k' \sin \varphi,$$

где k' и n' — прежние номера строки и столбца, которыми задано положение пиксела в исходном изображении. В отличие от сдвига поворот изображения относительно начала координат связан со значительными вычислительными затратами и, как следствие, занимает заметное время. Кроме того, при повороте изображения его отсчеты (пикселы) перемещаются на различную величину на плоскости изображения в зависимости от их расстояния от начала координат. Чем дальше отстоит отсчет от начала координат, вокруг которого делается поворот, тем на большее расстояние он перемещается. В результате этого сетка отсчетов после поворота уже не будет совпадать с сеткой отсчетов исходного изображения, поскольку направления строк и столбцов будут повернуты на угол ϕ относительно строк и столбцов в исходном изображении. Для приведения повернутого изображения к исходной сетке отсчетов, с которой работает компьютер, необходимо произвести его передискретизацию. Эту операцию можно выполнить без потерь, если применить метод, описанный нами в *главе 4*, однако на это потребуется дополнительное время. В используемых в настоящее время графических редакторах идут по более простому пути, осуществляя передискретизацию с использованием интерполяции первого порядка или бикубической, в связи с этим в изображения вносятся характерные искажения.

Для того чтобы осуществить поворот относительно произвольной точки изображения, достаточно путем сдвига изображения поместить эту точку в начало координат, затем выполнить поворот и, наконец, путем сдвига изображения в обратном направлении вернуть эту точку в исходное положение.

Изменение масштаба изображения

При изменении масштаба новые координаты точки в аналоговом изображении после изменения его масштаба определяются следующими выражениями: x = ax', y = by', где x' и y' координаты точки в исходном изображении, a и b коэффициенты изменения масштаба (увеличения или уменьшения) в направлении соответствующих координат. Значения а и b могут быть как одинаковыми при изменении масштаба изображения без нарушения пропорций, так и различаться между собой, обеспечивая тем самым растяжение (сжатие) изображения в одном из направлений. В случае изменения масштаба цифрового изображения новые координаты отсчетов (номера строк и столбцов) k, n находятся следующим образом: k = bk', n = an', где k' и n' — номера строк и столбцов, определяющих положение отсчета (пиксела) в исходном изображении. Изменение масштаба цифрового изображения приводит к тому, что и при увеличении, и при уменьшении масштаба в нецелое число раз сетка отсчетов преобразованного изображения не совпадает с сеткой отсчетов исходного изображения, что приводит к необходимости передискретизации преобразованного изображения для устранения этого несоответствия. Необходимость передискретизации возникает и при изменении масштаба в целое число раз, однако в этом случае алгоритм передискретизации оказывается проще. Поскольку изменение масштаба связано с операциями умножения и с передискретизацией изображения, то это приводит к неизбежному увеличению временных затрат.

Рассмотренные геометрические преобразования изображения удобно выполнять, используя для этой цели матричный аппарат путем перемножения матрицы преобразования (матрицы сдвига, поворота, изменения масштаба) на матрицу-строку, задающую координаты каждого отсчета изображения, дополненную единицей. Так в случае сдвига изображения имеем

$$\begin{vmatrix} y' & x' & 1 \end{vmatrix} \times \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ y_0 & x_0 & 1 \end{vmatrix} = \begin{vmatrix} (y' + y_0) & (x' + x_0) & 1 \end{vmatrix}.$$

MATDUIIA- MATDUIIA MATDUIIA-

матрица-	матрица	матрица-
строка	сдвига	строка

В случае поворота изображения координаты преобразованного изображения определяются таким образом:

$$\begin{vmatrix} y' & x' & 1 \end{vmatrix} \times \begin{vmatrix} \cos \varphi & -\sin \varphi & 0 \\ \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} (x' \sin \varphi + y' \cos \varphi) & (x' \cos \varphi - y' \sin \varphi) & 1 \end{vmatrix}.$$

И, наконец, новые координаты отсчетов изображения после изменения его масштаба находятся следующим образом:

$$|y' x' 1| \times \begin{vmatrix} b & 0 & 0 \\ 0 & a & 0 \\ 0 & 0 & 1 \end{vmatrix} = |by' ax' 1|.$$

Все три матрицы преобразования могут быть сведены к одной, которая включает в себя параметры, определяющие величину сдвига, поворота и изменения масштаба.

$$\begin{vmatrix} y' & x' & 1 \end{vmatrix} \times \begin{vmatrix} b\cos\varphi & -b\sin\varphi & 0 \\ a\sin\varphi & a\cos\varphi & 0 \\ y_0 & x_0 & 1 \end{vmatrix} = \\ = \begin{vmatrix} (ax'\sin\varphi + by'\cos\varphi + y_0) & (ax'\cos\varphi - by'\sin\varphi + x_0) & 1 \end{vmatrix}.$$

Аналогичным образом может быть записано выражение для преобразования цифрового изображения

$$\begin{vmatrix} k' & n' & 1 \end{vmatrix} \times \begin{vmatrix} b\cos\varphi & -b\sin\varphi & 0 \\ a\sin\varphi & a\cos\varphi & 0 \\ k_0 & n_0 & 1 \end{vmatrix} = \\ = \begin{vmatrix} (an'\sin\varphi + bk'\cos\varphi + k_0) & (an'\cos\varphi - bk'\sin\varphi + n_0) & 1 \end{vmatrix}.$$

Простейшим, но не лучшим алгоритмом передискретизации преобразованного изображения является округление значений координат отсчетов преобразованного изображения с заполнением "пустот", которые могут образовываться при увеличении изображения путем применения интерполяции нулевого порядка. Однако если желательно дополнительные искажения, возникающие в результате передискретизации, свести к минимуму, то лучше использовать метод, описанный в *главе 4*.

7.3. Дисторсия и методы ее коррекции

Дисторсия (от лат. *distorsio* — искривление) — погрешность изображения в оптических системах, при которой нарушается геометрическое подобие между объектом и его изображением; одна из аберраций оптических систем. Заключается в геометрических искажениях изображения по сравнению с объектом из-за изменения линейного увеличения объектива от центра к краю поля. Характерный пример искажений, возникающих в оптической системе, обладающей дисторсией, демонстрирует приведенное на рис. 7.3 изображение квадрата.



Рис. 7.3. Примеры геометрических искажений, обусловленных дисторсией: *а* — подушкообразные искажения; *б* — отсутствие геометрических искажений; *в* — бочкообразные искажения

На этом рисунке слева показана подушкообразная (Pincushion Distortion), или положительная дисторсия, справа — бочкообразная (Barrel Distortion), или отрицательная. В сложных объективах характер дисторсии может иметь смешанный, более сложный вид. Дисторсия не нарушает резкости изображения.

Количественно дисторсия оптической системы характеризуется так называемой относительной дисторсией $v = \frac{\beta}{\beta_0 - 1}$, где β_0 — линейное увеличение идеальной системы без дисторсии, а β — увеличение, имеющее место в действительности. Относительная дисторсия выражается в процентах (%).

Как правило, дисторсия возникает при съемке широкоугольными и телеобъективами. Особенно велика дисторсия (бочкообразная) у сверхширокоугольных объективов типа "фиш-ай" ("рыбий глаз"), поле зрения которых составляет около 180°. Дисторсия преувеличивает ощущение глубины.

Дисторсия наиболее заметна при съемке архитектурных объектов. Другой случай, когда с искажениями такого рода мириться нельзя, — панорамные изображения, получаемые из нескольких последовательных снимков.

Особенно вредна дисторсия в фотографических объективах, применяемых в геодезии или фотограмметрии. Для хороших фотообъективов относительная дисторсия близка к 0,5%. В объективах, применяемых при аэрофотосъемке, дисторсия равна ~0,01%. Величина дисторсии может быть существенно уменьшена путем применения в объективах двух и большего количества линз, а также посредством применения асферических линз. В отдельных случаях (симметричные фотообъективы или зрительные трубы) дисторсию можно практически свести к нулю. Одним из методов компенсации дисторсии является цифровая обработка изображений в компьютере. Однако разработка алгоритма, посредством которого можно было бы полностью компенсировать геометрические искажения, обусловленные дисторсией, оказалась довольно сложной задачей. Кроме того, одной из проблем, с которыми здесь приходится встречаться, является измерение величин искажений, вызванных дисторсией, по всему полю изображения.

Частично ослабить геометрические искажения изображения, обусловленные дисторсией, можно сравнительно несложным путем, например, используя для этой цели опцию фильтра **Distort | Spherize** (Искажение | Сферизация) графического редактора Photoshop. Однако используя такой простой прием, не удается скомпенсировать искажения по всему полю.

Недавно появились сообщения о том, что южнокорейскими учеными разработаны широкоугольные объективы, в которых практически отсутствует дисторсия. В основе решения, предложенного южнокорейскими учеными, лежит использование зеркально-линзовой оптической системы, ключевым элементом которой является коническая зеркальная поверхность. В этой системе свет проходит сквозь куполообразную линзу, сделанную из поликарбоната и представляющую собой внешний корпус системы, затем отражается от конической поверхности и попадает на конечный оптический элемент системы, которым является преломляющая линза, направляющая световой поток на светочувствительную матрицу видеокамеры.

7.4. Перспективные искажения изображений и их коррекция

При съемке сцены на светочувствительную матрицу камеры проецируются объекты, составляющие снимаемую сцену. В силу закона перспективы близко расположенные объекты создают изображения больших размеров, чем объекты удаленные, благодаря чему у зрителя создается ощущение глубины наблюдаемой сцены. Рисунок 7.4 демонстрирует это. По этой причине железнодорожные рельсы, уходящие вдаль, на изображении кажутся сходящимися на горизонте. Из сказанного следует, что проекции объектов, т. е. их изображения передаются в искаженном виде.

Перспективные искажения носят следующий характер.

- □ В изображении не сохраняется соотношение длин и площадей.
- □ В изображении параллельные прямые изображаются сходящимися в одной точке прямыми. Рисунок 7.5 поясняет сказанное.
- □ Прямая на объекте проецируется в прямую на изображении, но с неравномерной шкалой. Например, движение точки с постоянной скоростью по прямой на объекте отображается ее движением на изображении тоже по прямой, но с неравномерной скоростью. Чем больше точка удаляется, тем меньше становится видимая скорость ее движения. Мы не приводим здесь формулы, устанавливающие связь между координатами точки на объекте и на изображении в силу их громоздкости, при необходимости читатель найдет их в [75, 82].



Рис. 7.4. Пример перспективных искажений при фотосъемке



Рис. 7.5. Рисунок, демонстрирующий характерные особенности перспективных искажений



а

б

Рис. 7.6. К пояснению метода коррекции геометрических искажений

Эти искажения наиболее сильно проявляются при съемке короткофокусными объективами с близкого расстояния.

В ряде случаев возникает необходимость в коррекции перспективных искажений. Это не всегда возможно, но в некоторых случаях это удается сделать довольно просто. Например, при неправильной съемке картины, как это показано на рис. 7.1, для устранения перспективных искажений необходимо осуществить коррекцию трапецеидальных искажений, а затем восстановить правильное соотношение высоты и ширины изображения. Эти искажения можно скорректировать, используя инструмент **Crop** (Рамка) графического редактора Photoshop. С этой целью необходимо инструментом **Crop** выделить все корректируемое изображение, затем в появившейся верхней панели установить флажок **Perspective** (Перспектива). После этого установить маркеры инструмента **Crop** по углам искаженного изображения, как это показано на рис. 7.6, a, а затем нажать клавишу <Enter>. После этого необходимо установить правильное отношение высоты изображения к его ширине, которое должно быть известно заранее, используя для этой цели опцию **Image | Image Size** (Изображение | Размер изображения) при снятом флажке **Constrain Proportions** (Сохранять пропорции). Результат коррекции показан на рис. 7.6, d.

К сожалению, такой простой способ коррекции не всегда применим.

7.5. Коррекция геометрических искажений изображений в общем случае

Рассмотрим теперь общий случай коррекции геометрических искажений изображений. Примером таких искажений могут служить искажения, возникающие при аэросъемке и космической съемке поверхности Земли. В этом случае результирующие геометрические искажения складываются из искажений, вносимых оптической системой устройства, производящего съемку, перспективных искажений и искажений, обусловленных кривизной земной поверхности. Величина этих искажений по полю изображения не постоянна и, как правило, неизвестна.

Переходя к рассмотрению метода коррекции геометрических искажений, обратимся к рис. 7.7, на котором показана проекция изображения на светочувствительную поверхность датчика сигнала изображения.

На этом рисунке незакрашенными квадратами обозначены точки взятия отсчетов при дискретизации изображения в процессе его оцифровки. В том случае, если геометрические искажения отсутствуют, значения яркости изображения в точках взятия отсчетов будут равны L_1 , L_2 , L_3 и т. д. Если же геометрические искажения имеют место, изображение деформируется, и в точки взятия отсчетов 1, 2, 3 и т. д. сдвигаются точки, показанные на рисунке черными квадратами, значения яркости в которых составляют L_1' , L_2' , L_3' и т. д. Если такое изображение воспроизвести, то оно будет иметь вид, показанный на рис. 7.8. При этом в точках взятия отсчетов мы будем иметь неверные значения яркости, соответствующие точкам, которые

на проекции изображения на светочувствительную поверхность датчика сигнала изображения показаны залитыми квадратами, а правильные значения яркости окажутся смещенными (рис. 7.8).



Рис. 7.7. К пояснению метода коррекции геометрических искажений в общем случае



Рис. 7.8. К пояснению метода коррекции геометрических искажений в общем случае

Задачей коррекции является определение правильных или близких к ним значений яркости в точках взятия отсчетов неискаженного изображения, т. е. значений яркости в незакрашенных квадратах на рис. 7.7. Для решения этой задачи необходимо располагать функциями, которые устанавливают связь между координатами одно-

именных точек в искаженном x_i' , y_i' и неискаженном x_i и y_i изображениях. Несколько позже мы рассмотрим, как определяются эти функции, а сейчас будем считать, что они монотонные и нам известны:

$$x_i' = \Phi_x \left(x_i, y_i \right), \tag{7.1}$$

$$y_i' = \Phi_v \left(x_i, y_i \right), \tag{7.2}$$

где i = 1, 2, 3, ... Будем также считать, что нам известны значения яркости в точках некорректированного изображения L_1 , L_2 , L_3 , ..., поскольку они оказались в точках дискретных отсчетов (см. рис. 7.8). Для того чтобы скорректировать эти значения, необходимо выполнить следующее.

- 1. Найти координаты точек, которые вследствие геометрических искажений сползли из точек отсчетов в новые места. На рис. 7.8 эти точки показаны неза-крашенными квадратами.
- 2. Поскольку в дискретном искаженном изображении правильные значения яркостей L₁', L₂', L₃', ..., которые следует поместить в точки дискретных отсчетов скорректированного изображения, неизвестны, то их следует определить по известным значениям яркости ближайших соседей этих точек, расположенных в точках дискретных отсчетов искаженного изображения, используя для этой цели один из методов интерполяции. Эти точки в виде не залитых квадратов показаны на рис. 7.8.
- 3. Заменить неверные значения яркости в точках дискретных отсчетов на интерполированные значения.

Поясним сказанное на примере коррекции яркости в точке 8 на рис. 7.8.

1. Вначале используя формулы (7.1) и (7.2), находим смещенные координаты точки 8:

$$x'_8 = \Phi_x(x_8, y_8),$$

 $y'_8 = \Phi_y(x_8, y_8).$

- Затем, зная координаты точек 2, 3, 7, 8 в искаженном изображении (координаты точек, обозначенных черными квадратами) и яркости этих точек L₂, L₃, L₇, L₈, определяем, например, методом билинейной интерполяции значение L̂₈.
- 3. Заменяем в искаженном изображении значение L_8 на интерполированное значение \hat{L}_8 . Поступая таким образом, шаг за шагом мы корректируем геометрические искажения.

Чаще всего для целей коррекции используют билинейную интерполяцию, поскольку ее применение требует меньших вычислительных ресурсов. Однако лучшими видами интерполяции являются интерполяция с использованием функций sinc и функций Бесселя, близкие к ним результаты получаются при использовании бикубической интерполяции.

7.6. Метод определения функций, описывающих геометрические искажения

Переходя к определению функций $\Phi_x(x_i, y_i)$, $\Phi_y(x_i, y_i)$, отметим, что, как прави-

ло, не удается подобрать для них достаточно простые выражения, которые позволяли бы описать геометрические искажения во всей плоскости изображения. Поэтому все изображение разбивают на отдельные области и для каждой из них находят свою пару функций, описывающих искажения. С этой целью используют так называемые реперные точки, положение которых заранее известно как на неискаженном, так и на искаженном изображении, и с их помощью разделяют изображение на отдельные области. На рис. 7.9, *а* показаны четыре реперные точки и границы определяемой ими прямоугольной области в неискаженном изображении, а на рис. 7.9, *б* — эти же реперные точки и определяемые ими точки в искаженном изображении.



Рис. 7.9. Расположение реперных точек и определяемых ими границ: *а* — на неискаженном изображении; *б* — на искаженном изображении

Если величины четырехугольных областей выбраны не чрезмерно большими, то процесс деформации изображения внутри них можно описать выражениями

$$x'_{i} = \Phi_{x}(x_{i}, y_{i}) = a_{1}x_{i} + a_{2}y_{i} + a_{3}x_{i}y_{i} + a_{4},$$
(7.3)

$$y'_{i} = \Phi_{y}(x_{i}, y_{i}) = b_{1}x_{i} + b_{2}y_{i} + b_{3}x_{i}y_{i} + b_{4}, \qquad (7.4)$$

где a_1 , a_2 , a_3 , a_4 , b_1 , b_2 , b_3 , b_4 — постоянные коэффициенты, которые определяются путем подстановки в формулы (7.3) и (7.4) значений x_i , y_i , x'_i , y'_i для каждой из реперных точек 1, 1', 2, 2', 3, 3', 4, 4', показанных на рис. 7.9, и совместного решения получившихся уравнений. Таким образом, необходимо иметь множество реперных точек для того, чтобы покрыть ими все изображение. В качестве репер-

ных точек могут быть использованы какие-либо хорошо различимые на снимаемой сцене мелкие детали. Другой способ определения коэффициентов a_1 , a_2 , a_3 , a_4 , b_1 , b_2 , b_3 , b_4 заключается в использовании специальных тестовых изображений с изображением сетки реперных точек.

7.7. Наложение текстуры на полигональную оболочку объекта

Одной из часто встречающихся задач при работе с 3D-изображениями является задача наложения текстур на оболочки трехмерных изображений объектов, представленных в виде полигональных моделей.¹ Обратим внимание на то, что при использовании двумерного дисплея на его экран выводится не само трехмерное изображение, а его двумерная проекция в отличие от действительно трехмерного изображения, которое выводится при использовании специального трехмерного дисплея. В качестве примера на рис. 7.10, *а* показана полигональная оболочка 3Dизображения заготовки глобуса, на которую нужно нанести текстуру, представляющую собой географическую карту, показанную на рис. 7.10, *б*.



Рис. 7.10. Изображения: *а* — полигональной модели заготовки изображения глобуса; *б* — текстуры

Будем считать, что каждая линия на рис. 7.10, *а* сопоставляется соответствующей линии долготы или широты, показанной на рис. 7.10, *б*. При нанесении на оболочку изображения текстуры оно должно быть подвергнуто необходимым геометрическим преобразованиям для того, чтобы в каждый полигон была полностью вписана

¹ О полигональной модели см. в разд. 10.4.

соответствующая часть текстуры, заключенная в прямоугольнике, который образуется при пересечении соответствующих линий широты и долготы. На рис. 7.11, *а* показан заполняемый полигон, а на рис. 7.11, *б* прямоугольная область, содержащая ту часть текстуры, которой должен быть заполнен полигон.



Рис. 7.11. Изображения: *а* — одного из полигонов оболочки; *б* — фрагмента текстуры, которая должна быть вписана в полигон

При заполнении полигона нам необходимо для каждой точки изображения полигона, координаты которой обозначим X_n , Y_n , найти координаты x_T , y_T соответствующей точки в изображении текстуры. При решении этой задачи используем линейную интерполяцию.

Определим текстурные координаты, соответствующие точкам A и Б, которые обозначим как x_{TA} , y_{TA} и x_{TE} , y_{TE} соответственно, путем деления отрезков 1—4 и 2—3 пропорционально Y. Составим очевидные пропорции

$$\frac{x_{\mathrm{TA}} - x_{\mathrm{T1}}}{x_{\mathrm{T4}} - x_{\mathrm{T1}}} = \frac{y_{\mathrm{TA}} - y_{\mathrm{T1}}}{y_{\mathrm{T4}} - y_{\mathrm{T1}}} = \frac{Y_n - Y_1}{Y_4 - Y_1}, \ \frac{x_{\mathrm{T5}} - x_{\mathrm{T2}}}{x_{\mathrm{T3}} - x_{\mathrm{T2}}} = \frac{y_{\mathrm{T5}} - y_{\mathrm{T2}}}{y_{\mathrm{T3}} - y_{\mathrm{T2}}} = \frac{Y_n - Y_2}{Y_3 - Y_2},$$

откуда следует, что

$$x_{\text{TA}} = x_{\text{T1}} + \left(x_{\text{T4}} - x_{\text{T1}}\right) \frac{Y_n - Y_1}{Y_4 - Y_1}, \ y_{\text{TA}} = y_{\text{T1}} + \left(y_{\text{T4}} - y_{\text{T1}}\right) \frac{Y_n - Y_1}{Y_4 - Y_1},$$
$$x_{\text{T5}} = x_{\text{T2}} + \left(x_{\text{T3}} - x_{\text{T2}}\right) \frac{Y_n - Y_2}{Y_3 - Y_2}, \ y_{\text{T5}} = y_{\text{T2}} + \left(y_{\text{T3}} - y_{\text{T2}}\right) \frac{Y_n - Y_2}{Y_3 - Y_2}$$

Для нахождения текстурных координат для точки на изображении полигона с координатами X_n , Y_n разделим отрезок АБ пропорционально X и составим пропорцию

$$\frac{x_{\rm T} - x_{\rm TA}}{x_{\rm TE} - x_{\rm TA}} = \frac{y_{\rm T} - y_{\rm TA}}{y_{\rm TE} - y_{\rm TA}} = \frac{X_n - X_A}{X_E - X_A},$$

откуда имеем окончательно

$$x_{\rm T} = x_{\rm TA} + \left(x_{\rm TB} - x_{\rm TA}\right) \frac{X_n - X_{\rm A}}{X_{\rm B} - X_{\rm A}}, \ y_{\rm T} = y_{\rm TA} + \left(y_{\rm TB} - y_{\rm TA}\right) \frac{X_n - X_{\rm A}}{X_{\rm B} - X_{\rm A}}.$$

Задача решается аналогично и в случае, когда строка пересекает другие ребра полигона.

Обратим внимание на то, что интерполяция координат при наложении текстуры аналогична интерполяции интенсивностей отраженного света при использовании метода Гуро (*см. разд. 10.7*)¹.

Заметим, что поскольку мы используем интерполяцию в описанном методе наложения текстуры, то тем самым вносим погрешность, однако если количество полигонов в 3D-изображении велико и, как следствие, размеры полигонов малы, то эта погрешность мала, и она тем меньше, чем меньше размер полигона.

Одним из недостатков рассмотренного метода наложения текстуры является большое количество вычислительных операций, а также необходимость выполнять деление, которое является медленной операцией.



Рис. 7.12. Триангуляция

Метод, свободный от этих недостатков, заключается в следующем. 3Dизображение представляется в виде треугольных полигонов. Достигается это путем деления полигонов с большим числом вершин на части так, например, как это показано на рис. 7.12, *а*. Изображение текстуры также представляется в виде набора областей треугольной формы, как это показано на рис. 7.12, *б*. Затем посредством аффинного преобразования устанавливают связь между координатами на изображении полигона и изображении текстуры

$$\begin{aligned} x_{\mathrm{T}} &= a X_n + b Y_n + c , \\ y_{\mathrm{T}} &= d X_n + e Y_n + f , \end{aligned}$$

¹ Если в алгоритм заполнения встроить интерполяцию по методу Гуро или Фонга, то можно получить более реалистичное текстурированное изображение, в котором будет учитываться отражение света.

где коэффициенты a, b, c, d, e и f остаются неизменными в течение всего цикла наложения текстуры на данный полигон. Значения этих коэффициентов определяются путем совместного решения шести уравнений, устанавливающих связь между координатами вершин треугольных полигонов и треугольных областей на изображении текстуры. Например, для треугольника на рис. 7.12 это будет

$$\begin{split} x_{\mathrm{T1}} &= a X_1 + b Y_1 + c \; , \\ x_{\mathrm{T2}} &= a X_2 + b Y_2 + c \; , \\ x_{\mathrm{T3}} &= a X_3 + b Y_3 + c \; , \\ y_{\mathrm{T1}} &= d X_1 + e Y_1 + f \; , \\ y_{\mathrm{T2}} &= d X_2 + e Y_2 + f \; , \\ y_{\mathrm{T3}} &= d X_3 + e Y_3 + f \; . \end{split}$$

Отметим, что использовать этот метод следует только при большом количестве граней вследствие значительной методической погрешности. Достоинством этого метода наложения текстуры является простота вычислительных операций. Более подробные сведения по этому вопросу читатель найдет в [82]. Для треугольников часто используют алгоритмы, оптимизированные по быстродействию.

7.8. Увеличение изображений

При увеличении изображений возникает необходимость обеспечивать нужную интерполяцию программным путем. Делается это путем введения в изображения дополнительных строк и пикселов (отсчетов) в строках. Так, например, при увеличении линейных размеров изображения в два раза необходимо в него ввести дополнительные строки, разместив их между уже имеющимися строками, а также ввести дополнительные пикселы, поместив их между уже существующими на изображении пикселами. При этом необходимо определить величины яркости вставляемых в изображение пикселов. Эти значения находятся путем применения того или иного вида интерполяции. На практике чаще всего применяют интерполяцию нулевого порядка, билинейную интерполяцию или бикубическую интерполяцию.

Интерполяция нулевого порядка

При увеличении изображения в него добавляются новые пикселы, яркость которых необходимо определить. Рисунок 7.13 поясняет сказанное. Слева на этом рисунке показано исходное изображение, а справа — увеличенное в два раза, причем вновь вставленные пикселы выделены серым цветом.

В простейшем случае необходимые значения яркости находят, применяя интерполяцию нулевого порядка (Nearest Neighbor), когда значения яркости вновь вставляемых пикселов делают равными значениям яркости пикселов, к которым они примыкают. В нашем примере это пикселы, расположенные слева, слева внизу и внизу от исходных пикселов.



Рис. 7.13. К пояснению интерполяции нулевого порядка

Однако такой способ определения яркости вставляемых в изображения пикселов не является наилучшим, т. к. приводит к повышению заметности растровой структуры изображений, которая при больших увеличениях существенно снижает их качество, а также делает наклонные границы на изображениях зазубренными. На рис. 7.16, *а* в качестве примера приведено изображение, полученное в результате четырехкратного увеличения с использованием интерполяции нулевого порядка.

Обращаясь к рассмотрению механизма этого явления, будем считать, что исходное изображение перед его пространственной дискретизацией удовлетворяло теореме отсчетов и, следовательно, после дискретизации оно свободно от шума, обусловленного наложением спектров. Будем также считать, что интервал пространственной дискретизации изображения Δ одинаков в обоих направлениях. Кроме того, для простоты рассмотрения примем, что распределение прозрачности в апертуре воспроизводящего устройства постоянно, а сама апертура имеет форму квадрата со стороной Δ . Рассмотрение удобно провести в спектральной области. На рис. 7.14 приведена зависимость спектральной интенсивности дискретизированного изображения $S_{cq}(\omega_x, \omega_y)$ от круговой пространственной частоты ω_x , а также амплитудночастотная характеристика апертуры воспроизводящего устройства с равномерным распределением прозрачности, описывающая эффект фильтрации изображения этой апертурой

$$K(\omega_x, \omega_y) = \frac{\sin(\pi\omega_x/\omega_{\pi})}{(\pi\omega_x/\omega_{\pi})},$$

где $\omega_{\rm d} = 2\pi/\Delta$ — частота пространственной дискретизации.



Рис. 7.14. К пояснению возникновения искажений в процессе преобразования дискретизированного изображения в непрерывное

Из рассмотрения спектра дискретизированного изображения и амплитудночастотной характеристики апертуры воспроизводящего устройства следует, что при его воспроизведении, когда изображение преобразуется из дискретного вида в непрерывный вид, возникают два вида искажений. Во-первых, побочные спектры дискретизированного изображения полностью не отфильтровываются, создавая на экране характерную растровую структуру в виде квадратов с постоянной яркостью. Во-вторых, $K(\omega_x, \omega_y)$ заметно ослабляет верхние пространственные частоты основного спектра, который на рисунке показан штриховкой. Если размер изображения невелик, то растровая структура мало заметна в силу ограничений со стороны зрительной системы. Увеличение же линейных размеров изображения в n раз с использованием интерполяции нулевого порядка приводит к увеличению в *n* раз размеров вновь формируемых пикселов, которые образуются путем объединения смежных вставляемых пикселов, имеющих ту же яркость, что и исходный. Увеличение размера пикселов делает их более заметными для зрителя, ухудшая общий вид изображения. Обращаясь к спектральному представлению изображения, обратим внимание, что в рассматриваемом случае увеличение изображения влечет за собой сужение его пространственного спектра в n раз, а также сжатие по оси ω_r

в это же число раз $K(\omega_x, \omega_y)$ вследствие уменьшения в *n* раз частоты пространственной дискретизации, обусловленной увеличением размеров вновь формируемых пикселов. Другими словами, график, показанный на рис. 7.14, в рассматриваемом случае сжимается в *n* раз по горизонтальной оси, не претерпевая каких-либо других изменений. Из изложенного следует, что увеличение изображения при использовании интерполяции нулевого порядка не приводит к каким-либо дополнительным его искажениям, видимое же снижение качества всецело обусловлено изменившимися условиями наблюдения, при которых ограничения со стороны остроты зрения оказались ослабленными. Этот же результат будет иметь место, если увеличение изображения осуществить оптическим способом. В целях ослабления искажений такого типа были разработаны более совершенные методы интерполяции изображений.

Билинейная интерполяция

Переходя к рассмотрению билинейной интерполяции, обратимся к рис. 7.15, на котором показана прямоугольная решетка, в узлах которой расположены точки дискретных отсчетов яркости исходного изображения. В результате интерполяции нам требуется найти интерполирующую функцию для области изображения, ограниченной отсчетами с номерами 6, 7, 10, 11 с тем, чтобы посредством этой функции определить значения яркости вновь вставляемых пикселов, которые будут располагаться в этой области. На рисунке точки этих отсчетов обозначены черными квадратиками. Реализация билинейной интерполяции состоит в следующем.



Рис. 7.15. Прямоугольная решетка отсчетов

Вначале, используя значения яркостей в точках дискретных отсчетов 6-7 и 10-11, находятся одномерные интерполирующие функции $L_{y1}(x)$ и $L_{y2}(x)$, представляющие собой интерполированные значения яркости на интервалах, заключенных между этими отсчетами.

Затем, используя как опорные значения яркостей, найденных посредством интерполирующих функций $L_{y1}(x)$ и $L_{y2}(x)$, для любого значения x, лежащего в пределах интерполируемой области изображения, находится искомая двумерная интерполирующая функция $\hat{L}_{c}(x, y)$, которая имеет вид

$$\hat{L}_{c}(x,y) = \frac{L_{c}(0,0)}{\Delta\Delta}(\Delta - x)(\Delta - y) + \frac{L_{c}(\Delta,0)}{\Delta\Delta}x(\Delta - y) + \frac{L_{c}(0,\Delta)}{\Delta\Delta}(\Delta - x)y + \frac{L_{c}(\Delta,\Delta)}{\Delta\Delta}xy.$$

Аналогичным образом находятся интерполирующие функции для других областей изображения.

Применение билинейной интерполяции для определения значения яркости вновь вставляемых в изображение пикселов при его увеличении позволяет существенно ослабить зазубренность наклонных границ, однако эти границы на изображении сильно размываются. На рис. 7.16, δ в качестве примера приведено изображение, полученное в результате четырехкратного увеличения с применением билинейной интерполяции.

Бикубическая интерполяция

Переходя к рассмотрению бикубической интерполяции, обратимся к рис. 7.15, на котором показана прямоугольная решетка, в узлах которой расположены точки дискретных отсчетов яркости изображения. В результате интерполяции нам требуется найти интерполирующую функцию для области изображения, ограниченной отсчетами с номерами 6, 7, 10, 11, с тем, чтобы посредством этой функции определить значения яркости вновь вставляемых пикселов. На рисунке точки этих отсчетов обозначены черными квадратиками. Один из способов выполнения бикубической интерполяции состоит в следующем.

Вначале находятся одномерные интерполирующие функции $L_{y1}(x)$, $L_{y2}(x)$, $L_{y3}(x)$, $L_{y4}(x)$ для четырех строк, включающих в себя отсчеты 1, 2, 3, 4 (для первой строки), 5, 6, 7, 8 (для второй строки), 9, 10, 11, 12 (для третьей строки), 13, 14, 15 и 16 (для четвертой строки). Эти функции позволяют найти интерполированные значения яркости соответствующих строк на интервалах 2-3, 6-7, 10-11 и 14-15.

Затем, используя как опорные значения яркостей, найденных посредством интерполирующих функций, для любого значения x, лежащего в пределах интерполируемой области изображения, аналогичным образом находим искомую двумерную интерполирующую функцию $\hat{L}_{c}(x, y)$, которая имеет вид

$$\hat{L}_{c}(x, y) = \sum_{n=0}^{3} \sum_{k=0}^{3} a_{nk} (x/\Delta)^{n} (y/\Delta)^{k} ,$$

где a_{nk} — коэффициенты. Для определения коэффициентов a_{nk} составим следующую систему уравнений:

$$L_{c}(0,0) = a_{00};$$

$$\hat{L}_{c}(1,0) = a_{00} + a_{10} + a_{20} + a_{30};$$

$$\hat{L}_{c}(0,1) = a_{00} + a_{01} + a_{02} + a_{03};$$

$$\hat{L}(1,1) = \sum_{n=0}^{3} \sum_{k=0}^{3} a_{nk};$$

1

i

1

(

$$\frac{\partial L(0,0)}{\partial x} = a_{10};$$

$$\frac{\partial \hat{L}_{c}(1,0)}{\partial x} = a_{10} + 2a_{20} + 3a_{30};$$

$$\frac{\partial \hat{L}_{c}(0,1)}{\partial x} = a_{10} + a_{11} + a_{12} + a_{13};$$

$$\frac{\partial \hat{L}_{c}(0,1)}{\partial x} = \sum_{n=1}^{3} \sum_{k=0}^{3} a_{nk}n;$$

$$\frac{\partial \hat{L}_{c}(0,0)}{\partial y} = a_{01};$$

$$\frac{\partial \hat{L}_{c}(0,0)}{\partial y} = a_{01} + a_{11} + a_{21} + a_{31};$$

$$\frac{\partial \hat{L}_{c}(0,1)}{\partial y} = a_{01} + 2a_{02} + 3a_{03};$$

$$\frac{\partial \hat{L}_{c}(1,1)}{\partial y} = \sum_{n=0}^{3} \sum_{k=1}^{3} a_{nk}k;$$

$$\frac{\partial^{2} \hat{L}_{c}(0,0)}{\partial x \partial y} = a_{11} + 2a_{21} + 3a_{31};$$

$$\frac{\partial^{2} \hat{L}_{c}(0,1)}{\partial x \partial y} = a_{11} + 2a_{12} + 3a_{31};$$

$$\frac{\partial^{2} \hat{L}_{c}(0,1)}{\partial x \partial y} = a_{11} + 2a_{12} + 3a_{13};$$

решение которой дает их значения.

В дискретном случае, а именно он нас интересует, поскольку мы имеем дело с цифровым изображением, значения производных заменяются значениями конечных разностей, для вычисления которых необходимо располагать значениями яркости в точках отсчетов 1—16. Таким образом, при бикубической интерполяции в качестве опорных значений, в отличие от билинейной интерполяции, используется не 4, а 16 опорных точек. Аналогичным образом находятся интерполирующие функции для других областей изображения.

На рис. 7.16, *в* в качестве примера приведено изображение, полученное в результате четырехкратного увеличения и применения бикубической интерполяции. Из рисунка видно, что на световых границах в увеличенном изображении практически отсутствуют ступеньки, характерные для интерполяции нулевого порядка, а само изображение оказывается более четким, чем это было при интерполяции нулевого порядка и при билинейной интерполяции. Следует обратить также внимание на то, что применение бикубической интерполяции при увеличении изображений дает результаты очень близкие к тем, которые получаются при использовании интерполяции посредством функций типа sin(x)/x.





б



Рис. 7.16. Изображения после четырехкратного увеличения с использованием: *а* — интерполяции нулевого порядка; *б* — билинейной интерполяции; *в* — бикубической интерполяции

Интерполяция посредством функций sin x/x

Заметность растровой структуры на изображении можно свести к минимуму, если яркость вставляемых пикселов определять посредством более совершенных методов интерполяции. Действительно, поскольку исходное дискретизируемое изображение согласно принятому допущению удовлетворяет условиям теоремы отсчетов, то, используя формулу (4.5), можно абсолютно точно определять значения яркости изображения в любых точках, в том числе и в точках вновь вставляемых пикселов при его увеличении. В этом случае мы будем иметь дело только с увеличением размера изображения без сопутствующего ему увеличения размеров пикселов, т. к. яркость каждого вновь вставляемого пиксела будет определяться действительной яркостью исходного не дискретизированного изображения в этой точке. На рис. 7.17 показаны спектр дискретизированного изображения для этого случая и амплитудно-частотная характеристика апертуры воспроизводящего устройства.



Рис. 7.17. К пояснению метода уменьшения ошибок интерполяции при увеличении цифровых изображений

Из рисунка следует, что увеличение изображения при использовании интерполяции, определяемой формулой (4.5), приводит только к сужению его пространственного спектра (в данном примере в 3 раза) без какого-либо изменения амплитудно-частотной характеристики. Из приведенного рисунка также следует, что благодаря сужению пространственного спектра изображения, обусловленного его увеличением, амплитудно-частотные искажения, вносимые в изображение апертурой воспроизводящего устройства, существенно уменьшаются. Существенно уменьшаются на изображении и искажения, связанные с неполным подавлением побочных спектров, т. к. они оказываются в той области частотного пространства, где значения $K(\omega_x, \omega_y)$ невелики. Однако на практике для получения интерполяции, близкой к предельно возможной, при использовании этого метода необходимо, чтобы число слагаемых в двойной сумме формулы (4.5) было бы около 100. Это является недостатком рассмотренного метода.

Применяемые в настоящее время графические редакторы, например Photoshop, предлагают использование трех видов интерполяции при увеличении изображений: интерполяцию нулевого порядка (Nearest Neighbor), интерполяцию первого порядка (Bilinear) и бикубическую интерполяцию (Bicubic) [35]. Из трех предлагаемых методов последний метод, являясь наилучшим, тем не менее, несколько уступает рассмотренному нами методу интерполяции с использованием функции отсчетов.

В Photoshop есть еще два способа интерполяции: **Bicubic smoother** (Бикубическая, глаже) и **Bicubic sharper** (Бикубическая, четче).¹ В них влияние "далеких" пикселов ослаблено и усилено соответственно.

¹ Эти способы интерполяции доступны после выбора в меню **Image** (Изображение) команды **Image Size** (Размер изображения), а затем в появившемся диалоговом окне разворачивания списка внизу окна.

В заключение отметим, что аналогичные по своему существу методы можно использовать и используют в целях уменьшения заметности мельканий движущихся изображений, например в кино, "размножая" проецируемые на экран кадры путем перекрывания их проекции обтюратором.

7.9. Метод триангуляции

Рассмотренные ранее методы интерполяции, применяемые при увеличении изображений, обладают одним общим недостатком, а именно при воспроизведении наклонных яркостных/цветовых границ на них проявляется эффект ступенчатости. Наиболее сильно этот недостаток появляется при использовании интерполяции нулевого порядка (Nearest Neighbor), в меньшей степени в случае применения билинейной интерполяции (Bilinear Interpolation) и в еще меньшей степени при использовании бикубической интерполяции (Bicubic Interpolation). Рисунок 7.18, на котором показано увеличенное в два раза изображение наклонной яркостной границы при использовании интерполяции нулевого порядка, демонстрирует этот артефакт.

На этом рисунке кружками обозначены отсчеты яркости в исходном увеличиваемом изображении, малыми квадратами — отсчеты, которые вставляются в увеличенное изображение, для которых путем интерполяции требуется определить значение яркости, клетками обозначены пикселы изображения.





Рис. 7.18. К пояснению эффекта ступенчатости

Рис. 7.19. К пояснению метода интерполяции по трем отсчетам

Анализ показывает, что ступенчатость на наклонной границе обусловлена тем, что в рассматриваемых методах направления, в которых выполняется интерполяция, привязаны к направлениям строк и столбцов, в то время как направления яркост-

ных/цветовых границ не всегда совпадают с этими направлениями. Для того чтобы уменьшить зазубренность наклонных границ на изображениях при их увеличении, было предложено отказаться от интерполяции по четырем опорным отсчетам, которые используются при интерполяции нулевого порядка, билинейной и бикубической интерполяций, а использовать для этой цели только три отсчета. На рис. 7.19 в качестве примера показаны три отсчета в исходном увеличиваемом изображении, обозначенные окружностями, расположенными в вершинах образуемого при этом треугольника, по которым путем интерполяции вычисляются значения яркости трех новых вставленных при увеличении отсчетов. В рассматриваемом частном случае этих отсчетов три и расположились они на линиях, соединяющих вершины треугольника 1, 2, 3, поскольку увеличение равно двум. В случае больших увеличений количество вставляемых новых отсчетов будет больше и располагаться они будут не только на сторонах треугольника, показанного на рисунке, но и внутри него. При этом интерполирующая формула в случае интерполяции отсчетов чернобелого полутонового изображения будет иметь вид:

$$L = ax + by + c ,$$

где L — яркость точки с координатами x и y; a, b и c — коэффициенты, постоянные в пределах интерполирующего треугольника. Значения этих коэффициентов легко находятся путем решения системы из трех уравнений:

$$L_{1} = ax_{1} + by_{1} + c,$$

$$L_{2} = ax_{2} + by_{2} + c,$$

$$L_{3} = ax_{3} + by_{3} + c,$$

где L_1 , L_2 , L_3 — яркости опорных отсчетов 1, 2 и 3, показанных на рис. 7.19, а x_1 , y_1 , x_2 , y_2 и x_3 , y_3 — их координаты.

Сопоставляя рис. 7.18 и 7.19, видим, что в последнем случае ступенчатость существенно уменьшилась, благодаря тому, что в последнем случае направление границы и направление, в котором выполняется интерполяция, совпадают. Это совпадение при использовании описываемого метода выполняется для горизонтальных, вертикальных и наклонных границ, идущих под углом 45° к направлению строк, но не выполняется для границ, угол наклона которых отличается от 45°. Поэтому для границ с другим углом наклона проблема ступенчатости границ рассмотренным методом не решается.

Решение этой проблемы было предложено в работе [124]. Авторами этой работы предложенный метод был назван data-dependent triangulation или DDT (триангуляция, зависящая от данных). Сущность метода DDT заключается в том, что опорные отсчеты таким образом объединяются в интерполирующие треугольники, что, по крайней мере, направление одной из сторон треугольника и направление границ на интерполируемом изображении оказываются максимально близкими. При этом выполняются следующие условия:

□ все отсчеты изображения используются для интерполяции;

🗖 интерполирующие треугольники не пересекаются.



Рис. 7.20. К пояснению метода триангуляции, зависящей от данных (DDT)

Рисунок 7.20, на котором показано разбиение плоскости изображения на интерполирующие треугольники, поясняет сказанное. На рисунке виден ряд треугольников, одна из сторон которых вытянута в определенном направлении — это направление границы на изображении, которое на рисунке не показано.

Мы здесь не будем касаться алгоритма формирования треугольных областей интерполяции. Читатель, интересующийся этой проблемой, найдет необходимые сведения в литературе [106, 124, 142, 178].

Отметим, что описанные в этом разделе методы направлены, главным образом, не на то, чтобы минимизировать среднеквадратичную ошибку интерполяции, а на то, чтобы минимизировать видность этой ошибки в значительной степени путем придания ей менее заметной структуры.

В настоящее время описанные методы нашли применение в ряде коммерческих программ.

7.10. Уменьшение изображений

Уменьшение изображений достигается путем прореживания его строк и пикселов в строках. Так, например, при уменьшении изображения в два раза из него устраняется каждая вторая строка и каждый второй пиксел вдоль каждой строки. Поскольку при этом количество отсчетов, которыми представляется изображение, сокращается, то это может повлечь за собой нарушение условий теоремы Котельникова (теоремы отсчетов) и, как следствие, возникновение на изображении шума пространственной дискретизации. На рис. 7.21, *а* в качестве примера показано изображение, которое было получено путем уменьшения исходного изображения в два раза. На рис. 7.21, *а* хорошо заметны полосы муара на здании с лоджиями. Для того чтобы избежать возникновения шума пространственной дискретизации, необходимо перед уменьшением изображения согласовать количество отсчетов (пикселов) на уменьшенном изображении с шириной его пространственного спектра. Достигается это путем пространственной фильтрации изображения фильтром, амплитудночастотная характеристика которого имеет прямоугольную форму с крутым срезом.



Рис. 7.21. Изображения, полученные после уменьшения в два раза: *а* — без предварительной фильтрации; *б* — с предварительной фильтрацией

Другими словами, для этой цели необходимо осуществить фильтрацию путем свертки изображения с импульсной функцией

$$h(n',k') = \frac{\sin(\omega_{xrp}n'/M)}{(\omega_{xrp}n'/M)} \frac{\sin(\omega_{yrp}k'/M)}{(\omega_{yrp}k'/M)}$$

где ω_{xrp} , ω_{yrp} — как и ранее граничные, пространственные частоты, M — кратность сокращения ширины спектра. Кратность сокращения ширины спектра должна

быть согласована с кратностью уменьшения размера изображения. Так, если в исходном изображении количество отсчетов на единицу длины было выбрано оптимальным образом, то кратность ограничения ширины пространственного спектра посредством фильтрации перед уменьшением изображения должна равняться кратности уменьшения его линейного размера. Распределение яркости в профильтрованном изображении $L_{cO}(k,n)$ находится по формуле

$$L_{c\Omega}(k,n) = \sum_{k'=-\frac{K-1}{2}}^{\frac{K-1}{2}} \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_{c}(k+k',n+n')h(n',k'),$$

где $L_{c}(k,n)$ — распределение яркости в исходном изображении, N и K — протяженности импульсной функции, которые рекомендуется брать равными 21—25.

На рис. 7.21, б в качестве примера приведено изображение, которое получено из того же исходного изображения, что и изображение, показанное на рис. 7.21, a, в результате двукратного уменьшения, но с применением предварительной фильтрации. При выполнении предварительной фильтрации было принято N = 25, K = 25. Из рассмотрения рисунка видно, что в данном случае на уменьшенном изображении отсутствует шум пространственной дискретизации.

Выше был рассмотрен случай, когда изображение уменьшалось в целое число раз. Для уменьшения изображения в не целое число раз, например в M_1/M_2 раз, где M_1 и M_2 — целые числа и, кроме того, $M_1 > M_2$, необходимо вначале исходное изображение увеличить в M_1 раз с применением метода, описанного в предыдущем разделе, затем профильтровать с целью сокращения ширины его пространственного спектра и, наконец, уменьшить в M_2 раз.

7.11. Интерполяция при использовании матриц Байера

При воспроизведении цветных изображений, полученных посредством матриц, в которых применен фильтр Байера, возникает необходимость восстанавливать значения отсутствующих отсчетов цветовых компонентов, поскольку каждая ячейка матрицы, как это видно на рис. 7.22 и ЦВ.7, может генерировать сигнал только одного из трех цветовых компонентов. Из рис. 7.22 также следует, что количество отсчетов зеленого компонента в два раза превышает количество отсчетов красного или синего компонентов. Значения отсутствующих отсчетов могут быть найдены одним из методов интерполяции.

Простейшим методом интерполяции является интерполяция нулевого порядка, при этом отсутствующие значения отсчетов зеленого компонента находятся как среднее значений четырех ближайших имеющихся отсчетов этого компонента. Для нахождения значений отсутствующих отсчетов красного компонента можно применить двухступенчатую процедуру. Вначале найти значения отсчетов, местоположение которых совпадает с местоположением отсчетов синего компонента путем определения среднего по четырем соседним отсчетам красного компонента, а затем найти значения отсчетов, местоположение которых совпадает с местоположением отсчетов зеленого компонента путем усреднения значений отсчетов четырех ближайших отсчетов красного компонента. Аналогичным образом определяются значения отсутствующих отсчетов синего компонента. Однако этот метод приводит к возникновению на изображении заметного шума пространственной дискретизации и размытию границ, как это видно из рис. 7.23.



Рис. 7.22. Фильтр Байера





Рис. 7.23. Пример восстановления отсутствующих отсчетов методом интерполяции нулевого порядка: *а* — исходное изображение; *б* — изображение, полученное методом интерполяции нулевого порядка¹

¹ Изображения на рис. 7.22, 7.23, 7.25, 7.26 цитируются по работе D. Darian Muresan and Thomas W. Parks Optimal Recovery Approach to Image Interpolation // School of Electrical and Computer Engineering, Cornell University Ithaca, NY 14853 darian, помещенной в Интернете по адресу: http://focus.ti.com/ pdfs/univ/09-Poster.pdf.

Ситуацию можно несколько исправить, если принять во внимание то, что в пределах объектов, составляющих сцену, как правило, цвет мало изменяется, а изменяется в основном яркость. Это обусловлено условиями освещения, например величиной угла между отражающей свет поверхностью объекта и направлением на источник света, который в разных точках поверхности объекта оказывается различным. Принимая во внимание, что для большинства изображений яркость в значительной степени определяется интенсивностью зеленого компонента, а, значит, как это следует из предыдущего, отношение интенсивностей красного компонента к интенсивности зеленого компонента, а также отношение интенсивности синего компонента к интенсивности зеленого компонента в пределах изображения объекта приблизительно постоянно, модернизируем метод следующим образом.

Вначале найдем значения отсутствующих отсчетов зеленого компонента. Поскольку имеющихся отсчетов зеленого компонента в обрабатываемом изображении больше, чем отсчетов красного и синего компонентов, то зеленый компонент изображения может быть получен с большим разрешением.

Затем найдем значения отсчетов красного компонента путем перемножения значения отсчетов зеленого компонента, которые теперь известны для всех пикселов, на отношения значений отсчетов красного компонента к значениям отсчетов зеленого компонента, которые также известны для пикселов, расположенных рядом. Аналогичным образом находятся значения отсчетов синего компонента. Описанная модернизация метода заметно улучшает восстанавливаемое изображение, однако его границы остаются несколько размытыми (рис. 7.24), что обусловлено несовершенством метода интерполяции, который используется при определении значений отсутствующих отсчетов зеленого компонента.



Рис. 7.24. Пример восстановления отсутствующих отсчетов модернизированным методом интерполяции

P_1	P_2	<i>P</i> ₃
P_4	P_5	P_6
P_7	P_8	P_9

Рис. 7.25. К пояснению модернизации метода

Поэтому в рассматриваемый метод была введена дополнительная модернизация, которая заключается в том, что интерполяция выполняется вдоль границ, а не поперек границ.

Для пояснения существа этой модернизации обратимся к рис. 7.25. Обозначим через p_i функцию, имеющую смысл вероятности того, что пиксел P_5 и пиксел P_i принадлежат одной и той же границе. Если P_5 и P_i находятся оба с одной стороны границы, то p_i должна быть близкой к единице, в противном случае p_i должна быть близкой к нулю. Алгоритм дополнительной модернизации при этом будет таким.

1. Вначале находятся отсутствующие значения отсчетов зеленого компонента изображения путем интерполяции посредством формулы

$$G_5 = \frac{p_2G_2 + p_4G_4 + p_6G_6 + p_8G_8}{p_2 + p_4 + p_6 + p_8},$$

где G_2 , G_4 , G_5 , G_6 , G_8 — значения отсчетов зеленого компонента для пикселов P_2 , P_4 , P_5 , P_6 , P_8 . Затем путем интерполяции находятся отсутствующие значения отсчетов красного и синего компонентов.

В свою очередь, процедура нахождения значений отсчетов красного и синего компонентов включает в себя два этапа.

На первом этапе находятся отсутствующие значения отсчетов красного компонента изображения, местоположение которых совпадает с местоположением отсчетов синего компонента путем интерполяции посредством формулы

$$R_5 = G_5 \frac{p_1 \frac{R_1}{G_1} + p_3 \frac{R_3}{G_3} + p_7 \frac{R_7}{G_7} + p_9 \frac{R_9}{G_9}}{p_1 + p_3 + p_7 + p_9},$$

где G_1 , G_3 , G_5 , G_7 , G_9 — значения отсчетов зеленого компонента для пикселов P_1 , P_3 , P_5 , P_7 , P_9 ; R_1 , R_3 , R_5 , R_7 , R_9 — значения отсчетов красного компонента для пикселов P_1 , P_3 , P_5 , P_7 , P_9 .

На втором этапе находятся отсутствующие значения отсчетов красного компонента изображения, местоположение которых совпадает с местоположением отсчетов зеленого компонента путем интерполяции посредством формулы

$$R_5 = G_5 \frac{p_2 \frac{R_2}{G_2} + p_4 \frac{R_4}{G_4} + p_6 \frac{R_6}{G_6} + p_8 \frac{R_8}{G_8}}{p_2 + p_4 + p_6 + p_8},$$

где R_2 , R_4 , R_5 , R_6 , R_8 — значения отсчетов красного компонента для пикселов P_2 , P_4 , P_5 , P_6 , P_8 .

Аналогичным образом находятся отсутствующие значения отсчетов синего компонента изображения.

2. Следующий шаг алгоритма состоит в выборе функции p_i . В своей работе [121] Кок (Cok) предложил функцию p_i выбрать следующим образом.

При определении значения отсчета для зеленого компонента пиксела P₅:

$$\left(\begin{array}{l} p_2 = p_8 = 0; \ p_4 = p_6 = 1, \ \text{если} \ \left| G_4 - G_6 \right| < P$$
 и $\left| G_2 - G_8 \right| > P, \ p_2 = p_8 = 1; \ p_4 = p_6 = 0, \ \text{если} \ \left| G_4 - G_6 \right| > P$ и $\left| G_2 - G_8 \right| < P, \$ иначе $p_2 = p_4 = p_6 = p_8 = 1, \$

где *Р* — величина порога, который подбирается экспериментально.

При определении значения отсчета для синего компонента пиксела P₅:

$$p_1 = p_3 = p_7 = p_9 = 1;$$

 $\begin{cases} p_2 = p_8 = 1; \ p_4 = p_6 = 0, \ \text{если } P_2 \ \text{и } P_8 - \text{синие}, \\ p_2 = p_8 = 0; \ p_4 = p_6 = 1, \ \text{если } P_4 \ \text{и } P_6 - \text{синие}. \end{cases}$

Для красного компонента все аналогично.

На рис. 7.26 приведено изображение, полученное в результате использования описанного метода. Сравнивая это изображение с изображением рис. 7.24, видим, что в данном случае шум пространственной дискретизации хотя и уменьшился, но все еще велик.



Рис. 7.26. Пример восстановления отсутствующих отсчетов на изображении с применением дополнительно модернизированного метода интерполяции

Дальнейшее усовершенствование рассматриваемых методов интерполяции было сделано Киммелом (Kimmel) [136], который предложил более удачные варианты функции p_i , применив градиентные оценки, вычисляемые через конечные разности.

В соответствии с рис. 7.25 им были определены градиенты на изображении в точке отсчета пиксела P_5 для направлений осей x и y, а также для диагональных направлений следующим образом

$$D_{x}(I_{5}) = \frac{I_{4} - I_{6}}{2} , D_{y}(I_{5}) = \frac{I_{2} - I_{8}}{2} , D_{xd}(I_{5}) = \frac{I_{3} - I_{7}}{2\sqrt{2}} , D_{yd}(I_{5}) = \frac{I_{1} - I_{9}}{2\sqrt{2}}$$

где I_i — интенсивность красного, зеленого или синего компонентов изображения пиксела P_i . Обратим внимание на то, что при вычислении разности всегда используют компоненты одного цвета. Так если интенсивность I_5 является интенсивностью зеленого компонента, то интенсивностями зеленого компонента являются и интенсивности $I_1 - I_9$. Функции p_i были определены Киммелем [136] следующим образом

$$p_i = \frac{1}{\sqrt{1 + D(I_5)^2 + D(I_i)^2}},$$

где D выражается через разность в направлении I_i . Поясним сказанное двумя примерами, а именно

$$p_{6} = \left[1 + D_{x}(I_{5})^{2} + D_{x}(I_{6})^{2}\right]^{-1/2} \times p_{3} = \left[1 + D_{xd}(I_{5})^{2} + D_{xd}(I_{3})^{2}\right]^{-1/2}$$

На рис. 7.27 показано изображение, полученное путем применения описанного алгоритма. Сопоставляя это изображение с изображением, приведенным на рис. 7.26, видим, что оно менее искажено шумом пространственной дискретизации, чем изображение на рис. 7.26.



Рис. 7.27. Пример восстановления отсутствующих отсчетов на изображении с применением метода Киммела. Изображение заимствовано из [136]



Морфологические операции и их применение

8.1. Морфологические операции

Первоначально морфологические операции были развиты применительно к обработке бинарных изображений, но в дальнейшем морфологические алгоритмы были расширены на полутоновые изображения. Морфологические операции применяют для изменения формы изображений объектов. Например, для устранения разрывов в контурных линиях, устранения отдельных точек и штрихов при получении контурных препаратов, для изготовления масок и т. д. Другим примером применения этих операций может служить использование морфологической фильтрации, например, в дактилоскопии. Морфологические операции также широко используют в алгоритмах анализа изображений и распознавания образов. Эти операции являются нелинейными. В основе морфологических операций лежит учет связности пикселов в обрабатываемых изображениях. Для описания алгоритмов, в соответствии с которыми выполняется та или иная морфологическая операция, используют различные подходы, а именно теорию множеств, метод шаблонов, а также другие методы [23, 102, 166, 167]. Исходя из соображений достижения наибольшей наглядности изложения, мы будем использовать тот или иной метод. Следуя сложившейся традиции, рассмотрение морфологических операций начнем применительно к обработке бинарных изображений. Поэтому вначале рассмотрим, как получаются бинарные изображения.

8.2. Бинарные изображения и их характеристики

В отличие от полутонового изображения в бинарном (двухградационном) изображении яркость пикселов может принимать только два значения — черное или белое. Бинарное изображение получается из цветного изображения посредством его преобразования вначале в полутоновое изображение, а затем полутонового изображения в бинарное путем сравнения яркости каждого из пикселов с заранее выбранным порогом. В том случае, если яркость пиксела превышает значение установленного порога, ему присваивается значение "единица", он считается белым, в противном случае ему присваивается нулевое значение, и он считается черным.



Рис. 8.1. Исходное (а) и бинаризированное (б) изображения

Принято при представлении объектов на бинарном изображении пикселам, представляющим объекты, присваивать значения "единица" (белое), а пикселам, представляющим фон, присваивать значения "ноль" (черное). Эта операция называется *бинаризацией изображения*. На рис. 8.1 показаны соответственно исходное, в данном примере полутоновое, изображение и это же изображение после его бинаризации.

Разумеется, бинарное изображение содержит существенно меньшее количество информации о представленных на нем объектах, поскольку содержит лишь их силуэты, однако, в ряде случаев, исключение из изображений излишней, не используемой при решении данной конкретной задачи информации, может быть полезным. Примером таких задач, где работать с бинарными изображениями предпочтительнее, являются задачи определения площади объектов, задачи, в которых измеряется положение объекта на изображении, и ряд других.

Пользуясь терминологией математической морфологии, можно сказать, что объекты на бинарном изображении представляются множествами всех пикселов соответствующего цвета (в данном случае белого). Задание такого множества является вариантом полного морфологического описания изображения.

Рассмотрим методы нахождения основных геометрических характеристик бинарных изображений объектов.

Площадь бинарного изображения объекта *S* по определению равна количеству пикселов, составляющих силуэт объекта, т. е.

$$S = \sum_{k=1}^{K} \sum_{n=1}^{N} L(k,n),$$

где: L(k,n) — яркость пиксела, занимающего *k*-ю строку и *n*-й столбец в бинарном изображении, состоящем из *K* строк и *N* столбцов. Напомним, что по условию яркость белого пиксела бинарного изображения принята равной единице, а черного пиксела равной нулю.

Положение силуэта объекта k_{ob} , n_{ob} на бинарном изображении часто определяют как координаты "центра тяжести" ("центра масс"), которые вычисляются по формулам

$$k_{\rm of} = \frac{\sum_{k=1}^{K} \sum_{n=1}^{N} kL(k,n)}{S}, \quad n_{\rm of} = \frac{\sum_{k=1}^{K} \sum_{n=1}^{N} nL(k,n)}{S}.$$

Однако если нам известна форма и размеры трехмерного объекта, а также характеристики отображающей системы, то, используя бинарное изображение, можно определить не только положение центра тяжести объекта на плоскости этого изображения, но и его положение в пространстве. Поясним сказанное примером. На рис. 8.2 показано силуэтное изображение самолета, идущего на посадку.



Рис. 8.2. Силуэтное изображение самолета

Измеряя расстояние между концами крыльев на изображении в пикселах, а также зная размах крыльев реального объекта и характеристики телевизионной камеры (фокусное расстояние объектива, размеры фоточувствительной поверхности датчика сигнала изображения, размеры пиксела), посредством которой было получено исходное изображение, не составляет труда посредством геометрических построений определить расстояние до объекта. Далее, исходя из формы реального объекта и его видимого силуэта, контуры которого совпадают с контурами его проекции, элементарно находятся углы, определяющие его положение в пространстве.

К бинарным изображениям принадлежат также контурные изображения. Они используются для представления рукописного или печатного текста, рабочих и сборочных чертежей конструкторской документации, в художественных графических работах.

8.3. Логические операции над бинарными изображениями

Основными логическими операциями являются: логическое умножение (AND), логическое сложение (OR) и отрицание (NOT). Эти три операции представляют функционально полный класс, все другие логические операции могут быть получены путем комбинирования только этих трех операций. В табл. 8.1 приведены результаты выполнения четырех наиболее часто используемых при обработке изображений
логических операций AND, OR, NOT и XOR (исключающее ИЛИ). Через L_1 и L_2 обозначены операнды, участвующие в логических операциях, при этом операция NOT выполняется над операндом L_1 . При обработке изображений логические операции применяются попиксельно, т. е. над соответствующими пикселами двух изображений (исключение составляет только операция NOT, которая выполняется над пикселами одного изображения, инвертируя значения их яркостей). На рис. 8.3 показаны результаты применения приведенных логических операций над бинарными изображениями.

Таблица	8.1
---------	-----

L_1	L_2	AND	OR	NOT	XOR
0	0	0	0	1	0
0	1	0	1	1	1
1	0	0	1	0	1
1	1	1	1	0	0



Рис. 8.3. Иллюстрация применения логических операций к бинарным изображениям

8.4. Понятие связности

При обработке бинарных изображений широко применяется понятие *связности* пикселов, относительно которых предполагается, что они принадлежат одному классу (объекту). Рассмотрим рис. 8.4, *а*. На нем хорошо видны 3 области: черный контур прямоугольника, белые пикселы внутри этого контура и белые пикселы

снаружи. Пикселы, относящиеся к каждой из областей, связаны друг с другом. На рис. 8.4, б ситуация не столь однозначна. Пикселы черного цвета можно отнести как к четырем разным прямолинейным отрезкам, так и к единому контуру прямоугольника. Для разрешения данной ситуации следует определить понятие связности.



Рис. 8.4. Иллюстрация к понятию связности пикселов

Для используемого в компьютерных дисплеях ортогонального растра определяют два вида связности пикселов: четырехсвязность и восьмисвязность. Пиксел A_8 , показанный на рис. 8.5, интенсивность которого равна единице, называют четырехсвязным, если среди окружающих его пикселов имеется хотя бы один с интенсивностью, равной единице, который бы своей "стороной" касался пиксела A_8 .

A_0	A_1	A_2
A_7	A_8	A_3
A_6	A_5	A_4

Рис. 8.5. Иллюстрация четырехсвязности и восьмисвязности пикселов

Другими словами, для того чтобы пиксел A_8 был бы четырехсвязным, необходимо, чтобы хотя бы один из пикселов A_1 , A_3 , A_5 , A_7 имел интенсивность, равную единице. Этот же пиксел называют восьмисвязным, если среди окружающих его пикселов A_0 , A_1 , A_2 , A_3 , A_4 , A_5 , A_6 , A_7 хотя бы один имеет интенсивность, равную единице. Сделанные определения касаются определения связности объектов. Аналогичные определения можно сделать относительно фона. При этом четырехсвязность фона автоматически означает восьмисвязность объектов и наоборот.



Рис. 8.6. Иллюстрация к критерию связности объектов

Алгоритмы целого ряда морфологических обработок бинарных изображений критичны к критерию связности. Проиллюстрируем это следующим примером. На рис. 8.6 показано бинарное изображение, на котором имеются пикселы, принадлежащие и фону, и объектам. Если объекты, приведенные на этом рисунке, рассматривать как четырехсвязные, то в нем будет три объекта, а если их рассматривать как восьмисвязные, то их будет только два.

Если в поле зрения находится более одного объекта, то можно определить топологические характеристики имеющейся совокупности объектов: например, разность между числом объектов и числом отверстий (число Эйлера).

Нетрудно также пометить отдельные объекты и вычислить геометрические характеристики для каждого из них в отдельности. Наконец, перед дальнейшей обработкой изображение можно упростить, постепенно модифицируя его итеративным образом.

8.5. Логическая апертура

Поясним морфологическую обработку изображений методом использования так называемой логической апертуры следующим образом. При морфологической обработке бинарного изображения логической апертурой оно сканируется окном, размеры которого составляют $N \times M$ пикселов. В качестве примера на рис. 8.7 показано такое окно размером 3×3 пиксела. Здесь через L_0 , L_1 , L_2 , L_3 , L_4 , L_5 , L_6 , L_7 , L_8 обозначены интенсивности пикселов обрабатываемого изображения, которые оказались в пределах сканирующего окна.

$$M$$

$$L_0 \quad L_1 \quad L_2$$

$$N \quad L_7 \quad L_8 \quad L_3$$

$$L_6 \quad L_5 \quad L_4$$

Рис. 8.7. Окно размером 3×3 пиксела

В процессе сканирования изображения при каждом положении сканирующего окна из обрабатываемого изображения считываются значения пикселов, оказавшихся в его пределах, и на основании анализа значений их интенсивностей в соответствии с правилами, принятыми для реализуемой морфологической операции, центральному пикселу присваивается то или иное значение.

Рассмотрим пример использования логической апертуры для описания алгоритма простейшей морфологической операции, заключающейся в *удалении одиночных точек* (*clean*), которая уничтожает на двухградационных изображениях объекты размером в один пиксел, т. е. одиночные белые точки на черном фоне. В этом случае правило, реализуемое логической апертурой при выполнении рассматриваемой морфологической операции, может быть записано следующим образом:

$$L_8 = \begin{cases} 0, & \text{если } L_0 = L_1 = L_2 = L_3 = L_4 = L_5 = L_6 = L_7 = 0, \\ L_8 & \text{в противном случае.} \end{cases}$$

Другими словами, если все 8 соседей центрального пиксела принадлежат фону, то и интенсивность центрального пиксела нужно сделать равной нулю, т. е. преобразовать его в фоновый пиксел, в противном случае значение центрального пиксела следует оставить неизменным. На рис. 8.8, a приведено исходное бинарное изображение, которое, в частности, содержит две белые точки размером в один пиксел, а на рис. 8.8, δ показано это же изображение, но после его обработки морфологической операцией *clean*. Рассмотрение рис. 8.8, δ показывает, что имеющиеся на рис. 8.8, a две белые одиночные точки на рис. 8.8, δ исчезли, в то время как изломанная линия толщиной в один пиксел при этом сохранилась. На этом рисунке сохранились также изображения крестов толщиной в один пиксел и буквы, толщина линий которой составляет 3 пиксела.

Рассмотренная морфологическая операция может быть использована для удаления одиночных импульсных помех на бинарном изображении, а также для удаления одиночных точек, возникающих при изготовлении контурных препаратов из полутоновых изображений.

В качестве второго примера рассмотрим использование логической апертуры для описания алгоритма морфологической операции, заключающейся в *удалении внутренних пикселов объектов (remove)*. Эта операция на бинарных изображениях в отличие от морфологической операции *clean* устанавливает в 0 пикселы объектов, у которых интенсивности четырех соседних по горизонтали и по вертикали пикселов равны единице. В этом случае правило, реализуемое логической апертурой при выполнении рассматриваемой морфологической операции, может быть записано следующим образом:

$$L_8 = \begin{cases} 0, & \text{если } L_1 = L_3 = L_5 = L_7 = 1, \\ L_8 & \text{в противном случае.} \end{cases}$$

Рисунок 8.8, *в* демонстрирует результат применения к изображению, представленному на рис. 8.8, *а*, морфологической операции *remove*.



Рис. 8.8. Исходное бинарное изображение (a), изображения после выполнения морфологических операций *clean* (b), *remove* (b)

Обратим внимание на то, что морфологические операции действуют на пикселы обрабатываемого изображения избирательно, при этом результат их воздействия определяется как видом морфологической операции, так и значениями интенсивности пикселов в области, примыкающей к обрабатываемому пикселу. Другими словами, морфологические операции относятся к группе так называемых *пространственных (матричных)* преобразований.

Понятие логической апертуры позволяет легко распространить морфологические операции на полутоновые, а также на трехмерные (объемные) изображения.

8.6. Морфологические операции дилатации и эрозии бинарных изображений

Дилатация (наращивание) и эрозия являются базовыми морфологическими операциями, которые часто применяются при обработке изображений. Их используют и для построения других морфологических операций. Ограничимся вначале рассмотрением простейшего случая, когда бинарное изображение объекта сканируется окном размером 3×3 пиксела.

В этом случае алгоритм операции дилатации (*dilate*) заключается в том, что при сканировании обрабатываемого изображения окном размером 3×3 пиксела центральный пиксел исходного изображения устанавливается в состояние 1, если хотя бы один из пикселов его восьмисвязной окрестности равен единице, в противном случае он сохраняет свое прежнее нулевое значение. Этот алгоритм может быть записан следующим образом:

$$L_8 = \begin{cases} 1, & \text{если } (L_0 = 1) \lor (L_1 = 1) \lor (L_2 = 1) \lor (L_3 = 1) \lor \\ & \lor (L_4 = 1) \lor (L_5 = 1) \lor (L_6 = 1) \lor (L_7 = 1), \\ L_8 & \text{в противном случае,} \end{cases}$$

где символ V означает логическую операцию ИЛИ.



Рис. 8.9. К пояснению операции дилатации: *а* — исходное изображение; *б* — изображение после выполнения дилатации

В результате выполнения операции дилатации на объект, представленный на бинарном изображении, наращивается слой пикселов, имеющих значение "единица". Рисунок 8.9, на котором слева показан исходный объект (рис. 8.9, *a*), а справа (рис. 8.9, *б*) этот же объект, подвергнутый дилатации, поясняет сказанное.

Операция эрозии является операцией, обратной по отношению к операции дилатации (наращивания).

В рассматриваемом случае алгоритм эрозии (*erode*) заключается в том, что интенсивность пиксела исходного изображения устанавливается в ноль, если интенсивность хотя бы одного из пикселов изображения, оказавшихся внутри окна размером 3×3 пиксела, равна нулю, в противном случае его интенсивность сохраняет свое значение. Этот алгоритм может быть записан следующим образом:

$$L_8 = \begin{cases} 0, & \text{если } (L_0 = 0) \lor (L_1 = 0) | \lor (L_2 = 0) \lor (L_3 = 0) \lor \\ & \lor (L_4 = 0) \lor (L_5 = 0) \lor (L_6 = 0) \lor (L_7 = 0), \\ L_8 & \text{в противном случае.} \end{cases}$$
(8.1)

В результате выполнения эрозии с объекта, представленного на бинарном изображении, снимается внешний слой пикселов, принадлежащих объекту, т. е. пикселов, имеющих значение "единица", а также сглаживаются выступы границ объекта размером в один пиксел. Кроме того, при эрозии уничтожаются все малые объекты (линии толщиной в один или два пиксела, точки, размеры которых меньше размера окна, и т. д.). Рисунок 8.10 поясняет сказанное. На этом рисунке слева (рис. 8.10, *a*) показаны исходные объекты, а справа (рис. 8.10, *б*) — результат их преобразования операцией "эрозия".

Морфологические операции дилатации и эрозии, как впрочем и ряд других, могут быть *n*-кратными. Под *n*-кратной операцией понимается последовательное применение *n* раз данной операции к обрабатываемому изображению. Рисунок 8.11 поясняет сказанное. На рис. 8.11, *a* приведено исходное изображение, на рис. 8.11, *б*

показан результат применения к нему однократной операции дилатации, а на рис. 8.11, *в* двукратное применение этой операции. Рисунок 8.11, *г* и *д* соответственно демонстрируют применение к исходному изображению однократной и двукратной операции эрозии.



Рис. 8.10. Исходное изображение (*a*), изображение после выполнения операции эрозии (*б*)



Рис. 8.11. К пояснению *n*-кратности выполнения операций дилатации и эрозии: исходное изображение (*a*), результаты однократного выполнения этих операций ((б) и (*c*) соответственно), результаты двукратного выполнения этих операций ((*в*) и (*d*) соответственно)

Как уже было отмечено, морфологические операции над изображениями часто рассматривают на основе теории множеств [23, 102]. Рассмотрим этот вопрос подробнее. Пусть L_c и **В** представляют собой множества из пространства Z^2 . Первое множество \mathbf{L}_{c} является обрабатываемым изображением, а второе — так называемым, *структурообразующим множеством* **В** (структурообразующим элементом, или примитивом (structuring element)). Дилатация множества \mathbf{L}_{c} по множеству **В** обозначается как $\mathbf{L}_{c} \oplus \mathbf{B}$ и определяется следующим образом

$$\mathbf{L}_{c} \oplus \mathbf{B} = \left\{ z \left| \left(\hat{\mathbf{B}} \right)_{z} \cap \mathbf{L}_{c} \neq \emptyset \right\} \right\}.$$
(8.2)

В написанном выражении использованы следующие обозначения: \bigcap — символ пересечения множества пикселов, составляющих обрабатываемое изображение \mathbf{L}_c , с множеством пикселов, составляющих примитив \mathbf{B} ; \emptyset — символ пустого множества; символ \wedge над \mathbf{B} — это символ центрального отражения множества, представляющего примитив \mathbf{B} ; ()_z — символ параллельного сдвига множества, заключенного в круглые скобки.

Напомним кратко значение этих операций:

- □ пересечением двух множеств A и B называется множество C, элементы которого принадлежат одновременно двум пересекающимся множествам, и обозначается следующим образом: C = A ∩ B;
- □ множество называется пустым, если оно не содержит ни одного элемента;
- центральным отражением множества называется его зеркальное отражение относительно своих вертикальной и горизонтальной осей;
- \Box под параллельным сдвигом или переносом множества **A** понимается его перенос в точку *z* и обозначается как (**A**)_{*z*}.

Таким образом, выражение (8.2) описывает следующие действия, выполняемые над обрабатываемым изображением. Вначале множество **B**, представляющее собой примитив, подвергается центральному отражению относительно своего начала координат (относительно центра **B**) и затем сдвигается в точку *z*. После этого выполняется оценка результирующего значения пиксела изображения. Для того чтобы выполнить оценку значений пикселов всего массива L_c , обрабатываемое изображение сканируется центральным отражением примитива. При этом значения пикселов в обрабатываемом изображении устанавливаются равными единице в том случае, если единичные значения отсчетов обрабатываемого изображения и центрального отражения примитива совпадают хотя бы в одном элементе. В противном случае значения пикселов в изображении не изменяются.

Учитывая сказанное, выражение (8.2) можно представить в следующем виде

$$\mathbf{L}_{c} \oplus \mathbf{B} = \left\{ z \left| \left(\hat{\mathbf{B}} \right)_{z} \cap \mathbf{L}_{c} \subseteq \mathbf{L}_{c} \right\} \right\}.$$
(8.3)

В написанной формуле использован символ \subseteq , который обозначает, что подмножество вновь образованных единиц принадлежит множеству пикселов изображения L_c .

Помимо выражения (8.2) в современной литературе по морфологии изображений можно встретить и другие определения дилатации, однако это выражение удобнее других своей наглядностью, если рассматривать примитив как маску свертки. Несмотря на то, что дилатация относится к операциям над множествами, а сверт-ка — к арифметическим операциям, основной процесс, состоящий в отражении множества **B** относительно его центра и затем в сканировании по множеству L_c (т. е. по изображению), по существу, аналогичен процессу свертки, рассмотренно-му в *разд. 5.2*.

Обращаясь к операции эрозии множества L_c , представляющего собой бинарное изображение, по примитиву **B**, которая обозначается как $L \ominus B$, запишем

$$\mathbf{L}_{c} \ominus \mathbf{B} = \left\{ z \, \middle| \, \left(\mathbf{B} \right)_{z} \subseteq \mathbf{L}_{c} \right\}. \tag{8.4}$$

Другими словами, эрозия множества L_c по примитиву **В** представляет собой множество всех пикселов, при сдвиге в которые множество **В** полностью содержится в множестве L_c . Как и в случае дилатации, выражение (8.4) не является единственно возможным определением эрозии (см., например, [23]). Однако этому определению обычно отдается предпочтение при использовании морфологических методов по тем же причинам, о которых уже упоминалось выше. Нетрудно видеть, что дилатация и эрозия являются двойственными операциями по отношению к теоретико-множественным операциям дополнения и центрального отражения.

Часто в качестве примитива используют массив единиц размером 3×3. Понятно, что вследствие симметрии этого массива относительно центра в данном частном случае отражение примитива относительно центра теряет смысл.

Примечание

В общем случае форма и размеры примитива могут быть выбраны любыми в зависимости от решаемой задачи. В этом случае в зависимости от формы и размеров примитива при дилатации толщина наращиваемого слоя пикселов, имеющих значение "единица", в различных направлениях может быть различной. Аналогично для операции "эрозия".

8.7. Использование шаблонов при выполнении морфологических операций

В ряде случаев при описании алгоритма выполнения морфологической операции удобно использовать так называемые шаблоны, размеры которых совпадают с размерами сканирующего окна. Этот метод близок к методу, в котором использованы примитивы. Поясним применение шаблонов на простейшем, уже рассматривавшемся ранее примере удаления одиночных точек (операция *clean*) на бинарных изображениях, а затем используем их при описании других алгоритмов. В рассмат-

риваемом случае обрабатываемое изображение сканируется окном размером 3×3 пиксела, показанным на рис. 8.7. При этом область изображения, перекрываемая сканирующим окном, попиксельно сравнивается с шаблоном, показанным на рис. 8.12.



Рис. 8.12. Пример шаблона размером 3×3 пиксела

В том случае, если интенсивности пикселов изображения, оказавшихся в пределах сканирующего окна, совпадают с интенсивностями соответствующих пикселов шаблона, интенсивность пиксела изображения, оказавшегося в центре окна, приравнивается нулю, т. е. он преобразуется в фоновый пиксел, в противном случае интенсивность его не изменяется. Это можно записать следующим образом:

$$L_8 = \begin{cases} 0, & \text{если } \sum_{k=0}^{8} (L_k - L_{\text{ш}k})^2 = 0, \\ L_8 & \text{в противном случае,} \end{cases}$$

где через L_0 , L_1 , L_2 , L_3 , L_4 , L_5 , L_6 , L_7 , L_8 обозначены значения интенсивности пикселов изображения, оказавшихся в пределах сканирующего окна размером 3×3 пиксела, а через $L_{\rm m0}$, $L_{\rm m1}$, $L_{\rm m2}$, $L_{\rm m3}$, $L_{\rm m4}$, $L_{\rm m5}$, $L_{\rm m6}$, $L_{\rm m7}$, $L_{\rm m8}$ — соответственно значения интенсивностей пикселов шаблона, пронумерованных в том же порядке, что и пикселы окна.

Морфологическая операция удаления одиночных точек является слишком простой, для того чтобы на ее примере можно было бы продемонстрировать достоинства использования шаблонов, заключающиеся в удобстве описания и реализации алгоритмов различных морфологических операций. С целью демонстрации возможностей рассматриваемого приема рассмотрим еще несколько морфологических операций на примерах, когда размер применяемых шаблонов равен 3×3 пиксела.

Морфологическая операция дилатации (dilate). Алгоритм этой морфологической операции при использовании метода шаблонов может быть описан следующим образом.

$$L_8 = \begin{cases} 1, & \text{если } \sum_{k=0}^{8} (L_k - L_{\text{ш}k}) > 0 \\ L_8 & \text{в противном случае.} \end{cases}$$

На рис. 8.13 приведено изображение шаблона для рассматриваемого случая, который представляет собой массив, состоящий из нулей. На рисунке черным обозначены элементы шаблона, равные нулю.



Рис. 8.13. Пример шаблона, состоящего из нулей

Морфологическая операция эрозии (erode), которая, как уже отмечалось, является операцией, обратной по отношению к операции дилатации. Алгоритм этой морфологической операции, уже рассмотренной ранее, при использовании метода шаблонов может быть описан следующим образом:

$$L_{8} = \begin{cases} 0, & \text{если } \sum_{k=0}^{8} (L_{k} - L_{\text{m}k}) < 9, \\ L_{8} & \text{в противном случае.} \end{cases}$$
(8.5)

В этом случае также применяется шаблон, показанный на рис. 8.13. Правило (8.5), следуя которому реализуется алгоритм эрозии, несколько проще, чем правило, представленное формулой (8.1).

Ранее был рассмотрен случай, когда окно и соответственно шаблон имели одинаковые размеры в обоих направлениях, равные 3×3 пиксела. В общем случае эти размеры могут быть большими и неодинаковыми, если этого требует вид реализуемой морфологической операции.

При рассмотрении морфологических операций дилатации и эрозии оказалось достаточным использование только одного сравнительно простого шаблона, в других случаях, как это будет видно из дальнейшего, количество используемых шаблонов может быть существенно больше, а вид их более сложным.

Вопрос о том, какой метод выполнения морфологических операций следует выбрать в каждом частном случае, решается отдельно, исходя из соображений простоты получающегося при этом алгоритма.

Рассмотрим теперь некоторые другие наиболее употребительные морфологические операции.

8.8. Морфологические операции бинарного открытия и бинарного закрытия

Последовательное применение операций эрозии и дилатации называется операцией бинарного открытия (open), или размыкания [23]. Эта операция на первом шаге своего выполнения (на шаге выполнении операции эрозии) устраняет мелкие детали, точки и линии, объединяя области фона, разъединенные ранее узкими линиями. На втором шаге выполнения (на шаге выполнении операции дилатации) операция бинарного открытия восстанавливает размеры объектов, сохранившихся после выполнения первого шага. В результате применения операции бинарного открытия к бинарному изображению на этом изображении сглаживаются контуры объектов, ликвидируются выступы небольшой ширины, обрываются узкие перешейки.

В качестве иллюстрации на рис. 8.14, *а* приведено исходное бинарное изображение, на рис. 8.14, *б* показано это же изображение после выполнения первого шага алгоритма (т. е. эрозии), а на рис. 8.14, *в* приведен результат выполнения всей операции открытия, т. е. второго заключительного шага этой операции (дилатации).



Рис. 8.14. Иллюстрация выполнения операции бинарного открытия: а — исходное изображение; б — изображение, получившееся после выполнения 1-го шага алгоритма; в — изображение, получившееся после выполнения 2-го шага алгоритма

Последовательное применение операций дилатации и эрозии называется операцией *бинарного закрытия (close)*, или замыкания [23]. Эта операция на первом шаге своего алгоритма (на шаге выполнении операции дилатации) "заливает" узкие разрывы и щели малой ширины в объектах, кроме того, эта операция удаляет малые дыры в объектах и заполняет промежутки в их контурах.

На втором шаге алгоритма (на шаге выполнении операции эрозии) операция бинарного закрытия восстанавливает размеры объектов, изменившиеся после выполнения первого шага алгоритма. В результате применения операции бинарного закрытия к бинарному изображению на этом изображении также сглаживаются контуры объектов, но это достигается за счет "заливки" узких разрывов.





б — изображение, получившееся после выполнения 1-го шага алгоритма;
 в — изображение, получившееся после выполнения 2-го шага алгоритма

В качестве иллюстрации на рис. 8.15, *а* приведено исходное бинарное изображение, на рис. 8.15, *б* показано это же изображение после выполнения первого шага алгоритма (т. е. дилатации), а на рис. 8.15, *в* приведен результат выполнения всей операции закрытия, т. е. второго заключительного шага этой операции (эрозии).

Операции бинарного открытия и закрытия могут быть *n*-кратными. Под *n*-кратными операциями бинарного открытия и закрытия понимаются такие операции, когда вначале применяется *n* раз операция эрозии (во втором случае дилатации), а затем *n* раз операция дилатации (во втором случае эрозии). При использовании *n*-кратных операций бинарного открытия и закрытия из изображения в первом случае устраняются детали, а во втором случае — дыры, размеры которых определяются кратностью применяемых операций.

Часто, как уже было отмечено, морфологические операции над изображениями рассматривают на основе теории множеств [23, 96]. В этом случае размыкание множества L_c по примитиву **В** может быть записано в виде

$$\mathbf{L}_{c} \circ \mathbf{B} = (\mathbf{L}_{c} \ominus \mathbf{B}) \oplus \mathbf{B}.$$

Из этой записи следует, что вначале над изображением L_c выполняется эрозия по примитиву **B**, а затем над полученным результатом выполняется дилатация по тому же примитиву. Часто в качестве примитива используется массив единиц размером 3×3 .

Аналогичным образом операция замыкания множества L_c по примитиву **B** может быть представлена как

$$\mathbf{L}_{c} \bullet \mathbf{B} = (\mathbf{L}_{c} \oplus \mathbf{B}) \ominus \mathbf{B}.$$

Трактовка этого преобразования аналогична трактовке предыдущего — вначале над изображением L_c выполняется дилатация по примитиву **B**, а затем над полученным результатом выполняется операция эрозии по тому же примитиву. Размыкание и замыкание также являются двойственными операциями.

Использование даже только рассмотренных морфологических операций уже позволяет решать целый ряд задач по обработке изображений. Так, например, совместное использование морфологических операций открытия и закрытия позволяет устранить с бинарного изображения инвертирующую импульсную помеху, т. е. помеху, обращающую белые пикселы в черные и наоборот. В качестве примера проиллюстрируем устранение такой помехи с изображения отпечатка пальца, показанного на рис. 8.16. Вероятность появления импульса помехи в рассматриваемом случае составляла p = 0,08. На рисунке хорошо заметны черные точки помехи на белом фоне и белые точки на черных папиллярных линиях изображения.

Для ослабления помехи на изображении в описываемом эксперименте были последовательно применены две морфологические операции — вначале операция открытия, а затем операция закрытия. Использованный примитив представлял собой массив единиц размером 3×3. Поскольку операция открытия состоит из последовательно выполняемых операций эрозии и дилатации, то рассмотрим, как эти операции проявляются на обрабатываемом изображении. При выполнении эрозии белые точки на папиллярных линиях, обусловленные воздействием помехи, исчезают, но при этом, как и следовало ожидать, увеличиваются в своем размере черные точки помехи на белом фоне. При выполнении морфологической операции дилатации над изображением, полученным в результате эрозии, размеры черных точек помехи на белом фоне сокращаются до своих первоначальных размеров. Таким образом, в результате последовательного выполнения операций эрозии и дилатации, составляющих морфологическую операцию открытия, получается изображение, показанное на рис. 8.17.



Рис. 8.16. Зашумленное изображение отпечатка пальца



Рис. 8.17. Изображение, полученное в результате обработки зашумленного изображения морфологической операцией "открытие"

Далее, в соответствии с рассматриваемым алгоритмом, результат, полученный на предыдущем этапе, обрабатывается морфологической операцией "закрытие", которая состоит из двух последовательно выполняемых операций — операции дилатации и операции эрозии. При выполнении дилатации на белом фоне исчезают чер-

ные точки помехи, а также уменьшаются в размерах черные детали изображения. При выполнении операции эрозии, составляющей вторую часть алгоритма закрытия, размеры изображения восстанавливаются. На рис. 8.18 показан результат обработки зашумленного изображения путем последовательного применения морфологических операций открытия и закрытия. Сравнивая это изображение с зашумленным изображением, показанным на рис. 8.16, видим, что на обработанном изображении помеха отсутствует, но вместе с тем появились разрывы, обусловленные малой толщиной папиллярных линий на изображении.

Заканчивая этот раздел, отметим, что устранение инвертирующей помехи с зашумленного изображения можно также осуществить путем последовательного применения морфологических операций открытия и закрытия, но в порядке, обратном тому, который был описан ранее, т. е. вначале выполнить морфологическую операцию закрытия, а затем уже морфологическую операцию открытия. Этот случай иллюстрируется рис. 8.19 и 8.20, на которых показаны изображения, полученные после выполнения первого и второго этапов реализации этого способа.



Рис. 8.18. Изображение, полученное в результате обработки зашумленного изображения последовательным применением морфологических операций "открытие" и "закрытие"



Рис. 8.19. Изображение, полученное в результате обработки зашумленного изображения морфологической операцией "закрытие"



Рис. 8.20. Изображение, полученное в результате обработки зашумленного изображения последовательным применением морфологических операций "закрытие" и "открытие"

8.9. Морфологические операции утончения и утолщения бинарных изображений

Утончение (thin) — это морфологическая операция, в результате выполнения которой происходит утончение изображения объекта до ширины одного пиксела. Алгоритм выполнения этой операции заключается в том, что изображение циклически сканируется окном размером 3×3 пиксела. При выполнении каждого цикла для каждого положения сканирующего окна анализируются значения яркостей пикселов, попавших в это окно. В том случае, если одновременно выполняются следующие условия:

- □ значение яркости центрального пиксела равно единице;
- хотя бы один из пикселов внутри этого окна равен нулю;
- обращение центрального пиксела окна в нуль не разрывает восьмисвязную цепочку пикселов, составляющих изображение объекта,

яркость центрального пиксела устанавливается в нуль, в противном случае значение его яркости сохраняется равным единице.

Циклическое сканирование изображения продолжается до тех пор, пока толщина объекта, т. е. толщина цепочки пикселов, не станет равной одному пикселу. На рис. 8.21, *а* показано исходное изображение, на рис. 8.21, *б* — промежуточная фаза его обработки, а на рис. 8.21, *в* — изображение объекта, полученное в результате выполнения описанной операции.



Рис. 8.21. Иллюстрация выполнения морфологической операции утончения бинарного изображения: *а* — исходное изображение;
 б — промежуточная фаза обработки изображения;
 в — изображение, полученное в результате выполнения всей операции

Операцию утончения используют для предобработки изображений текста, в частности, рукописного, перед его распознаванием, поскольку эта операция позволяет устранить из изображений букв все несущественное и мешающее, например утолщения, и вместе с тем сохранить непрерывность линий. На рис. 8.22, а приведены изображения рукописного текста до его обработки посредством морфологической операцией "утончение" и после нее (рис. 8.22, б). Из рисунка видно, что после обработки каждая линия, составляющая букву, стала иметь толщину в один пиксел. При распознавании текста на таком изображении элементы букв, такие как разветвления, пересечения и т. д. распознаются гораздо проще, чем в случае, когда толщина линий букв непостоянная.

5. Retholobations cruitations rapidirency & reofige.	5. Похробованые списталь и сарейники в наруш.
найах TV шушайора: если на шушкаторе	набах TV индичающья: ест на индичаторе
РАС, сперьвательно и иминальтора, рормируется	PAC, cugebaters a inuitant of a, populy etric
HENCENARDENHING RAAL MEETHOEDING, SO MOHHO STUD.	HEULEVERGENHARD EVER WEELHDEREN LO MOHAD ELVE -
zatebre our infecretra napional & genapitrabai enc-	zavabre são repeçientra nativos o genativabai no
Terre 6 cutpony PAU 6 noutpitai cuciterre c	Terese & cityfory PALL & completent encitence c
no engriouscu beybhaitton & genationaby currency	последнощим возвраютом в депарокору систему
индинальтора. Влеето этого кандой последова-	unquiaitabala, Bueero stoco nanigai tochegaba-
Тельной чогие индинатора с поетрогным памоу-	Tersition Torue inquinatopa c noetportition premay-
roration personneur crabuters à coordination	raisettu pyrochernen crasmor & coato heatenthue
Toring na jenne, T.C. Na Kapiere. Oyenning the	Tonuy na june, r.e., xa kapare. Oyennis to
SKOHOMULO BULLERENEI;	How Burrenertui,
Raponuñepegno que PAU queochegregrego renerretreghy popunposativo questripaques incroques cupestor	Каранитерицио зля РЛИ дугообщино тенестуру Фовницобаль дниклорацией историой спрывый
rapidiustrice quittifican à coordinational pypoisies men.	Kapitalite quittifair à coatretente prover and
r- uai (metogon 6119 (um Aganafa)). 3gell,	Y- vai (mesogon 6119 (un Apanafa)). 3gecs,
a	б

Рис. 8.22. Изображения рукописного текста до обработки посредством морфологической операцией утончения (*a*) и после нее (δ)

Утолщение (thicken) — это морфологическая операция, которая является обратной по отношению к морфологической операции "утончение" со всеми вытекающими отсюда последствиями. Отметим, что алгоритм морфологической операции "утолщение" на практике реализуется редко. Обычно это делается следующим образом. Вначале изображение инвертируется, при этом области, занятые объектами, становятся черными, а области фона белыми. Затем выполняется операция утончения, которая, естественно, утончает белые области, поскольку алгоритм утончения реагирует на них как на объекты, при этом черные области утолщаются. Затем полученный результат снова инвертируется. В зависимости от вида изображений объектов описанная процедура иногда приводит к возникновению отдельных изолированных белых пикселов в обработанном изображении, поэтому заключительным этапом обработки является удаление изолированных точек, например, методом открытия.

8.10. Морфологические операции: усечение, мост и удаление центрального пиксела в конфигурациях с *Н*-связностью бинарных изображений

Морфологическая операция усечения (spur). Эта морфологическая операция является важным дополнением к другим морфологическим операциям и, в частности, к операции утончения. Рассмотрим эту операцию на примере задачи отсечения ложных ветвей. Как известно, в основе автоматического распознавания рукописных текстов лежит анализ скелета каждого символа. При получении скелета часто в результате различных причин на изображении символа возникают паразитные отростки. Пример такого изображения показан на рис. 8.23, *a*.

В случае морфологической операции усечения (*spur*) под паразитными отростками понимаются пикселы объекта, у которых только один из восьми соседних пикселов имеет интенсивность, равную единице, а остальные равны нулю. Отличие алгоритма морфологической операции усечения (*spur*) от алгоритма морфологической операции усечения (*spur*) от алгоритма морфологической операции удаления одиночных точек (*clean*) состоит в том, что содержимое сканирующего окна сравнивается не с одним шаблоном, как в предыдущем случае, а последовательно с группой шаблонов, приведенных на рис. 8.24.





Рис. 8.23. Исходное изображение (*a*), изображение, полученное в результате выполнения морфологической операции усечения (б)



Рис. 8.24. Шаблоны, используемые для выполнения морфологической операции усечения (*spur*)

Если заполнение сканирующего окна совпадает с одним из шаблонов, показанных на рис. 8.24, то центральный пиксел окна обнуляется, т. е. преобразуется в фоновый пиксел, в противном случае его интенсивность не изменяется, т. е.

$$L_{8} = \begin{cases} 0, & \text{если для одного из шаблонов } \sum_{k=0}^{8} (L_{k} - L_{\text{ш}k})^{2} = 0, \\ L_{8} & \text{в противном случае.} \end{cases}$$
(8.6)

На рис. 8.23, *б* показан результат обработки изображения, представленного на рис. 8.23, *a*, морфологической операцией *spur*. В целях демонстрации выполнения морфологической операции усечения (*spur*) на рис. 8.25, *a* приведена также группа исходных тестовых изображений, а на рис. 8.25, *б* показан результат их обработки рассматриваемой морфологической операцией.



Рис. 8.25. Иллюстрация к морфологической операции усечения *(spur)*: *а* — исходные тестовые изображения; *б* — изображения, полученные в результате выполнения данной операции

Морфологическая операция "мост" (bridge) соединяет ранее разъединенные пикселы фрагментом фона толщиной в один пиксел. Рисунок 8.26 демонстрирует результат выполнения этой операции. На рис. 8.26, *а* показан ряд исходных бинарных изображений объектов, а на рис. 8.26, *б* приведены изображения этих же объектов, но после их преобразования операцией "мост".



Рис. 8.26. Иллюстрация к морфологической операции "мост" (*bridge*): *а* — исходные изображения; *б* — изображения, полученные в результате выполнения данной операции



Рис. 8.27. Шаблоны, используемые для выполнения морфологической операции "мост" (bridge)

На рис. 8.27 приведена группа шаблонов, которые были использованы при выполнении этой операции. Если заполнение сканирующего окна совпадает с одним из шаблонов, показанных на рис. 8.27, то центральный пиксел окна приобретает яркость, равную единице, в противном случае его интенсивность не изменяется, т. е.

$$L_8 = \begin{cases} 1, & \text{если для одного из шаблонов } \sum_{k=0}^8 (L_k - L_{\text{ш}k})^2 = 0, \\ L_8 & \text{в противном случае.} \end{cases}$$

Морфологическая операция hbreak удаляет центральные пикселы в конфигурациях с *H*-связностью. Алгоритм этой морфологической операции при использовании метода шаблонов может быть описан следующим образом. Обрабатываемое изображение сканируется окном. Если заполнение сканирующего окна совпадает с одним из шаблонов, то центральный пиксел окна обнуляется, т. е. преобразуется в фоновый пиксел, в противном случае его интенсивность не изменяется, т. е.

$$L_8 = \begin{cases} 0, & \text{если для одного из шаблонов } \sum_{k=0}^8 (L_k - L_{\text{ш}k})^2 = 0, \\ L_8 & \text{в противном случае.} \end{cases}$$

На рис. 8.28, *а* показан ряд исходных бинарных изображений объектов, а на рис. 8.28, *б* приведены изображения этих же объектов, но после их преобразования операцией *hbreak*.





Рис. 8.28. Иллюстрация к морфологической операции *hbreak*: *a* — исходные изображения; *б* — изображения, полученные в результате выполнения данной операции;

в — группа шаблонов, которые были использованы при выполнении этой операции

На рис. 8.28, в приведена группа шаблонов, которые были использованы при выполнении этой операции.

В заключение этого раздела отметим, что применение шаблонов хотя и связано, как правило, с необходимостью использовать их большие наборы, вместе с тем этот метод предоставляет очень широкие возможности конструирования различных морфологических операций. Так, выбирая тот или иной набор шаблонов и используя их для обработки изображения, можно исключать из него или, наоборот, выделять на нем самые разнообразные конфигурации пикселов, образующих прямые линии, углы, разветвления и т. д. При этом в ряде случаев могут потребоваться окна, размер которых может существенно превышать 3×3 пиксела, например, при исключении (обнаружении) кривых линий и более сложных фрагментов. Понятно, что время обработки в этом случае возрастает.

8.11. Заполнение (заливка) областей бинарных изображений

Алгоритм заполнения области внутри замкнутого 8-связного контура (*fill*) включает в себя следующие шаги. Вначале внутри замкнутого контура задается точка, яркость которой по умолчанию равна яркости окружающего ее контура.

Рисунок 8.29, *а* поясняет сказанное. На этом рисунке в виде сетки показаны отдельные пикселы изображения, белым цветом изображен контур, а также точка, помещенная внутри контура. Затем в режиме итерации, начиная с этой точки, выполняется четырехсвязная дилатация. При этом в процессе обработки каждого пиксела исходного изображения значение его яркости устанавливается в единицу только при выполнении следующих двух условий, а именно:

- если хотя бы один из пикселов в его четырехсвязной окрестности равен единице;
- если его яркость до этого равнялась нулю, т. е. если до преобразования он принадлежал фону, а не контуру.

На рис. 8.29, *б* показан результат дилатации после первой итерации. На рис. 8.29, *в* приведен результат дилатации после четвертого шага итерации. Из рисунка видно, что при данном алгоритме дилатация не распространяется за пределы контура. Дилатация продолжается до полного заполнения области (рис. 8.29, *г*). Описанный алгоритм с соответствующими изменениями может быть применен и для заполнения областей в полутоновых и цветных изображениях.



Рис. 8.29. Иллюстрация к алгоритму заполнения областей бинарных изображений: *а* — изображение с точкой, помещенной внутри контура; *б* — результат дилатации после первой итерации; *в* — результат дилатации после четвертого шага; *г* — результат обработки (полное заполнение контура)

8.12. Построение скелета объекта на бинарном изображении

Одним из способов описания объекта является выделение из него так называемого *скелета* (морфологическая операция *skel*). Существует ряд методов получения *скелета* объекта, из которых наиболее наглядным, на наш взгляд, является метод *степного пожара*. Этот метод выделяет также дополнительную информацию, которая вместе со скелетом дает возможность восстановить описываемый объект. Сначала в целях наглядности объяснения рассмотрим случай, когда изображение объекта задано в аналоговом виде. Представим себе изображение объекта в виде области, покрытой травой, а изображение фона — в виде области, где трава отсутствует. На рис. 8.30 объект показан в виде прямоугольника. Будем считать, что огонь возник по всему периметру объекта одновременно.



Рис. 8.30. Иллюстрация к морфологической операции skel

В этом случае огонь будет распространяться с постоянной скоростью в направлении центра объекта. На рис. 8.30 штриховой линией показаны фронты огня, распространяющиеся от каждой из четырех границ объекта. При этом фронт огня, распространяющийся от одной границы объекта, будет встречаться с фронтом огня, распространяющимся от другой границы, и эти два фронта огня будут гасить друг друга. Линии гашения огня образуют скелет объекта, который на рис. 8.30 показан жирными линиями.

При построении скелета объекта вводят в рассмотрение функцию гашения. Функция гашения представляет собой зависимость минимального расстояния d от точки скелета с координатами x и y до границы объекта. Располагая скелетом объекта и функцией гашения, можно восстановить изображение объекта следующим образом. Описывая окружности радиусом d из каждой точки скелета и проводя к ним касательную, мы получаем точное изображение объекта. Этот же результат можно получить, используя морфологическую MATLAB-функцию skel с параметром inf.

Ранее был рассмотрен способ построения скелета объекта применительно к аналоговому изображению. Построение скелета цифровых изображений принципиально не отличается от рассмотренного ранее метода, однако имеет специфические особенности, связанные с применением эвклидовой метрики. Подробные сведения по этому вопросу приведены в литературе [23, 150, 151].

8.13. Производные морфологические операции над бинарными изображениями

Ряд морфологических операций могут быть реализованы в результате нахождения разности двух изображений, одно из которых является исходным, а второе было получено в результате применения к исходному изображению одной из морфологических операций. Приведем примеры.

Морфологическая операция выделения периметра (remove) реализуется следующим образом. Вначале двоичное изображение объекта $L_1(k,n)$ подвергается эрозии, в результате которой получается изображение $L_2(k,n)$. После этого из изображения $L_1(k,n)$ вычитается изображение $L_2(k,n)$. Получившееся в результате вычитания распределение яркости будет представлять собой изображение границы (периметра) $L_{rp}(k,n)$ толщиной в один пиксел:

$$L_{rp}(k,n) = L_1(k,n) - L_2(k,n).$$



Рис. 8.31. Иллюстрация к морфологической операции выделения периметра: *а* — исходное изображение; *б* — изображение периметра

Исходное изображение и результат его обработки посредством морфологической операции выделения периметра представлены на рис. 8.31, *а* и б соответственно. Этот же результат может быть получен непосредственно путем применения к исходному изображению $L_1(k,n)$ MATLAB-операции remove, удаляющей внутренние пикселы объекта, у которых соседние пикселы по горизонтали и вертикали равны 1.

Этот подход был описан в *разд*. 8.5, как пример использования логической апертуры. Можно также получить этот результат, применив метод шаблонов.

На рис. 8.32 приведена группа шаблонов, которые используются при непосредственном выполнении операции геточе и формулы (8.6).



Рис. 8.32. Шаблоны, используемые для выполнения морфологической операции *remove*

Этот же результат может быть получен и на основе теории множеств и записан следующим образом:

$$\beta(\mathbf{L}_{c}) = \mathbf{L}_{c} \setminus (\mathbf{L}_{c} \ominus \mathbf{B}),$$

где $\beta(\mathbf{L}_c)$ — граница множества \mathbf{L}_c , \ — символ вычитания из множества \mathbf{L}_c множества $(\mathbf{L}_c \ominus \mathbf{B})$, т. е. результата операции эрозии множества \mathbf{L}_c по примитиву **B**. В том случае, если применить *n*-кратную эрозию, толщина границы составит *n* пикселов.

Морфологическая операция "низ шляпы" (bothat) реализуется путем вычитания исходного изображения из его бинарного закрытия $(\mathbf{L}_{c} \bullet \mathbf{B}) \setminus \mathbf{L}_{c}$.

Морфологическая операция "верх шляпы" (tophat) реализуется путем вычитания из исходного изображения его бинарного открытия $L_c \setminus (L_c \circ B)$.

Более подробное рассмотрение двух последних морфологических операций приведено в *разд. 8.16* применительно к полутоновым изображениям.

8.14. Использование "интегральных" шаблонов при выполнении морфологических операций над бинарными изображениями

Особенностью рассмотренного ранее метода выполнения морфологических операций с использованием шаблонов является то, что, как правило, он требует для своей реализации применения большого количества шаблонов. Так, например, для реализации морфологической операции spur требуется 8 шаблонов, для операции bridge необходимо использовать 30 шаблонов, для операции remove — 16 шаблонов и т. д. Исключение составляют только самые простые операции, например, операция удаления одиночных 8-связных точек (clean). В ряде случаев этого можно избежать, если модифицировать этот метод. Модифицированный вариант этого метода, который мы назовем методом "интегральных" шаблонов, заключается в следующем. В этом методе используется один-единственный шаблон, паттерн (от англ. pattern) которого представляет собой изображение удаляемого объекта, например отрезок прямой заданной длины и ориентации, окружность заданного диаметра и т. д. На рис. 8.33 в качестве примера приведены два таких шаблона. Шаблон, показанный на рис. 8.33, а предназначен для удаления из изображения отрезков линий длиной в 5 пикселов, ориентированных к направлению строк под углом 45°, а на рис. 8.33, б показан шаблон, предназначенный для удаления из изображения окружностей заданного диаметра.





Рис. 8.33. Шаблоны для удаления из изображения определенных объектов: *а* — отрезков линий длиной в 5 пикселов, ориентированных к направлению строк под углом 45°; *б* — окружностей заданного диаметра

При реализации модифицированного варианта метода обрабатываемое изображение сканируется окном, размеры которого совпадают с размерами шаблона. Как и ранее, область изображения, перекрываемая сканирующим окном, попиксельно сравнивается с шаблоном. Отличие модифицированного варианта метода состоит в том, что, если интенсивности пикселов изображения, оказавшихся в пределах сканирующего окна, совпадают с интенсивностями соответствующих пикселов шаблона, все пикселы изображения, оказавшиеся в пределах сканирующего окна, а не только центральный, как это было ранее, обнуляются, т. е. преобразуются в фоновые пикселы. В противном случае значения интенсивностей пикселов изображения, оказавшихся в пределах окна, сохраняются. Достоинством этого метода является простота и наглядность. Его удобно применять в тех случаях, когда из изображения требуется удалить объекты, форма, размеры и ориентация которых заранее известны.

8.15. Морфологические операции дилатации, эрозии, открытия и закрытия полутоновых изображений

В полутоновых цифровых изображениях каждый пиксел как элемент множества, представляющего изображение, описывается тремя координатами, две из них описывают координаты пиксела в пространстве, а третья представляет дискретное значение яркости.

Распространим рассмотренные ранее базовые морфологические операции эрозии и дилатации на полутоновые изображения.

Морфологическая операция дилатации полутонового изображения выполняется следующим образом. Обрабатываемое изображение сканируется окном, размеры которого равны размерам примитива. При каждом положении окна на обрабатываемом изображении вычисляются суммы путем сложения интенсивности каждого пиксела, оказавшегося в пределах окна, с интенсивностью накрывающего его элемента примитива. Затем среди полученных сумм находится наибольшая. Значение этой наибольшей суммы присваивается пикселу обработанного изображения, расположенному в центре сканирующего окна. В результате дилатации, при условии, что все значения элементов примитива положительные, все пикселы полутонового изображения становятся светлее. Степень этого осветления зависит от значений яркости пикселов изображения, а также от формы и значений интенсивностей элементов примитива.

Используя ранее рассмотренный подход к определению морфологических операций на основе теории множеств, приведем математическую запись полутоновой дилатации изображения L_c по примитиву B, которая имеет вид

$$L_{c} \oplus B = \max \left\{ L_{c} \left(k - k', n - n' \right) + B \left(k', n' \right) | \left(k - k', n - n' \right) \in D_{L}; \left(k', n' \right) \in D_{B} \right\},\$$

где D_L и D_B — области, в которых определены изображение и примитив соответственно, k, k', n и n' — номера строк и столбцов элементов массивов изображения и примитива. Обратим внимание на то, что в данном случае L_c и B представляют собой функции, а не множества.









Рис. 8.34. Иллюстрация к морфологическим операциям эрозии и дилатации полутонового изображения: *а* — исходное изображение; *б* — примитив; *в* — результат дилатации исходного изображения; *г* — результат эрозии исходного изображения

На рис. 8.34, *а* в качестве примера показано исходное изображение, на рис. 8.34, *в* — результат его дилатации, а на рис. 8.34, *б* приведен примитив, который при этом был использован.

Морфологическая операция эрозии полутонового изображения выполняется похожим образом. Обрабатываемое изображение также сканируется окном, размеры которого должны быть равны размерам примитива. При каждом положении окна на обрабатываемом изображении вычисляются разности интенсивностей пикселов, оказавшихся в пределах окна, и интенсивностей элементов примитива, накрывающих эти пикселы, а затем среди полученных разностей находится наименьшая. Значение этой наименьшей разности присваивается пикселу обработанного изображения, расположенному в центре сканирующего окна. В результате эрозии, при условии, что все значения элементов примитива положительные, яркость светлых деталей полутонового изображения, если их площадь меньше площади примитива, ослабляется. Степень этого ослабления зависит как от значений яркости пикселов изображения, окружающих эти детали, так и от формы самого примитива и значений интенсивностей его элементов. На рис. 8.34, *г* в качестве примера показан результат эрозии исходного изображения, а на рис. 8.34, *б* приведен примитив, который при этом был использован.

Используя ранее рассмотренный подход к определению морфологических операций на основе теории множеств, приведем математическую запись полутоновой эрозии изображения L_c по примитиву B, которая имеет вид

$$L_{c} \ominus B = \min \{ L_{c}(k+k', n+n') - B(k', n') | (k+k', n+n') \in D_{L}; (k', n') \in D_{B} \}.$$

Морфологическая операция открытия полутонового изображения, как и в случае бинарного изображения, выполняется следующим образом: вначале полутоновое изображение подвергается операции эрозии, а затем полученный результат подвергается операции с использованием одного и того же примитива.

При подходе к описанию морфологической операции открытия полутонового изображения L_c по примитиву *B* ее математическое описание имеет вид [23, 96]

$$L_{\rm c} \circ B = (L_{\rm c} \ominus B) \oplus B$$

Морфологическая операция закрытия полутонового изображения выполняется следующим образом. Вначале полутоновое изображение подвергается операции дилатации, а затем полученный результат подвергается операции эрозии с использованием одного и того же примитива. Аналогично при описании морфологической операции открытия полутонового изображения L_c по примитиву *В* имеем [23, 96]

$$L_{\rm c} \bullet B = (L_{\rm c} \oplus B) \ominus B.$$

На рис. 8.35 в качестве примера приведены результат открытия (рис. 8.35, *a*) и результат закрытия (рис. 8.35, *б*) изображения, показанного на рис. 8.34, *a*.



а

б

Рис. 8.35. Результаты выполнения морфологических операций открытия (*a*) и закрытия (*б*)

На практике *операция открытия* применяется для удаления из полутонового изображения небольших по сравнению с размерами примитива светлых деталей. Достигается это благодаря тому, что на первом этапе этой операции, когда выполняется эрозия, небольшие светлые детали удаляются из изображения, при этом несколько уменьшается яркость всего изображения. На втором этапе операции, когда выполняется дилатация, яркость сохранившихся на изображении деталей восстанавливается, но при этом детали, удаленные на первом этапе, не восстанавливаются.

Операция закрытия на практике применяется для удаления из полутонового изображения небольших по сравнению с размерами примитива темных деталей. Достигается это благодаря тому, что на первом этапе этой операции, когда выполняется дилатация, небольшие темные детали удаляются из изображения, при этом несколько увеличивается яркость всего изображения. На втором этапе операции, когда выполняется эрозия, яркость сохранившихся на изображении деталей восстанавливается, но при этом детали, удаленные на первом этапе, не восстанавливаются.

8.16. Морфологические операции *tophat*, *bothat* и морфологический градиент при фильтрации полутоновых изображений

Рассмотрим морфологические функции *tophat* и *bothat*, которые применяют при фильтрациях полутоновых изображений.

Морфологическая функция *tophat* (верх шляпы) представляет собой вычитание из обрабатываемого изображения L_c результата его открытия $(L_c \circ B)$, при этом результат обработки может быть записан следующим образом:

$$L_{th} = L_{\rm c} - \left(L_{\rm c} \circ B\right).$$

Поскольку при открытии из исходного изображения удаляются небольшие светлые детали, то результатом операции *tophat* является выделение из исходного изображения этих небольших светлых деталей, которые при дальнейших этапах обработки изображений могут быть использованы путем их включения в обрабатываемое изображение с тем или иным весовым коэффициентом. На рис. 8.36 приведены исходное изображение (рис. 8.36, *a*) и результат его обработки рассматриваемой операцией (рис. 8.36, δ).

Морфологическая функция *bothat* (низ шляпы) представляет собой вычитание исходного изображения L_c из результата его закрытия $(L_c \bullet B)$, при этом результат обработки может быть записан следующим образом:

$$L_{bh} = (L_{c} \bullet B) - L_{c}.$$



Рис. 8.36. Иллюстрация выполнения морфологической операции *tophat*: *а* — исходное изображение; *б* — результат обработки



Рис. 8.37. Иллюстрация выполнения морфологической операции *bothat*: *а* — исходное изображение; *б* — результат обработки

Поскольку при закрытии из исходного изображения удаляются небольшие темные детали, то результатом операции *bothat* является выделение из исходного изображения этих небольших темных деталей, которые при дальнейших этапах обработки изображений могут быть использованы путем их включения в обрабатываемое изображение с тем или иным весовым коэффициентом. На рис. 8.37 приведены исходное изображение (рис. 8.37, a) и результат его обработки рассматриваемой операцией (рис. 8.37, b).



Рис. 8.38. Исходное изображение (*a*), изображение с увеличенными резкостью световых границ и контрастом мелких деталей (б)

На практике использование морфологических операций *tophat* и *bothat* совместно с исходным изображением L(k,n) в соответствии со следующим выражением

$$L_{\rm Bbix} = L_{\rm c} + \left[L_{\rm c} - \left(L_{\rm c} \circ B \right) \right] - \left[\left(L_{\rm c} \bullet B \right) - L_{\rm c} \right],$$

позволяет увеличить резкость световых границ и контраст его мелких деталей. На рис. 8.38 приведены исходное изображение (рис. 8.38, a) и результат его обработки описанным способом (рис. 8.38, δ) при использовании примитива, приведенного на рис. 8.34, δ .

Достоинством этого метода обработки по сравнению с рассмотренным ранее в *разд. 5.11* методом нерезкого маскирования является то, что в рассматриваемом случае на изображении отсутствуют окантовки [50]. Возможности этого метода можно расширить, если суммирование и вычитание результатов операций *tophat* и *bothat* производить с весовыми коэффициентами.

Рассмотрим еще один метод морфологической фильтрации, а именно морфологический градиент. Эта морфологическая операция реализуется путем вычитания из результата дилатации исходного изображения результата его эрозии, выполненных с использованием одного и того же примитива. На рис. 8.39 приведены исходное изображение (рис. 8.39, a) и его морфологический градиент (рис. 8.39, b) при использовании примитива размером 3×3 пиксела, заполненного единицами.

Одной из возможных областей применения данного типа фильтрации является художественная графика, где этот прием может оказаться достаточно выразительным инструментом.



Рис. 8.39. Здесь: *а* — исходное изображение; *б* — результат его фильтрации методом "морфологический градиент"

В заключение заметим, что морфологическая обработка предоставляет широкие возможности для реставрации и улучшения изображений, поскольку она имеет дело непосредственно с формой тех или иных деталей изображения. Подбирая соответствующие примитивы, можно избирательно влиять на обработку избранных фрагментов обрабатываемых изображений.

8.17. Фильтрация полутоновых изображений посредством логической апертуры

Эффективным методом фильтрации импульсных помех на изображении является их фильтрация логической апертурой. Существо этого метода состоит в том, что в процессе фильтрации вычисляется разность яркости каждого из пикселов исходного изображения и средней яркостью группы окружающих его пикселов, и если абсолютная величина этой разности превышает установленный порог, то в выходной массив вместо яркости данного пиксела записывается средняя яркость окружающих его пикселов. В противном случае в выходной массив записывается первоначальное значение яркости пиксела. В соответствии с этим алгоритмом яркость пикселов в профильтрованном изображении определяется выражением

$$L_{\Omega}(k,n) = \begin{cases} L(k,n) \text{ при } \left| L(k,n) - \overline{L} \right| < P, \\ \overline{L} \quad \text{при } \left| L(k,n) - \overline{L} \right| \ge P, \end{cases}$$

где

$$\overline{L} = \frac{1}{8} \left[\sum_{k'=-1}^{1} \sum_{n'=1}^{1} L(k+k', n+n') - L(k, n) \right].$$

L(k-1, n-1)	L(k-1, n)	L(k-1, n+1)	
L(k, n-1)	L(k, n)	L(k, n+1)	
L(k+1, n-1)	L(k+1, n)	L(k+1, n+1)	

Рис. 8.40. Пояснение фильтрации логической апертурой



Рис. 8.41. Фильтрация изображения логической апертурой: *а* — изображение с импульсной помехой; *б* — изображение после фильтрации логической апертурой

Рисунок 8.40 поясняет изложенное ранее.

Фильтрация изображений посредством логической апертуры оказывается особенно эффективной при ослаблении импульсных помех, возникающих в цифровых каналах связи. Эффективность рассмотренного метода демонстрирует рис. 8.41. На рис. 8.41, *а* показано изображение, искаженное импульсной помехой, а на рис. 8.41, *б* — это же изображение, но после его фильтрации логической апертурой.

Метод фильтрации изображений логической апертурой легко распространяется на случай движущихся изображений. В этом случае при вычислении средней яркости окружающих пикселов необходимо использовать соответствующие пикселы как в предыдущих, так и в последующих кадрах по отношению к фильтруемому кадру изображения.

При фильтрации цветных изображений логической апертурой каждая цветовая составляющая изображения должна фильтроваться отдельно. Глава 9



Сегментация изображений

9.1. Сегментация изображений на отдельные области

Целью сегментации изображений является выделение из них, в зависимости от решаемой задачи, отдельных деталей, областей или изображаемых объектов.

Областями применения этого вида обработки являются задачи, связанные с анализом и опознаванием изображений. Типичным примером использования методов сегментации изображений на отдельные области может служить сегментация изображений, полученных с космических аппаратов, осуществляющих дистанционное зондирование Земли. Применение этих методов позволило прогнозировать урожаи зерновых в районах земного шара, являющихся их основными производителями, контролировать лесное хозяйство, городское землепользование, а также использовать в других областях человеческой деятельности. Другим примером сегментации является выделение изображений актеров, снимаемых на однородном цветном фоне, обычно синем, при реализации рир-проекции. При сегментации изображения должны выполняться следующие условия:

- □ в результате сегментации изображение разделяется на ряд областей таким образом, чтобы каждый его пиксел входил бы в одну из областей;
- области, которые получаются в результате сегментации, не должны пересекаться, другими словами, каждый пиксел изображения может входить только в одну область;
- все пикселы, отнесенные к одной области, должны обладать одними и теми же свойствами. Например, яркость или цвет пикселов, отнесенных к одной области, должны лежать в пределах, определенных для данной области.

Обычно алгоритмы сегментации изображений используют следующие базовые свойства изображения:

однородность выделяемых областей в отношении признака, по которому выполняется сегментация, например, однородность по яркости, по цвету, или по какому-нибудь другому признаку;

- наличие скачкообразного изменения какого-нибудь признака, например, скачка яркости, отделяющего одну область изображения от другой;
- изменение во времени каких-либо характеристик изображения, обусловленных, например, его движением.

В настоящее время разработан ряд методов сегментации изображений, некоторые из них рассматриваются ниже.

В качестве признаков, на основе которых пикселы относятся к той или иной области сегментации, часто используют:

- яркость пиксела в случае не цветных изображений;
- 🛛 цвет;
- 🗖 дисперсию,
- а также ряд других признаков.

9.2. Обнаружение на изображении точек и отрезков прямых линий

Обнаружение отдельных изолированных точек на изображении является одной из простейших операций. Для обнаружения изолированной точки изображение сканируется окном, размеры которого равны 3×3 пиксела, при этом для каждого положения окна вычисляется сумма

$$L_{\Sigma} = 8L_8 - \left(L_0 + L_1 + L_2 + L_3 + L_4 + L_5 + L_6 + L_7\right), \tag{9.1}$$

где L_0 , L_1 , L_2 , L_3 , L_4 , L_5 , L_6 , L_7 , L_8 — значения яркостей пикселов, оказавшихся в пределах окна, показанного на рис. 9.1.

L_0	L_1	L_2
L_7	L_8	L_3
L_6	L_5	L_4

Рис. 9.1. Сканирующее окно

Затем абсолютное значение вычисленной суммы L_{Σ} сравнивается с порогом P. В том случае, если это значение превышает порог или равно ему, т. е. если

$$\left|L_{\Sigma}\right| \geq P$$
,

считается, что на изображении обнаружена изолированная точка, в противном случае считается, что в данной позиции изолированная точка отсутствует.
Немногим более сложной процедурой является обнаружение линий толщиной в один пиксел. При обнаружении линий изображение также сканируется окном размером 3×3 пиксела, но при этом вычисляется уже не одна сумма, а четыре, поскольку обнаруживаемые отрезки линий, оказавшихся в пределах окна, могут иметь четыре ориентации. Этими суммами будут

$$\begin{split} &L_{\Sigma 1} = 2 \big(L_3 + L_7 + L_8 \big) - \big(L_0 + L_1 + L_2 + L_4 + L_5 + L_6 \big), \\ &L_{\Sigma 2} = 2 \big(L_0 + L_4 + L_8 \big) - \big(L_1 + L_2 + L_3 + L_5 + L_6 + L_7 \big), \\ &L_{\Sigma 3} = 2 \big(L_1 + L_5 + L_8 \big) - \big(L_0 + L_2 + L_3 + L_4 + L_6 + L_7 \big), \\ &L_{\Sigma 4} = 2 \big(L_2 + L_6 + L_8 \big) - \big(L_0 + L_1 + L_3 + L_4 + L_5 + L_7 \big). \end{split}$$

Далее необходимо выяснить, проходит ли вообще какая-нибудь линия через центр окна, а если проходит, то какая из четырех возможных: горизонтальная, под углом к направлению строки 45°, вертикальная или линия под углом к строке 135°. С этой целью вначале абсолютные значения вычисленных сумм сравниваются между собой, и определяется та сумма $L_{\Sigma i}$, для которой выполняется неравенство

$$|L_{\Sigma i}| > |L_{\Sigma j}|$$
 для $j = 1, 2, 3, 4$ при условии, что $j \neq i$.

Затем абсолютное значение суммы $L_{\Sigma i}$ сравнивается с порогом P:

$$|L_{\Sigma i}| \ge P$$

В том случае, если это значение превышает порог, принимается решение о принадлежности анализируемого пиксела, расположенного в центре окна линии, и он каким-либо способом выделяется на изображении, в противном случае считается, что линия не проходит через центр окна. Описанная методика может быть применена и для обнаружения линий другой ориентации, однако при этом необходимо использовать окна больших размеров.

9.3. Обнаружение перепадов яркости на изображении

Перепады яркости на изображениях могут представлять собой как границы объектов, так и градиентные изменения яркости, вызванные плавными изменениями освещенности, коэффициентов отражения или углового положения оболочек самих объектов. При сегментации представляют интерес перепады яркости, обусловленные границами объектов, поскольку целью сегментации, как правило, является выделение объектов на изображениях [73, 84]. Перепады этого типа в идеале представляют собой резкие скачки яркости, однако на практике чаще приходится иметь дело с размытыми световыми границами вследствие апертурных искажений вносимых системой. На практике применяют два метода обнаружения перепадов яркости, образуемых световыми границами: *градиентный метод* и *метод с использованием лапласиана*.





На рис. 9.2, *а* в качестве примера показан перепад яркости L_c в направлении, перпендикулярном вертикальной границе на аналоговом изображении. На рис. 9.2, *б* представлена зависимость градиента этого перепада от координаты *x*, который в рассматриваемом случае вырождается в первую производную L_c по *x*. На рис. 9.2, *в* приведена зависимость лапласиана этого перепада от *x*, который в рассматриваемом случае вырождается во вторую производную L_c по *x*. Обратим внимание на то, что ни градиент, ни лапласиан непосредственно не указывают на положение средины перепада яркости: градиент отличен от нуля на протяжении всего перепада, а лапласиан отличен от нуля в начале и в конце перепада. В том случае, если необходимо определить точки, расположенные на срединах перепадов яркости, например, для того чтобы провести через них контур, нужно принять во внимание следующее. В первом случае эти точки будут располагаться на средине интервала, где градиент отличен от нуля, а во втором случае положение этих точек определится местом пересечения нулевого уровня линиями, соединяющими точки, в которых значения лапласиана отличны от нуля. На рис. 9.2, г показан идеально резкий перепад яркости для вертикальной границы на цифровом изображении, а на рис. 9.2, ∂ размытый перепад яркости для вертикальной границы на этом же изображении. На рис. 9.2, е показан градиент перепада яркости размытой вертикальной границы на цифровом изображении. Как и в случае аналогового изображения, здесь градиент указывает лишь на те участки изображения, где имеет место перепад. В том случае, если мы хотим определить положение контуров, разделяющих области, между которыми имеет место перепад яркости, как и в предыдущем случае необходимы хотя и не сложные, но дополнительные вычисления. Сказанное справедливо и применительно к использованию для этой цели лапласиана.

Рассмотрим теперь методы нахождения градиента и лапласиана.

По определению *градиент* аналогового изображения $L_c(x, y)$ в точке x, y представляет собой вектор **G**, ориентированный в направлении максимального изменения яркости, модуль которого равен

$$\left|\mathbf{G}\right| = \sqrt{G_x^2 + G_y^2} , \qquad (9.2)$$

где $G_x = \partial L_c(x, y)/\partial x$, $G_y = \partial L_c(x, y)/\partial y$ — компоненты этого вектора. Этот вектор с осью *x* составляет угол $\alpha(x, y)$, который, как известно, равен

$$\alpha(x, y) = \operatorname{arctg}(G_y/G_x).$$

В случае дискретных изображений частные производные G_x и G_y определяются приближенно одним из следующих способов.

Перекрестный градиентный оператор Робертса

Согласно этому способу дискретное изображение сканируется окном размером 2×2 пиксела, которое показано на рис. 9.3.

$L_{\rm c}(k,n)$	$L_{\rm c}(k, n+1)$
$L_{\rm c}(k+1,n)$	$L_{\rm c}(k+1,n+1)$

Рис. 9.3. Сканирующее окно при использовании оператора Робертса

Для каждого положения окна вычисляются значения G_x и G_y по формулам

$$G_{x} = L_{c}(k+1, n+1) - L_{c}(k, n), \qquad (9.3)$$

$$G_{y} = L_{c}(k+1, n) - L_{c}(k, n+1), \qquad (9.4)$$

а затем по формуле для пиксела, расположенного в k-й строке и в n-м столбце, вычисляется модуль градиента.

Градиентный оператор Превитта

В отличие от оператора Робертса при использовании оператора Превитта дискретное изображение сканируется окном размером 3×3 пиксела, которое показано на рис. 9.1, при этом для каждого положения окна значения G_x и G_y вычисляются по формулам

$$G_x = (L_2 + L_3 + L_4) - (L_0 + L_6 + L_7), \qquad (9.5)$$

$$G_{y} = (L_{4} + L_{5} + L_{6}) - (L_{0} + L_{1} + L_{2}), \qquad (9.6)$$

где по-прежнему L_0 , L_1 , L_2 , L_3 , L_4 , L_5 , L_6 , L_7 , L_8 — значения яркостей пикселов, оказавшихся в пределах окна, показанного на рис. 9.1. Далее вычисляется модуль градиента по формуле (9.2) для пиксела изображения, находящегося в центре сканирующего окна.

Градиентный оператор Собела

Оператор Собела отличается от оператора Превитта тем, что при вычислении компонентов градиента G_x и G_y значения отсчетов яркости L_1 , L_3 , L_5 и L_7 берутся с весовым коэффициентом 2, а значения G_x и G_y для каждого положения окна вычисляются по формулам

$$G_x = (L_2 + 2L_3 + L_4) - (L_0 + L_6 + 2L_7), \qquad (9.7)$$

$$G_{y} = (L_{4} + 2L_{5} + L_{6}) - (L_{0} + 2L_{1} + L_{2}).$$
(9.8)

После этого по формуле (9.2) вычисляется модуль градиента для пиксела изображения, находящегося в центре сканирующего окна. Увеличение веса отсчетов яркости L_1 , L_3 , L_5 и L_7 при использовании оператора Собела позволяет несколько уменьшить влияние шума на результат вычисления градиента.

Сравнивая три рассмотренных метода нахождения градиента, отметим, что в случае применения метода Робертса интервал, на котором градиент отличен от нуля, на один отсчет уже, чем в случае применения метода Превитта или метода Собела. Достоинством же методов Превитта и Собела является симметрия сканирующих окон относительно анализируемого пиксела. Обратим внимание на то, что во всех рассмотренных случаях на участках изображения, где яркость постоянна, вычисленные значения компонентов градиента G_x и G_y равны нулю.

На практике часто в целях сокращения вычислительных затрат модуль градиента вычисляют не по формуле (9.2), а по приближенной формуле

$$\left|\mathbf{G}\right| \approx \left|G_{x}\right| + \left|G_{y}\right|. \tag{9.9}$$

Лапласиан аналогового изображения $L_{c}(x, y)$ в точке x, y определяется выражением

$$\Delta(x, y) = \partial^2 L_{\rm c}(x, y) / \partial x^2 + \partial^2 L_{\rm c}(x, y) / \partial y^2 .$$

Лапласиан дискретного изображения $L_{c}(k,n)$ для пиксела, расположенного в k-й строке и в n-м столбце определяется приближенно одним из следующих двух способов

$$\Delta(k,n) = 4L_8 - (L_1 + L_3 + L_5 + L_7)$$
(9.10)

или

$$\Delta(k,n) = 8L_8 - (L_0 + L_1 + L_2 + L_3 + L_4 + L_5 + L_6 + L_7).$$
(9.11)

В первом случае результат оказывается инвариантным к повороту на углы, кратные 90°, во втором случае инвариантным к повороту на углы 45°.

Недостатком метода обнаружения перепадов яркости с использованием лапласиана по сравнению с градиентным методом является его большая чувствительность к шуму, который всегда имеется на изображениях.

9.4. Методы выделения контуров на изображениях

Как известно, семантические изображения легко и быстро различаются зрительной системой человека. Объясняется это тем, что различение семантических изображений, как уже отмечалось в *главе 3*, происходит по контурам, которые являются наиболее устойчивыми признаками изображений наблюдаемых объектов. Это, в частности, явилось причиной широкого распространения контурных изображений как в изобразительном искусстве, так и в технических приложениях. Когда художник изображает реальные предметы в виде контурного рисунка, определение контуров рисуемых объектов не вызывает у него каких-либо затруднений, зрительная система легко справляется с этой задачей. Однако при попытках автоматизации этого процесса, когда мы поручаем эту работу компьютеру, возникают проблемы. Несмотря на то, что было разработано много алгоритмов, направленных на решение данной проблемы, она до настоящего времени так и остается не решенной до конца. Рассмотрим некоторые из этих алгоритмов.

Процедура выделения контуров включает в себя два этапа. Вначале на изображениях обнаруживаются яркостные перепады, образующие контуры, а затем результаты сравниваются с порогом [23, 73, 84, 97, 102]. Если результат обнаружения превышает порог, то считается, что этот пиксел изображения принадлежит контуру и ему присваивается значение яркости контура $L_{\rm k}$, в противном случае ему присваивается значение яркости контура $L_{\rm k}$, в противном случае ему присваивается значение яркости фона $L_{\rm p}$ в формируемом контурном изображении. В результате такой обработки изображения получается его контурный препарат. Выделенные контуры на контурном препарате обычно разорваны во многих местах, кроме того, на нем имеются точки и штрихи, которые воспринимаются зри-

тельной системой как помеховый фон. На рис. 9.4 показаны: исходное изображение

(рис. 9.4, a) и его контурный препарат (рис. 9.4, δ), на котором хорошо видны отмеченные особенности.



Рис. 9.4. Изображение (а) и его контурный препарат (б)

Выбирая величину порога, можно несколько уменьшить эти неприятные артефакты, однако совсем от них избавиться не удается. Чем выше порог, тем меньше на контурном изображении будет отдельных точек и штрихов, не являющихся элементами выделяемых контуров, однако при этом в выделенных контурах увеличивается количество и протяженность разрывов. Уменьшение величины порога приводит к обратной картине. В некоторой степени количество возникающих на контурном изображении элементов, не принадлежащих контурам, может быть уменьшено путем применения медианной или логической фильтрации, но добиться их полного устранения не удается. Многочисленные попытки решить задачу выделения контуров привели к разработке целого ряда методов, которые рассматриваются далее.

Выделение контуров с использованием лапласиана. В рассматриваемом случае яркостные перепады выделяются путем вычисления значений лапласиана по форму-

лам (9.10) или (9.11), а затем найденное значение лапласиана $\Delta(k,n)$ сравнивается с порогом *P*. Если это значение превышает порог, т. е.

$$\Delta(k,n) > P ,$$

то считается, что через этот пиксел проходит контур и ему присваивается значение яркости $L_{\rm k}$, в противном случае считается, что пиксел принадлежит фону и ему присваивается значение яркости $L_{\rm b}$. Вычисление лапласиана по формулам может также рассматриваться как свертка исходного изображения с импульсными функциями

$$\begin{vmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{vmatrix}, \begin{vmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{vmatrix}, \begin{vmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{vmatrix}.$$

Обратим внимание на то, что сумма элементов импульсных функций равна нулю.

Как уже отмечалось, использование лапласиана для обнаружения перепадов яркости делает метод выделения контуров чувствительным к шуму на изображении. В этом отношении градиентные методы обнаружения перепадов яркости обладают преимуществом по сравнению с методом обнаружения границ на основе вычисления лапласиана, поэтому рассмотрим некоторые из них.

□ Метод Робертса. В этом методе [50, 73, 84] для обнаружения перепадов яркости применяется перекрестный градиентный оператор Робертса. Результат вычисления модуля градиента $|\mathbf{G}|$ по формулам (9.3, 9.4, 9.9) сравнивается с порогом *P*, который подбирается экспериментально, и если выполняется условие $|\mathbf{G}| > P$, пиксел считается принадлежащим контуру и ему присваивается значение яркости $L_{\rm k}$, в противном случае пиксел считается принадлежащим фону и ему присваивается значение яркости $L_{\rm th}$.

Приблизительную ориентацию выделяемого контура можно определить по положению в окне того из четырех пикселов, который имеет наибольшее значение яркости.

- □ Метод Превитта. В этом методе [50, 73, 84] для обнаружения перепадов яркости применяется рассмотренный выше градиентный оператор Превитта. Результат вычисления модуля градиента $|\mathbf{G}|$ по формулам (9.5), (9.6), (9.9), как и ранее, сравнивается с порогом *P*, который подбирается экспериментально, и, если выполняется условие $|\mathbf{G}| > P$, пиксел считается принадлежащим контуру и ему присваивается значение яркости $L_{\rm k}$, в противном случае пиксел считается принадлежащим фону и ему присваивается значение яркости $L_{\rm th}$.
- □ Метод Собела. Отличие этого метода от метода Превитта заключается только в том, что для обнаружения перепадов яркости в данном случае применяется рассмотренный выше градиентный оператор Собела, и, следовательно, для вычисления модуля градиента |G| используются формулы (9.7)—(9.9).

- Метод пересечения нулевого уровня. В этом методе для обнаружения перепадов яркости применяется оператор Лапласа (лапласиан), который в случае дискретного изображения, как уже было сказано в разд. 9.3, вычисляется по одной из формул: (9.10) или (9.11). При пересечении границы яркости оператор Лапласа меняет свой знак, проходя через нулевое значение. Значения номера строки и номера столбца, где это происходит, в методе пересечения нулевого уровня принимаются за координаты точки, принадлежащей контуру.
- Метод Кирша. Этот метод [50, 84] отличается от метода Собела видом оператора, посредством которого обнаруживаются перепады яркости на контурах. В этом случае применяется оператор с окном размером 3×3 пиксела, показанном на рис. 9.1, который определяется следующим образом

$$\left|\mathbf{G}\right| = D(i, k) = \max\left\{1, \max_{i=0 \div 7} \left[\left|5S_i - 3T_i\right|\right]\right\},\$$

где

$$S_i = L_{i+0} + L_{i+1} + L_{i+2},$$

$$T_i = L_{i+3} + L_{i+4} + L_{i+5} + L_{i+6} + L_{i+7}.$$

Индексы у слагаемых L_i в этих формулах подсчитываются по модулю 8. Как и в предыдущем случае, решение о принадлежности данного пиксела к контуру принимается по результатам сравнения $|\mathbf{G}|$ с порогом. Пиксел считается принадлежащим контуру в том случае, если $|\mathbf{G}|$ превышает значение порога. Обратим внимание на то, что оператор Кирша дает максимальное значение курсового градиента в некоторой точке изображения без учета значения L_8 .

Метод Уоллиса. Нелинейный метод обнаружения перепадов яркости Уоллиса [50, 84] использует гомоморфную обработку изображения. В соответствии с этим методом считается, что точка находится на перепаде яркости, если значение логарифма от яркости в этой точке превышает среднее значение логарифмов яркостей четырех ближайших соседних пикселов на некоторое фиксированное значение (порог). Пиксел контрастированного изображения определяется следующим образом:

$$|\mathbf{G}| = \log L_8 - \frac{1}{4}\log(L_1) - \frac{1}{4}\log(L_3) - \frac{1}{4}\log(L_5) - \frac{1}{4}\log(L_7),$$

или, что то же самое,

$$|\mathbf{G}| = \frac{1}{4} \log \left\{ \frac{[L_8]^4}{L_1 L_3 L_5 L_7} \right\}.$$

Нет необходимости при реализации этого метода каждый раз вычислять логарифм, т. к. сравнение $|\mathbf{G}|$ с верхним и нижним порогами эквивалентно сравнению дроби, стоящей в фигурных скобках с видоизмененными порогами. Главное достоинство логарифмического детектора перепадов помимо простоты вычислений заключается в том, что он не чувствителен к мультипликативным изменениям уровня яркости.

Логарифмический метод контрастирования можно рассматривать как линейное контрастирование изображения посредством оператора Лапласа, но предварительно подвергнутое операции логарифмирования.

Кроме рассмотренных был разработан еще ряд методов выделения контуров, среди них метод Розенфельда, метод Хюккеля и некоторые другие [84], но наиболее удачным из них, безусловно, является метод Канни, к рассмотрению которого и перейдем.

Метод Канни является одним из лучших методов выделения контуров [68, 119]. Особенностью этого метода является использование морфологических операций. Алгоритм этого метода включает в себя следующие действия над изображением, из которого выделяются контуры.

Сглаживание изображения посредством фильтрации, с использованием импульсной функции, описываемой гауссовым законом

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right),$$

где σ — параметр, определяющий степень сглаживающего действия импульсной функции на присутствующий в изображении шум. Чем больше σ , тем сильнее сглаживание.

□ Нахождение градиента изменения яркости в изображении путем вычисления квадратного корня из суммы квадратов производных яркости изображения L_c(x, y) по координатам x и y

$$\mathbf{G}\big|=\sqrt{G_x^2+G_y^2}\,,$$

где $G_x = \partial L_c(x, y) / \partial x$, $G_y = \partial L_c(x, y) / \partial y$.

- Пороговая обработка результатов вычисления градиентов в каждой точке изображения. При этой обработке осуществляется так называемое "немаксимальное" подавление скачков яркости изображения (non-maximal suppression), в результате которого сохраняются только те значения вычисленных градиентов, которые превышают значения градиентов в двух соседних точках на изображении по направлению градиента изображения. Другими словами, сохраняются только значения градиентов в точках максимальной крутизны изменения яркости на световых границах.
- Морфологическая обработка результатов, полученных на предыдущем шаге алгоритма. При этой обработке задаются два порога: нижний и верхний. При формировании контуров все точки, превысившие верхний порог, сохраняются. Что же касается точек, превысивших нижний порог, то сохраняются только те точки, которые непосредственно соседствуют с точками, превысившими верхний порог, прочие же исключаются из формируемого контурного изображения.



а







)





Рис. 9.5. Изображение и его контурные препараты: *а* — исходное изображение. Контурные препараты, полученные методами: *б* — Робертса; *в* — Превитта; *г* — Собела; *д* — пересечения нулей; *е* — Канни Достоинствам метода Канни является то, что он обеспечивает лучшее выделение контуров, чем другие рассмотренные методы, даже при наличии небольшого уровня шума. К недостаткам этого метода следует отнести его большую сложность, а также некорректную обработку *У*-разветвлений.

Приведенное выше описание алгоритма сделано на примере аналогового изображения. Понятно, что в случае выделения контуров из цифрового изображения методом Канни модуль градиента необходимо вычислять так, как это описано в *разд. 9.3*.

На рис. 9.5 для сравнения приведены исходное изображение и его контурные препараты, полученные посредством описанных выше методов.

Сопоставляя между собой показанные на рис. 9.5 контурные препараты, видим, что в отличие от препарата, полученного посредством метода Канни, на всех других имеются множественные разрывы контуров, а также отдельные точки и штрихи.

В заключение этого раздела заметим, что вычисление контуров применительно к цветному изображению не представляет каких-либо проблем. В этом случае контуры могут вычисляться как для яркостного компонента, так и для компонентов цвета R, G и B, используя описанные выше методы, в зависимости от требований решаемой задачи.

9.5. Метод сегментации путем связывания пирамиды

Связывание пирамиды представляет собой эффективный метод сегментации и вычисления признаков изображений. Этот метод и реализующий его алгоритм был предложен Бюртом (Burt) [118] и может быть эффективно использован для выделения световых границ на зашумленных изображениях [101]. Рассмотрим его на примере выделения размытой световой границы на зашумленном изображении. На рис. 9.6, *а* показано распределение яркости вдоль части строки, которую пересекает размытая световая граница, имеющая небольшой контраст. Уровни яркости пикселов, составляющих эту строку, представлены номерами квантовых уровней. Таким образом, левая часть строки, как это видно из рисунка, имеет в среднем меньшую яркость, чем правая часть.

Рассматриваемый алгоритм включает в себя три этапа.

1. На первом этапе строится *пирамида Гаусса*. При построении пирамиды Гаусса строка сканируется окном (импульсной функцией), протяженность которой вдоль строки составляет 4 отсчета (пиксела). Особенностью этого сканирования является то, что окно при своем перемещении вдоль строки совершает сдвиги, составляющие 2 отсчета. При каждом положении окна вычисляется средняя яркость пикселов изображения, оказавшихся в пределах этого окна. Это вычисленное значение яркости присваивается отсчету более высокого уровня, число которых, как это видно из рисунка, в два раза меньше, чем число пикселов в строке. После того как сканирование строки изображения будет завершено и будет сформирован слой отсчетов более высокого уровня, на рисунке он обозначен как $G^{(1)}$, выполняется

отсчетов оолее высокого уровня, на рисунке он ооозначен как $G^{(7)}$, выполняется сканирование этого слоя отсчетов рассмотренным выше окном и формируется следующий слой отсчетов, обозначенный на рисунке как $G^{(2)}$.



Рис. 9.6. Пояснение метода сегментации посредством связывания пирамиды на примере выделения контура на зашумленном изображении: *a* — пирамида Гаусса;
 б — процедура связывания узлов; *в* — процедура перевычисления средних уровней яркости;
 г — окончательный результат после выполнения нескольких итераций шагов (б) и (в)

Процесс продолжается до тех пор, пока во вновь сформированном слое не окажется всего один отсчет. Полученная при этом структура называется пирамидой Гаусса. Исходные пикселы изображения и сформированные в результате построения отсчеты в промежуточных слоях называются узлами. Отсчет, находящийся в вершине, называется корневым узлом или корнем. Отсчеты, находящиеся в промежуточных слоях по отношению к отсчетам, находящимся в вышестоящем слое, называются дочерними узлами. Эти же отсчеты по отношению к отсчетам, находящимся в нижестоящих слоях, являются родительскими. Линии, соединяющие узлы, называются ребрами.

- 2. Сегментация посредством связывания пирамиды является вторым шагом алгоритма. На этом этапе выполняется перегруппировка пирамиды, которая заключается в переназначении связей (ребер) между дочерними и родительскими узлами. Делается это следующим образом. Сначала для узлов самого нижнего уровня $G^{(0)}$ (т. е. пикселов изображения) находятся ближайшие по величине яркости узлы следующего уровня (в данном случае уровня $G^{(1)}$) и они соединяются ребрами с соответствующими узлами уровня $G^{(1)}$. При этом должно выполняться требование, заключающееся в том, что все пикселы уровня $G^{(0)}$, соединенные с уровнем $G^{(1)}$, должны следовать друг за другом, без внедрения пикселов, связанных с другими узлами уровня $G^{(1)}$ (т. е. без пересечения связей). Эта процедура повторяется по всем уровням пирамиды. В результате, связи в пирамиде образуют новую структуру данных. Начиная с вершины пирамиды, каждый отсчет (узел) соединяется с несколькими отсчетами (узлами) на следующем, более низком уровне. Созданная таким образом структура называется деревом. Рисунок 9.6, δ поясняет сказанное.
- 3. Усреднение связанных узлов является третьим шагом алгоритма. На этом шаге алгоритма полученная структура связей используется для пересчета средних уровней яркости на этот раз с использованием только связанных на самом низком уровне и продолжается по всем уровням пирамиды вверх. Последние два шага алгоритма итерационно повторяются до тех пор, пока не достигается устойчивый результат, показанный на рис. 9.6, *г*. Анализ дерева связей демонстрирует результат рассмотренной процедуры сегментации. На рис. 9.6, *б* можно видеть два поддерева, которые имеют корни на третьем уровне пирамиды. Далее на следующем, более низком уровне возникают четыре поддерева. Но различия в уровнях яркости на этом уровне значительно меньше. Из этого можно сделать заключение, что структура уровней яркости отчетливо разделяется на две области. Окончательный результат сегментации получается путем последовательного переноса уровней к уровню $G^{(0)}$. Эти значения показаны на рис. 9.6, *г* в виде чисел, заключенных в круглые скобки.

На рис. 9.7 показан двумерный вариант построения пирамиды Гаусса, который может быть использован при реализации рассматриваемого метода.



Рис. 9.7. Двумерный вариант построения пирамиды Гаусса

Обратим внимание на то, что посредством этой процедуры даже сильно зашумленные изображения могут быть успешно сегментированы. Не существует также какихлибо ограничений на форму сегментируемой области.

Процедура связывания пирамиды объединяет сегментацию и эффективное сглаживание признаков в процессе построения дерева, благодаря чему ослабляется влияние шума, имеющегося на изображении. Преимуществом этого метода является также то, что при его использовании не нужно заранее знать количество уровней сегментации. Эти данные заключены уже в самой структуре дерева. Дополнительную информацию, касающуюся рассмотренного метода сегментации, читатель может найти в работах Бюрта и др. (Burt) [117], Питикейнена (Pietikainen) и Розенфельда (Rosenfeld) [157].

9.6. Пороговая обработка

Рассмотрим сегментацию изображений, в основу которой положена пороговая обработка изображений с использованием глобального порога. Этот вид сегментации может быть применен в том случае, если в изображении яркость выделяемых объектов отличается от яркости остальной части изображения, которая рассматривается при этом как фон. Это различие находит отражение в гистограмме распределения яркости на изображении в виде наличия ярко выраженных максимумов. В качестве примера на рис. 9.8, *а* приведено изображение аэрофотографии города, через который протекает река, впадающая в залив.

На рисунке вода изображена темным цветом, а городские строения белым и серым цветами. На рис. 9.8, б показана гистограмма этого изображения, на которой видны два максимума: один в области темных полутонов, а другой в области светлых полутонов. Первый максимум формируется пикселами, изображающими водные

поверхности, а второй максимум формируется остальной частью изображения. В том случае, если объектами, которые следует выделить при сегментации, являются водные поверхности, достаточно подвергнуть изображение пороговой обработке, поместив порог P между двумя максимумами. При этом пикселы, яркость которых L ниже порога, т. е. L < P, следует отнести к выделяемым объектам, т. е. водным поверхностям, а пикселы, яркость которых выше порога или равна ему, следует отнести к фону, которым в рассматриваемом примере являются городские застройки. На рис. 9.8, e приведено изображение, полученное в результате сегментации изображения, представленного на рис. 9.8, e, рассматриваемым методом. Следует заметить, что в данной конкретной задаче положение порога не критично.



Рис. 9.8. К пояснению метода сегментации, основанного на пороговой обработке изображения

Рассмотренный метод сегментации изображений путем их пороговой обработки в чистом виде редко применяется, поскольку он эффективен только для сравнительно небольшого круга изображений, в которых объекты и фон четко различаются по яркостному признаку, и, кроме того, яркость фона мало изменяется в пределах изображения. В том случае, если яркость фона заметно изменяется по изображению, используют вариант этого метода — метод пороговой обработки с локальным порогом. Особенность этого метода заключается в том, что значение порога адаптивно изменяется с изменением яркости фона [23]. Обычно метод пороговой обработки применяется в сочетании с другими более эффективными методами сегментации, например, при сегментации путем выделения контурных признаков.

9.7. Метод выращивания областей

Принцип, положенный в основу выращивания областей, заключается в следующем [23]. Области, на которые разделяется изображение при сегментации, должны характеризоваться каким-либо признаком. Таким признаком, например, может быть условие, состоящее в том, что яркость пиксела полутонового черно-белого изображения заключена в некоторых заранее заданных пределах, другими словами, условие принадлежности яркости пиксела к фиксированному интервалу уровней серого. В одном из вариантов алгоритма выращивания областей, начиная с произвольного пиксела изображения, производится проверка каждого из его соседей на предмет, обладает ли он тем же самым признаком, что и исходный пиксел. Если да, то этот пиксел включается в данную область, после чего проверяются его соседи. После того как больше не останется пикселов, смежных с данной областью и характеризующихся данным признаком, процедура формирования области завершается и начинается процедура формирования следующей области, начиная с любого другого пиксела, не вошедшего в сформированную область. Сегментация завершается после того, как все пикселы изображения будут включены в какую-либо область.



Рис. 9.9. Исходное изображение (а) и результат его сегментации (б)

Возможны варианты описанного здесь алгоритма сегментации. Например, значения яркостей пикселов, вокруг которых производится наращивание областей, могут быть заранее заданы, это так называемые *"центры кристаллизации"*, или, например, в процессе сегментации значение яркости, с которым сравнивается каждый новый проверяемый пиксел, может каждый раз вычисляться как среднее значение яркости, уже выращенной к данному моменту области. Выбор того или иного способа определяется конкретной решаемой задачей. В качестве примера на рис. 9.9 приведены исходное изображение (*a*) и результат сегментации этого изображения (*б*).

Обычно области, на которые разделено изображение, маркируются в соответствии с тем признаком, по которому в них был включен тот или иной пиксел.

9.8. Метод разделения и слияния областей

Альтернативный подход к решению задачи сегментации изображений заключается в том, что вначале производится разбиение изображения на множество произвольных непересекающихся первичных областей в соответствии с выбранным признаком. Процесс разбиения продолжается до тех пор, пока не сформируются однородные по выбранному признаку области. После этого смежные первичные области пикселов, для которых значения признака, по которому они были разделены, совпадают, объединяются. Это так называемый *метод с использованием квадродерева*.

Поясним это примером. На рис. 9.10, *а* схематически показано изображение (обозначим его через *O*), подлежащее сегментации по тому или иному признаку. В соответствии с описываемым алгоритмом область *O*, т. е. само изображение, разделяем на 4 квадратные первичные области O_1 , O_2 , O_3 , O_4 (см. рис. 9.10, δ) и производим проверку на их однородность. Другими словами, проверяем, все ли пикселы, составляющие эти области, удовлетворяют признакам, в соответствии с которыми производится сегментация изображения.

Предположим, что для первичных областей O_1 , O_2 , O_3 это условие выполняется, а для первичной области O_4 нет. В этом случае следующим шагом алгоритма является разделение первичной области O_4 на четыре части O_{41} , O_{42} , O_{43} , O_{44} . Этот процесс продолжается до тех пор, пока полученные в результате разделения все первичные области не окажутся однородными в смысле признака, в соответствии с которым производится разделение. В предельном случае это может быть один пиксел. Возвращаясь к рассматриваемому примеру, предположим, что все первичные области O_{41} , O_{42} , O_{43} , O_{44} (см. рис. 9.10, ϵ) оказались однородными и поэтому процесс разделения изображения на первичные области завершается. После этого переходим к объединению смежных первичные областей, полученных для одинаковых признаков, в соответствии с которыми происходило разделение изображения на первичные области. Предположим, что первичные области O_3 , O_{41} и O_{43} соответствуют одинаковым значениям признака, по которому происходило разделение. Кроме того, предположим, что области O_2 и O_{42} также получены при одинаковых, но других значениях этого признака. Тогда, следуя описанному правилу, первичные области O_{41} и O_{43} объединяются с областью O_3 , а первичная область O_{42} объединяются с областью O_2 . На этом процесс сегментации завершается. На рис. 9.10, *г* по-казан результат сегментации. На рис. 9.10, ∂ приведено квадродерево, полученное на этапе разделения изображения на первичные области.



Рис. 9.10. К пояснению метода разделения и слияния областей

9.9. Метод сегментации на основе движения

Метод сегментации изображения на основе движения позволяет выделить изображения движущихся объектов из изображения неподвижного фона. В основе этого метода лежит принцип попиксельного сравнения изображений сцены, представленных в различных кадрах телевизионного изображения, с изображением неподвижного фона. В случае черно-белых полутоновых изображений при выделении изображения движущегося объекта из неподвижного фона вычисляются абсолютные значения разностей яркостей пикселов для изображения текущего кадра и изображения фона, с последующим сравнением их с порогом.

$$\Delta L = \left| L_{\rm c}\left(k,n,i\right) - L_{\rm c}\left(k,n\right) \right|,$$

где $L_{c}(k, n, i)$ — распределение яркости в кадре номер i; $L_{\phi}(k, n)$ — распределение яркости в фоновом изображении; k — номер строки; n — номер столбца. Если абсолютное значение вычисленной разности превышает порог P, то принимается решение о принадлежности пиксела движущемуся объекту и значение яркости пиксела записывается в массив выделяемого изображения $L_{OF}(k, n, i)$, в противном случае считается, что пиксел принадлежит фону. Это правило может быть записано следующим образом:

$$L_{OF}(k, n, i) = \begin{cases} L_{c}(k, n, i) & \text{при } \Delta L > P, \\ 0 & \text{при } \Delta L \le P. \end{cases}$$

Величина порога определяется экспериментально и зависит от уровня флуктуационного шума, который всегда присутствует. Чем выше уровень шума, тем выше должен быть выбран порог, с которым производится сравнение абсолютного значения разности. Этот случай легко распространяется на цветное изображение путем вычисления абсолютных значений разностей R-, G-, В-компонентов цветного изображения. В том случае, если имеется не один, а ряд движущихся объектов, алгоритм работает аналогично. При реализации этого алгоритма изображение фона получается путем предварительного фотографирования сцены, содержащей только неподвижные объекты в отсутствие движущихся объектов, которые в дальнейшем появляются и перемещаются в пределах этой сцены.

Недостатком описанного алгоритма является то, что в естественных условиях получения движущихся изображений освещение сцены может хотя и медленно, но существенно изменяться с течением времени, при этом будут изменяться интенсивности как пикселов, которые представляют движущийся объект, так и тех пикселов, которые представляют фон, а это будет приводить к ошибкам. Для того чтобы избежать этого недостатка, был разработан усовершенствованный вариант этого алгоритма, в котором реализуется регенерация фонового изображения. В соответствии с этим алгоритмом в результате принятия решения яркость анализируемого пиксела либо записывается в массив выделяемого движущегося изображения $L_{OE}(k, n, i)$, если $\Delta L > P$, либо в массив регенерируемого фона $L_{\phi}(k, n, i)$, если $\Delta L > P$. Это правило может быть записано следующим образом:

$$L_{OF}(k, n, i) = \begin{cases} L_{c}(k, n, i) & \text{при } \Delta L > P, \\ 0 & \text{при } \Delta L \le P. \end{cases}$$
$$L_{\phi}(k, n, i) = L_{c}(k, n, i) & \text{при } \Delta L \le P. \end{cases}$$

Описанный алгоритм надежно работает, но, как следует из его описания, требует предварительного фотографирования сцены, содержащей только неподвижные объекты, для получения фонового изображения.

Однако этот недостаток не является фатальным, его можно избежать, если фоновое изображение формировать в процессе съемки сцены, содержащей движущиеся

объекты. В этом случае первый кадр изображения, содержащего движущиеся объекты, рассматривается как нулевое приближение к фоновому изображению и сохраняется в массив фонового изображения. Далее, по мере поступления последующих кадров, фоновое изображение не только регенерируется, но и корректируется путем исключения из него пикселов, принадлежащих изображениям движущихся объектов, при этом освободившиеся места заполняются пикселами, принадлежащими тем частям фона, которые ранее заслонялись ушедшими с этих мест объектами. Формирование фонового изображения полностью завершается после того, как движущийся объект сместится на величину своего размера и полностью откроет ту часть фона, которую он раньше заслонял.

На практике для повышения надежности выделения движущихся изображений осуществляют попиксельно сравнение интенсивностей R, G и B не в двух соседних, а в ряде кадров, следующих друг за другом. При этом количество используемых кадров может быть достаточно большим (до 8 и более).

Обратим внимание на то, что при реализации алгоритма выделения объекта из фона необходимо работать с несжатыми файлами или с файлами, сжатыми без потерь информации. Если не выполнить это условие, то помеха преобразования, возникающая в процессе компрессии изображений, приводит к тому, что их фоновый компонент от кадра к кадру немного изменяется, в результате чего не происходит полной компенсации фона при выделении движущихся объектов.

Рассмотренные алгоритмы применяются в задачах охранной сигнализации, задачах контроля дорожного движения, задачах сжатия движущихся изображений, при использовании технологий 2,5D-изображений и в ряде других. Стандарт MPEG-4 предусматривает возможность разделения изображения на медиаобъекты, частными случаями которых являются фоновое изображение и изображения движущихся объектов на прозрачном фоне.

9.10. Сегментация, основанная на применении морфологических операций эрозии и наращивания

Как известно, текстурный компонент изображения является важным информативным признаком. Поэтому часто при сегментации изображения в качестве признака, в соответствии с которым оно разделяется на области, используют его текстуру. Один из алгоритмов сегментации такого рода основан на последовательном применении морфологических операций эрозии и наращивания. Этот алгоритм позволяет разделить области изображения, которые различаются между собой тем, что текстура одних областей имеет более крупную структуру, чем других, как, например, в случае изображения, показанного на рис. 9.11, *а*.

В соответствии с этим алгоритмом первый шаг обработки изображения состоит в последовательном применении к нему эрозии *m*1 раз. Величина *m*1 выбирается

таким образом, чтобы в результате эрозии были бы полностью подавлены элементы мелкоструктурной текстуры. Естественно, что при этом размеры элементов крупноструктурной текстуры оказываются также уменьшенными. На рис. 9.11, δ показан результат применения эрозии по отношению к исходному изображению m1 = 4 раза.



Рис. 9.11. К пояснению метода сегментации, основанного на применении морфологических операций эрозии и наращивания

Второй шаг алгоритма заключается в применении к изображению, полученному в результате выполнения первого шага алгоритма (см. рис. 9.11, δ), операции наращивания последовательно m^2 раз. Значение параметра m^2 выбирается таким образом, чтобы элементы крупноструктурной текстуры в результате наращивания слились в сплошную область. На рис. 9.11, ε приведено изображение, полученное в результате такого наращивания. В нашем случае было принято $m^2 = 18$.

Третий шаг алгоритма заключается в нахождении границы, разделяющей области изображения с различной текстурой. С этой целью изображение $L_1(k,n)$, полученное после выполнения второго шага алгоритма, подвергается однократной эрозии,

в результате которой получается изображение $L_2(k,n)$. После этого из изображения $L_1(k,n)$ вычитается изображение $L_2(k,n)$:

$$L_{\rm rp}(k,n) = L_1(x, y) - L_2(x, y).$$

Получившееся в результате вычитания распределение яркости будет представлять собой изображение границы толщиной в один пиксел.

На рис. 9.11, г показано изображение границы, разделяющей области с различной текстурой, которая наложена на исходное изображение.

Рассмотренный алгоритм можно использовать и в том случае, когда изображение включает в себя не два типа текстур, различающихся между собой размерами составляющих их элементов, а большее. В этом случае описанную обработку применяют к исходному изображению не один раз, как в рассмотренном примере, а большее число раз с тем, чтобы при каждом ее применении находить границы между областями с различающимися по размеру элементов текстурами. После того как все границы будут найдены, они совмещаются, в результате чего мы получаем окончательный результат.

9.11. Особенности сегментации изображений в случае, когда признак, по которому она выполняется, является векторной величиной

Рассмотренные ранее методы сегментации изображений относятся к случаю, когда признак, по которому производится сегментация, является скалярной величиной, т. е. характеризуется одним числом. Типичными примерами таких признаков являются: яркость пиксела, принадлежность пиксела к изображению неподвижного или движущегося объекта, размер элемента, составляющего текстуру и т. д. Однако часто на практике признак, по которому сегментируют изображение, является векторным, т. е. представляется не одним, а рядом чисел. Простейшим примером такого признака является цвет пиксела, который определяется тремя компонентами, а именно: интенсивностью красного компонента R, интенсивностью зеленого компонента G и интенсивностью синего компонента B. Другим примером векторного признака, применяемого при сегментации, является ряд центральных моментов, представляющих распределение текстуры, о чем речь пойдет далее. Векторные признаки являются более информативными и поэтому чаще используются.

Сегментацию с использованием векторных признаков рассмотрим на примере сегментации цветного изображения. В этом случае все цветовое пространство разделяется на области, каждая из которых охватывает определенный диапазон его значений, при этом, как и ранее, при сегментации должны выполняться следующие требования:

□ в результате сегментации изображение разделяется на ряд областей таким образом, чтобы каждый его пиксел входил в одну из областей;

- области, которые получаются в результате сегментации, не должны пересекаться, другими словами, каждый пиксел изображения может входить только в одну область;
- все пикселы, отнесенные к одной области, должны обладать одними и теми же свойствами, а именно: цвет пикселов, отнесенных к одной области, должен лежать в определенных для нее пределах.

Выбор используемой цветовой модели при этом определяется конкретной решаемой задачей. Так, например, если признаком, используемым для целей сегментации, является именно цвет, независимо от значения яркости, то удобно воспользоваться колориметрической системой L*a*b*, преобразовав предварительно изображение в это цветовое пространство и в дальнейшем использовать координаты a^* и b^* . Первая из этих координат представляет соотношение красного и зеленого цветов, а вторая — соотношение синего и желтого цветов. Если же в качестве признака используются все три характеристики цвета, то в этом случае может быть использована любая колориметрическая система.

Рассмотрим пример, полагая для конкретности, что работа происходит в цветовой модели (колориметрической системе) R, G, B, для сегментации используется метод выращивания областей, а признак принадлежности пиксела к той или иной области задан табл. 9.1.

Таблица 9.1

Номер области і	1	2	3	4
R_{Π}	$r_1 = 0 - 126$	$r_2 = 0 - 126$	$r_3 = 127 - 255$	$r_4 = 127 - 255$
G_{Π}	$g_1 = 0 - 99$	$g_2 = 100-255$	$g_3 = 0 - 99$	$g_4 = 100-255$
B _n	$b_1 = 0 - 255$	<i>b</i> ₂ = 0—255	$b_3 = 0 - 255$	<i>b</i> ₄ = 0—255

В этой таблице использованы следующие обозначения: r_1 , r_2 , r_3 , r_4 — множества значений красного компонента; g_1 , g_2 , g_3 , g_4 — множества значений зеленого компонента, b_1 , b_2 , b_3 , b_4 — множества значений синего компонента, R_{Π} , G_{Π} , B_{Π} — значения соответствующих цветовых компонентов пиксела. В этом случае условие принадлежности проверяемого пиксела к выращиваемой области *i* имеет вид:

$$R_{\Pi} \in r_i \wedge G_{\Pi} \in g_i \wedge B_{\Pi} \in b_i =$$
true,

где \land — символ логического умножения.

Использование цвета в качестве признака, по которому производится сегментация, широко используется в космических системах мониторинга природных ресурсов.

9.12. К-метод сегментации изображений

В общем случае *К*-метод сегментирует изображение на *К* различных кластеров (областей), расположенных на возможно больших расстояниях друг от друга по заданному признаку. В качестве такого признака может быть выбран цвет (значения всех трех компонентов RGB одновременно), цвет и геометрическое расстояние одновременно и т. д. По умолчанию при реализации этого метода для измерения расстояний применяется Эвклидова метрика.

Метод сегментации "К-метод" реализуется посредством двухэтапного итеративного алгоритма, который минимизирует сумму расстояний "точка — центроид", полученную путем суммирования по всем К кластерам. Другими словами, целью работы алгоритма является минимизация изменчивости внутри кластеров и максимизация изменчивости между кластерами.

Алгоритм начинает свою работу с *К* случайно выбранных положений центроидов кластеров, а затем изменяет принадлежность точек (объектов) к кластерам, т. е. перемещает точки из одних кластеров в другие для того, чтобы получить наиболее значимый результат.

Во время первого этапа при каждой итерации все точки сразу перегруппировываются таким образом, чтобы они возможно ближе располагались к своим центроидам, после чего пересчитываются координаты центроидов каждого кластера. Эта часть алгоритма позволяет найти быстро, но только приблизительное решение задачи сегментации, которая является отправной точкой для второго этапа.

Во время второго этапа алгоритма точки индивидуально подвергаются перегруппировке в том случае, если это приводит к уменьшению суммы расстояний, а координаты центроидов кластеров пересчитываются после перегруппировки каждой точки. Каждая итерация во время второго этапа состоит только из одного прохода по всем точкам.

По окончании выполнения описанного алгоритма программа сегментации может выдавать такие дополнительные данные, как:

суммы расстояний "точка — центроид";

🗖 координаты центроидов,

а также некоторые другие данные.

Алгоритм *К*-метода может сходиться к локальному оптимуму, когда при разделении точек перемещение любой точки к другому кластеру увеличивает результирующую сумму расстояний. Эта проблема может быть решена только путем разумного (удачного) выбора начальных точек.

9.13. Сегментация изображений на основе статистических характеристик текстуры

Рассмотрим сегментацию изображений применительно к случаю, когда в качестве признаков, по которым производится сегментация, используются центральные моменты распределения и функция автокорреляции текстур, а текстура рассматривается как стохастическое поле [25, 31]. Заметим, что, в отличие от зрительной системы, современные алгоритмы способны различать текстурные поля, отличающиеся друг от друга не только дисперсиями, коэффициентами автокорреляции и одномерными плотностями распределения вероятностей яркости, но и центральными моментами более высокого порядка.

Рассмотрим признаки, которые часто используются для сегментации полей.

 Локальный второй центральный момент текстурного поля, который вычисляется по формуле

$$M_{2} = \frac{1}{KN} \sum_{-K/2}^{K/2} \sum_{-N/2}^{N/2} \left[L_{c}(k, n) - \overline{L_{c}} \right]^{2},$$

где K и N — размеры в пикселах по вертикали и горизонтали области, в которой выполняется суммирование; $\overline{L_c}$ — среднее значение яркости в области, по которой производится суммирование; $L_c(k,n)$ — значение яркости пиксела, расположенного в k-й строке и n-м столбце. Второй центральный момент, или дисперсия характеризует степень шероховатости поверхности изображаемого объекта, обусловленной текстурой.

Локальный третий центральный момент текстурного поля

$$M_{3} = \frac{1}{KN} \sum_{-K/2}^{K/2} \sum_{-N/2}^{N/2} \left[L_{c}(k, n) - \overline{L_{c}} \right]^{3},$$

который характеризует асимметрию распределения отклонений яркости, обусловленных текстурой, от среднего значения.

Локальный четвертый центральный момент текстурного поля

$$M_{4} = \frac{1}{KN} \sum_{-K/2}^{K/2} \sum_{-N/2}^{N/2} \left[L_{c}(k,n) - \overline{L_{c}} \right]^{4},$$

который характеризует остроту распределения отклонений яркости, обусловленного текстурой, от среднего значения.

Обратим внимание на то, что все перечисленные выше признаки, которые используются при сегментации изображений, а именно: центральные моменты распределения и функция автокорреляции, носят локальный характер, поскольку они вычисляются в ограниченной области. Поэтому перед тем как приступать собственно к сегментации изображения, необходимо предварительно вычислить их значения и записать в многомерный массив чисел. Процесс вычисления значений признаков, по которым выполняется сегментация изображений на области, состоит в следующем. Окном размером $K \times N$ сканируют изображение, при каждом его положении вычисляют значения признаков и записывают эти значения в двумерные массивы, количество которых равно числу вычисляемых признаков. Количество строк и столбцов у этих массивов равно количеству строк и столбцов сканируемого изображения. При этом для того чтобы результаты измерений не искажались бы световыми границами в результате их попадания в сканирующее окно, можно использовать один из следующих приемов:

- после окончания сканирования исходного изображения скорректировать искаженные результаты путем их замены значениями граничащих с ними неискаженных величин, используя в качестве критерия, по которому определяются искажения, превышение вторым центральным моментом экспериментально подобранного порога;
- □ перед сканированием изображение подвергнуть предварительной обработке с целью выделения из него только текстурного компонента (*см. разд. 6.9*).

После того как признаки, по которым производится сегментация изображения на области, получены, выполняют собственно сегментацию изображения по одному из описанных выше алгоритмов с учетом того, что, как правило, признак, по которому выполняется сегментация, является векторной величиной.

9.14. Интерактивная сегментация изображений

Ранее был рассмотрен ряд методов автоматической сегментации изображений. К сожалению, посредством этих методов и реализующих их алгоритмов не удается решать произвольные задачи сегментации с гарантированным результатом, например, выделять из сложного фона изображения тех или иных сложных объектов. В связи с этим были разработаны и продолжают разрабатываться методы интерактивной сегментации. В качестве примера рассмотрим два таких метода.

Первым и наиболее простым методом интерактивной сегментации изображений и реализующим его алгоритмом является применение инструмента **Magic Wand** (Волшебная палочка). Он существует в графическом редакторе Photoshop. При использовании этого метода пользователь в интерактивном режиме указывает пиксел, принадлежащий объекту, яркость которого составляет L_{c1} , и задает порог P. После этого алгоритм выделяет все области на сегментируемом изображении, яркость пикселов которых L_c удовлетворяет условию

$$L_{c1} - P \le L_c \le L_{c1} + P$$

В случае сегментации цветного изображения алгоритм работает аналогичным образом, с той разницей, что при этом написанное выше неравенство должно быть выполнено для всех трех цветовых компонентов изображения. Однако этот алгоритм может быть использован для выделения объектов из фона только в том случае, если выделяемый объект и фон, из которого он выделяется, не содержат одинаковых по интенсивности и цвету пикселов. Если же это условие не выполнено, то наряду с пикселами объекта будут выделены пикселы фона.

В качестве второго примера рассмотрим метод сегментации, основанный на теории графов [23]. Описываемый далее метод лежит в основе ряда алгоритмов, реализованных во многих графических редакторах, таких, как, например, Photoshop.

Переходя к описанию метода, напомним, что граф включает в себя вершины n_1 , n_2 , n_3 , ... и ребра (n_i, n_j) , которыми эти вершины соединены. Если ребру графа приписано направление, то оно называется дугой. Вершина, из которой исходит дуга, называется начальной вершиной дуги (родителем), а вершина, на которой заканчивается дуга, называется конечной вершиной (потомком). Выделим в графе единственную вершину, которую назовем начальной (корневой), и множество вершин, которые назовем концевыми (целевыми) вершинами. Припишем каждой дуге (n_i, n_j) некоторую стоимость $c(n_i, n_j)$. Последовательность вершин $n_1, n_2, ..., n_k$, в которой каждая последующая вершина n_i является потомком предыдущей n_{i-1} , назовем путем от n_1 до n_k . При этом стоимость всего пути составит

$$c = \sum_{i=2}^{k} c(n_{i-1}, n_i).$$

Для определенности дальнейших рассуждений определим элемент контура как границу между двумя пикселами p и q, находящимися в одной строке или в одном столбце, как это видно на рис. 9.12. Полный контур является последовательностью соединенных друг с другом элементов.



Рис. 9.12. Элемент контура, заключенный между пикселами р и q



Рис. 9.13. К пояснению метода сегментации изображений, основанного на теории графов: *а* — фрагмент изображения размером 3×3 пиксела; *б* — стоимости элементов контура; *в* — контур, соответствующий пути с минимальной стоимостью

Теперь покажем, каким образом приведенные ранее понятия могут быть использованы в задаче обнаружения контуров на изображении, для чего обратимся к изображению, представленному на рис. 9.13, *а*. На рисунке обозначены номера строк и столбцов фрагмента изображения. В квадратных скобках приведены значения яркостей пикселов.

Каждому элементу контура, заданному пикселами p и q, поставлена в соответствие стоимость, которая определена следующим образом:

$$c(p,q) = L_{\text{make}} - (L_p - L_q), \qquad (9.12)$$

где $L_{\text{макс}}$ — максимальное значение яркости в изображении (в рассматриваемом примере $L_{\text{макс}} = 7$), а L_p и L_q — значения яркостей пикселов p и q соответственно. По соглашению, пиксел p находится справа относительно направления движения по контуру. Элемент контура (1, 2)—(2, 2) согласно рис. 9.13, δ расположен между пикселами с координатами (1, 2) и (2, 2). Здесь первое число в скобках обозначает номер строки, а второе номер столбца. Если двигаться по элементу контура слева направо, то пиксел p будет иметь координаты (2, 2), а пиксел q — координаты (1, 2). В этом случае стоимость элемента контура составит c(p,q)=7-(7-6)=6. Это значение стоимости записано в рамке под элементом контура (под стрелкой). Если же двигаться по элементу контура справа налево, то под пикселом p следует понимать пиксел с координатами (1, 2), а под пикселом q — пиксел с координатами (2, 2) и стоимость в этом случае будет равна 8. На рис. 9.13, δ это значение стоимости записано в рамке над элементом контура.



Рис. 9.14. Граф для изображения, представленного на рис. 9.13. Пунктирной линией показан путь минимальной стоимости

Для простоты дальнейших рассуждений будем считать, что контуры начинаются в верхней строке изображения и заканчиваются в нижней, при этом первый элемент контура может находиться или между пикселами с координатами (1, 1) и (1, 2), или между пикселами с координатами (1, 2) и (1, 3). Аналогично последний элемент контура может находиться или между пикселами с координатами (3, 1) и (3, 2), или между пикселами с координатами (3, 2) и (3, 3). На рис. 9.14 приведен граф, относящийся к рассматриваемой задаче. Вершины графа на этом рисунке показаны в виде прямоугольников, а дуги — прямыми линиями. Каждая вершина графа соответствует одному из элементов контура на рис. 9.13. Между двумя вершинами имеется дуга только в том случае, если соответствующие два элемента контура, будучи соединенными подряд, могут являться участком контура. На рис. 9.14 рядом с дугами в прямоугольных рамках показаны стоимости каждого элемента контура, вычисленные по формуле (9.12). Целевые вершины, в которых может окончиться контур, показаны на рисунке темным цветом. После того как построен граф и определены стоимости элементов контура, находится путь, для которого стоимость минимальна, и по этому пути проводится контур. На рис. 9.14 пунктирной линией показан путь, которому соответствует минимальная стоимость, а на рис. 9.13, в — контур, соответствующий этому пути. Простейший, но не лучший способ отыскания пути минимальной стоимости — это метод перебора всех возможных вариантов пути. Недостаток этого метода состоит в том, что он требует больших вычислительных затрат. В связи с этим были разработаны более экономные алгоритмы поиска, рассмотрение которых выходит за рамки этой книги.

В заключение данного раздела заметим, что за последнее десятилетие в области разработки интерактивных алгоритмов сегментации изображений были получены значительные успехи, среди которых следует назвать разработку алгоритма GraphCut [114], а также ряда других.

Помимо рассмотренных выше методов сегментации в настоящее время применяется еще целый ряд других методов сегментации, подробные сведения о которых читатель найдет в литературе [115, 116, 128, 129, 130, 148, 153, 154, 175].

9.15. Гранулометрия

Гранулометрия — это раздел техники, одной из задач которой является измерение распределения размеров частиц на сложных изображениях. Пример такого изображения показан на рис. 9.15, *а*. В качестве примера применения гранулометрии можно назвать медицину, где она применяется при анализе крови для подсчета количества эритроцитов, тромбоцитов и т. д., промышленность, где она используется для количественного анализа зерен в зернистых структурах (сталях, сплавах и т. п.), а также в ряде других областей.

В основе этой технологии лежит тот или иной вид обработки изображения, в результате которого из изображения последовательно исключаются объекты все больших и больших размеров и при этом каждый раз измеряется площадь, занимаемая объектами, которые в нем остались. Не трудно видеть, что изменение этой площади ΔS_i после *i*-го шага будет пропорционально произведению площади одного объекта, оказавшегося в группе исключенных объектов S_{di} , на количество объектов N_i , которые были исключены во время этого шага, т. е.

$$\Delta S_i = S_{di} N_i \, .$$



Рис. 9.15. Иллюстрация применения гранулометрии: *а* — изображение частиц с различными размерами; *б* — график зависимости количества частиц на изображении от их площади

Наиболее сложной частью этого метода является разработка алгоритма, производящего исключение объектов из изображения по критерию их размера. Алгоритмы такого рода оказываются, как правило, весьма чувствительны к таким характеристикам объектов, как их форма, ориентация, а если изображение является полутоновым, то и к распределению яркости по поверхности объекта.

Одним из наиболее простых методов разделения объектов по критерию их размера является следующий метод. К изображению применяется морфологическая операция эрозии последовательно i раз, а затем морфологическая операция дилатации (наращивания) последовательно такое же количество раз. Измеряя изменения площади, занимаемой объектами на изображении, ΔS_i , которые происходят при каждом шаге i, строят зависимость

$$\Delta S_i = f(i).$$

Для того чтобы получить зависимость количества объектов каждого размера N_i , присутствующих в изображении, от величины их площади S_{di} , т. е.

$$N_i = \frac{\Delta S_i}{S_{di}} = f_1(i),$$

необходимо определить их площади. Это можно сделать либо непосредственно, измерив их площади на изображении, либо определить приближенно косвенным

путем исходя из того, что площадь объекта пропорциональна шагу *i*, при котором он был исключен из изображения. На рис. 9.15, *б* приведена зависимость $N_i = f_2(\sqrt{S_{di}})$, полученная первым способом для изображения, приведенного на рис. 9.15, *a*.

9.16. Обнаружение и различение на изображении объектов известной формы

Обнаружение на изображении с равномерным фоном объекта, все параметры которого априорно известны, а именно форма, размеры, координаты, ориентация, распределение яркости в объекте, может быть выполнено путем использования для этой цели шаблона, представляющего неискаженное изображении этого объекта [49]. Алгоритм обнаружения заключается в сопоставлении участка изображения, где предполагается наличие объекта, с шаблоном путем их попиксельного сравнения. Критерием обнаружения объекта является равенство нулю суммы квадратов попиксельных разностей изображения и шаблона, т. е.

$$\sum_{k'=0}^{K-1} \sum_{n'=0}^{N-1} \left[L_{\rm c} \left(k_0 + k', n_0 + n' \right) - L_{\rm magn} \left(k', n' \right) \right]^2 = 0,$$

где $L_{c}(k,n)$ — значение яркости пиксела в *k*-й строке и *n*-м столбце исходного изображения, $L_{\text{шабл}}(k,n)$ — значение яркости пиксела в *k*-й строке и *n*-м столбце шаблона, *K* и *N* — размеры шаблона в пикселах в направлении строк и столбцов, k_0 и n_0 — номера строки и столбца, определяющих положение пиксела, расположенного в левом верхнем углу изображения объекта, которые в рассматриваемом случае считаются априорно известны, поскольку априорно известно положение изображения объекта на изображении.

В том случае, если изображение искажено помехами, но все параметры объекта попрежнему априорно известны, сумма средних квадратов попиксельных разностей должна сравниваться с порогом P, величина которого определяется уровнем помех и заданным значением вероятности ложной тревоги. Если она меньше порога или равна ему, считается, что на изображении присутствует данный объект:

$$\sum_{k'=0}^{K-1} \sum_{n'=0}^{N-1} \left[L(k_0 + k', n_0 + n') - L_{\text{шабл}}(k', n') \right]^2 \le P.$$
(9.13)

Достоинством описанного метода является то, что при белом гауссовом шуме на изображении он обеспечивает максимально возможную вероятность правильного обнаружения объекта, которая совпадает с вероятностью правильного обнаружения предсказываемой теорией статистических решений.

Однако на практике часть параметров, описывающих объект на изображении, где его присутствие требуется обнаружить, как правило, бывает заранее неизвестна. К таким параметрам могут относиться: точные координаты объекта, его ориентация, размеры, распределение яркости в объекте, а в более сложных случаях и другие. Так в случае, если априорно неизвестными параметрами являются координаты объекта на изображении k_0 и n_0 , то алгоритм его обнаружения включает в себя перебор всех возможных значений k_0 и n_0 с проверкой для каждого значения k_0 и n_0 , условия (9.13).

Если неравенство выполняется, принимается решение о наличии объекта на изображении, причем за его положение принимаются координаты k_0 и n_0 , при которых левая часть неравенства минимальна. В данном случае количество вычислительных операций по сравнению с предыдущим случаем заметно возрастает. Еще более возрастает количество вычислительных операций, если априорно неизвестными оказываются и другие параметры, например, ориентация и т. д.

Задача опознавания (различения) ряда отличающихся между собой объектов решается аналогичным образом. Проиллюстрируем это простым примером. Будем считать, что нам априорно известно, что на изображении присутствует один из Mобъектов, и требуется определить, какой именно из M объектов присутствует на изображении. Кроме того, нам априорно известны все параметры изображений этих объектов, включая их координаты. В этом случае для определения, какой именно из M объектов присутствует, необходимо вычислить M сумм и принять решение в пользу изображения *j*-го объекта, для которого выполняется условие

$$\sum_{k'=0}^{K-1} \sum_{n'=0}^{N-1} \left[L_{c}\left(k_{0}+k',n_{0}+n'\right) - L_{\text{шабл}}\left(k',n',j\right) \right]^{2} \leq \sum_{k'=0}^{K-1} \sum_{n'=0}^{N-1} \left[L_{c}\left(k_{0}+k',n_{0}+n'\right) - L_{\text{шабл}}\left(k',n',i\right) \right]^{2},$$
(9.14)

где i = 1, 2, ..., M, $i \neq j$. Другими словами, решение принимается в пользу того объекта, шаблон изображения которого меньше всего отличается от его изображения на анализируемом изображении.

Этот алгоритм может быть легко распространен на случаи, когда ряд параметров изображений объектов априорно не известны, подобно тому, как это было сделано в задаче обнаружения на изображении объектов, что очевидно и в пояснении не нуждается.

Глава 10



Объемные изображения

10.1. Стереоскопические изображения

При наблюдении близко расположенных предметов двумя глазами зритель ощущает их объемность (стереоскопичность). Объясняется это тем, что при рассматривании близкорасположенных объектов зритель взглядом как бы охватывает наблюдаемый объект с разных сторон. Рисунок 10.1 поясняет сказанное.



Рис. 10.1. К пояснению стереоскопического эффекта

Чем ближе к наблюдателю расположен объект, тем больше угол α между оптическими осями глаз, и тем большую часть объекта видит наблюдатель. При наблюдении удаленных объектов оси зрения почти параллельны, изображения объекта на сетчатках глаз почти идентичны и стереоэффект исчезает. Таким образом, стереоэффект проявляется только при наблюдении близко расположенных объектов, в чем не трудно убедиться. Кроме того, вследствие неидентичности изображений наблюдаемого объекта на сетчатках правого и левого глаз благодаря явлению параллакса возникает дополнительная возможность оценки расстояния до него. Поясним это следующим примером, для чего обратимся к рис. 10.2. При наблюдении точечного объекта, удаленного от наблюдателя на расстояние l, его изображения на сетчатках глаз вследствие явления параллакса будут смещены на величины a_1 и a_2 относительно их положения в случае его бесконечного удаления. На этом рисунке использованы следующие обозначения: СЛ и СП — соответственно сетчатки левого и правого глаз, ЗЛ и ЗП — зрачки левого и правого глаз, N — точечный объект, M — плоскость, в которой расположен точечный объект, n_1 и n_2 — проекции точечного объекта на сетчатки глаз, b — расстояние между зрачками, так называемый базис, f — фокусное расстояние зрачка.



Рис. 10.2. К пояснению параллакса

Линейный параллакс при этом, как не трудно видеть, будет равен

$$p = A - b = |a_1| + |a_2|.$$

Из подобия треугольника с вершинами n_1 , N, n_2 и треугольника с вершинами ЗЛ, N, ЗП следует, что b/A = l/(l + f), а, значит,

$$l = bf / (A - b).$$

Принимая во внимание выражение для параллакса *р*, получим окончательно

$$l = bf / p$$
.

Из полученного выражения следует, что величина параллакса при известных базисе и фокусном расстоянии позволяет определять расстояние до наблюдаемого объекта. На этом свойстве основаны методы измерения расстояния до наблюдаемых объектов посредством использования стереотруб. В этих приборах для увеличения точности определения расстояния используют большие базисы и большие фокусные расстояния применяемой оптики.

Обратим внимание на то, что величина угла между оптическими осями глаз позволяет судить о расстоянии до наблюдаемого объекта, но это не то же самое, что стереоэффект. В этом легко убедиться, если взять два идентичных изображения, смещенные друг относительно друга, и поместить их в стереоскоп. При наблюдении такой "стереопары" мы будем видеть плоскую картинку, отодвинутую от наблюдателя на расстояние, которое зависит от величины взаимного смещения изображений, составляющих "стереопару". Чем меньше взаимное смещение, тем дальше, и соответственно, крупнее наблюдатель будет видеть картинку.



Рис. 10.3. Стереоскопическое окно совмещено с ближайшими объектами сцены



Рис. 10.4. Стереоскопическое окно размещено впереди всех объектов сцены



Рис. 10.5. Стереоскопическое окно размещено таким образом, что часть объектов сцены выступают впереди него, а часть расположена позади

Стереоэффект используют для того, чтобы фотографии и киноизображения сделать более натуральными. С этой целью был разработан ряд технологий получения стереоизображений и их демонстрации.

При работе со стереоизображениями вводится понятие *стереоскопического окна*. Под стереоскопическим окном понимается пространственная граница рисунка (плоскость), через которую мы рассматриваем стереоскопическое изображение. По желанию это окно можно установить в любой плоскости изображения, например, в одной плоскости с ближайшим объектом сцены, позади сцены или внутри нее. Хотя большинство стереоизображений делаются таким образом, чтобы ближайшие объекты, составляющие сцену, находились бы в плоскости стереоскопического окна, однако иногда значительную часть сцены помещают за стереоскопическим окном и только отдельные объекты этой сцены выступают вперед. В ряде случаев, поступая таким образом, можно добиться интересных эффектов, но далеко не всегда. Проиллюстрируем это примерами.

В первом случае (рис. 10.3) для того чтобы дерево и начало дорожки находились в плоскости стереоскопического окна, а остальные объекты сцены позади него, необходимо сделать так, чтобы a = a' и b = b'. Из этого можно сделать вывод, что когда ближайшие предметы находятся на одном и том же расстоянии от левого края изображения на правом и левом рисунке, то они лежат в плоскости стереоскопического окна, при этом выступающих объектов нет.

Во втором случае (рис. 10.4) для того чтобы все объекты сцены находились позади плоскости окна, необходимо выполнить условия a' > a и b' > b. Из этого можно сделать вывод, что когда ближайший объект на правой половине стереопары больше удален от левого края, чем соответствующий ему объект на левой половине, тогда все объекты сцены будут размещены позади стереоскопического окна.

В третьем случае (рис. 10.5) для того чтобы дерево и начало дорожки выступали из стереоскопического окна, необходимо выполнить условия a > a' и b > b'. При этом объекты, которые касаются рамки окна, такие, как дерево, будут выглядеть неестественно, те же, которые не касаются рамки, такие, как дорожка, будут висеть в воздухе и создавать иногда желательный, но не всегда оправданный эффект.

Методы получения стереоизображений

Получение стереоизображений путем фотографирования объекта не представляет сложности. Для того чтобы получить стереопару, необходимо одну и ту же сцену снять дважды. Для получения стереоэффекта после съемки первого кадра фотоаппарат должен быть смещен на некоторое расстояние и повернут так, чтобы снимаемый объект оказался в кадре на том же месте, что и на предыдущем снимке. При наблюдении такой стереопары наблюдатель увидит объект, расположенный в плоскости экрана. Другие же объекты, которые при фотографировании были расположены ближе к месту, из которого проводилась съемка, будут казаться выдвинутыми вперед относительно плоскости экрана, а объекты, которые были располо-
жены дальше, будут казаться расположенными за плоскостью экрана. На рис. 10.6 показана стереопара, при съемке которой объектом, на который наводилась камера, были часы.



Рис. 10.6. Стереопара

Обычно для усиления стереоэффекта расстояния между двумя точками, из которых ведется съемка (оно называется базой), берется большим, чем расстояние между зрачками глаз (около 65 мм). Однако следует иметь в виду, что при выборе чрезмерно большой базы у зрителя сначала появляется ощущение дискомфорта, а затем стереоэффект исчезает и изображения стереопары видятся как два отдельных изображения.

Методы наблюдения стереоизображений

Для того чтобы у зрителя создавалось впечатление рельефности наблюдаемого объекта, т. е. его объемности, необходимо, чтобы на сетчатку каждого из его глаз проецировались соответствующие изображения стереопары. В настоящее время известен ряд методов, посредством которых решается эта задача.

Анаглифный метод представления стереоизображения

Этот метод был описан Шарлем Д'Альмейда в 1858 г. и позже в 1891 г. доработан Луи Дюко дю Ороном. Существо метода заключается в том, что оба изображения стереопары изготовляются на одном носителе, но в дополнительных цветах. Этим носителем может быть бумага, проекционный экран, экран дисплея и т. д. Например, изображение, предназначенное для наблюдения левым глазом, делается красным, а изображение, предназначенное для наблюдения правым глазом, делается голубым, т. е. в дополнительном к красному цвете, как это показано на рис. 10.7 и ЦВ.8, на которых приведено анаглифное изображение, полученное из изображений, представленных на рис. 10.6.



Рис. 10.7. Анаглифное изображение

Для наблюдения такой стереопары используются специальные очки с цветофильтрами. В рассматриваемом случае цветофильтр для левого глаза должен быть красным, а цветофильтр для правого глаза голубым. При наблюдении изображения, показанного на рис. 10.7, через очки с цветофильтрами левый глаз будет видеть только красное изображение, которое для него предназначено, а правый глаз только голубое. Наблюдаемое изображение воспринимается как объемное. Этот метод представления стереоизображений просто реализуется, но обладает двумя недостатками. Используя этот метод, невозможно передать все цветовые оттенки объекта. Метод позволяет передавать зелень листвы, золото и серебро ювелирных изделий, оттенки человеческой кожи, но не может передать цвет ярко-красной розы или цвет ярко-синего бархата. Кроме того, для реализации этого метода необходимы очки с цветофильтрами, что создает определенные неудобства при групповом просмотре стереоизображений.

Метод оптического совмещения изображений стереопары (стереоскопы)

Существо этого метода заключается в том, что посредством несложной оптической системы, состоящей из призм и линз, показанной на рис. 10.8, каждое из изображений стереопары представляется для наблюдения соответствующему глазу зрителя.

Достоинством метода является то, что он позволяет наблюдать цветные изображения, без ограничений на воспроизведение каких-либо цветов. Одно время стереоскопы и наборы стереоизображений к ним были широко распространены. Однако у этого метода имеется существенный недостаток, заключающийся в том, что одновременно наблюдать изображение может только один зритель. Кроме того, необходимость использовать стереоскоп создает для зрителя неудобства, и конечно, это неудобно при групповом просмотре стереоизображений.



Рис. 10.8. Стереоскоп

Вместе с тем этот метод с успехом реализован в специальных очках, предназначенных для погружения в виртуальную реальность, которое по своей сущности индивидуально. В настоящее время такие очки выпускаются промышленностью. Примером могут служить виртуальные очки CyberEyes 3D Vision Deluxe. Параметры этих очков следующие: разрешение — 640×480 пикселов, количество отображаемых цветов — 16,7 млн.

Метод наблюдения стереопары через поляризационные очки

Существо метода заключается в том, что левое и правое изображения проецируются на экран через поляризаторы, плоскость поляризации которых повернута друг относительно друга на угол 90°. При наблюдении изображения зритель использует очки с поляризаторами, плоскость поляризации которых также повернута друг относительно друга на угол 90°. При этом поляризатор перед левым глазом пропускает только левое изображение, у которого плоскость поляризации совпадает с плоскостью поляризации левого поляризатора очков и естественно не пропускает правое изображение с перпендикулярной плоскостью поляризации. Аналогичным образом работает поляризатор очков перед правым глазом. В результате зритель видит перед собой объемное изображение. Недостатками этого метода являются: потеря части светового потока в результате прохождения света через поляризаторы при проекции изображений стереопары на экран, а также сама необходимость использования очков с поляризаторами, что создает неудобства при групповом просмотре стереоизображений.

Метод наблюдения стереопары через жидкокристаллические затворные очки

В рассматриваемом случае для получения объемного изображения, например, на экране компьютера, используют жидкокристаллические "затворные" очки. В этих очках роль стекол играют затворы на жидких кристаллах, которые управляются

сигналами, поступающими от дисплея и делающими "стекла" очков попеременно прозрачными или же не прозрачными. Во время работы системы на экран дисплея также попеременно выводятся изображения, предназначенные для показа левому и правому глазу зрителя. Благодаря тому что переключение очков синхронизировано с выводом левого и правого изображений стереопары, зритель одновременно видит только одно изображение, предназначенное соответствующему глазу. В результате создается иллюзия наблюдения объемного изображения. Как это следует из описания, при использовании данного метода частота кадров дисплея должна быть в два раза выше, чем обычно для того, чтобы обеспечить отсутствие мельканий экрана, поскольку каждый глаз в данном случае видит только половину передаваемых кадров. Метод обеспечивает хорошие результаты, однако необходимость работать с повышенной частотой кадров является его серьезным недостатком. Другим его недостатком является неудобство при групповом просмотре стереоизображений.

Растровый метод представления стереоизображения

Принцип, на котором основан этот метод, несложен. В качестве элемента, посредством которого разделяются и направляются в глаза зрителя каждое из изображений стереопар, используется линзовый растр, показанный на рис. 10.9.



Рис. 10.9. К пояснению растрового метода отображения стереоскопических изображений

Каждое из изображений стереопары разделяется на полоски, количество которых равно количеству цилиндрических микролинз растра и помещается с нижней стороны растра в их фокальной плоскости, так чтобы под каждой микролинзой размещалось бы две полоски: одна — принадлежащая левому изображению, а вторая — правому. Размещение полосок показано на рис. 10.9, *а*. Полоски, принадлежащие левому изображению, не затушеваны, а полоски, принадлежащие правому изображению, затушеваны. Применительно к жидкокристаллическому дисплею полоски изображению стереопары формируются из пикселов, принадлежащих тому или иному изображению стереопары. Луч света, от каждого пиксела, проходя через микролинзу, отклоняется от своего первоначального направления так, как это показано на рис. 10.9, *б*. При этом образуются зоны, из которых видно либо правое, либо левое изображение стереопары. Путем выбора геометрических размеров и материала, из которого изготавливается цилиндрический растр, расстояние между

этими зонами делают приблизительно равным среднему расстоянию между глазами зрителя, которое равно 65 мм. При реализации этого метода, так же, как и при реализации методов, описанных ранее, зритель видит объемное изображение, снятое под одним-единственным ракурсом. Заметим, что разрешение при этом уменьшается в 2 раза, поскольку и на правое, и на левое изображение стереопары теперь приходится только по половине пикселов экрана дисплея.

Принцип, на котором основан описанный метод, позволил разработать многоракурсную систему наблюдения стереоизображений. Особенность этой системы заключается в том, что при съемке делается не пара взаимно смещенных изображений, а серия из 8—9 штук, которая разделяется на полоски и размещается под линзовым растром. При реализации эта система позволяет наблюдать изображения объекта под различными ракурсами путем изменения точки, из которой ведется наблюдение. Естественно, что разрешение при этом соответственно уменьшается. Недавно компанией Philips были продемонстрированы два прототипа 3D-дисплеев: 14,5-дюймовый на 8 ракурсов и 20-дюймовый на 9 ракурсов. При этом разрешение каждого ракурса составляло 341×256 пикселов для 15-дюймовой модели и 533×400 пикселов — для 20-дюймовой модели.

10.2. 2,5 О-изображения

В основе 2,5D-технологии лежит принцип представления изображения в виде фоновой составляющей, на которую наложен ряд изображений, выполненных на прозрачных слоях. Каждый из слоев соответствует одному из планов изображаемой сцены и может независимо от других слоев перемещаться в горизонтальном и вертикальном направлениях относительно фонового компонента, на котором изображен задний план. При рассматривании 2,5D-изображения, зритель видит сквозь части прозрачных подложек, которые не заполнены изображениями объектов, изображения объектов, расположенные на удаленных слоях, а также фоновое изображение. Таким образом, изображения объектов на слоях, расположенных ближе к зрителю, заслоняют собой объекты, изображенные на слоях, расположенных дальше от зрителя. Благодаря этой особенности 2,5D-изображения при имитации изменения точки наблюдения вследствие параллакса у зрителя создается иллюзия глубины сцены. Рисунок 10.10 поясняет принцип организации 2,5D-изображения. На этом рисунке обозначены: a — фоновый компонент изображения, соответствующий заднему плану, δ — компонент, на котором представлен передний план.

На рис. 10.11 показано 2,5D-изображение таким, как видит его зритель. Количество слоев может быть достаточно большим.

Технология 2,5D позволяет создавать иллюзию движения в трехмерной сцене путем взаимного перемещения слоев, например, иллюзию проплывающей местности, наблюдаемой из окна вагона движущегося поезда. Достоинством этой технологии является то, что при ее применении отпадает необходимость в передаче заново каждого нового кадра, вместо этого можно передать один раз изображения фоновой компоненты и слоев, а затем передавать величины смещения слоев и фона друг относительно друга. При этом на передачу движения потребуется очень небольшая затрата двоичных единиц кода. Это достоинство технологии определило область ее применения в Интернете с узкополосными каналами, а также в первых компьютерных играх, в те времена, когда производительность компьютеров была невелика. Приложение Adobe Flash, предоставляющее широкие возможности работы со слоями, может служить примером этого. Оно широко используется в Web-технологиях. В настоящее время описанная технология иногда применяется на киностудиях при создании кинофильмов, предварительно снимаемых видеокамерами в целях упрощения применяемых декораций. Иногда количество слоев, на которых изображаются промежуточные планы, превышает сотню.



Рис. 10.10. К пояснению устройства 2,5D-изображения



Рис. 10.11. Такой видит картинку зритель, рассматривая 2,5D-изображение

К недостатку технологии 2,5D-изображений относится то, что при большом смещении слоев изображения промежуточных планов остаются плоскими, и это становится заметным зрителю.

10.3. Воксельная модель представления трехмерных изображений

Рассмотрение методов представления трехмерных изображений объектов начнем с так называемой воксельной модели [82, 102]. Воксельная модель разбивает весь объем трехмерного изображения на ячейки — вокселы, создавая трехмерный (3D) растр. Воксел — это элемент объема (voxel — volume element). Таким образом, воксельная модель представляет собой растровую модель для трехмерного изображения. На рис. 10.12 приведен трехмерный растр, составленный из вокселов.



Рис. 10.12. Трехмерный растр, составленный из вокселов

Каждый воксел, в случае цветного изображения с использованием моды RGB, характеризуется цветом, для представления которого расходуется 24 бита. Кроме того, каждый воксел характеризуется прозрачностью, которая может изменяться в пределах от нуля до единицы. Для представления прозрачности также расходуется 24 бита. Чем больше вокселов содержит трехмерный растр и, следовательно, чем меньше размеры самих вокселов, тем с большим разрешением представляются объекты в трехмерном пространстве. Однако здесь мы встречаемся с очень серьезной проблемой. Поскольку уменьшение размера вокселов влечет за собой возрастание количества вокселов, заполняющих отображаемый объем, обратно пропорционально третьей степени их линейного размера Δ , то это приводит к сильному увеличению объема памяти, требуемой для сохранения трехмерного изображения. В данном случае требуемый объем памяти составит M = 48N бит, где N — количество вокселов в изображении. Проиллюстрируем это примером. Пусть мы имеем дело с цветным трехмерным изображением, размеры которого равны N = 1000×1000×1000. Проводя вычисления, находим, что требуемый объем памяти в данном случае составит $M = 48 \times 10^9$ бит, т. е. около 6 Гбайт! Вторым недостатком, обусловленным большим количеством данных, используемых для представления изображений в воксельной модели, является малая скорость обработки таких изображений.

Несмотря на отмеченные недостатки, воксельная модель в настоящее время считается перспективной, поскольку имеет следующие достоинства.

Она удобна для использования в медицине, например, при сканировании томографом (computer tomography), поскольку получающиеся при этом изображения срезов сканируемого объекта объединяются в трехмерное изображение для дальнейшего анализа. Воксельная модель применяется также в геологии, сейсмологии и других областях, правда, пока при построении трехмерных изображений с небольшим разрешением.

- Эта модель допускает простое выполнение топологических операций над отдельными объектами, а также всей сценой. Например, для показа разреза достаточно те вокселы, которые заслоняют разрез, сделать прозрачными.
- □ Кроме того, ее достоинством является простота описания сложных объектов и сцен.

Легко заметить, что воксельная модель за исключением некоторых случаев является избыточной. Так, например, если среда, окружающая сцену, прозрачна, то нет необходимости передавать (сохранять) все вокселы сцены, а достаточно ограничиться только теми, которые формируют поверхности объектов, составляющих сцену. Такой способ представления сцены обеспечивает большую экономию в требуемом объеме памяти. Проиллюстрируем это примером. Предположим, трехмерный объект представляет собой шар радиусом *R*. В этом случае при прочих равных условиях количество вокселов, размещающихся в его объеме, будет пропорционально $\frac{4}{3}\pi R^3$, в то время как количество вокселов, располагающихся

в поверхностном слое сферы, будет пропорционально $4\pi R^2$, т. е. в R/3 раз меньше. Если учесть, что при таком методе представления трехмерных изображений, назовем его модифицированным воксельным, на представление пустого пространства вообще не затрачивается память, то выигрыш получается впечатляющим.

Дальнейшим развитием этого метода является векторная полигональная модель. Главной особенностью векторной полигональной модели является то, что она описывает только поверхности объектов, составляющих сцену.

10.4. Векторная полигональная модель представления трехмерных изображений

Векторная полигональная модель в настоящее время находит наиболее широкое применение для представления трехмерных изображений объектов [82]. Ее используют в системах автоматизированного проектирования, компьютерных играх, тренажерах, системах виртуальной реальности и в других областях. При использовании этой модели оболочка объекта формируется набором плоских граней — полигонов. Рисунок 10.13, *а* поясняет сказанное. Часто грани оболочки имеют треугольную форму, но они могут быть также многоугольниками. Таким образом, в памяти компьютера создается трехмерное изображение оболочки объекта, которое можно представить себе как его скульптурное изображение.

Положение в пространстве и форма каждой грани однозначно описываются положением ее границ, которые задаются набором векторов, образующих полигональную сетку (wireframe), показанную на рис. 10.13, δ . В свою очередь, положение векторов определяется положением вершин (vertex) — точек, определяющих начало и конец вектора. Из изложенного следует, что оболочка объекта, в конечном счете, описывается набором координат вершин, которые расположены в точках начала и конца векторов (см. рис. 10.13, δ).



Рис. 10.13. Векторная полигональная модель представления трехмерных изображений: *а* — оболочка объекта; *б* — полигональная сетка

Массив вершин

Индексы вершин	0	1	2	3	4	5	6	7	8	9
Координата х										
Координата у										
Координата <i>z</i>										

Массив граней

Номера граней	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Индексы вершин	0	1	2	0	1	1	3	0	2	4	3	5	8	6	4	7
Индексы вершин	2	3	5	6	5	4	8	8	6	7	7	9	9	9	9	9
Индексы вершин	1	0	1	2	4	3	0	6	5	3	8	4	6	5	7	8

Рис. 10.14. К пояснению способа хранения данных для описания оболочек объектов

Часто для описания оболочек объектов используют два массива. В первый трехмерный массив записывают декартовы координаты всех вершин, принадлежащих объекту. Во второй же массив, размерность которого равна числу вершин, формирующих грань, сохраняют индексы вершин, определяющих грани. Рисунок 10.14 поясняет изложенное для случая, когда грани имеют треугольную форму. Возможны и другие варианты. Размер граней оболочки определяет разрешение трехмерного изображения подобно тому, как размер пиксела, в случае двумерного изображения, или размер воксела в трехмерном изображении. Чем меньшего размера грани (соответственно, большее количество вершин) используются для аппроксимации оболочки объекта, тем больше будет разрешение трехмерного изображения, и тем более точно будет представлена форма объекта. Однако увеличение количества вершин влечет за собой увеличение объема памяти, необходимого для сохранения изображения оболочки объекта.

Векторная полигональная модель позволяет по сравнению с воксельной моделью более экономно описывать трехмерные изображения объектов.

Главным достоинством этой модели является то, что она описывает не весь объем, занимаемый объектом, а лишь его оболочку, что обеспечивает громадную экономию в затрате единиц двоичного кода по сравнению с воксельной моделью, в которой описывается весь объем. Это особенно сильно проявляется при увеличении разрешения трехмерного изображения. Если в воксельной модели требуемый при этом объем памяти растет обратно пропорционально третьей степени линейного размера элемента (воксела), то в рассматриваем случае это возрастание будет обратно пропорционально только второй степени линейного размера разрешаемого элемента.

Кроме того, для сокращения требуемого объема памяти вершины по поверхности оболочки располагают неравномерно. Там, где кривизна поверхности оболочки мала, их располагают на большем расстоянии друг от друга, а там, где кривизна велика, вершины помещают более тесно, как показано на рис. 10.13, *б*. При таком расположении вершин размеры граней на участках с большой кривизной будут малыми, что и обеспечивает приемлемую точность представления оболочки объекта при умеренной затрате двоичных единиц кода.

Требуемый объем памяти в рассматриваемом случае может быть рассчитан следующим образом:

$$M = NV \times CV + NF \times CF \times P,$$

где NV — количество вершин, CV — количество единиц двоичного кода, используемых для записи трех координат каждой вершины, NF — количество граней, CF — количество единиц двоичного кода, используемых для записи индекса одной вершины, P — количество вершин, образующих грань. Для представления одной координаты вполне достаточно четырех байт (float), а для представления номера грани — двух байт (unsigned short). Количество вершин, образующих грань, может быть равно 3, 4, 5 или даже 6.

Сравнивая векторную полигональную модель с воксельной моделью по величине затрат единиц двоичного кода на описание объекта, приведем такие данные: для описания оболочки трехмерного изображения головы человека, занимающего объем 250×250×290 вокселов, в случае воксельной модели потребовалось 18,12 Мбайт, в то время как в случае векторной полигональной модели потребовалось всего лишь 199 Кбайт, т. е. в 91 раз меньше. Для сравнения отметим, что для представления двумерного изображения этого объекта потребовалось 215 Кбайт.

Помимо рассмотренной векторной полигональной модели известна родственная ей аналитическая модель, в которой описание поверхности задается математическими формулами, а также ряд других, которые мы здесь рассматривать не будем.

Завершая этот раздел, обратим внимание на то, что до сих пор речь шла только о представлении оболочек объектов, которые, как уже отмечалось, можно уподобить изображению скульптур изображаемых объектов. Реальные же объекты помимо формы (оболочки) характеризуются и таким свойством, как способность отражать свет, что собственно и делает их видимыми. Поэтому в дальнейшем оболочки мы будем рассматривать как набор плоских граней, которые свет отражают, рассеивают, поглощают, а также пропускают, преломляя его.

10.5. Модель освещения

Отражение света от плоской поверхности, в рассматриваемом случае, от грани оболочки, происходит в соответствии с хорошо известным законом отражения, который гласит, что:

- падающий и отраженный лучи света, а также нормаль к отражающей поверхности лежат в одной плоскости;
- угол между падающим лучом и нормалью равен углу между нормалью и отраженным лучом (рис. 10.15);
- □ кроме того, интенсивность падающего *I* и отраженного *I_s* лучей одинаковы, т. е. при идеальном отражении не происходит потерь энергии.



Рис. 10.15. Зеркальное отражение света

Этот закон выполняется для идеально зеркальной поверхности. Такая поверхность не имеет собственного цвета. Если же отражающая поверхность не идеальна, то интенсивность отраженного луча может быть рассчитана по эмпирической формуле Фонга:

$$I_s = IK_s \cos^p \alpha \,,$$

где K_s — коэффициент пропорциональности ($0 < K_s < 1$), p — показатель степени, который лежит в пределах от 1 до 200.

Если поверхность шероховатая (матовая), имеет место диффузное отражение. При диффузном отражении луч рассеивается равномерно во все стороны. Диффузное отражение описывается законом Ламберта:

$$I_d = IK_d \cos \theta$$
,

где I — по-прежнему интенсивность падающего луча, I_d — интенсивность отраженного луча, K_d — коэффициент, учитывающий свойства материала отражающей поверхности ($0 < K_d < 1$), θ — угол между нормалью и падающим лучом (рис. 10.16).



Поверхность

Рис. 10.16. Диффузное отражение света

В случае диффузного отражения отражающая поверхность имеет свой собственный цвет, т. е. коэффициент K_d зависит от длины волны. При этом наблюдаемый цвет диффузно отражающей поверхности определяется как цветом источника освещения, так и цветом самой поверхности.

Поскольку поверхности реальных непрозрачных объектов не являются ни идеально зеркальными, ни идеально матовыми, то интенсивность лучей, отраженных от таких поверхностей, можно рассчитать по формуле

$$I_{\rm orp} = I \left(K_d \cos \theta + K_s \cos^p \alpha \right). \tag{10.1}$$

Учитывая, что в реальных случаях всегда имеет место подсветка объекта светом, отраженным от других объектов, то для расчетов интенсивности отраженных лучей $I_{\text{отр}}$ применяют формулу, учитывающую это обстоятельство, а именно:

$$I_{\text{orp}} = I_a K_a + \frac{I\left(K_d \cos\theta + K_s \cos^p \alpha\right)}{R+k}, \qquad (10.2)$$

где I_a — интенсивность рассеянного света, K_a — константа, R — расстояние от центра проекции до поверхности, k — константа.

В случае, когда объект является прозрачным, необходимо учитывать закон преломления. Согласно закону преломления:

падающий и преломленный лучи света, а также нормаль к отражающей поверхности лежат в одной плоскости;



Рис. 10.17. Преломление светового луча

□ для углов падения и преломления выполняется следующее соотношение:

$$n_1 \sin \alpha_1 = n_2 \sin \alpha_2 \,,$$

где n_1 и n_2 — абсолютные показатели преломления первой и второй сред соответственно, а α_1 и α_2 — углы падения и преломления (рис. 10.17).

10.6. Определение нормали к поверхности и вектора отражения

Как уже было показано в *разд. 10.5*, для того чтобы определить интенсивность света отражаемого освещаемой поверхностью, необходимо знать косинус угла, заключенного между нормалью N к этой поверхности и источником освещения, а также косинус угла между направлением зеркального отражения света, падающего на поверхность от источника, и линией наблюдения. В случае полигональной модели изображения отражающими поверхностями будут ее грани, поэтому найдем интенсивность света, исходящего от одной из граней, для чего обратимся к рис. 10.18, на котором показана эта грань. Рассмотрим случай, когда грань освещается параллельным пучком света, т. е. от удаленного источника.



Рис. 10.18. Определение интенсивности света, отражаемого гранью

Сначала определим вектор нормали к грани. В целях упрощения решения задачи выполним это следующим образом. Введем новую систему координат x', y', z', которая отличается от прежней x, y, z лишь тем, что ее начало совмещено с одной из вершин рассматриваемой грани, обозначенной на рис. 10.18 цифрой 1, при этом направление координатных осей в обеих системах одинаково. В новой системе координат вектор нормали будет исходить из ее начала. Как известно, координаты вектора нормали могут быть найдены путем вычисления векторного произведения любых двух векторов, лежащих в плоскости рассматриваемой грани. В качестве таких векторов можно использовать любые два ребра грани, например ребра 1-2 и 1-3. Назовем ребро 1-2 вектором **A**, а ребро 1-3 — вектором **B**, как показано на рис. 10.18. Выражая координаты вектора нормали **N** через координаты вершин грани в старой системе координат, имеем:

$$\begin{aligned} x'_{\mathbf{N}} &= (y_2 - y_1)(z_3 - z_1) - (z_2 - z_1)(y_3 - y_1), \\ y'_{\mathbf{N}} &= (z_2 - z_1)(x_3 - x_1) - (x_2 - x_1)(z_3 - z_1), \\ z'_{\mathbf{N}} &= (x_2 - x_1)(y_3 - y_1) - (y_2 - y_1)(x_3 - x_1). \end{aligned}$$

В том случае, если грань не треугольная, а, например, четырехугольная, то расчет нормали можно выполнять по любым трем вершинам грани.

Найдем косинус угла, заключенного между нормалью N к грани и вектором, направленным на источник света S. Как известно, скалярное произведение двух векторов S и N равно

$$\mathbf{S} \cdot \mathbf{N} = |\mathbf{S}| |\mathbf{N}| \cos \theta \,, \tag{10.3}$$

где |S| и |N| — длины векторов S и N соответственно. Кроме того, известно, что

$$\mathbf{S} \cdot \mathbf{N} = x_{\mathbf{S}} x_{\mathbf{N}} + y_{\mathbf{S}} y_{\mathbf{N}} + z_{\mathbf{S}} z_{\mathbf{N}}, \qquad (10.4)$$

где x_{S} , y_{S} , z_{S} и x_{N} , y_{N} , z_{N} — компоненты векторов S и N соответственно. Решая совместно (10.3) и (10.4), находим

$$\cos \theta = \frac{x_{\mathbf{S}} x_{\mathbf{N}} + y_{\mathbf{S}} y_{\mathbf{N}} + z_{\mathbf{S}} z_{\mathbf{N}}}{|\mathbf{S}| |\mathbf{N}|}.$$

Для упрощения вычислений вектор S целесообразно взять единичной длины.

Найдем косинус угла, заключенного между вектором отраженного луча и вектором, направленным на наблюдателя. Рассмотрим единичные векторы \mathbf{R}_1 , \mathbf{S}_1 и \mathbf{N}_1 . Так как вектор нормали к грани, а также векторы падающего и отраженного лучей лежат в одной плоскости, то

$$\mathbf{R}_1 + \mathbf{S}_1 = \mathbf{N}',$$



Рис. 10.19. Определение косинуса угла между вектором отраженного луча и вектором, направленным на наблюдателя

где N' — вектор, представляющий собой диагональ ромба, показанного на рис. 10.19, который по направлению совпадает с нормалью. При этом, как не трудно видеть из рисунка,

$$\mathbf{N'} = \mathbf{N}_1 2 \cos \theta \,,$$

или

$$\mathbf{R}_1 + \mathbf{S}_1 = \mathbf{N}_1 2 \cos \theta \, .$$

Из написанного выражения следует, что

$$\mathbf{R}_1 = \mathbf{N}_1 2 \cos \theta - \mathbf{S}_1 = \frac{\mathbf{N}}{|\mathbf{N}|} 2 \cos \theta - \frac{\mathbf{S}}{|\mathbf{S}|}.$$

Значение cos θ, как и ранее, найдем из скалярного произведения

$$\cos \theta = \frac{\mathbf{N} \cdot \mathbf{S}}{|\mathbf{N}||\mathbf{S}|}$$

Делая подстановку найденного значения $\cos \theta$ в выражение для \mathbf{R}_1 , получим

$$\mathbf{R}_1 = 2\mathbf{N} \frac{\mathbf{N} \cdot \mathbf{S}}{\left|\mathbf{N}\right|^2 \left|\mathbf{S}\right|} - \frac{\mathbf{S}}{\left|\mathbf{S}\right|}.$$

Считая, что при отражении луча не происходит его ослабления, а, следовательно, $\mathbf{R} = \mathbf{R}_1 |\mathbf{S}|$, запишем

$$\mathbf{R} = 2\mathbf{N}\frac{\mathbf{N}\cdot\mathbf{S}}{\left|\mathbf{N}\right|^{2}} - \mathbf{S}$$

при этом для компонентов вектора **R** будем иметь

$$x_{\mathbf{R}} = 2x_{\mathbf{N}} \frac{x_{\mathbf{N}}x_{\mathbf{S}} + y_{\mathbf{N}}y_{\mathbf{S}} + z_{\mathbf{N}}z_{\mathbf{S}}}{x_{\mathbf{N}}^2 + y_{\mathbf{N}}^2 + z_{\mathbf{N}}^2} - x_{\mathbf{S}},$$

$$y_{\mathbf{R}} = 2y_{\mathbf{N}} \frac{x_{\mathbf{N}}x_{\mathbf{S}} + y_{\mathbf{N}}y_{\mathbf{S}} + z_{\mathbf{N}}z_{\mathbf{S}}}{x_{\mathbf{N}}^2 + y_{\mathbf{N}}^2 + z_{\mathbf{N}}^2} - y_{\mathbf{S}},$$

$$z_{\mathbf{R}} = 2z_{\mathbf{N}} \frac{x_{\mathbf{N}} x_{\mathbf{S}} + y_{\mathbf{N}} y_{\mathbf{S}} + z_{\mathbf{N}} z_{\mathbf{S}}}{x_{\mathbf{N}}^2 + y_{\mathbf{N}}^2 + z_{\mathbf{N}}^2} - z_{\mathbf{S}} .$$

При этом косинус угла, заключенного между вектором отраженного луча и вектором, направленным на наблюдателя, найдем, используя скалярное произведение вектора отраженного луча \mathbf{R} и вектора \mathbf{K} , направленного на наблюдателя,

$$\cos \alpha = \frac{\boldsymbol{K} \cdot \boldsymbol{R}}{|\boldsymbol{K}||\boldsymbol{R}|} = \frac{x_{\mathbf{K}} x_{\mathbf{R}} + y_{\mathbf{K}} y_{\mathbf{R}} + z_{\mathbf{K}} z_{\mathbf{R}}}{\sqrt{x_{\mathbf{K}}^2 + y_{\mathbf{K}}^2 + z_{\mathbf{K}}^2} \sqrt{x_{\mathbf{R}}^2 + y_{\mathbf{R}}^2 + z_{\mathbf{R}}^2}}.$$

Для упрощения вычислений векторы S, N и K целесообразно брать единичной длины.

10.7. Методы закраски граней

После того как найдены косинусы углов, определяющих взаимные положения нормали к плоскости грани, источника освещения и наблюдателя, можно вычислить интенсивность лучей, отраженных от каждой из граней, по формуле (10.2).



Рис. 10.20. Иллюстрация к методам закраски граней на примерах сферы и цилиндра: *a* — сфера с "примитивной" закраской; *б* — сфера с закрашиванием граней по методу Гуро; *в* — сфера с закрашиванием граней по методу Фонга;

г — цилиндр с "примитивной" закраской;

д — цилиндр с закрашиванием граней по методу Гуро;

е — цилиндр с закрашиванием граней по методу Фонга

Однако непосредственное применение описанной методики расчета приводит к тому, что на изображении оказываются легко различимыми грани отображаемых объектов. Этот эффект демонстрируется рис. 10.20, *а* и *г*, на которых изображены сфера и цилиндр. Для получения иллюзии гладкой поверхности количество граней векторной полигональной модели должно быть сильно увеличено, однако это влечет за собой увеличение требуемого объема памяти и замедление скорости рисования объекта. Эта проблема решается посредством использования метода Гуро.

Метод Гуро

Метод Гуро состоит в том, что каждая плоская грань объекта закрашивается плавно изменяющимся цветом, который определяется путем интерполяции цветов примыкающих граней. В соответствии с этим методом закрашивание грани выполняется в четыре этапа. Вначале вычисляются нормали к каждой грани. Затем находятся нормали **N** в каждой вершине путем усреднения нормалей, примыкающих к вершине граней. В случае треугольных граней это нормали N_1 , N_2 и N_3 . Рисунок 10.21 поясняет изложенное.

$$N = (N_1 + N_2 + N_3)/3.$$

Используя найденные таким образом нормали в вершинах и формулу (10.2), находятся значения интенсивностей в вершинах. Заключительным этапом метода является закраска граней путем линейной интерполяции значений интенсивностей в вершинах.

Недостатки метода Гуро проявляются при имитации криволинейных зеркальных поверхностей, перпендикулярных линии наблюдения, что хорошо заметно на рис. 10.20, б и д, на которых, особенно на рис. 10.20, б, отсутствует световой блик, обусловленный отражением. Для объяснения причин, обусловивших этот недостаток, обратимся к рис. 10.22, на котором изображена полигональная модель цилиндра, освещаемого из точки, расположенной позади наблюдателя.

Ν₃ **Рис. 10.21.** Закрашивание по методу Гуро

Рис. 10.22. К пояснению недостатка закрашивания, свойственного методу Гуро

На этом рисунке закраской выделена грань, которая ориентирована перпендикулярно к линии наблюдения. Поскольку источник света расположен на линии наблюдения и поскольку согласно законам оптики угол падения равен углу отражения, на изображении этой грани должен присутствовать блик, но он отсутствует.





Обусловлено это принципом, положенным в основу метода Гуро. Действительно, поскольку косинусы углов между нормалями N_a , N_6 , N_B , N_{Γ} и вектором, направленным на источник света, равны в силу очевидной симметрии, интенсивности диффузно отраженных лучей в вершинах также будут равны между собой. Из этого следует, что в результате интерполяции найденные интенсивности диффузно отраженных лучей будут равны между собой и на ребрах, а, следовательно, грань будет закрашена одним тоном. От этого недостатка свободен метод Фонга.

Метод Фонга

Метод Фонга отличается от метода Гуро тем, что для определения интенсивности отраженных лучей света в каждой точке грани интерполируются не интенсивности отраженного света, а векторы нормалей. Согласно алгоритму вначале, как и в случае метода Гуро, определяются нормали к каждой грани полигональной модели объекта. Затем находятся нормали N в каждой вершине модели путем усреднения нормалей, примыкающих к вершине граней. Далее, в каждой точке закрашиваемой грани опять-таки путем линейной интерполяции находится интерполированный вектор нормали. И, наконец, по направлению векторов полученных таким образом нормалей определяется интенсивность отраженных лучей света по формуле (10.2). Этот метод более сложен и для своей реализации требует больше вычислительных операций, чем метод Гуро, однако он дает лучшие результаты, что следует из рассмотрения рис. 10.20, в и e.

10.8. Трассировка лучей

В предыдущих разделах рассматривались случаи, когда имелся один-единственный объект, который освещался одним-единственным источником света, создающим поток параллельных лучей. Однако в реальности все обстоит намного сложнее. Как правило, отображаемая сцена включает в себя множество объектов, обладающих различными оптическими свойствами, которая освещается несколькими источниками света, имеющими различные спектральные характеристики. В этом случае объекты освещаются не только источниками света, которые мы назовем первичными, но также и светом, отраженным от других объектов, интенсивность и спектральный состав которого будет также зависеть от отражающих свойств этих объектов. Для решения этой проблемы были разработаны два метода, которые называются *методом прямой трассировки лучей* и *методом обратной трассировки лучей*.

Метод прямой трассировки лучей

Этот метод заключается в прослеживании (трассировке) лучей, исходящих от источников света, которые называются первичными. Часть из них, не встречая на своем пути объектов, покидают сцену и по этой причине из дальнейшего рассмотрения исключаются. Другая часть лучей встречает на своем пути объекты, от которых зеркально или диффузно отражается или преломляется, образуя вторичные лучи. Некоторые из вторичных лучей попадают в глаз наблюдателя, другие, не встретив

на своем пути объектов, покидают сцену и должны быть исключены из дальнейшего рассмотрения, а некоторые, встретив на своем пути объекты, зеркально или диффузно отражаются от них или преломляются, порождая новое поколение лучей. Алгоритм построения изображения рассматриваемым методом сводится к перебору всех лучей, сопровождаемому при этом отбором только лучей, попадающих в глаз наблюдателя. Недостатком этого метода является большое количество вычислительных операций, связанных с расчетом лучей, которые в дальнейшем не используются. В связи с этим был разработан альтернативный метод — *метод обратной трассировки лучей*.

Метод обратной трассировки лучей

Данный метод заключается в прослеживании (трассировке) лучей, исходящих не от источников света, а в обратном направлении — от точки наблюдения. В этом методе такие лучи называются первичными. При таком подходе к решению проблемы в расчет принимаются только те лучи, которые вносят вклад в формирование изображения, поэтому этот метод позволяет существенно сократить перебор лучей и, как следствие, количество вычислительных операций, связанных с расчетом лучей. При этом методе прослеживаются только первичные лучи, которые, исходя из точки наблюдения, попадают на объекты. Все другие лучи с самого начала исключаются из рассмотрения. Поскольку на объектах лучи поглощаются, зеркально или диффузно отражаются или же преломляются, то определяются, так называемые, вторичные лучи, которые могли породить первичные лучи, исходящие из точки наблюдения. Другими словами, задача состоит в том, чтобы найти лучи, которые, попав на поверхность объекта, породили лучи, попадающие в точку наблюдения. Затем процесс повторяется путем прослеживания (трассировки) только тех вторичных лучей, которые, в свою очередь, при распространении попадут на объекты. Процесс завершается, когда трассируемые лучи достигнут источников света.

Поскольку законы геометрической оптики носят "симметричный" характер, то при реализации метода обратной трассировки лучей каких-либо проблем с этой стороны не возникает. Проблемы технического характера возникают, когда трассируемый луч попадает на диффузно отражающую поверхность, которая одинаково отражает во всех направлениях. В этом случае источником первичного луча может быть любой луч, пришедший от любого объекта, видимого из данной точки, что приводит к необ-ходимости выполнять большое количество вычислительных операций.

При практической реализации метода обратной трассировки лучей вводят ряд ограничений, которые позволяют уменьшить количество вычислительных операций. Примерами таких ограничений могут быть:

- 🗖 ограничение количества итераций, после которых трассировка завершается;
- отражающие поверхности описываются суммой двух компонентов диффузного и отражающего;
- 🗖 при диффузном отражении учитываются только лучи от источников света;
- источники света могут только излучать свет, но не могут его отражать или преломлять, а также некоторые другие ограничения.

10.9. Проекции трехмерных изображений на плоскость

Как уже было отмечено, в основе зрительного восприятия окружающего нас мира лежат центральные проекции наблюдаемых трехмерных объектов на сетчатки глаз. Рассмотрим центральную проекцию трехмерного объекта на плоскость, при этом координатную ось x декартовой системы координат совместим с осью камеры, а оси y и z — с плоскостью проекции, так, как это показано на рис. 10.23, на котором приведена так называемая модель камеры с передней плоскостью изображения [29].



Рис. 10.23. Модель камеры с передней плоскостью изображения

На рисунке использованы следующие обозначения: x_0 , y_0 , z_0 — декартовы координаты точки A, принадлежащей объекту, x_{Π} , y_{Π} , z_{Π} — координаты проекции этой точки на плоскость, f — фокусное расстояние объектива камеры. Такую проекцию можно представить себе как изображение на стекле, совмещенном с передней плоскостью изображения, через которое наблюдатель рассматривает объект. Из подобия треугольников, образованных осью x, проецирующим лучом и координатой z_0 точки A, принадлежащей объекту, в первом случае, и осью x, проецирующим лучом и координатой z_{Π} точки A', принадлежащей проекции объекта, во втором случае, следует, что

$$x_{\rm fi} = 0$$
, (10.5)

$$y_{\rm II} = \frac{y_{\rm o}f}{x_{\rm o} + f},\tag{10.6}$$

$$z_{\rm II} = \frac{z_0 f}{x_0 + f}.$$
 (10.7)

Приведем основные свойства центральной проекции (ее также называют перспективной проекцией). При центральной проекции:

□ не сохраняется отношение длин отрезков и площадей;

прямые линии отображаются прямыми линиями;

□ параллельные прямые отображаются прямыми, сходящимися в одной точке.

В предельном случае, когда фокусное расстояние f объектива камеры стремится к бесконечности, центральная проекция вырождается в параллельную проекцию, а если к тому же, как в рассмотренном случае, передняя плоскость изображения (плоскость проецирования) перпендикулярна оси x, центральная проекция вырождается в аксонометрическую проекцию.

Не касаясь других видов проекций, отметим, что любая проекция принципиально содержит меньше информации, чем трехмерное изображение объекта. Объясняется это тем, что при проецировании теряется информация о глубине (координате x_0). По этой причине при изготовлении чертежей на них приводят как минимум две или больше проекций объекта. Вместе с тем интересно обратить внимание на то, что при рассматривании фотографий или картин, т. е. по существу проекций объектов на плоскость, у зрителя не возникает проблем с определением расстояний до того или иного изображенного объекта. Объясняется это тем, что зритель почти всегда располагает априорной информацией о возможном расположении объектов в изображенной сцене, которую и использует при определении координаты глубины (*см. разд. 3.8 и 10.12*).

10.10. 3D-сканеры

При разработке трехмерных (3D) сцен возможны два подхода. В первом случае все объекты, составляющие сцену, рисуются, например, с помощью таких пакетов программ, как 3ds Max или Maya. Достоинством этого подхода является относительная простота выполнения. Этот подход широко применяется при разработке компьютерных игр, а также сцен виртуальной реальности. Недостаток же его состоит в том, что нарисованная сцена выглядит, как правило, не натуральной, она отличается от реальной сцены так же, как рисунок отличается от фотографии, или же, как мультипликационный фильм от фильма, снятого с натуры.

Второй подход состоит в том, что 3D-изображения объектов сцены получают путем сканирования реальных объектов, благодаря чему достигается реалистичность трехмерных изображений. Этот способ создания трехмерных сцен отличается большой трудоемкостью и высокими денежными затратами ввиду высокой стоимости 3D-сканеров и большой затраты труда разработчиков.

3D-сканирование реальных объектов применяют:

в развлекательной индустрии. В настоящее время ни одна крупная студия не обходится без сканирования объектов реквизита и, главное, без сканирования ведущих актеров, что позволяет заменять их дублерами при чрезвычайных обстоятельствах. Так, например, в ходе съемок фильма "Гладиатор" умер ведущий актер Оливер Рид, но съемки не были остановлены, поскольку умершего актера заменил дублер с оцифрованным лицом Оливера Рида;

- при разработке сцен виртуальной реальности;
- □ в системах идентификации личности;
- 🗖 при выполнении геодезических работ;
- 🗖 для сканирования ювелирных украшений.

Существующие технологии 3D-сканирования можно подразделить на две большие группы — это технологии, использующие контактные методы сканирования, и технологии, использующие бесконтактные методы сканирования. Рассмотрим кратко эти технологии и реализующие их сканеры.

Контактный метод сканирования и реализующие его сканеры

В основе принципа действия контактного сканера лежит обводка поверхности сканируемого объекта оператором, которая выполняется вручную. Сканеры, основанные на этой технологии, снабжены специальным механическим приспособлением, которое называется сенсором и представляет собой щуп. Перед сканированием на сканируемый объект наносится сетка, размер ячеек который выбирается таким образом, чтобы в областях высокой кривизны поверхности сканируемого объекта размер ячеек был бы минимальным, а в областях с малой кривизной поверхности — наибольшим. В точках, где линии сетки пересекаются, посредством сенсора производятся измерения их координат, которые затем вводятся в компьютер.

Основным достоинством и одновременно недостатком этого типа сканеров является то, что трехмерная модель объекта создается в памяти компьютера оператором вручную, что приводит к значительным временным затратам. Вторым недостатком бесконтактных сканеров является невозможность считывать текстуру. Кроме того, сканеры этого типа не могут применяться для сканирования больших объектов, поэтому применение контактных сканеров при создании фильма или компьютерной игры невозможно. Однако имеются области, где они с успехом находят свое применение ввиду их невысокой стоимости, например, для сканирования скульптур, моделей корпусов всевозможных небольших деталей, где не требуется считывания текстуры сканируемого объекта, и т. д.

Бесконтактные методы сканирования и реализующие их сканеры

Бесконтактные методы сканирования разделяются на:

- □ лазерные методы сканирования;
- 🗖 оптические методы сканирования;
- 🗖 ультразвуковые методы сканирования.

В свою очередь, оптические методы сканирования разделяются на методы, в которых используется:

🗖 метод фокусного ряда;

- □ стереоэффект;
- □ структурированный свет;
- 🗖 диффузное отражение света сканируемыми объектами.

Что касается ультразвуковых методов сканирования, то здесь мы о них только упоминаем, поскольку в литературе отсутствует сколько-нибудь подробное их описание.

Обращаясь к рассмотрению бесконтактных методов сканирования трехмерных объектов, следует отметить наличие так называемой проблемы сшивания частей получаемого в результате сканирования трехмерного изображения. Существо проблемы заключается в том, что при сканировании сканирующее устройство "видит" лишь часть сканируемого объекта, и для получения 3D-изображения всего объекта необходимо просканировать его со всех сторон, а затем соединить (сшить) полученные части 3D-изображений вместе. Поскольку в большинстве рассматриваемых далее методов координата глубины измеряется лишь для отдельных точек сканируемого объекта, а для остальных находится путем интерполяции, и, следовательно, на различных, обычно слегка перекрывающихся частях изображений, полученных путем сканирования при различных взаимных положениях объекта и сканирующего устройства, ее значения немного расходятся, возникает необходимость ручного согласования. Это приводит к существенному увеличению трудоемкости.

Рассмотрим кратко перечисленные выше методы и реализующие их сканеры бесконтактного сканирования.

Лазерные сканеры

При использовании этого метода объект сканируется лазерным лучом. В то время как горизонтальная и вертикальная координаты сканируемой точки объекта определяются, исходя из направления лазерного луча, третья координата, координата глубины *x*, находится путем измерения времени задержки *t* сигнала, отраженного от поверхности сканируемого объекта

$$x=\frac{ct}{2},$$

где c — скорость распространения света. В сканерах этого типа используются лазерные датчики, а также цифровая фототехника. Применение цифровой фототехники позволяет создавать 3D-модели объектов с текстурами. Достоинством лазерных сканеров является возможность работать "в автоматическом режиме", а также возможность сканировать крупные объекты, что, как уже было отмечено, невозможно при использовании контактных сканеров. Слова "в автоматическом режиме" взяты в кавычки, поскольку в полном смысле автоматических 3D-сканеров в настоящее время не существует. Причиной этому является наличие проблемы сшивания изображений. Результаты сканирования требуют обычно трудоемкого последующего редактирования, на которое уходит много времени. Лазерные сканеры в настоящее время применяются для сканирования только крупных объектов, при решении таких задач, как:

- трехмерное сканирование промышленных объектов (сложные сооружения, цеха, вышки, башни);
- 🗖 съемка заводских территорий, железнодорожных станций, аэропортов;
- 🗖 съемка улиц и площадей;
- 🗖 съемка в целях профилирования дорог, съемка мостов;
- съемка в целях реконструкции и строительства зданий, а также археологических памятников;
- съемка в целях архивации трехмерных данных для объектов исторического наследия;
- 🗖 съемка объектов в горной промышленности и тоннелестроении.

В качестве примера сканеров этого типа можно назвать лазерный сканер Trimble GX.

Основными недостатками сканеров этого типа являются:

- слишком большая относительная погрешность измерения глубины при сканировании небольших объектов;
- 🗖 высокая стоимость сканеров этого типа.

Сканеры, в которых используется метод фокусного ряда

Как известно, любая оптическая система обеспечивает максимальное разрешение только в одной плоскости. При удалении от этой плоскости на расстояние d разрешение уменьшается пропорционально квадрату этого расстояния. Чем меньше глубина разрешения, тем сильнее выражена эта зависимость. В рассматриваемом методе это свойство использовано для определения координаты глубины сканируемого объекта [102, 155]. Существо метода заключается в том, что вначале получается ряд изображений при различных настройках оптической системы, каждая из которых обеспечивает максимальное разрешение в своей плоскости, удаление этой плоскости от оптической системы точно известно, в результате получается так называемый фокусный ряд. Затем для каждой точки изображения (пиксела, применительно к цифровым изображениям) методом сравнения значений сигнала во всех изображениях фокусного ряда находится та плоскость, в которой изображение мелких деталей объекта воспроизводится с максимальным разрешением, что соответствует максимальному значению сигнала от мелкой детали объекта, расположенной в этом месте. Зная удаление найденной плоскости от оптической системы, находят удаление (третью координату) данной точки (пиксела) 3D-изображения объекта. В тех областях изображений объекта, где мелкие детали отсутствуют, третья координата находится путем интерполяции при использовании в качестве опорных точек тех пикселов, для которых она определена.

Недостатками рассмотренного метода являются:

- □ недостаточная точность получаемых результатов;
- □ проблема определения третьей координаты для больших бесструктурных областей сканируемого объекта, т. е. для областей, в которых отсутствуют мелкие детали;
- □ а также "проблема сшивания".

Сканеры, в которых используется стереоэффект¹

В этом методе для измерения координаты глубины 3D-изображения используются две одинаковые камеры, оптические оси которых направлены параллельно друг другу, а расстояние (базис) между которыми равно *b*. Рисунок 10.24 поясняет сказанное.



Рис. 10.24. К пояснению измерения расстояния до наблюдаемого объекта

При расстоянии l между точкой объекта, до которой измеряется расстояние (глубина), и линией, соединяющей центры левого и правового объективов камер O_{π} и O_{π} , положения изображений точки объекта, создаваемые правой и левой камерами на их светочувствительных поверхностях СП_л и СП_п, будут смещены на величины a_1 и a_2 относительно центральных осей. На этом рисунке использованы следующие обозначения: n_1 и n_2 — проекции точечного объекта на изображениях левой и правой камер на их светочувствительные поверхности, N — точечный объект, M — плоскость, в которой расположен точечный объект, b — расстояние между объективами камер, так называемый базис, f — фокусное расстояние объективов. Линейный параллакс при этом, как не трудно видеть, будет равен

$$p = A - b = |a_1| + |a_2|.$$
(10.8)

¹ См. [102, 135].

Из подобия треугольников $n_1 N n_2$ и $O_{\pi} N O_{\pi}$ следует, что b/A = l/(l+f), а, следовательно,

$$l = bf / (A - b).$$

Принимая во внимание выражение для параллакса р, получим окончательно

$$l = bf / p$$
.

Из полученного выражения следует, что величина параллакса при известном базисе и фокусном расстоянии камер позволяет однозначно определять расстояние до любой точки наблюдаемого объекта, отличающейся по яркости и/или цвету от соседних точек.

Недостатки технологии, основанной на использовании стереоизображений, заключаются в том, что при ее применении трудно обеспечить точное соответствие для пикселов на изображениях объекта, полученных с двух камер в бесструктурных областях, вследствие чего невозможно обеспечить достаточную точность получаемых 3D-изображений, а также наличие "проблемы сшивания".

Метод сканирования, основанный на использовании структурированного света

Проблема, возникающая при сканировании бесструктурных областей объектов, с которой встречаются при использовании метода фокусного ряда и метода, основанного на стереоскопическом эффекте, решается путем применения так называемого метода структурированного света [113, 152]. Рассмотрим принцип, положенный в основу этого метода, для чего обратимся к рис. 10.25, на котором показан сканируемый объект (скульптурное изображение головы), содержащий большие по площади области, в которых отсутствуют мелкие детали.



Рис. 10.25. К пояснению метода сканирования, основанного на использовании структурированного света (проецируется точка)

Отсутствие мелких деталей в этих областях можно возместить, создав их искусственно, путем проецирования на них мелких деталей посредством проектора. В рассматриваемом случае проектор проецирует на объект изображение яркой точки.

Далее изображение сканируемого объекта фотографируется посредством двух фотокамер. Оптическая ось первой из фотокамер совпадает с оптической осью проектора, а оптическая ось второй фотокамеры по отношению к оптической оси первой фотокамеры расположена под углом 90°. Кроме того, оптические оси обеих фотокамер расположены в одной плоскости, а расстояние фотокамер от фотографируемого объекта выбрано одинаковым и достаточно большим для того, чтобы перспективными искажениями можно бы было пренебречь. На рис. 10.26 показаны фотографии, полученные посредством двух фотокамер, которые представляют вид спереди (рис. 10.26, a) и вид слева от нас (рис. 10.26, b).



Рис. 10.26. Изображения, полученные двумя камерами (проекция точки): *а* — вид спереди; *б* — вид слева

Имея изображение точки на этих изображениях, не составляет труда определить ее положение в пространстве. Так рис. 10.26, a позволяет определить ее положение по вертикали и горизонтали, а рис. 10.26, δ позволяет найти ее положение по глубине относительно первой фотокамеры. Таким образом, помещая точку в различные места бесструктурных областей объекта, можно легко определить все три их координаты. Однако решение задачи можно существенно упростить, используя тот факт, что на виде слева одна из координат, а именно координата по вертикали, повторяет координаты, а именно координаты глубины, проецировать на сканируемый объект не точку, а прямую линию, как это показано на рис. 10.27.

На рис. 10.28 показаны изображения, получаемые с двух фотокамер: вид слева и вид спереди. Не трудно видеть, что поскольку изображения соответствующих точек линии на том и другом изображениях находятся на одной высоте, то для каждой точки поверхности, на которую спроецирована линия, легко определить все три ее координаты.



Рис. 10.27. К пояснению метода сканирования, основанного на использовании структурированного света (проецируется прямая линия)



Рис. 10.28. Изображения, полученные двумя камерами (проекция прямой линии): *а* — вид спереди; *б* — вид слева



Рис. 10.29. Решетка



Рис. 10.30. К пояснению метода сканирования, основанного на использовании структурированного света (проецируется решетка)



Рис. 10.31. Изображения, полученные двумя камерами (проекция решетки): *а* — вид спереди; *б* — вид слева

На практике при сканировании трехмерных объектов идут еще дальше, проецируя на сканируемый объект не одиночную линию, а решетку, состоящую из ряда линий, подобную показанной на рис. 10.29.

При этом схема измерения выглядит так, как это показано на рис. 10.30, а изображения, полученные с первой и со второй фотокамер, т. е. изображения вида спереди и вида слева, имеют вид, показанный на рис. 10.31, *а* и *б* соответственно.

В этом случае при сканировании важно с самого начала правильно задать соответствие линий на обеих фотографиях, что несложно, начав, например их нумерацию на обоих изображениях. Таким образом, при использовании метода структурированного света находятся координаты точек той части поверхности, на которую спроектирована решетка, в независимости от того, имеются ли на ней мелкие детали, или же они отсутствуют. Координаты же остальных точек определяются методом интерполяции. Это является определенным недостатком, т. к. требует использования решеток с плотно расположенными тонкими линиями. Кроме того, при работе с объектами, различные части которых имеют сильно различающуюся детализацию, необходимо вручную добавлять точки в создаваемую модель, в областях высокой детальности.

Кроме описанной технологии, в которой используется решетка из прямых линий, известны еще технологии сканирования по зонам и по точкам, но они применяются сравнительно редко.

Ранее был рассмотрен метод сканирования трехмерных объектов, в результате которого получается трехмерное изображение его оболочки. При реализации этого метода обычно одновременно получается и текстура сканируемого объекта путем его фотографирования при отсутствии решетки.

Несмотря на кажущуюся простоту описанной технологии и рекламируемую изготовителями сканеров автоматичность режима, процесс получения 3D-изображений оказывается также весьма трудоемким и требует больших затрат ручного труда, что является серьезным недостатком этого метода. Здесь, как и ранее, при получении 3D-изображения объекта по нескольким проекциям необходимо производить сшивание этих изображений, а для этого нужно предварительно произвести калибровку камер, т. е. определить их взаимное расположение.

10.11. Трехмерное сканирование, основанное на диффузном отражении света сканируемыми объектами

Рассматриваемый далее метод 3D-сканирования для определения координаты глубины использует распределение светотеней на двумерных изображениях объектов. Это оказывается возможным потому, что количество света, попадающего в объектив камеры от каждой точки сканируемого объекта, определяется как отражающими свойствами поверхностей наблюдаемых объектов, так и взаимным расположением этих поверхностей и источников света. Особенно просто этот метод реализуется при сканировании объектов, поверхность которых диффузно отражает подающий на нее свет [44, 51, 59, 102].

Рассмотрим процесс образования светотеней на изображении объекта, при этом будем считать, что объект диффузно отражает падающий на него свет. Согласно закону косинусов Ламберта, сила отраженного света I_1 диффузно отражающей поверхностью одинакова во всех направлениях и равна

$$I_1 = IK\cos\theta, \tag{10.9}$$

где I — сила падающего света, K — коэффициент отражения поверхности (0 < K < 1), θ — угол между нормалью **N** к отражающей поверхности и направлением на источник света. Сам факт зависимости силы диффузно отраженного света

от углового положения отражающей поверхности по отношению к направлению, откуда происходит ее освещение, дает принципиальную возможность определять угловую ориентацию отражающей поверхности в различных точках пространства, а по ним и всю поверхность. Это обстоятельство может быть использовано в целях получения трехмерных изображений при сканировании реальных объектов.

Рассмотрим эту проблему подробнее, для чего обратимся к рис. 10.32. В целях простоты дальнейшего изложения будем считать, что источник параллельных лучей света расположен в одной из координатных плоскостей, например, в плоскости *xy*. Поместим камеру, посредством которой будем измерять силу света, диффузно отражаемого различными точками объекта, на оси *x*. Расстояние от камеры до сканируемого объекта выберем достаточно большим для того, чтобы можно было бы пренебречь перспективными искажениями.

На этом рисунке показана диффузно отражающая свет поверхность, которая помещена в начало координат, а также нормаль к этой поверхности N, которая составляет с направлением на источник света I угол θ .



Рис. 10.32. Диффузно отражающая свет поверхность, произвольно ориентированная в пространстве

Обратим внимание на то, что угол θ однозначно не определяет угловое положение в пространстве нормали к отражающей поверхности. Для того чтобы однозначно задать угловое положение в пространстве этой нормали, необходимо задать два угла, а именно: угол α — азимут и угол β — угол места. При этом, как не трудно видеть, имеет место очевидное соотношение

$$\cos\theta = \cos\beta\cos(\xi - \alpha). \tag{10.10}$$

Это соотношение может быть получено путем последовательного проецирования нормали N сначала на координатную плоскость xy, а затем проецирования проекции нормали A на линию I. При этом будем иметь

$$\mathbf{B} = \mathbf{N}\cos\beta\cos(\xi - \alpha) = \mathbf{N}\cos\theta.$$

Делая подстановку соs θ из формулы (10.10) в формулу (10.9) и выполняя несложные преобразования, получим:

$$I_1 = IK\cos(\xi - \alpha)\cos\beta, \qquad (10.11)$$

где ξ — угол между осью x и направлением на источник света. В этом выражении известными величинами считаются сила света I, которым освещается диффузно отражающая поверхность, сила света I_1 , диффузно отраженного этой поверхностью, а также угол ξ , поскольку они могут быть измерены. Неизвестными же величинами являются углы α , β и коэффициент отражения поверхности K. Имея всего одно уравнение и три неизвестных, невозможно определить эти неизвестные. Однако проблема может быть легко решена, если при измерениях воспользоваться дополнительными источниками света. Действительно, расположив дополнительный источник параллельных лучей света, например, на оси x и измерив силу света I_0 , отраженного рассматриваемой поверхностью от этого источника, можно составить дополнительное уравнение

$$I_0 = IK \cos\alpha \cos\beta. \tag{10.12}$$

Аналогичным образом можно получить еще одно недостающее уравнение, если дополнительный источник света поместить так, чтобы он располагался на линии, лежащей в координатной плоскости *xz*

$$I_2 = IK\cos(\zeta - \beta)\cos\alpha, \qquad (10.13)$$

где ζ — угол между осью *x* и направлением на источник света. Решая совместно уравнения (10.11)—(10.13), получим

$$tg\alpha = \frac{I_1/I_0 - \cos\xi}{\sin\xi},$$
 (10.14)

$$tg\beta = \frac{I_2/I_0 - \cos\zeta}{\sin\zeta},$$
 (10.15)

$$K = \frac{I_0}{I\cos\alpha\cos\beta}.$$
 (10.16)

Формулы (10.14) и (10.15) полностью определяют угловое положение в пространстве нормали к диффузно отражающей поверхности объекта в каждой ее точке. Располагая этими данными, не составляет труда найти тангенсы углов между плоскостями, касательными к поверхности объекта в этих точках, и координатными осями у и z, которые, как не трудно видеть, соответственно равны ctg α , ctg β . Процедура построения поверхности видимой части объекта заключается в том, что вначале для произвольной точки его поверхности с координатами y_0 и z_0 задается произвольное значение координаты x_0 , которое удобно принять равным нулю. Поскольку целью сканирования является определение формы объекта (его оболочки), а не его расположение в пространстве, то это не влияет на конечный результат. Затем, перемещаясь по произвольному пути из точки с координатами x_0 , y_0 , z_0 в точку с координатами y_1 , z_1 путем наращивания координаты x, определяется координата глубины x_1 в этой точке в соответствии с выражением

$$x_{1} = \int_{y_{0}}^{y_{1}} \operatorname{ctg}\alpha(y, z_{1}) dy + \int_{z_{0}}^{z_{1}} \operatorname{ctg}\beta(y_{1}, z) dz.$$
(10.17)

В случае дискретных изображений интегралы в формуле (10.17) заменяются суммами, а координаты у и z заменяются номерами столбцов и строк, в которых расположены пикселы исходных двумерных изображений

$$M = \sum_{n_0}^{N} \operatorname{ctga}(L, n) + \sum_{l_0}^{L} \operatorname{ctg}\beta(l, N).$$
(10.18)

В этой формуле использованы следующие обозначения: l и n — соответственно текущие номера строки и столбца изображения, полученного посредством камеры, l_0 и n_0 — номера строки и столбца начальной точки поверхности, для которой координата глубины принята равной нулю, а L и N — номера строки и столбца пиксела, для которого измеряется третья дискретная координата M, шаг дискретизации которой равен шагу дискретизации исходных двумерных изображений.

Таким образом, для того чтобы определить третью координату (глубину) любой точки сканируемого объекта на его двумерном изображении, необходимо в общем случае измерить яркость этой точки при трех различных расположениях источника освещения. Обратим внимание на то, что при этих измерениях освещение точки сканируемого объекта, для которой производится измерение, не должно быть затенено. В случае сканирования простейших объектов достаточно каждую их точку поочередно осветить с трех позиций. Более сложные объекты, например лицо человека, потребуют уже большего количества источников света, используемых попеременно для освещения различных его частей, т. к. отдельные его части могут быть затенены при некоторых расположениях источников освещения. При этом положения этих источников могут быть любыми удобными для измерений.

До сих пор речь шла об определении третьей координаты той части объекта, которая видна камере, т. е. только части всего объекта. Напомним, что в рассмотренном случае камера располагалась на оси *x*. Для получения трехмерного изображения всего объекта необходимо произвести его съемку со всех сторон и произвести сшивку всех полученных при этом трехмерных изображений его частей. В отличие от рассмотренных ранее методов сканирования, проблема сшивки в данном случае намного проще, что является серьезным достоинством этого метода. Это обусловлено тем, что в отличие от методов, рассмотренных выше, третья координата сканируемого объекта (глубина) в данном методе определяется для каждого пиксела исходных двумерных изображений, благодаря чему при сшивке отдельные фрагменты трехмерного объекта легко совмещаются. Недостатком рассмотренного метода, как, впрочем, и других методов 3D-сканирования, использующих свет, является то, что он ориентирован на работу с объектами, диффузно отражающими свет. Для сканирования объектов, имеющих блестящие, зеркальные и прозрачные поверхности, таких, как, например, окрашенный кузов автомобиля, их необходимо покрывать антибликовым составом (например, как это принято в используемых методах, путем распыления из баллончика), создающим белую матовую поверхность.

Далее в качестве иллюстрации приводятся результаты экспериментальной проверки описанного метода 3D-сканирования. В приводимом примере был выбран сравнительно простой объект, а именно конус, поверхность которого диффузно рассеивала свет. Конус располагался таким образом, что его вертикальная ось совпадала с координатной осью *z*. Простота объекта и его расположение позволили при определении третьей координаты обойтись освещением каждой точки его поверхности только двумя, а не тремя источниками света, например, фронтальным и боковым, как это показано на рис. 10.33. Так можно было поступить потому, что в данном случае яркость пикселов на контурах объекта априори постоянна и известна. При боковом освещении слева яркость левого контура максимальна и постоянна, а при фронтальном освещении яркость левого и правого контуров равна нулю.



Рис. 10.33. Изображения конуса, сделанные при неизменном положении камеры и объекта, различающиеся между собой только типом освещения: *а* — боковое освещение слева; *б* — боковое освещение справа; *в* — фронтальное освещение

В данном случае для расчета третьей координаты (глубины) 3D-изображения можно было воспользоваться сокращенным вариантом формулы (10.18)

$$M = \sum_{n_0}^{N} \operatorname{ctga}(L, n).$$
(10.19)

При этом мы задавали на контуре изображения конуса, показанного на рис. 10.33, *в*, одно и то же, заранее известное значение интенсивности, в рассматриваемом слу-

чае равное нулю, а затем перемещались вдоль строки в точку, координату глубины которой определяли. При выполнении эксперимента вначале были получены двумерные цифровые изображения конуса при трех различных положениях источников освещения: спереди, слева и справа, которые показаны на рис. 10.33. Эти изображения различались между собой только типом освещения, поскольку съемка делалась при неизменном положении камеры и объекта. Изображение конуса, показанное на рис. 10.33, а, было получено при боковом освещении слева, а изображение, показанное на рис. 10.33, б, было получено при боковом освещении справа. На этих изображениях фон, на котором виден конус, оставался неосвещенным, поскольку лучи от источников света по нему скользили, не освещая его. Изображение конуса, показанное на рис. 10.33, в, было получено при фронтальном освещении. В этом случае фон, на котором виден конус, освещен. Далее, для каждого пиксела изображения, показанного на рис. 10.33, а, и соответствующего ему пиксела изображения, показанного на рис. 10.33, в, а также для каждого пиксела изображения, показанного на рис. 10.33, б и в, измерялись интенсивности, величины которых прямо пропорциональны силам диффузно рассеянного света. После этого по формуле (10.14) определялся сначала тангенс, а затем котангенс угла α , после чего по формуле (10.19) находилась координата глубины M. Дискретные координаты L, *N* и *M* были использованы для построения 3D-изображения сканируемого объекта, которое показано на рис. 10.34, б. На рис. 10.34, а приведены результаты этих измерений в виде линий постоянной высоты, представляющие собой, как и следовало ожидать, окружности, радиус которых линейно убывает по мере приближения к вершине конуса.

На рис. 10.34, б показано аксонометрическое изображение конуса, построенное на основании результатов измерений посредством функции mesh системы MATLAB.



Рис. 10.34. Линии постоянной высоты (*a*) и аксонометрическое изображение объекта (*б*), полученные экспериментально

10.12. Определение координаты глубины по 2D-изображению

При рассматривании трехмерной сцены и составляющих ее объектов информацию о ней зритель получает из ее проекций на сетчатки глаз, которые по своей природе являются двумерными. Тем не менее, несмотря на это, в зрительной коре головного мозга формируется 3D-представление о наблюдаемой сцене и составляющих ее объектах [72]. При этом возникает кажущийся парадокс. С одной стороны, изображения на сетчатках глаз непосредственно не содержат данных о координатах глубины наблюдаемых объектов, поскольку являются двумерными, а с другой стороны, эти координаты каким-то образом восстанавливаются при формировании их представления в зрительной коре.

Обращаясь к анализу этой проблемы, сразу же исключим из рассмотрения бинокулярный фактор наблюдения, т. е. стереоэффект, поскольку при наблюдении объектов одним глазом, а также при наблюдении 2D-фотографий и картин зрение продолжает восстанавливать третью, утраченную координату. Заметим также, что ряд животных и птиц вообще лишены стереоскопического зрения, например, лошади, зайцы и т. д., поскольку поля зрения левого и правого глаз у них не перекрываются, но, несмотря на это, они ориентируются в пространстве. Можно также исключить из последующего рассмотрения различие в фокусировке глаз при наблюдении объектов, расположенных на различных расстояниях, как не определяющее явление, тем более, что оно проявляется на сравнительно небольших расстояниях. Из этого, конечно, не следует, что перечисленные выше факторы вообще не играют никакой роли при нахождении третьей координаты, однако они не являются определяющими.

Анализ проблемы приводит к выводу, что кажущийся парадокс разрешается, если учесть наличие априорной информации, которой располагает зритель при "дешифровке" 2D-изображений наблюдаемой сцены [41, 45]. Для пояснения сказанного обратимся к рис. 10.35, на котором изображена городская улица.

При рассматривании этого изображения у зрителя не возникает каких-либо затруднений ни в определении расстояний до изображенных на нем объектов, ни их размеров. Объясняется это тем, что при "дешифровке" наблюдаемого изображения зрительная система использует дополнительную информацию, заключающуюся в том, что все объекты, изображенные на этом рисунке, не парят в пространстве, а привязаны к плоскости проезжей части улицы и к плоскости тротуара. Здесь действует простое правило: объект, который изображен на большем расстоянии от линии горизонта вниз, находится ближе к наблюдателю, а объект, который изображен ближе к линии горизонта, находится дальше от наблюдателя. В общем случае поверхность, на которой находятся наблюдаемые объекты, может иметь более сложный вид, как, например, это показано на рис. 10.36, но это правило по-прежнему остается справедливым.


Рис. 10.35. Изображение городской улицы



Рис. 10.36. Изображение руин античного храма

Кроме того, при определении расстояний до наблюдаемых объектов зрительной системой может также использоваться и другая информация, часть которой зрителю обычно известна, например, информация о размерах наблюдаемых объектов: высоте фонарных столбов, автомобилей, среднем росте людей и т. д. Принцип

определения расстояний до объектов, составляющих наблюдаемую сцену, который используется зрительной системой, может быть также использован для получения 3D-изображения на основании 2D-изображения наблюдаемой сцены, полученного путем его простого фотографирования обычной цифровой камерой. Этот принцип позволяет отказаться от использования для этой цели дорогостоящего лазерного сканера.

Рассмотрим ряд простых, но достаточно типичных примеров определения расстояния до объекта по его 2D-изображению.

 Вначале обратимся к простейшему, но часто встречающемуся случаю, когда направление на фотографируемый объект совпадает с оптической осью камеры, а истинная высота объекта Z_{об} и фокусное расстояние объектива камеры f, посредством которой получено изображение, известны. В этом случае для вычисления расстояния до объекта X_{об} достаточно определить вертикальный размер проекции этого объекта Z_{пр} на светочувствительной поверхности камеры (на передней плоскости изображения) и воспользоваться известным соотношением (10.7)

$$Z_{\rm np} = \frac{Z_{\rm ob} f}{X_{\rm ob} + f},$$
 (10.20)

решив которое относительно X_{об}, находим

$$X_{\rm of} = f\left(\frac{Z_{\rm of}}{Z_{\rm np}} - 1\right). \tag{10.21}$$

2. Переходя к рассмотрению второго случая, будем считать, что 2D-изображение представляет собой изображение сцены, заполненной объектами, расстояния до которых требуется определить, при этом размеры самих объектов неизвестны. Известно лишь, что эти объекты расположены на плоской горизонтальной поверхности, а также известна высота *H*, с которой произведена съемка, фокусное расстояние объектива *f* и положение линии горизонта. Для решения этой задачи обратимся к рис. 10.37, на котором приведена модель камеры с передней плоскостью изображения. Для конкретности дальнейшего изложения под расстояниями до объектов будем понимать расстояния между проекцией центра объектива камеры на ось *x* и одной из точек, в которых объекты соприкасаются с горизонтальной плоскостью. В рассматриваемом примере будем искать расстояние *L* до точки *A*.

Из прямоугольного треугольника, образованного проекцией проецирующего луча OA на координатную плоскость xz, т. е. линии O-2, а также линии O-1 и линии 1-2, следует, что

$$\frac{H}{w} = \frac{X+f}{f} \,. \tag{10.22}$$



Рис. 10.37. Модель камеры с передней плоскостью изображения

Из рассмотрения прямоугольника, образованного проекцией проецирующего луча *OA* на координатную плоскость *xy*, т. е. линии *A*-3, а также линии стороны *2*-3 и стороны *A*-2, следует, что

$$\frac{Y}{v} = \frac{X+f}{f} \,. \tag{10.23}$$

Кроме того, как не трудно видеть,

$$L^{2} = Y^{2} + (X + f)^{2}.$$
 (10.24)

Решая уравнения (10.22—10.24) относительно L, найдем

$$L = \frac{H}{w} \sqrt{v^2 + f^2} , \qquad (10.25)$$

где w — расстояние на изображении между проекцией линии горизонта и проекцией точки A, v — расстояние на изображении между вертикальной линией, которая делит изображение пополам, и проекцией точки A. При $L \to \infty \ w \to 0$, т. е. бесконечно удаленный объект изображается на линии горизонта. Как правило, размер изображения на фотографии, экране монитора и т. д. отличается от размера изображения в передней плоскости, приведенной на рисунке модели камеры. Кроме того, при отображении изображения на мониторе эти размеры обычно выражаются в пикселах. Поэтому преобразуем формулу (10.25) к более удобному виду

$$L = \frac{H}{aw_{\rm H}} \sqrt{a^2 v_{\rm H}^2 + f^2} , \qquad (10.26)$$

где $aw_{\mu} = w$, $av_{\mu} = v$, а w_{μ} и v_{μ} — координаты изображения точки A на изображении, по которому производятся измерения, например, на экране монитора компьютера, a — коэффициент, согласующий размерности.

3. Рассмотрим теперь случай, когда фокусное расстояние объектива камеры неизвестно. С таким случаем мы можем встретиться при съемке камерой, имеющей объектив с переменным фокусным расстоянием (трансфокатор). Для решения задачи при данных условиях необходимо и достаточно знать расстояние L_1 до одного из изображенных объектов (более конкретно, расстояния до точки A соприкосновения одного из объектов с горизонтальной плоскостью, на которой он расположен) и тогда расстояния до других объектов могут быть найдены. Задача решается следующим образом. Вначале необходимо найти неизвестное фокусное расстояния f и делая подстановку в найденную формулу известного значения высоты, с которой произведена съемка H, и a, находится сначала фокусное расстояние

$$f = \sqrt{\frac{L_1^2 a^2 w_{\rm H1}^2}{H^2} - a^2 v_{\rm H1}^2} ,$$

а затем с помощью формулы (10.26) определяются расстояния до других объектов.

4. В том случае, если мы располагаем фотографией, для которой вообще неизвестны условия съемки, т. е. неизвестны *H* и *f*, а также положение линии горизонта, то для ее "дешифровки" потребуется априорное знание расстояний (или размеров) не менее трех объектов, расположенных на различных расстояниях до снимающей камеры.

Аналогичным образом решаются задачи применительно к случаю, когда поверхность, на которой расположены объекты, не является плоской и горизонтальной.

На рис. 10.38 показано изображение трехмерной сцены, сгенерированной посредством программы 3ds Max при использовании виртуальной камеры с объективом, имеющим фокусное расстояние f = 20 мм, и при высоте съемки H = 1,5 м.

По этому изображению были рассчитаны расстояния до изображенных на ней деревьев по формуле (10.26) описанным выше способом. Результаты измерения для деревьев слева направо составили: 51,9, 53,8, 100, 74,3, 34,1 и 123,3 м. Истинные расстояния при этом были 50, 50,5, 100, 71,6, 33,5 и 150 м соответственно. Расхождения между истинными расстояниями и измеренными объясняются недостаточным количеством строк в изображении, по которому производилось измерение. Другими словами, погрешность измерения обусловлена дискретностью изображения, использованного при измерениях. Снизить погрешность измерения можно путем увеличения частоты пространственной дискретизации изображения при съемке.



Рис. 10.38. Изображение трехмерной сцены, сгенерированное посредством программы 3ds Max

Для удобства практических измерений на изображение, по которому они выполняются, рекомендуется наносить линии постоянной дальности. Формула для расчета линий постоянной дальности может быть получена из формулы (10.26) путем ее решения относительно *w*_и

$$w_{\mu} = \frac{H}{aL} \sqrt{a^2 y_{\mu}^2 + f^2} \,.$$

Для построения линий постоянной дальности необходимо, задав значения L, H и f, изменять координату $v_{\rm u}$ и для каждого его значения определять координату $w_{\rm u}$. На рис. 10.38 показаны эти линии для ряда значений L.

10.13. 3D-печать

При 3D-печати на основе 3D-изображения (модели) создается реальный трехмерный объект. В основе технологий 3D-печати лежит послойное изготовление трехмерного объекта. На рис. 10.39 схематически показана часть объекта в процессе его изготовления (a), на которой выделен последний, только что созданный слой, представленный отдельно (δ).

Исторически первой технологией трехмерной печати является *стереолитография* (Stereo Lithography или SLA), которую в 1986 г. запатентовал Чарльз Халл (Charles Hull).



Рис. 10.39. Печать трехмерного объекта: *а* — объект в процессе его изготовления; *б* — последний, только что созданный слой

В соответствии с этой технологией процесс изготовления объекта протекает следующим образом. В рабочее пространство принтера помещается фотополимер в жидком состоянии, который обладает свойством застывать в твердый пластик при ультрафиолетовом облучении. На поверхность жидкого полимера, по очереди, точка за точкой, проецируется изображение формируемого слоя объекта, создаваемое ультрафиолетовым излучением, источником которого является либо лазер, либо обычная ультрафиолетовая лампа. Под воздействием ультрафиолетового излучения полимер застывает, образуя очередной слой объекта. После этого рабочий стол вместе со сформированной частью модели опускается на величину сформированного слоя, добавляется фотополимер до исходного уровня и наращивается следующий слой объекта. Этот процесс продолжается до тех пор, пока не будет изготовлен весь объект.

Стереолитография позволяет создавать объекты с точностью до десятых долей миллиметра, при этом обеспечивается хорошее воспроизведение мелких деталей и достаточно ровная поверхность объекта. В настоящее время эта технология наиболее широко распространена. Недостатком этой технологии является высокая цена и то, что набор обрабатываемых материалов ограничивается фотополимерами.

Рассмотренная технология была усовершенствована компанией Cubital Inc. и стала известна как технология Solid Ground Curing или SGC. Отличием этой технологии от описанной выше является то, что в ней засвечивается сразу вся поверхность формируемого слоя посредством ультрафиолетовой лампы через фотошаблон. Фотошаблон для каждого слоя печатается на стекле. Такой способ позволяет существенно увеличить производительность принтера за счет единовременного освещения всего слоя полимера вместо его поточечного сканирования. Компанией Envisiontec разработан 3D-принтер, в котором использована эта технология. Этот принтер обеспечивает разрешение для одного слоя 1280×1024 пиксела при размере пиксела в 150 или 90 микрон. Толщина слоя может варьироваться от 150

до 90 микрон. Посредством этого принтера можно изготавливать объекты размером до 190×152×230 мм со скоростью до 15 мм (в высоту) в час.

Похожий метод трехмерной печати — это выборочное лазерное спекание (Selective Laser Sintering, SLS). Этот метод отличается тем, что в нем в качестве рабочего материала используется не жидкий полимер, а порошок какого-либо легкоплавкого пластика. В рабочем объеме 3D-принтера порошок пластика нагревается до температуры, близкой к температуре плавления, при этом в рабочую зону подается жидкий азот для его предохранения от возгорания и окисления. Посредством сканирующего луча лазера на поверхность из пластикового порошка проецируется изображение сечения создаваемого объекта, при этом пластик нагревается выше температуры плавления и спекается. После этого сверху добавляется новый слой порошка, создаваемый объект опускается на величину созданного слоя и процедура повторяется. По окончании печати остатки неиспользованного порошка стряхиваются с готовой модели. Эта технология была разработана в 80-х годах прошлого века в Техасском университете в Остине и запатентована в 1989 г. Лазерное спекание обеспечивает высокое качество деталей, однако поверхность модели получается пористой. Применительно к этой технологии разработаны материалы, которые используются для изготовления металлических деталей. В этом случае вместо легкоплавкого пластикового порошка используются микрочастицы стали, покрытые слоем связующего пластика. В процессе изготовления детали спекание пластика происходит обычным образом, после чего модель помещается в печь, где пластик выгорает, а образовавшиеся пустоты заливаются бронзой. В результате получается деталь, состоящая на 60% из стали и на 40% из бронзы. Разработаны также материалы с керамической или стеклянной сердцевиной. Из них можно изготавливать детали, устойчивые к высоким температурам и агрессивным химическим веществам.

Компанией Helysis разработана технология трехмерной печати, которая известна как технология ламинирования. В соответствии с этой технологией в устройство по очереди вводятся тонкие листы материала, из которого лазером вырезаются слои создаваемого объекта. Затем изготовленные таким образом слои склеиваются друг с другом. В качестве материала может использоваться специальная бумага со слоем клеящего вещества, а также тонкий пластик, керамика и металлическая фольга.

Кроме лазерных технологий, используемых в 3D-принтерах, были разработаны, так называемые, струйные технологии. Основная идея таких технологий заключается в том, что специальная пишущая головка выдавливает на охлаждаемую платформуоснову капли разогретого термопластика. Капли термопластика, застывая и слипаясь друг с другом, формируют слои создаваемого объекта. Так же, как и ранее, печать здесь выполняется по слоям. Эта технология позволяет с высокой точностью (минимальная толщина слоя — 0,12 мм) изготовлять готовые к использованию детали размером до 600×600×500 мм. Основным производителем 3D-принтеров, использующих эту технологию, является компания Stratasys.

Похожая технология струйной печати разработана компанией Object Geometries. Здесь при печати посредством струйной головки используется фотополимерный

пластик. Как и ранее, объект печатается слой за слоем, разрешение в слое составляет 600×300 dpi, а толщина слоя может быть сделана всего 16 микрон. Под воздействием излучения ультрафиолетовой лампы каждый отпечатанный слой полимеризируется, превращаясь в твердый пластик

Известна еще одна технология струйной печати, но уже с использованием порошковых материалов. Разработана эта технология в Массачусетском технологическом институте. Производителем 3D-принтеров на основе этой технологи является компания Z Corporation. 3D-принтеры, производимые этой компанией, относительно недороги и работают заметно быстрее вышеописанных устройств.

Особенностью этого технологического процесса является то, что специальная струйная головка (адаптированная из струйных принтеров Hewlett-Packard) наносит на порошковый материал клеящее вещество. В качестве порошка используется гипс или крахмал. В тех местах, куда попало клеящее вещество, порошок склеивается и формирует модель. Как и в предыдущих случаях, печать выполняется послойно. По окончании процесса лишний порошок удаляется.

Существенной особенностью принтеров, использующих эту технологию, является то, что в них может применяться клеящая жидкость с добавлением пигментных красителей, что позволяет получать окрашенные объекты. Так, например, в цветном принтере от Z Corporation установлены четыре струйные головки с чернилами-клеем основных цветов. Получаемая детализация создаваемого объекта очень высокая.

10.14. Обработка оболочек трехмерных изображений

Обработка 3D-изображений включает в себя: обработку оболочек объектов, составляющих сцену, обработку их текстур, а также изменение условий освещения сцены и точки, из которой она наблюдается. Поскольку текстуры представляют собой 2D-изображения, то для их обработки могут быть применены рассмотренные ранее методы, а что касается вопросов, связанных с освещением сцены и изменением точки наблюдения, то они излагаются в *главе 14*. Поэтому здесь прямо перейдем к рассмотрению вопросов, связанных с обработкой оболочек объектов.

Обработка оболочки объекта. Здесь и в дальнейшем будем исходить из того, что трехмерное изображение объекта представляется векторной полигональной моделью. Напомним, что в этом случае оболочка (поверхность) объекта состоит из полигонов (многоугольников), положение которых задается вершинами. Часто в качестве полигонов применяют треугольники. Очевидно, для того чтобы внести какие-либо изменения в оболочку, необходимо изменить координаты вершин, которыми эта оболочка задана.

1. Простейший вид изменения заключается в изменении положения объекта в пространстве, а также в изменении его размеров. При этом новые декартовы

координаты для каждой *i*-й вершины X_i , Y_i , Z_i находятся следующим образом:

• при смещении в пространстве на величины x_0, y_0, z_0 :

$$X_i = x_i + x_0, \ Y_i = y_i + y_0, \ Z_i = z_i + z_0,$$

где x_i, y_i, z_i — прежние координаты вершины;

• при увеличении (уменьшении) размеров в k раз:

$$X_i = kx_i$$
, $Y_i = ky_i$, $Z_i = kz_i$;

- при повороте в пространстве:
 - \diamond вокруг оси x_i на угол ϕ

$$X_i = x_i$$
, $Y_i = y_i \cos \varphi - z_i \sin \varphi$, $Z_i = y_i \sin \varphi + z_i \cos \varphi$;

 \diamond вокруг оси y_i на угол Ψ :

$$X_i = x_i \cos \psi - z_i \sin \psi, \ Y_i = y_i, \ Z_i = x_i \sin \psi + z_i \cos \psi;$$

 \diamond вокруг оси z_i на угол γ :

$$X_i = x_i \cos \gamma - y_i \sin \gamma$$
, $Y_i = x_i \sin \gamma + y_i \cos \gamma$, $Z_i = z_i$.

На рис. 10.40 показана исходная система координат, относительно которой происходят описанные преобразования.



Рис. 10.40. Система координат

2. Изменения формы объекта могут носить и более сложный характер. В этом случае новые координаты вершин находятся следующим образом:

 $X_i = \Phi_x(x_i, y_i, z_i, a_1, a_2, ...), Y_i = \Phi_y(x_i, y_i, z_i, a_1, a_2, ...), Z_i = \Phi_z(x_i, y_i, z_i, a_1, a_2, ...),$ где $\Phi_x(.), \Phi_y(.), \Phi_z(.), a_1, a_2, ... — функции и параметры, определяющие характер преобразования.$

Приведем примеры, иллюстрирующие сказанное.

• Изменение масштаба, различное в разных направлениях:

$$X_i = k_x x_i, \ Y_i = k_y y_i, \ Z_i = k_z z_i,$$

где k_x , k_y , k_z — коэффициенты, определяющие изменение координат вершин в направлениях x, y, z соответственно. • Преобразование цилиндра, расположенного в начале координат, в конус может быть выполнено путем использования следующих формул

$$X_i = R\left(1 - \frac{1}{H}z\right), \ Y_i = R\left(1 - \frac{1}{H}z\right), \ Z_i = z,$$

где *R* — радиус цилиндра, *H* — высота цилиндра. Рисунок 10.41 поясняет изложенное.



Рис. 10.41. К пояснению преобразования цилиндра в конус: *а* — исходный цилиндр; *б* — полученный конус



Рис. 10.42. Примеры воздействия различного вида преобразований на параллелепипед

Преобразования такого рода применяются в 3D-графических редакторах. Приведенное в качестве примера преобразование цилиндра в конус включено в редактор 3ds Max под названием **Taper**. В этом редакторе имеется также множество аналогичных преобразований, например, **Bend** — изгиб на заданный угол в направлении заданной оси x, y, z, **Twist** — скручивание вокруг заданной координатной оси, **Stretch** и **Squeeze** — растягивание и сжатие вдоль задаваемых осей (т. е. изменение масштаба) и т. д. На рис. 10.42 показаны результаты воздействия перечисленных преобразований на параллелепипед.

3. Еще большие возможности дает преобразование 3D-объекта, выполняемое на уровне полигональной сетки. В этом режиме обработки изменяют координаты либо вершин, либо ребер, либо граней путем непосредственного доступа к ним.

Глава 11



Цифровое видео и телевидение

В нашей стране, насколько известно автору, первые работы в области цифрового телевидения, а именно разработки аналого-цифровых и цифроаналоговых преобразователей и исследования в области помехоустойчивости кодово-импульсной модуляции применительно к передаче телевизионных сигналов, были выполнены в Ленинградском институте авиационного приборостроения (ЛИАП) по хозяйственному договору с Всесоюзным научно-исследовательским институтом телевидения (ВНИИТ) в конце 50-х годов прошлого века. Со стороны ЛИАП эти работы возглавил автор этой книги, со стороны ВНИИТ данное направление курировал И. И. Цуккерман.

В период 60-70-х годов в области цифрового телевидения во ВНИИТ работали возглавляемые: И. И. Цуккерманом, П. Ф. Брацлавцем, научные коллективы, В. П. Мандражи, В. В. Чернышевым и др. В это же время в ЛИАП по заказам ВНИИТ в лабораториях, возглавляемых автором этой книги, работало несколько научно-исследовательских групп, результаты работы которых нашли отражение в публикациях тех лет [62, 63] и были отмечены центральной прессой [14]. В начале 70-х годов при ЛИАП Всесоюзным научно-исследовательским институтом телевидения была создана отраслевая научно-исследовательская лаборатория по этой тематике. Исследования, выполняемые этой лабораторией, были направлены на разработку методов аналого-цифрового преобразования и сжатия телевизионного сигнала (группа В. Я. Сорина), методов выделения телевизионного сигнала из шума (группа О. С. Астратова), методов создания цифровой памяти и магнитной записи (А. С. Горшков) [56] и др. Результатами работ коллективов ученых ЛИАП и ВНИИТ явился ряд совместных авторских свидетельств на изобретения, а также внедрение разработанных методов и устройств, в частности, в области цифрового космического телевиления.

11.1. Динамические изображения

Для создания ощущения движения объекта необязательно, чтобы его проекции на сетчатки глаз непрерывно смещались. Это ощущение не нарушается и в том случае, если на сетчатку глаза проецировать ряд быстро сменяющих друг друга изображений

объекта, на которых представлены последовательные фазы его движения. Именно это свойство зрения используется для создания иллюзии движения в кино, телевидении и компьютерном видео [93]. Для того чтобы движение воспринималось плавным, частота смены изображений с различными фазами движения должна быть достаточно большой, иначе движения будут казаться прерывистыми. Чем больше яркость экрана, тем выше должна быть эта частота (частота кадров). При используемых яркостях экрана компьютерного дисплея она составляет около 60—70 Гц, при яркостях экрана, применяемых в кино, она существенно ниже. Это накладывает определенные ограничения на допустимое значение величины угловых перемещений изображений объектов, т. е. перемещений в направлении перпендикулярной линии наблюдения. При нарушении этих ограничения движения воспринимаются как прерывистые. Эффект прерывистости движения хорошо заметен, например, при панорамной съемке опушки хвойного леса, когда на изображении имеется много вертикальных стволов, а панорамирование происходит достаточно быстро.

Недостаточно высокая частота кадров приводит также к другому неприятному явлению, которое было рассмотрено в *разд. 3.5*, — явлению мелькания наблюдаемого изображения. Критическая частота мельканий $f_{\rm kp}$, как показали исследования, за-

висит как от угловых размеров наблюдаемого мелькающего поля, так и от его средней яркости *L*. Для угловых размеров мелькающего поля, в данном случае, угловых размеров экрана, на котором демонстрируется движущееся изображение, эта зависимость описывается следующей эмпирической формулой

$$f_{\rm KD} = 101 {\rm g} L + 30$$
.

При проектировании устройств воспроизведения изображений, например дисплеев, знание этой величины очень важно, т. к. на ее основании определяется частота смены кадров. Так при используемых яркостях экранов компьютерных дисплеев $f_{\rm KD} \approx 60$ Гц, поэтому частота кадров в них выбирается выше этого значения.

При частотах мелькания, превышающих критическую частоту, мелькания незаметны, и яркость мелькающего поля L(t) кажется неизменной и пропорциональной своему среднему значению

$$L_{\rm cp} = \frac{1}{T} \int_0^T L(t) dt \,,$$

где T = 1/f — период мелькания. Эта формула представляет собой закон Тальбота.

11.2. Съемка исходного материала

В настоящее время цифровое представление видеоматериалов (телевизионные программы новостей, спортивные передачи, видеофильмы и т. д.) практически вытеснило их аналоговое представление. Обусловлено это рядом преимуществ цифровой техники по сравнению с аналоговой техникой. Основными преимуществами являются:

- □ удобство обработки и хранение изображений, составляющих видеозапись;
- возможность реализовать высокую помехоустойчивость при передаче и многократной перезаписи сигналов изображений в цифровой форме;
- возможность сжатия (компрессии) цифрового потока, которым передается видеосигнал, благодаря чему по одному стандартному каналу оказывается возможным передавать 4—5 телевизионных программ с профессиональным качеством;
- возможность упаковывать в транспортные потоки цифровые сигналы телевизионных передач совместно с другими цифровыми сигналами, имеющими совсем иную природу;
- возможность широкой автоматизации функций передающих камер, а также возможность обеспечения их стабильной работы;
- □ но самое главное, сохранение отснятого материала существенно более высокого качества при его многочисленных перезаписях [62, 81, 89].

Рассмотрим структуру типичного цифрового видео на примере видеофильма. Видеофильм представляет собой последовательность в общем случае цветных изображений (кадров), которые объединены в отдельные группы — клипы, представляющие тот или иной сюжет. Рисунок 11.1 поясняет изложенное.



Рис. 11.1. Последовательность кадров в видеофильме

При съемке движущихся объектов на каждом из кадров фиксируются отдельные фазы их движения. При воспроизведении этой последовательности кадров благодаря тому, что они быстро сменяют друг друга, создается иллюзия движения. В свою очередь, каждый из кадров, представляющий собой цветное неподвижное изображение, на самом деле состоит из трех цветоделенных неподвижных изображений, например, красного (R), зеленого (G) и синего (B). Рисунок 11.2 иллюстрирует изложенное.

Способ цветоделения может быть и другим, например, исходное цветное изображение может быть представлено в виде одного яркостного (Y) и двух цветоразностных (R-Y) и (B-Y) компонентов, при этом цветоразностные компоненты обычно формируются с меньшим пространственным разрешением, что допустимо благодаря особенностям нашего зрения и не приводит к потере качества изображения. (Подробнее о свойствах зрения см. в главе 3.) И, наконец, напомним, что цветоделенные компоненты представляют собой не что иное, как двумерные массивы целых чисел, над которыми совершаются математические операции при обработке изображения, и которые после соответствующего кодирования передаются по каналу связи либо записываются на диск или ленту при архивации. При создании цифрового видео независимо от того, делается ли телевизионный сериал, спортивная передача, передача новостей, развлекательная передача и т. п., вначале снимается исходный материал, из которого затем монтируется собственно создаваемая программа.



Рис. 11.2. Представление цветного изображения

Объем исходного материала обычно в несколько раз превышает объем уже смонтированной программы. Съемка исходного материала производятся посредством передающих телевизионных камер с использованием того или иного стандарта. В настоящее время для этой цели применяются, как правило, камеры на приборах с зарядовой связью (ПЗС), обладающие высокими качественными параметрами. В лучших образцах современных телевизионных камер используются матрицы с покадрово-строчным (FIT) переносом зарядов, что позволяет полностью избавиться от так называемого смаза, возникающего при попадании в кадр ярких объектов.

В принципе съемка может производиться либо посредством аналоговых передающих телевизионных камер, и тогда следующей операцией будет оцифровка изображений, либо посредством цифровых телевизионных камер, в которых аналогово-цифровые преобразователи (АЦП) включены сразу же на выходах матриц ПЗС. Последний случай является более предпочтительным, поскольку обеспечивает более высокое качество изображения и поэтому исключительно он и применяется в настоящее время. В этом случае отсутствуют промежуточные преобразования

дискретизированных во времени сигналов на выходах матриц в непрерывные сигналы, а затем их новая дискретизация в АЦП. В этих камерах матрицы ПЗС и включенные на их выходах АЦП работают на одной тактовой частоте. В современных цифровых телевизионных системах при съемке исходного материала применяют не менее, чем 10-разрядный двоичный цифровой код для представления каждого из компонентов цветного изображения, или даже 14-разрядное кодирование. Объясняется это тем, что при меньшем числе разрядов невозможно на требуемом уровне качества выполнить ряд важных операций, в частности, гаммакоррекцию, поскольку при малом числе уровней квантования возможно возникновение ложных контуров (*см. разд. 6.5*).

Снятый материал в цифровом виде записывается либо на ленту, либо на дисковый носитель, при этом осуществляется его сжатие (компрессия). Так как отснятый исходный материал в дальнейшем подлежит монтажу, то необходимо обеспечить доступ к каждому кадру записи, поскольку каждый кадр записи при монтаже может оказаться и первым, и последним. Из этого следует, что в данном случае не все виды компрессии могут быть применены, а только те, в которых не использована межкадровая избыточность, и, следовательно, обеспечен доступ к каждому кадру. В настоящее время для сжатия исходного отснятого материала применяют либо метод M-JPEG, либо MPEG-2, но только с *I*-кадрами. В последнем случае сжатие оказывается выше.

11.3. Форматы растров цветных изображений

Телевизионный сигнал может быть представлен как в компонентной, так и в композитной форме.

Композитный видеосигнал формируется из компонентного сигнала и передается потребителю в одном из трех форматов: NTSC, PAL или SECAM [93]. При оцифровке композитный видеосигнал подвергается временной дискретизации и квантованию по уровню яркости десятиразрядным кодом, т. е. квантуется на 1024 уровня. При этом на представление самого видеосигнала отводится около 70% уровней квантования, в то время как остальная часть квантовых уровней отводится на передачу синхроимпульсов. Скорость передачи данных для цифрового сигнала в системе NTSC составляет 143 Мбит/с, а в системе PAL — 177 Мбит/с.

Компонентный видеосигнал состоит из трех компонентов, например R, G, B или компонентов Y, R-Y (Cr) и B-Y (Cb). Особенностью оцифровки компонентного видеосигнала является то, что дискретизация, квантование и кодирование каждого из компонентов сигнала, например, яркости Y и двух цветоразностных сигналов R-Y (Cr) и B-Y (Cb) в этом случае выполняются раздельно. Определены ряд форматов представления компонентного сигнала в цифровой форме, которые различаются между собой структурой расположения отсчетов сигнала яркости и двух цветоразностных сигналов в растре изображения. Далее приводятся краткие сведения об этих форматах.

Формат 4:4:4

В этом формате все три компонента сигнала имеют одинаковую структуру расположения отсчетов, благодаря чему обеспечивается одинаковое пространственное разрешение по горизонтали и по вертикали как яркостное, так и цветовое. При этом формате частота временной дискретизации всех трех компонентов видеосигнала составляет 13,5 МГц, благодаря чему все три компонента передаются в полной полосе частот. В этом формате каждый компонент представлен 576 строками и 720 пикселами в каждой строке. Скорость цифрового потока при 10-битовом слове в данном случае составляет 405 Мбит/с. На рис. 11.3 приведены структуры растров для всех трех компонентов при использовании этого формата. Формат 4:4:4 используется при предварительной обработке сигнала, а также в тех случаях, когда в изображении требуется сохранить максимальное количество информации, например, в изображениях, используемых в научных исследованиях.

Рис. 11.3. Структура дискретизации 4:4:4: □ — отсчеты сигналов яркости; + — отсчеты цветоразностных сигналов

Формат 4:2:2

Он отличается от формата 4:4:4 тем, что в этом формате плотность отсчетов для цветоразностных сигналов в горизонтальном направлении уменьшена в 2 раза по сравнению с форматом 4:4:4, а в вертикальном направлении оставлена неизменной. Благодаря этому цветовое разрешение в горизонтальном направлении уменьшилось в 2 раза по сравнению с яркостным разрешением, при неизменном разрешении в вертикальном направлении. В результате уменьшения количества отсчетов, которым представляется изображение в этом формате, в 1,5 раза уменьшился цифровой поток, которым оно представляется. Однако вследствие того что цветовая острота зрения существенно ниже яркостной остроты зрения, о чем было сказано в разд. 3.11, зритель не замечает снижения качества изображении при переходе от формата 4:4:4 к формату 4:2:2. На рис. 11.4 приведены структуры растров для всех трех компонентов при использовании данного формата. В этом формате в активной части строки содержится 720 отсчетов яркостного сигнала и по 360 — каждого цветоразностного. Скорость цифрового потока при 10-битовом слове в данном случае составляет 270 Мбит/с. Формат 4:2:2 используется в качестве базового при оценке других вариантов дискретизации.

Ħ	Ħ	Ħ	Ħ	
Ħ	⊞	⊞	Ħ	
⊞	⊞	⊞	⊞	
Ħ	⊞	⊞	⊞	
Ħ	⊞	⊞	⊞	
⊞	⊞	⊞	⊞	
Ħ	Ħ	⊞	⊞	
Ħ	⊞	⊞	⊞	

Рис. 11.4. Структура дискретизации 4:2:2: □ — отсчеты сигналов яркости; + — отсчеты цветоразностных сигналов

Формат 4:2:0

Этот формат отличается от формата 4:4:4 тем, что в этом формате плотность отсчетов и в горизонтальном, и в вертикальном направлении уменьшена в 2 раза по сравнению с форматом 4:4:4. Благодаря этому, цветовое разрешение в обоих направлениях оказывается также уменьшенным в 2 раза по сравнению с яркостным разрешением, которое осталось неизменным. В результате уменьшения плотности расположения отсчетов сигналов цветности в растре в 2 раза в обоих направлениях цифровой поток также уменьшился в 2 раза. Другими словами, это привело к двукратному сжатию. Однако, как и в предыдущем случае, вследствие того, что цветовая острота зрения существенно ниже яркостной остроты зрения, зритель не замечает снижения качества изображения. В формате 4:2:0 компонент яркости Y в активной части кадра содержит 576 строк и в каждой строке по 720 отсчетов, а цветоразностные компоненты Сг и Сb в активной части кадра содержат по 288 строк и в каждой строке по 360 отсчетов.

На рис. 11.5 приведены изображения растровых структур для всех трех компонентов при использовании рассматриваемого формата. Формат 4:2:0 используется в современных стандартах сжатия как неподвижных, так и движущихся изображений, таких как JPEG, MJPEG, MPEG.

Ш □ Ш □ Ш □ Ш □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □ □
□ □ □ □ □
□ □ □ □ □
□ □ □ □
□ □ □ □
□ □ □ □
□ □ □ □
□ □ □
□ □ □
□ □ □
□ □ □
□ □ □
□ □ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □
□ □

Формат 4:1:1

Кроме перечисленных форматов находит применение также формат 4:1:1. В формате 4:1:1 компонент яркости Y в активной части кадра содержит 576 строк и в каждой строке по 720 отсчетов, а цветоразностные компоненты Cr и Cb в активной части кадра также содержат по 576 строк, но в каждой строке они содержат только по 180 отсчетов. На рис. 11.6 приведена рассматриваемая структура дискретизации цветного изображения.

Ħ		⊞		
Ħ		⊞		
⊞		Ħ		
⊞		⊞		
⊞		⊞		
Ħ		Ħ		
Ħ		Ħ		
Ħ		⊞		

Рис. 11.6. Структура дискретизации 4:1:1: □ — отсчеты сигналов яркости; + — отсчеты цветоразностных сигналов

Форматы 4:1:1 и 4:2:0 характеризуются одинаковой скоростью передачи данных, которая составляет 202,5 Мбит/с при длине кодового слова в 10 бит и 162 Мбит/с при 8 битах на слово. В том случае, если передавать только активную часть изображения (без обратного хода), то величина цифрового потока при 8 битах на слово сократится до 124 Мбит/с. Цифровые сигналы этих двух форматов получают из сигналов формата 4:2:2 путем предварительной обработки и прореживания отсчетов с целью сокращения скорости потока.

Формат 4:1:1 применяют в системах со стандартом разложения 525 строк и 60 кадр/с. Формат 4:2:0 применяют в системах со стандартом разложения 625 строк и 50 кадр/с. Это обусловлено тем, что потеря разрешения в вертикальном направлении более заметна в системах с меньшим числом строк, в то время как потеря разрешения в горизонтальном направлении более заметна в системах с большим числом строк.

11.4. Элементы нелинейного монтажа

Монтаж исходного материала, применяющийся в кино и телевидении, заключается в том, что вначале из всего отснятого материала, включающего в себя дубли и варианты, отбираются сцены (клипы), которые должны войти в монтируемую программу (фильм), т. е. выполняется так называемый черновой монтаж, целью которого является создание монтажного листа. После выполнения чернового монтажа переходят ко второй и заключительной части — к чистовому монтажу телевизионной программы или фильма. На этом этапе окончательно подбираются длительности отдельных клипов, включаются соединяющие их переходы (шторки, наплывы и т. п.), производится графическое редактирование отснятого материала, включение всевозможных эффектов и озвучивание. Как правило, объем исходного материала, из которого монтируется программа, как уже было отмечено, значительно превышает объем материала, который в конечном итоге оказывается включенным в нее. Так, например, исходного материала может быть 20 часов при конечном хронометраже передачи — 2 часа.

Традиционно в телевидении применялся так называемый линейный метод монтажа программы. Особенность этого метода заключалась в том, что как отснятый материал, так и смонтированный записывались на магнитную ленту, при этом для поиска и копирования нужной записи приходилось затрачивать много времени, которое расходовалось на перемотку ленты.

Развитие вычислительной техники и появление на рынке мощных компьютеров позволило решить эту проблему путем разработки новой технологии монтажа программ, так называемой технологии нелинейного монтажа. Принципиальная особенность этой технологии состоит в том, что исходный отснятый материал, состоящий из ряда клипов, в цифровом виде записывается на жесткий диск, а не на ленту, в виде отдельных файлов, что обеспечивает практически мгновенный доступ к каждому из них без необходимости перемотки. Благодаря этой особенности резко сокращается время, затрачиваемое как на черновой, так и на чистовой монтаж.

Технология нелинейного монтажа включает в себя следующие элементы:

- запись исходного отснятого материала на жесткий диск (в том случае, если съемка осуществлялась аналоговой камерой, материал предварительно оцифровывают);
- □ выполнение чернового монтажа, при котором из отснятого материала отбирается материал, включаемый в монтируемую телевизионную программу;
- чистовой монтаж, при котором уточняется длительность каждого клипа, вошедшего в создаваемую программу, включаются переходы между отдельными клипами, выполняется редактирование изображений, составляющих клипы, включение всевозможных эффектов и компьютерной графики, виртуальных сцен, титров, рир-проекции, мультипликации, озвучивание и т. д.;
- □ запись смонтированной программы на носитель (например, на ленту или диск) для ее демонстрации.

В настоящее время применяются системы нелинейного монтажа для различных платформ: компьютерная платформа на базе PC (AVID — нелинейный монтаж цифрового и вещательного качества, Matrox RT series — нелинейный монтаж цифрового качества, Matrox Axio — нелинейный монтаж SD- и HD-качества, Canopus — нелинейный монтаж цифрового качества SD & HD, AJA Xena — нелинейный монтаж HD/2K) компьютерная платформа Apple (Blackmagic Decklink — нелинейный монтаж SD/HD, AJA Kona — нелинейный монтаж SD/HD/2K, Matrox MXO — захват и мониторинг DVCAM/HDV/HD, MOTU V3HD — портативная монтажная система SD/HD) и др.

Для всех этих платформ разработано программное обеспечение, позволяющее осуществлять нелинейный монтаж. В качестве примера можно привести программу Adobe Premiere, обеспечивающую высокий уровень производительности в области профессиональной обработки видеоматериалов.

Целый ряд фирм в настоящее время выпускает специализированные станции для нелинейного монтажа, которые могут быть включены в единую компьютерную сеть. Поскольку черновой монтаж, как показывает практика, занимает в несколько раз больше времени, чем чистовой, количество станций, включенных в сеть и осуществляющих черновой монтаж, существенно превышает количество станций, осуществляющих чистовой монтаж. Благодаря этому в одной системе оказывается возможным одновременно выполнять работы, связанные и с черновым, и с чистовым монтажом без перезаписи видеоматериала, и тем самым осуществлять еще большую экономию времени. В качестве примера можно назвать станции фирмы Avid Technology Inc., которые являются самыми популярными в мире [94]. Одной из самых мощных систем обработки нелинейного видео на сегодняшний день является Avid Media Composer 5.0. Она применяется при создании фильмов, телевизионных шоу и рекламных роликов в современной кино- и видеоиндустрии, поскольку предоставляет удобный инструментарий для творческой работы, надежное управление медиаданными, высокое качество эффектов, возможность работы в составе производственного комплекса любого масштаба. Популярными в настоящее время готовыми компьютерными станциями на базе PC является, например, Newtek Toaster VT. Она представляет собой программно-аппаратный комплекс, предназначенный для редактирования и обработки "живого" видео, многоуровневого компоузинга, создания видеоэффектов, титрования, двумерного и трехмерного моделирования и анимации. Другим примером системы гибридного видеомонтажа является Canopus SD.

Отдельно следует остановиться на проблеме сжатия видеозаписей при их монтаже. При монтаже телевизионных программ возникает необходимость иметь доступ (вход в запись) к любому кадру, что невозможно осуществить при использовании метода сжатия MPEG-2, за исключением случая, когда используются только I-кадры. Это послужило причиной того, что в монтажных станциях применяют довольно часто сжатие по методу M-JPEG, в котором ограничиваются только внутрикадровым кодированием по методу JPEG и не используют для целей сжатия межкадровым кодированием по методу JPEG и не используют для целей сжатия межкадровым кодированием по методу JPEG и не используют для целей сжатия межкадровым кодированием по методу JPEG и стания опименения сжатия при редактировании и монтаже является опасность накопления ошибок. Дело в том, что применяемые в рассматриваемом случае методы сжатия являются методами сжатия с потерей информации, поэтому при каждой новой процедуре сжатия после очередного редактирования изображения ими вносятся дополнительные ошибки, что, в конечном счете, может привести к недопустимому уровню искажений (*см. разд. 12.10*). В связи с этим рекомендуется на стадии нелинейного цифрового монтажа каждую очередную запись осуществлять с малым коэффициентом сжатия.

И, наконец, в заключение остановимся на так называемом гибридном методе монтажа, включающем в себя как элементы линейного, так и элементы нелинейного монтажа. Впервые этот метод был предложен как метод, в котором линейный монтаж был дополнен отдельными элементами нелинейного монтажа в целях ускорения технологического процесса, но в дальнейшем он был практически вытеснен методом нелинейного монтажа. Однако в последнее время наметилась тенденция к включению в нелинейный монтаж процедуры записи готовой программы на ленту, т. е. процедуры, относящейся к линейному монтажу.

11.5. Разбивка видеоматериала на отдельные сцены

Разбивкой видеоматериала на отдельные сцены, или просто разбивкой на сцены называется операция, позволяющая непрерывный поток отснятого видеоматериала разделить на части, каждая из которых соответствует отдельной отснятой сцене, что упрощает дальнейший монтаж телевизионной программы, фильма и т. д. Обычно для каждой отдельной выделенной сцены создается значок, который, как правило, представляет собой уменьшенное изображение этой сцены. Разбивка может выполняться:

🗖 автоматически по времени съемки;

□ автоматически, через каждые *t* секунд;

автоматически по содержимому сцены

🗖 и, наконец, ручным способом.

В первом случае, при автоматической разбивке по времени съемки, при перезаписи материала с видеокамеры программа, используемая для монтажа видео, а следовательно, и для перезаписи, отслеживает временные метки на ленте и, обнаружив разрыв, начинает новую сцену. Этот вариант возможен только при перезаписи видеоматериала непосредственно с камеры.

Во втором случае программа разделяет отснятый видеоматериал на новые сцены в соответствии с заданным временным интервалом. Этот вариант разбивки на сцены оказывается удобным в том случае, если видеозапись содержит слишком длинные сюжеты.

Разбивка на сцены автоматически, по содержимому сцены осуществляется следующим образом. Программа, реализующая эту операцию последовательно, кадр за кадром, анализирует статистические характеристики изображений, например, их среднюю яркость $\overline{L_{c,l}}$, среднеквадратичное значение переменной составляющей яркости

 $\sigma_{c,l}$ и т. д., а затем вычисляет изменения этих значений $\Delta(\overline{L_{c,l}})$ и $\Delta(\sigma_{c,l})$ при переходе от кадра с номером l к кадру с номером l+1, например, по формулам

$$\overline{L_{c,l}} = \frac{1}{KN} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} L_{c,l}(k,n), \qquad \sigma_{c,l} = \sqrt{\frac{1}{KN} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} \left[L_{c,l}(k,n) - \overline{L_{c,l}} \right]^2}, \\ \Delta\left(\overline{L_{c,l}}\right) = \overline{L_{c,l}} - \overline{L_{c,l-1}}, \quad \Delta\left(\sigma_{c,l}\right) = \sigma_{c,l} - \sigma_{c,l-1},$$

где *k* и *n* — номера строк и столбцов пикселов изображения, а *K* и *N* — соответственно общее количество строк и столбцов в анализируемом изображении. После этого выполняется сравнение абсолютных значений вычисленных приращений $\Delta(\overline{L_{c,l}})$ и $\Delta(\sigma_{c,l})$ с соответствующими порогами $P_{\Delta \overline{L}}$ и $P_{\Delta \sigma}$. В том случае, если вычисленные значения абсолютных приращений превышают значение порога, т. е.

$$\left|\Delta\left(\overline{L_{c,l}}\right)\right| > P_{\Delta\overline{L}}, \ \left|\Delta\left(\sigma_{c,l}\right)\right| > P_{\Delta\sigma},$$

принимается решение о том, что произошел переход к новой сцене, в противном случае считается, что продолжается старая сцена.

В целях сокращения количества вычислительных операций, вместо анализа изменения среднеквадратичного значения яркости при переходе от кадра к кадру, целесообразно вычислять изменение средней величины абсолютного значения переменной составляющей яркости $\overline{|L_{c\sim, l}|}$, т. е.

$$\overline{L_{\mathbf{c}_{\sim,l}}} = \frac{1}{KN} \sum_{k=0}^{K-1} \sum_{n=0}^{N-1} \left| \left[L_{\mathbf{c}_{l}l}(k,n) - \overline{L_{\mathbf{c}_{l}l}} \right] \right|,$$

а затем уже эту величину сравнивать с соответствующим порогом

$$\left|\Delta\left(\overline{L_{c\sim,l}}\right)\right| > P_{\Delta\overline{L}\sim}\,,$$

результат при этом не изменится. Этот способ может плохо работать при неустойчивом освещении. Например, если видеоматериал был снят в ночном клубе при мигающем освещении, то каждая вспышка света в этом случае будет приводить к ложному обнаружению новой сцены.

И, наконец, разбивка видеоматериала на отдельные сцены ручным способом позволяет эту операцию выполнить наилучшим способом, используя в качестве критерия, по которому производится разбивка, не только статистические характеристики изображения, но и что важно, его смысловое содержание. Недостатком этого способа является большая трудоемкость.

11.6. Создание переходов

При монтаже программы создается последовательное соединение клипов, отобранных из исходного отснятого материала. Простейшим соединением двух клипов является стык, при котором два клипа непосредственно соединяются друг с другом. Однако этот способ соединения хотя и часто применяется, тем не менее, не всегда может удовлетворить художественный замысел режиссера. По этой причине были разработаны специальные виды переходов от одного клипа к другому. Несмотря на то, что существует большое количество самых разнообразных переходов, основная их масса использует либо эффект вытеснения одного изображения другим посредством применения для этой цели всевозможных шторок и диафрагм, либо эффект наплыва. Во втором случае замещаемое изображение постепенно гаснет, а на его месте высвечивается замещающее изображение. На рис. 11.7 приведен простейший пример перехода, основанный на эффекте вытеснения. Рисунок 11.7, *а* показывает изображение в клипе, которое должно быть вытеснено и заменено изображением, показанным на рис. 11.7, *в*. На рис. 11.7, *б* показана одна из промежуточных фаз вытеснения.



Рис. 11.7. Пример перехода, основанного на эффекте вытеснения

На рис. 11.8 приведен другой простейший пример перехода, основанный на эффекте наплыва. На этом рисунке так же, как и на предыдущем, показаны исходное изображение (a), конечное изображение (b) и промежуточное изображение (δ).



Рис. 11.8. Пример перехода, основанного на эффекте наплыва

Реализации этих переходов могут быть самыми разнообразными, например, при вытеснении шторка может быть заменена на фигурную открывающуюся диафрагму или на множество открывающихся диафрагм, а при наплыве в разных частях изображения наплыв может происходить с разной скоростью, однако в основу преобразований положен один и тот же принцип взаимодействия изображений из двух клипов с маскирующей функцией. Поясним изложенное, для чего распределение яркости в изображении первого клипа обозначим через $L_{c1}(k, n, t)$, распределение яркости в изображении второго клипа обозначим через $L_{c2}(k, n, t)$, а маскирующую функцию обозначим через $\Omega(k, n, t)$, где k и n — соответственно номера строк и столбцов в матрицах изображений, а t — время. Особенностью маскирующей функции является то, что значения ее прозрачности могут изменяться в пределах от 0 до 1. Распределение яркости в смонтированном изображении $L_{c\Sigma}(k, n, t)$, полученном в результате монтажного соединения первого и второго клипов, независимо от вида примененного перехода определяется следующим образом:

$$L_{c\Sigma}(k,n,t) = L_{c1}(k,n,t)\Omega(k,n,t) + L_{c2}(k,n,t)\left[1 - \Omega(k,n,t)\right].$$

Вид перехода при этом полностью определяется видом маскирующей функции. В случае вытеснения маскирующая функция может принимать лишь два значения: 0 и 1.

На рис. 11.9 показано распределение прозрачности в маскирующей функции, посредством которой осуществляется вытеснение одного изображения другим для трех значений времени *t*. На этом рисунке белый цвет соответствует значению прозрачности "единица", а черный — "ноль". В случае наплыва значения маскирующей функции последовательно принимают ряд промежуточных значений от 1 до 0.

На рис. 11.10 показано, как изменяется прозрачность маскирующей функции с течением времени при наплыве для пиксела, расположенного в *k*-й строке и *n*-м столбце изображения. Иногда рассмотренные методы перехода используют совместно с геометрическими преобразованиями изображений первого и второго клипов, при которых вместе с открытием шторки (диафрагмы) происходит движение заменяющих друг друга клипов, или же изображение одного клипа вырастает на фоне другого. Однако переходы этого типа скорее можно отнести к комбинации обычных переходов, совмещенных с так называемыми спецэффектами. Из изложенного ясно, что все многообразие переходов, которые можно создать, ограничивается только фантазией разработчика.



Рис. 11.9. Распределение прозрачности в маскирующей функции



Рис. 11.10. Зависимость прозрачности маскирующей функции от времени

В настоящее время любая программа, предназначенная для нелинейного монтажа, содержит наборы готовых переходов, которые исчисляются десятками. Примером тому могут служить программы: In:Sinc, SpeedRAZOR, Adobe Premiere.

11.7. Морфинг и варпинг

Морфинг (morphing) представляет собой такой вид обработки изображений, который позволяет создать иллюзию плавного преобразования одного изображения в другое. На рис. 11.11 приведен пример этого преобразования.



Рис. 11.11. Пример применения морфинга: изображение (*a*) преобразуется в изображение (*b*), изображение (*б*) является промежуточной версией

Для того чтобы осуществить такое преобразование, на исходном изображении, которое подлежит преобразованию, и на конечном изображении, в которое должно быть преобразовано исходное, выделяют соответствующие друг другу области, преобразуемые одна в другую. Этих областей может быть достаточно много. Морфинг может быть выполнен либо с одним изображением, либо с двумя изображениями. В первом случае преобразованное изображение является искаженной копией исходного. Однако на практике чаще применяется второй случай, при котором морфинг используется для создания иллюзии постепенного перехода одного изображения в другое. Обычно при морфинге в качестве исходного и конечного применяют неподвижные изображения, однако в общем случае могут использоваться и последовательности кадров, представляющие собой движущиеся изображения. В этом случае применение морфинга позволяет один движущийся объект, например автомобиль, плавно преобразовать в другой движущийся объект, например, в катящийся по дороге железнодорожный вагон. Рассмотрим в качестве примера морфинга неподвижных изображений преобразование одного лица в другое. В этом случае форма первого лица, а именно области изображения, представляющие собой овал лица, границы прически, глаза, уши, нос, рот и т. д., должны быть изменены так, чтобы они были согласованы с соответствующими областями в изображении второго лица. Это согласование соответствующих областей достигается благодаря их оконтуриванию посредством так называемых полевых линий (field lines). От этих линий не требуется, чтобы они были бы соединены между собой, более того, часто требуется как раз обратное для того, чтобы отдельные части изображения преобразовывались независимо.

Рассмотрим простейший случай, когда при морфинге используется всего одна полевая линия. Представим себе, что эта линия проведена через центр исходного изображения горизонтально. На изображении, в которое преобразуется исходное изображение, эта линия программой морфинга будет также помещена горизонтально. Теперь возьмем и повернем на конечном изображении эту линию вертикально. При выполнении программы поворот полевой линии приведет к повороту на 90° всего изображения. Изменяя длину полевой линии на конечном изображении по отношению к длине линии исходного изображения, можно его сжимать или растягивать. Обычно при морфинге используется не одна, а множество полевых линий. Предположим, мы хотим преобразовать посредством морфинга обычное овальное лицо в лицо квадратной формы, как это демонстрировалось в свое время по телевидению в одной из реклам. Для этого следует на исходном изображении оконтурить овал лица, а на конечном изображении полевые линии расположить так, чтобы они образовали квадрат, остальное выполнит программа. Таким образом, полевые линии являются тем инструментом, посредством которого мы можем изменять размеры и форму отдельных областей и даже всего изображения в целом. При работе программа морфинга автоматически создает ряд промежуточных изображений, благодаря чему при демонстрации образовавшейся последовательности кадров создается иллюзия плавного преобразования исходного изображение в конечное.

Варпинг (warping) представляет собой технологию преобразования изображения, напоминающую по своим конечным результатам морфинг, однако варпинг отличается от морфинга, прежде всего, математическими преобразованиями, используемыми для его реализации, а также выполняемыми при этом изменениями в изображении. При варпинге на преобразуемое изображение накладывается сетка, которая разделяет изображение на серию прямоугольных областей. При этом каждый узел сетки, образуемый пересечением горизонтальных и вертикальных линий, может быть перемещен посредством мыши относительно своего исходного положения. Перемещение узла сетки вызывает смещение пикселов изображения, на которое наложена сетка, причем чем ближе к узлу сетки пиксел, тем при прочих равных условиях он больше смещается. Получающийся при этом эффект напоминает деформацию изображения, нарисованного на резиновой мембране при ее местном растяжении. При варпинге так же, как и при морфинге, программа генерирует последовательность промежуточных изображений, благодаря которым создается плавный переход изображения к своей преднамеренно искаженной копии. В настоящее время имеется целый ряд программ, предназначенных для выполнения морфинга и варпинга, начиная с известной программы HSC Digital Morph.

11.8. Принципы анимации

Среди мультимедийных и интернет-приложений анимация занимает особое положение. Использование движения для представления ряда процессов делает эти процессы и их взаимосвязь для зрителя особенно выразительными, наглядными и понятными. Использование для этой цели только текста и графики, как это делается в книгах, менее эффективно. Конечно, наилучшим решением этой проблемы было бы использование видеопоследовательности, однако это, как правило, связано с необходимостью хранения большого объема информации и, следовательно, с более высокими требованиями к объему памяти. Использование для этих целей анимации является хорошим компромиссом. Кроме того, анимация, обладая своими, присущими только ей средствами выражения, имеет самостоятельное значение при создании художественных мультипликационных фильмов.

Принцип действия работы анимационной программы прост и заключается в том, что при предъявлении зрителю на экране монитора ряда быстро сменяющих друг друга изображений, на которых представлены различные фазы движения показываемого объекта, у зрителя создается иллюзия его движения. Для того чтобы движение казалось непрерывным, необходимо, чтобы в секунду предъявлялось более 20 фаз движения, в противном случае движение будет восприниматься прерывистым. В ряде случаев число различных фаз движения, показываемых за одну секунду, может быть меньшим, например, в периодически повторяющихся процессах. Хорошим примером этому может служить демонстрация принципа действия двигателя внутреннего сгорания. При повторяющихся движениях поршня вверх и вниз использование анимации создает полное впечатление работающего двигателя. Алгоритм работы простейшей анимационной программы заключается в том, что она работает с двумя классами графических объектов: спрайтами и средой. Спрайт представляет собой логически законченный движущийся на экране монитора элемент изображения, среда — это фон, по которому осуществляется движение. Движение спрайта происходит в соответствии с определенным сценарием. Известно множество различных способов, в соответствии с которыми реализуется движение спрайта, но все они могут быть сведены к двум группам: комбинационным и морфологическим. В основе комбинационных методов лежат различные варианты следующего алгоритма:

□ вывести на экран вариант спрайта;

□ стереть спрайт;

🗖 вывести на новое место экрана другой вариант спрайта

и т. д.

Поскольку этот процесс протекает достаточно быстро, то для зрителя создается иллюзия непрерывного движения. Морфологические методы реализации движения осуществляются обычно методами объектно-ориентированного программирования.

Различают два вида компьютерной анимации: 2D-анимация (обычная двумерная анимация) и 3D-анимация, в основе которой лежит трехмерное компьютерное мо-

делирование движущихся объектов. При 2D-анимации применяются традиционные методы покадровой анимации. При этом каждое новое изображение содержит изменение по сравнению с предыдущим. В случае 3D-анимации создаются трехмерные компьютерные модели объектов, с которыми в дальнейшем и протекает вся последующая работа.

В настоящее время имеется целый ряд программ, которые позволяют автоматизировать процесс создания последовательности изображений, каждое из которых фиксирует изменения состояния анимируемой сцены. В общем случае эти изменения касаются положения объектов и подобъектов, формы объектов, которая определяется действием различных модификаторов, свойств материалов объектов (цвет, блеск, прозрачность), состояния внешней среды, например, наличие дымки или тумана, а также многих других компонентов. Автоматизация процесса анимации заключается в том, что от пользователя требуется назначить значения анимируемых параметров лишь в некоторых кадрах, которые называются ключевыми. Между ключевыми кадрами могут находиться десятки промежуточных кадров. Значения анимируемых параметров в ключевых кадрах называются ключами анимации. Значения этих параметров в промежуточных кадрах программа рассчитывает автоматически. В ряде случаев можно обойтись и без ключевых кадров, если указать алгоритм изменения анимируемого параметра.

При анимации персонажей или механических устройств, которые состоят из ряда объектов, возникает необходимость осуществить анимацию преобразования объекта и сделать так, чтобы другие объекты повторяли это преобразование. Если, например, происходит поворот плечевой части руки персонажа, то необходимо, чтобы при этом перемещались вслед за плечом предплечье, кисть и пальцы. Для того чтобы этого добиться, нужно связать между собой отдельные объекты, сформировав иерархические цепочки. В этом случае объект, связываемый с другим объектом, становится дочерним, а объект, с которым осуществляется связывание, превращается в родительский. Анимацию преобразования части иерархической цепочки можно распространять в двух направлениях: вверх по цепочке, т. е. к началу, или вниз. Обычно по умолчанию при анимации связанных объектов действует метод так называемой прямой кинематики, при котором преобразования передаются от родительских объектов к дочерним объектам.

Анимационные технологии включают в себя такие эффектные приемы, как: морфинг и варпинг. Благодаря этому, имея, например, в своем распоряжении одноединственное изображение лица человека (фотографию, введенную в компьютер) и применяя варпинг к изображению губ, можно создать последовательность кадров, при просмотре которых возникает иллюзия говорящего персонажа.

Идея создания компьютерных моделей целых сцен не находила практического применения до тех пор, пока не были разработаны современные быстродействующие компьютеры и мощное программное обеспечение. В настоящее время эти технологии применяются в телевидении и кинематографии. Примерами применения этих технологий может служить создание фильма "Титаник" и ряда других. Интенсивное развитие Интернета привело к появлению анимационных программ, ориентированных на эту область использования.

Многие программы, предназначенные для создания мультимедиаприложений, включают в себя средства, позволяющие создавать двумерные анимации. Что касается трехмерных анимационных программ, то, безусловно, лидером здесь является Autodesk 3ds Max, разработанная под Windows.

В заключение этого раздела рассмотрим форматы файлов, применяемые для хранения результатов компьютерной анимации.

- AVI формат, применяемый для записи аудио-/видеоинформации, разработанный для операционной системы Windows. В этом формате используется от 8 до 24 бит на пиксел для передачи информации о цвете. Степень сжатия данных может изменяться.
- □ FLI формат записи компьютерной анимации, специально разработанный для программы Autodesk Animator. Этот формат поддерживается многими программами для анимации, дает возможность записывать изображения с разрешением 320×200 при 8-битной глубине цвета, сжатия нет.
- FLC этот формат, представляющий дальнейшее развитие формата FLI, разработан для записи компьютерной анимации и позволяет записывать изображения с разрешением 1280×1024 при 8-битной глубине цвета. Сжатия нет.
- □ MPEG этот формат предназначен для записи видеоинформации с глубиной цвета от 8 до 24 бит на пиксел, используется также для записи анимационных файлов.
- GIF89а формат записи растровой графики, позволяет записывать анимационные файлы, широко используется в Интернете.
- Shockwave/Flash формат распространяемой графики для хранения изображений и фильмов, разработанных в среде Flash, часто используется в Интернете

и др.

11.9. Спецэффекты

Спецэффекты наиболее часто применяются в рекламных роликах, а также при создании фантастических видеосцен. Спецэффекты подразделяются на двумерные (2D) и трехмерные (3D). Примером двумерных эффектов являются, например, такие широко известные эффекты, как "Мозаика", "Рябь", "Ветер" и т. д. Примером трехмерных эффектов являются "Вращения", "Листание страниц", "Скручивание" и т. п. Спецэффекты могут выполняться либо программным путем, либо аппаратно. Они реализуются посредством так называемых фильтров. Используются спецэффекты при редактировании как клипов, так и неподвижных изображений. В последнем случае для их редактирования применяются обычные графические редакторы, например, Photoshop, Corel PHOTO-PAINT и т. п. Обычно количество различных спецэффектов (фильтров), которые предоставляют в распоряжение создателя цифрового видео пакеты программ, в том числе предназначенные для нелинейного монтажа, составляет от нескольких десятков до нескольких сотен. Так, пакет Photoshop включает в себя 96 фильтров, Adobe Premiere предоставляет в распоряжение монтажера 64 фильтра, а комплект фирмы СОМО, основой которого является видеоплата для PC VideoX, предоставляет возможность в реальном времени выполнять до 400 спецэффектов.

В основе алгоритмов самых разнообразных спецэффектов лежат математические преобразования изображений, в результате которых происходит изменение цвета составляющих их пикселов в соответствии с тем или иным законом, изменение положения пикселов исходного изображения на координатной плоскости по заданному алгоритму, взаимодействие между массивом исходного изображения и массивом специальной маски. Возможно также применение других вариантов преобразования исходных изображений и их комбинаций. В качестве иллюстрации приведем несколько примеров спецэффектов, реализуемых посредством так называемых фильтров, графическим редактором Photoshop.

Размытие в движении. Применение этого фильтра к изображению создает эффект стремительного движения за счет того, что в изображение вводится смаз в направлении движения путем его свертки с прямоугольной импульсной характеристикой. В частном случае, когда имитируется движение вдоль координатной оси *x*, импульсная характеристика имеет вид

$$h(x) = \begin{cases} 1/\Delta x & \text{при } |x| \le \frac{\Delta x}{2}, \\ 0 & \text{при } |x| > \frac{\Delta x}{2}, \end{cases}$$

где Δx — параметр, определяющий величину смаза. Рисунок 11.12 демонстрирует действие фильтра **Motion Blur** (Размытие в движении).



Рис. 11.12. Пример действия фильтра **Motion Blur**: *а* — исходное изображение; *б* — преобразованное изображение

Волна. Применение этого фильтра к изображению вызывает смещение пикселов преобразуемого изображения в соответствии с выбранным законом (синусоидальным, треугольным, по закону меандра) либо в одном направлении, либо в двух взаимно перпендикулярных направлениях. Математически это преобразование для непрерывного изображения в случае синусоидального закона описывается следующим образом:

$$x_1 = x + x_0 \sin\left(\frac{2\pi y}{\lambda_x}\right), \ y_1 = y + y_0 \sin\left(\frac{2\pi y}{\lambda_y}\right),$$

где x и y — декартовы координаты произвольной точки в преобразуемом изображении, x_1 и y_1 — декартовы координаты этой же точки в преобразованном изображении, x_0 и y_0 — амплитуды, а λ_x и λ_y — длины искажающих волн в направлении координат x и y соответственно. Запись этих формул для дискретного изображения, каким собственно и является изображение, обрабатываемое в компьютере, очевидна. Рисунок 11.13 демонстрирует действие фильтра **Wave** (Волна) для случая, когда $y_0 = 0$.



Рис. 11.13. Пример действия фильтра **Wave**: *а* — исходное изображение; *б* — преобразованное изображение

Аналогичным образом работают фильтры **Twirl** (Вихрь), **ZigZag** (Зигзаг), **Ripple** (Рябь), **Spherize** (Сферизация) и ряд других, имеющиеся в графическом редакторе Photoshop, с тем отличием, что изменения координат произвольной точки преобразуемого изображения (для дискретных изображений это будет соответствовать произвольным пикселам) определяются параметрами соответствующих функций, значения которых устанавливаются в диалоге.

Мозаика. Применение этого фильтра к изображению создает эффект картинки, выполненной в технике мозаики, причем все элементы мозаики имеют форму квадратов и одинаковые размеры. Рисунок 11.14 иллюстрирует действие этого фильтра.

Достигается это преобразование весьма простыми средствами. Исходное изображение разбивается на квадратные фрагменты (элементы мозаики) заданного размера, а затем яркость пикселов каждого фрагмента в преобразованном изображении устанавливается равной средней яркости пикселов, составляющих этот фрагмент в исходном изображении. Разновидностью описанного спецэффекта является эффект **Crystallize** (Кристаллизация), по существу та же мозаика, но у которой элементы имеют неправильную форму, имитируя тем самым кусочки стекла. В этом случае границы фрагментов, на которые разбивается изображение, задаются посредством специальной маски.





Рис. 11.14. Пример действия фильтра **Mosaic**: *а* — исходное изображение; *б* — преобразованное изображение

Достаточно большую группу спецэффектов составляют эффекты, которые получаются в результате линейной фильтрации. К эффектам этого вида относятся всевозможные виды размытия и разнообразные виды увеличения резкости и подчеркивания контуров, которые получаются в результате свертки исходного изображения с той или иной маской. Иногда для достижения этих эффектов применяются комбинированные способы, включающие в себя сравнение с порогом компонентов изображения, получаемых в ходе нерезкого маскирования.

В заключение этого раздела отметим, что алгоритмы большинства спецэффектов, как правило, не слишком сложные.

11.10. Цифровая запись видеоданных

Цифровые технологии на этапах съемки видео и монтажа практически вытеснили аналоговые технологии. Это в полной мере касается и видеозаписи. В настоящее время для записи видеоданных используются следующие типы носителей: магнитная лента, оптические диски, RAID-системы, Flash-память и др. До сих пор идет интенсивная полемика о перспективах использования того или иного вида носителя. При этом принимаются во внимание как их технические, так и экономические характеристики. Поскольку магнитная запись имеет существенные отличия от других видов, то в данном разделе рассмотрим именно ее.

На первых этапах развития цифрового телевидения существовали сильные опасения относительно того, что для осуществления магнитной записи потребуется очень большой расход магнитной ленты. К счастью, эти опасения не оправдались, поскольку при цифровой записи оказалось возможным существенно увеличить ее плотность. Благодаря этому расход магнитной ленты при цифровой записи не только не увеличился, но он оказался еще и меньше, чем при аналоговой записи. Обусловлено это тем, что требуемая помехоустойчивость в случае цифрового сигнала может быть обеспечена при значительно меньшем отношении сигнала к шуму, чем в случае аналогового сигнала.

Важный шаг в создании цифровой видеозаписи был сделан фирмой Sony, которая еще в восьмидесятых годах прошлого столетия разработала цифровой кассетный видеомагнитофон первого цифрового формата D1. В 1986 г. D1 получил статус международного стандарта. Этот формат используется и в настоящее время. Поскольку принципиальные технические решения, положенные в основу формата D1, применяются во всех выпускаемых в настоящее время цифровых видеомагнитофонах, то все последующие рассмотрения проведем на его примере. Этот формат предназначен для записи на магнитную ленту размером 3/4 дюйма компонентного сигнала, соответствующего стандарту 4:2:2. Телевизионное изображение при записи разбивается на сегменты, состоящие из 50 телевизионных строк каждый, и записывается на ленту в виде наклонных строк записи. Такая запись называется на телевизионном растре, а на рис. 11.16 — расположение одного из сегментов на телевизионном растре, а на рис. 11.16 —

Один сегмент при записи занимает две строчки записи, при этом полный кадр, в стандарте 625/25, занимает 12 строчек записи. Запись осуществляется посредством 4-х головок, при этом записываемый сигнал делится между четырьмя каналами, каждый из которых соответствует своей головке.



Рис. 11.15. Расположение одного из сегментов на телевизионном растре

Рис. 11.16. Структура записи на магнитной ленте

Это разделение сигнала между четырьмя головками приводит к тому, что каждый сегмент при записи оказывается разделенным на 4 сектора, которые записываются на четырех различных строчках. На рис. 11.16 в качестве примера показаны эти секторы, которые выделены темной закраской. Такое разделение сигнала между головками приводит к так называемому перемешиванию данных, благодаря которому пикселы, расположенные на изображении рядом, на ленте оказываются разнесенными на сравнительно большие расстояния. Наличие перемешивания позволяет ослаблять заметность помех, обусловленных, например, выпадениями, при которых возникают протяженные пакеты ошибок, уничтожающие до 20 000-50 000 расположенных рядом битов данных. Эти ошибки могут иметь различное происхождение. Причиной, в результате которой поражается только один сектор внутри каждого сегмента, может быть загрязнение или отказ одной из головок. Более протяженные пакеты ошибок, поражающие два и более секторов, возникают вследствие царапин и дефектов магнитного покрытия. Поскольку растровые элементы, пораженные помехой, в восстановленном изображении оказываются разнесенными в пространстве, как это показано на рис. 11.17, то заметность помехи на изображении можно существенно ослабить, заменяя яркость искаженных помехой элементов на величину, полученную в результате интерполяции значений яркости элементов, окружающих их. Такая операция носит название маскирования. В связи с этим при записи цифрового телевизионного сигнала на магнитную ленту применяют не только межсекторное перемешивание, но и перемешивание внутрисекторное, путем использования для этой цели буферной памяти.



Рис. 11.17. Распределение ошибок на изображении: *а* — без перемешивания; *б* — с перемешиванием

Другим мощным методом борьбы с ошибками воспроизведения является применение специальных кодов, позволяющих их обнаруживать и исправлять. Принцип действия этих кодов заключается в том, что к информационным байтам кода, который нужно защитить, добавляется некоторое количество проверочных байтов, благодаря чему оказывается возможным обнаруживать и исправлять ошибки. Проиллюстрируем это на простейших примерах. Для обнаружения одиночной ошибки вполне достаточно к каждому кодовому слову добавлять по одному биту так, чтобы сумма всех единиц в образованном новом кодовом слове всегда представляла собой четное число. При передаче по шумящему каналу связи закодированного таким образом сигнала возникновение одиночной ошибки приведет к нарушению четности, и ошибка будет обнаружена. Для того чтобы иметь возможность не только обнаруживать, но и исправлять одиночные ошибки, требуется к защищаемому кодовому слову добавлять большее число проверочных битов. Это можно сделать путем представления данных каждого информационного слова в виде двумерного массива и введения дополнительных разрядов четности, размещаемых в дополнительных колонке и строке (рис. 11.18, *a*). При возникновении одиночной ошибки для ее исправления достаточно определить строку и столбец, в которых произошли нарушения четности числа единиц (см. рис. 11.18, δ), и заменить символ, расположенный по данному адресу, на обратный. В случае возникновения двойной ошибки дополнительные разряды кода позволяют обнаружить сам факт ее возникновения, но не позволяют ее исправить, поскольку для определения неправильно принятых символов кода информации оказывается недостаточно.



Рис. 11.18. Исправление одиночной ошибки

Не останавливаясь более подробно на этом вопросе, отметим, что для эффективной защиты цифровой видеозаписи от ошибок применяют весьма эффективные, так называемые коды-произведения, которые способны обеспечить защиту не только от одиночных ошибок, но также обеспечить защиту и от пакетных ошибок большой протяженности.

Далее в целях ослабления заметности не скорректированных ошибок в стандарте D1 применяется кодирование источника, в результате которого цифровой код заменяется кодом ASE (Adapted Spectral Energy Code). Недостатком цифрового кода является то, что величина ошибки, которая возникает в результате неверно принятого значения разряда, зависит от веса этого разряда. Так ошибка в младшем разряде кодового слова приводит к ошибке при воспроизведении в один квантовый уровень, в то время как ошибка в старшем разряде приводит к ошибке, составляющей 128 уровней квантования. Применение ASE-кода позволяет существенно уменьшить заметность однобитовых и двухбитовых ошибок.

Далее при записи символов кода на ленту применяют так называемое канальное кодирование. При канальном кодировании обеспечивается согласование спектраль-
ного состава кодируемого сигнала с частотной характеристикой канала записивоспроизведения, а также производится запись сигналов синхронизации и идентификации, без которых декодирование считываемого сигнала невозможно.

И, наконец, следует отметить, что при записи на магнитную ленту сигнала изображения одновременно с ним осуществляется запись звука, для которой отводится место на общей дорожке. При этом значительная часть обработки выполняется и для кода изображения, и для кода звука единообразно с использованием одних и тех же технических средств. Поскольку звуковые данные записываются на малой части наклонной дорожки, это делает их более уязвимыми к царапинам и прочим дефектам ленты. В целях повышения помехоустойчивости при записи данные звука дублируются. Кроме того, они размещаются в центральной, более защищенной области ленты. Все перечисленные обработки цифровых сигналов выполняются процессорами записи и воспроизведения, которые представляют собой достаточно сложные устройства, их описанию посвящена специальная литература.

В заключение отметим, что в настоящее время разработан целый ряд различных форматов цифровой видеозаписи, которые различаются между собой размером применяемой магнитной ленты (кроме стандарта 3/4 дюйма используются также ленты 1/2 и 1/4 дюйма), видом записываемого сигнала (компонентный и композитный), а также целым рядом других параметров. В настоящее время находят применение следующие форматы записи:

□ для компонентного сигнала: D1, D5, Digital Betacam, Betacam SX, Digital-S, DV, DVCPRO, DVCPRO50, DVCAM;

□ для композитного сигнала: D2, D3.

Сведения об этих форматах, а также сведения о видеомагнитофонах, в которых они применяются, можно найти в [27, 100].

11.11. Мобильное телевидение и видео

Под *мобильным телевидением* понимается технология, позволяющая просматривать телевизионные передачи на таких устройствах воспроизведения, как: сотовые телефоны, смартфоны и карманные компьютеры, которые имеют экраны небольшого размера и разрешения [18, 24].

Интенсивные дискуссии о мобильном телевидении начались в 2004 г. В настоящее время среди систем мобильного телевидения лидером считается система DVB-H, основанная на стандарте DVB-T, на примере которой рассмотрим основные черты систем мобильного телевидения. К мобильному телевидению предъявляется ряд специфических требований.

Во-первых, система мобильного телевидения должна обеспечивать устойчивый прием изображений не только с неподвижных, но также и с движущихся платформ, а также энергосберегающий режим.

Система DVB-H была разработана для применения в сетях передачи данных сотовой телефонной связи третьего поколения 3G и построена по двухуровневой иерархической схеме. Уровни системы DVB-H соответствуют двум нижним уровням модели взаимодействия открытых систем ISO/OSI, а именно физическому и канальному, при этом в качестве модулей физического уровня в ней используются передатчик и приемник системы DVB-T. Система DVB-H, как известно, обладает высокой помехозащищенностью в условиях многолучевого распространения телевизионного сигнала, а также при движении приемника, что обусловлено используемой системой модуляции — COFDM (частотное уплотнение с ортогональными несущими).

На канальном уровне системы DVB-Н дополнительно выполняются кодирование и временное секционирование. Дополнительное кодирование делается в целях улучшения защиты от импульсных помех и искажений, обусловленных доплеровским смещением частотного спектра принимаемого сигнала вследствие быстрого движения приемника. Временное секционирование (time slicing) используется для снижения энергопотребления в приемнике. Так, например, в мультиплексе DVB-T предусмотрено использование потока 4 Мбит/с для вещания на мобильные приемники по системе DVB-H, при этом поток одной TV-программы составляет 400 Кбит/с. Но данные каждой TV-программы передаются отдельными пакетами, или секциями. Например, если 2 Мбит данных в одной секции следуют со скоростью 4 Мбит/с, то в этом случае каждая секция будет передана и принята в пределах интервала времени, составляющего 0,5 секунд. Этого вполне достаточно для воспроизведения изображения и звука одной программы в течение 5 секунд, а, следовательно, приемник мобильного аппарата может быть переведен в энергосберегающий режим работы на 4,5 секунды из каждых пяти секунд. Благодаря этому удается добиться существенного сокращения энергопотребления приемника.

На канальном уровне системы DVB-H выполняется также многопротокольная инкапсуляция данных. Это означает, что передаваемые пакеты данных могут быть, например, IP-дейтаграммами, пакетами, содержащими данные мультимедийных служб или служб передачи файлов [92].

Другое специфическое требование, предъявляемое к мобильному телевидению, следует из того, что поскольку мобильный телефон (смартфон, карманный компьютер) является персональным устройством, то и просмотр программ на его экране носит индивидуальный характер, а значит услуги, предоставляемые мобильным телевидением, должны удовлетворять потребностям каждого конкретного потребителя. Из этого непосредственно следует, что каждой социальной группе людей должен быть обеспечен свой пакет услуг.

И, наконец, для систем мобильного телевидения необходимо создавать свой контент, который бы учитывал специфику этой технологии, и в частности, тот факт, что разрешение в этой системе существенно ниже, чем в системе вещательного телевидения. Поэтому просто копирование программ традиционного телевидения с простым уменьшением размера и разрешения для трансляции на мобильные терминалы не представляется возможным. Действительно, при простом копировании трансляций футбольных матчей их невозможно будет смотреть, поскольку при этом будет не видно мяча, а передачи из театра потеряют смысл, если невозможно будет различить выражений лиц актеров.

Мобильное видео отличается от мобильного телевидения только тем, что в нем отсутствует канал связи.

11.12. Объемное телевидение и видео

После того как рядом компаний были разработаны и начали выпускаться 3Dмониторы, а также другое оборудование, предназначенное для съемки и демонстрации стереоскопических изображений, значительно возрос интерес к 3D-видео и телевидению. При этом под 3D-видео и телевидением подразумевается передача и демонстрация движущихся стереоскопических изображений. В настоящее время в ряде стран уже ведутся опытные 3D-телепередачи (в указанном смысле), а также появились кинотеатры, демонстрирующие стереоскопическое видео с использованием для этой цели поляризационных очков (*см. разд. 10.1*). По оценкам специалистов настоящее объемное телевидение, основанное на применении многоракурсных и голографических систем, может появиться не ранее, чем через 10—20 лет.

Существуют три основные проблемы на пути создания систем объемного телевидения. Это проблема получения 3D-изображений, проблема их передачи и хранения, а также проблема их отображения.

Для стереоскопических систем эти проблемы в основном решены. Для съемки стереоскопических изображений используются специальные передающие камеры, имеющие по два объектива, разнесенных в пространстве на величину базы. С помощью таких камер снимаются изображения для левого и правого глаз (изображения стереопары).

Что же касается передачи и хранения стереоизображений, то здесь имеются некоторые проблемы. Дело в том, что для сохранения двух изображений, составляющих стереопару, без применения каких-либо специальных приемов, требуется в 2 раза больше памяти, чем для сохранения одного изображения. К счастью, эта проблема имеет как минимум два решения.

Первое решение проблемы основано на использовании сильной взаимной корреляционной связи между распределением яркости и цвета в изображениях для левого и правого глаз. Это делает возможным передавать одно из изображений как базовое, а для второго передавать только сигнал, представляющий его отличие от базового изображения, используя для этого дифференциальную кодово-импульсную модуляцию (ДКИМ), а также координаты смещения между наиболее близкими блоками в обоих изображениях (векторы смещения — нечто вроде векторов движения).

Второе решение заключается в том, что из стереопары формируют собственно изображение передаваемой сцены, цветное или полутоновое черно-белое, в зависимости от того, какими являются оригиналы, а также "карту глубины", или Z-изображение. Последнее можно представить себе в виде полутонового черно-белого изображения, где градациями серого представляется удаленность каждого из элементов первого изображения от снимающей камеры. Поскольку зрительная система имеет пониженную чувствительность к погрешности, с которой передается координата глубины, то Z-изображение можно формировать с меньшей точностью и на этом сэкономить в используемом для его представления цифровом потоке. При этом в случае стереоскопического телевидения цифровой поток увеличивается всего на 25—30% по сравнению с цифровым потоком, который требуется для передачи обычного 2D-телевидения. Сформированные таким образом данные сохраняются или передаются в место приема. Описанный формат представления данных называется форматом 2D+Z. На рис. 11.19 в качестве примера приведены 2Dизображение и сопутствующее ему Z-изображение.



Рис. 11.19. К пояснению формата 2D+Z: *а* — собственно изображение; *б* — карта глубины

Для воспроизведения стереоскопических изображений принятый цифровой поток должен быть преобразован из формата 2D+Z к формату, представляющему исходную стереопару (точнее, близкую к исходной), путем обратного пересчета.

Формат 2D+Z является естественным продолжением концепции представления информации о телевизионном изображении в виде компонентов. Как известно, в современных системах телевидения изображение представляется в виде яркостной и двух цветовых составляющих. Добавление еще одного компонента, компонента глубины, который характеризует "объемность" изображения, логично и хорошо согласуется с принципами совместимости. При использовании этого формата обычные телевизоры будут воспроизводить двумерные изображения, в то время как "3D"-телевизоры (стереоскопические) будут воспроизводить стереоизображения.

Не является проблемой и отображение стереоизображений. В настоящее время для этой цели разработаны специальные 3D-мониторы и проекционные устройства. В частности, компания Philips промышленно выпускает 3D-мониторы, основанные на линзово-растровом принципе. Входными данными таких мониторов являются сформированные из стереопары обычное изображение и карта глубины (формат 2D+Z).

Восстановление стереопары происходит путем интерполяции исходного изображения с учетом карты глубины, и только после этого изображения для левого и правого глаз могут быть выведены на экран.

По сообщениям СМИ в настоящее время многие мировые киностудии уже стали снимать новые фильмы в формате стереопары, т. е. сразу двумя камерами.

Систему действительно объемного телевидения принципиально можно реализовать, используя 3D-мониторы, формат 2D+Z передачи и многоракурсную камеру. Такая система позволит зрителю изменять позицию наблюдения и рассматривать изображения объектов под разными ракурсами [34].

11.13. Виртуальные студии

В основе технологии виртуальной студии лежит технология рирпроекции, которая вначале была разработана в кино, а затем ее стали широко применять в телевидении. Эта технология заключается в том, что результирующее изображение, включающее в себя живых персонажей (актеров, дикторов и т. д.) и фоновое изображение составляется из двух изображений: например, изображения играющих актеров и изображения декорации, на фоне которой разворачивается действие. Вначале выполняется съемка актеров, во время которой их снимают на однородном голубом, синем, или зеленом фоне. Поскольку фон однородный, а цвет его отличается от цвета одежды и открытых частей тела актеров, то по этому признаку легко отделить изображения актеров от изображения фона и сформировать их силуэтные изображения, так называемую маску, которая сохраняется в альфа-канале. Фоновое изображение, например изображение декорации, снимается отдельно или изготавливается посредством компьютерных приложений, например Adobe Photoshop, 3ds Max. На последнем этапе изображения актеров впечатываются в изображение фона, используя для этой цели масочное изображение. Делается это посредством следующего простого алгоритма. Одновременно сканируются три изображения: изображение маски $L_{M}(k,n)$, фоновое изображение $L_{d}(k,n)$ и изображение с актерами $L_{a}(k, n)$. Здесь, как и ранее, через k обозначен номер строки, а через n номер столбца, определяющие положение пиксела в растровом изображении. В том случае, если пиксел на изображении маски находится вне пределов силуэта одного из актеров, который будем считать черным, в результирующее изображение $L_{c}(k,n)$ включается пиксел фонового изображения, в противном случае в него включается пиксел изображения одного из актеров.

$$L_{c}(k,n) = \begin{cases} L_{\phi}(k,n) & \text{при } L_{M}(k,n) > 0, \\ L_{a}(k,n) & \text{при } L_{M}(k,n) = 0. \end{cases}$$

В настоящее время технология рирпроекции широко применяется в телевидении, например при передаче новостей, прогнозов погоды и т. д.

Технологии, применяемые в виртуальных студиях, можно рассматривать как логическое развитие технологии рирпроекции [17]. Важным в этих технологиях является возможность перемещать камеру при съемке в пределах студии или переключаться с одной камеры в студии на другую. Однако при перемещении камеры или замене одной камеры на другую изменяется не только ракурс снимаемого персонажа, но также изменяются ракурсы, под которыми видны предметы, составляющие декорацию. Поэтому при смене плана или ракурса должно соответственно меняться и изображение фона. Рирпроекция такой возможности не обеспечивает. Для того чтобы обеспечить необходимое изменение фонового изображения, генерируемого, например приложениями 3ds Max или Maya, в этих приложениях создаются так называемые виртуальные камеры и виртуальные источники освещения, которые обладают теми же характеристиками, что и реальные, установленные в студии. Эти виртуальные камеры "помещаются" в тех же пространственных координатах генерируемой приложением сцены, что и реальные камеры в реальной студии, и повторяют их перемещения, изменения фокусных расстояний объективов и т. д. Благодаря этому снимаемые виртуальными камерами изображения сгенерированной сцены точно вписываются в результирующее изображение. Для определения координат реальных камер и передачи полученных данных в систему моделирования и визуализации виртуальных объектов служат датчики, которые устанавливаются на штативах, пьедесталах или других приспособлениях для установки камер. В соответствии с вводимыми данными параметры виртуальных камер обновляются, в результате чего корректируются характеристики виртуальных объектов, а именно их положение относительно виртуальных камер, освещенность, отбрасываемые тени и т. д. Все это позволяет создать на экране телевизора реалистичное изображение.

Источником информации о положении камер в пространстве сцены может служить также сама ткань для рирпроекции с нанесенной на нее специальной сеткой (Pattern Recognition). Эта сетка используется для определения координат камеры специальной системой, входящей в комплект устройства распознавания координат. В рассматриваемом случае данные о положении камеры определяются посредством анализа видеосигнала. При использовании этого метода оказывается возможным использовать обычные штативы и объективы без специальных датчиков.

Для того чтобы реальный объект видеосъемки, например актер, мог бы перемещаться по всей глубине виртуальной сцены, в том числе занимая позиции, при которых некоторые объекты виртуальной сцены экранируют его, применяется 2,5Dтехнология (*см. разд. 10.2*). При этом результирующее изображение компонуется из ряда слоев, и объект видеосъемки в зависимости от своего положения по глубине сцены может оказываться между слоями компьютерных декораций.

В настоящее время виртуальные студии выпускаются сравнительно небольшим количеством компаний, это зарубежные компании Datavideo, Orad, FOR-A, Vizrt, Ultimatte и некоторые другие, а также отечественные "СофтЛаб-НСК" и "Эра".

Следует отметить, что роль виртуальных студий возрастает по мере перехода от телевидения стандартного разрешения к телевидению высокой четкости.

В заключение, в качестве примера, приведем несколько типов виртуальных студий, выпускаемых в настоящее время, достаточно подробные характеристики которых читатель найдет в [17].

Datavideo MS-DVK1, двухканальная виртуальная студия, выпускаемая компанией Datavideo, построенная на базе аппаратных устройств рир-проекции DVK100, позволяющих совмещать видео и сигнал фона. Достоинством Datavideo MS-DVK1 является ее сравнительная простота.

Компанией Orad выпускается оборудование и программное обеспечение для создания виртуальных студий. Виртуальная студия ProSet является примером наиболее универсальной и мощной студии, выпускаемой этой компанией. В своей работе эта студия использует такое программное обеспечение, как 3ds Max и Maya.

SmartSet является примером недорогой виртуальной студии, разработанной этой же компанией. SmartSet создана для оформления виртуальных декораций и предназначена для использования в любом виде трансляций, таких как новости, спортивные передачи, финансовые отчеты, развлекательные шоу и многое другое. При использовании SmartSet небольшая студия размером $3 \times 3 \text{ м}^2$ может быть превращена в большую виртуальную студию.

Компания FOR-A в сотрудничестве с фирмой Brainstorm Multimedia разработала виртуальную студию digiStorm, которая представляет собой мощную систему виртуальной реальности с развитыми графическими функциями и богатым инструментарием. Одним из достоинств этой студии является легкость освоения ее персоналом телекомпании.

Виртуальная студия VRCAM, выпускаемая компанией FOR-A, отличается отсутствием датчиков движения и сравнительно невысокой стоимостью.

В заключение назовем еще виртуальные студии семейства "Фокус" компании "СофтЛаб-НСК".

11.14. Аппаратные средства и пакеты программ для работы с цифровым видео

Для нелинейного монтажа необходимо иметь устройство ввода/вывода видео в реальном времени, достаточно быстродействующий компьютер, большую оперативную память, а также накопитель в виде жесткого диска большой емкости. Еще совсем недавно каждое из этих устройств налагало существенные ограничения на характеристики создаваемого видео, поскольку быстродействие плат ввода/вывода и компьютеров были небольшими, величина оперативной памяти составляла 8—16 Мбайт, а емкость жестких дисков не превышала 4 Гбайт. Однако с течением времени ситуация быстро меняется. В настоящее время персональные компьютеры с высокой производительностью не являются чем-то особенным, оперативная память в 1024 Мбайт и более вполне обычна, а емкость жестких дисков перешагнула рубеж 2 Тбайт. И это уже не говоря о компьютерах, специально ориентированных на работу с цифровым видео. Аналогичный прогресс имел место и в отношении устройств ввода/вывода видео. Одновременно с развитием аппаратных средств имело место и развитие программного обеспечения, в результате чего были созданы системы нелинейного монтажа. Приведем краткие данные о некоторых из них.

Система нелинейного монтажа Night Suite создана на базе IBM PC-совместимого компьютера с операционной системой Windows. В ней применен метод сжатия данных в соответствии с алгоритмом JPEG с переменным коэффициентом сжатия от 100:1 до 2:1. Запись изображения и звука производится на отдельные жесткие диски. Количество используемых для записи жестких дисков может составлять 35 для видео и 7 для звука. В системе имеется генератор рир-проекции по цвету, яркости и альфа-каналу. Титры вводятся посредством пакета Inscriber CG. Поддерживается пакетная оцифровка материала и автоматическая сборка программы по монтажному листу. Ввод звука осуществляется с двух аналоговых и двух цифровых входов. Частота дискретизации звука 44,1 и 48 кГц при 16 битах на отсчет.

Система Media Illusion, созданная на базе рабочих станций Silicon Graphics, позволяет осуществлять цифровой монтаж, постобработку, вводить спецэффекты и редактировать изображения. Система позволяет также создавать собственные эффекты. Кроме того, имеются возможности анимации освещения, управления 3D-камерой, создания эффектов морфинга и варпинга. Система Media Illusion включает в себя большинство известных технологий рир-проекции, а также имеет большой набор настраиваемых фильтров, среди которых: мягкая фокусировка, регулировка яркости, удаление и добавление фактуры изображения и т. д.

Фирмой EYEON Software выпущен программный продукт Digital Fusion, предназначенный для создания кино- и видеопрограмм, который имеет широкие возможности в отношении создания спецэффектов. Пакет Digital Fusion предназначен для работы в среде Windows на платформах Intel Pentium и DEC. Пакет поддерживает прямой интерфейс с распространенными платами ввода/вывода, в том числе с платой Hollywood фирмы DPS, которая работает без компрессии в 10-битовом формате D1. В этом пакете имеются следующие цифровые спецэффекты: псевдотрехмерные деформации (вращения, перемещения, изменения масштаба), имитация стекающих капель, вихревого движения и т. д., ореол и имитация бликов в оптике, смаз, цветокоррекция. Одним из наиболее интересных инструментов пакета является возможность отслеживать последовательность изображений, которая позволяет автоматически определять положение заданного участка изображения в очередном кадре видеопоследовательности, исходя из заданных цветовых и яркостных характеристик этого участка. Благодаря этому можно привязывать определенный спецэффект к области, изменяющей свое положение от кадра к кадру.

Упоминавшийся уже программный продукт Speed Razor Mach 3.5 компании In:sync предназначен для нелинейного монтажа видео и звука на IBM PC-совместимых компьютерах и DEC Alpha. Этот пакет имеет прямой интерфейс с такими устройствами ввода/вывода видео в реальном времени, как: DPS PAR, Perception, Hollywood, Truevision TARGA 1000/2000 Pro, TARGA 2000 DTX/RTX, Intergraph Video Engine и Studio Z. Программный продукт Speed Razor может работать практически с неограниченным числом видео и звуковых слоев и с разрешением от 720×576 до 4000×4000. Имеется большое число всевозможных переходов и эффектов. Фильтры этого пакета включают в себя размывание изображения, подкрашивание, цветовую и яркостную коррекцию, инверсию цвета и яркости, замораживание кадра, свечение, изменение линейных размеров изображения, скручивание, имитацию бликов на линзах, струй дождя и т. п. Можно использовать комбинирование видеофрагментов по альфа-каналу, применяя статичные и движущиеся маски. Каждый из перечисленных эффектов можно применять к неограниченному числу слоев. Частота дискретизации звука, используемая в пакете Speed Razor, составляет 44,1 кГц при 16 битах на отсчет, при этом возможно микширование до 20 стереодорожек в реальном времени.

Информацию о других пакетах программ, предназначенных для работы с цифровым видео, и используемых при этом аппаратных средствах можно найти в обзоре [94].

Глава 12



Сжатие изображений без потерь информации

12.1. Проблема сжатия изображений

Известно, что для записи изображений требуются достаточно большие объемы памяти, часто в единицы и даже десятки мегабайтов.

Так, например, отсканированное фотографическое цветное изображение размером 1732×1165 пикселов при 24 битах на пиксел (режим True Color) требует 5,8 Мбайт. Такие ресурсы оказываются недостижимыми, если изображение необходимо записать на дискету или передать по Интернету. Однако наличие специально разработанных форматов записи графических файлов, практически каждый из которых базируется на том или ином, а иногда и на нескольких алгоритмах сжатия изображений, снимает остроту проблемы. Посредством сжатия (компрессии) изображений удается в несколько раз и даже в ряде случаев в десятки раз сократить цифровой поток, представляющий изображение. Для вышеупомянутого изображения при записи его в формате TIFF требуется приблизительно 4,9 Мбайт, а если использовать формат JPEG, то можно получить разные степени сжатия в зависимости от жесткости требований, предъявляемых к качеству изображения. В данном случае максимально высокого качества изображение будет занимать всего 2,4 Мбайт, среднего (вполне приемлемого для большинства задач) качества 0,2 Мбайт, а применение максимального сжатия позволит довести эту цифру до 0,07 Мбайт.

Подчеркнем, что при записи изображений всегда остается важным вопрос сохранения качества изображения. Как правило, проблема эффективного сжатия изображений решается либо без потери качества, либо с минимальными потерями, практически незаметными для зрителя. Это оказалось возможным благодаря свойственным изображениям статистической и психофизической избыточности, что позволяет произвести такое кодирование изображения, при котором для его записи потребуется существенно меньше двоичных единиц кода. Методы сжатия изображений, основанные на устранении избыточности обоих типов, будут рассмотрены в последующих разделах.

12.2. Избыточность изображений

Статистическая избыточность сигнала неподвижного изображения обусловлена наличием сильных статистических связей между его смежными пикселами, а также тем, что различные уровни их яркости имеют разную вероятность. Применительно к последовательности кадров, в случае видео, имеется дополнительный источник статистической избыточности, который обусловлен наличием сильных статистических связей между значениями яркости соответствующих пикселов в смежных кадрах. На изображении это проявляется в том, что соседние пикселы имеют одинаковые или близкие яркость и цветовой тон. Резкие скачки этих параметров наблюдаются только при наличии контуров, в случае статических изображений, или при наличии перемещающихся объектов в последовательности кадров. Статистическая избыточность сигнала, которым представлено изображение, как неподвижное, так и движущееся, может быть устранена или сильно уменьшена, а следовательно, уменьшен цифровой поток путем его соответствующей перекодировки (сжатия). Эта перекодировка включает в себя два этапа — вначале декорреляцию сигнала, а затем представление часто встречающихся значений сигнала более короткими кодовыми комбинациями, а редко встречающихся значений — более длинными кодовыми комбинациями.

При этом не происходит потери информации, поскольку исходное изображение может быть точно восстановлено. Вследствие этого методы, реализующие этот принцип сжатия изображений, называются *методами сжатия без потери информации*, или энтропийными методами (от слова "энтропия" — одного из основных понятий теории информации).

Согласно Шеннону энтропия является мерой, устанавливающей среднее количество информации на символ сообщения (в данном случае на пиксел изображения). Для последовательности из m статистически независимых символов, появляющихся с вероятностями p_i , энтропия выражается в следующем виде:

$$H = -\sum_{i=1}^{m} p_i \log_2 p_i , \qquad (12.1)$$

где \log_2 — двоичный логарифм, *i* — номер символа.

Если вероятность появления некоторого символа сообщения равна единице, а остальных нулю, т. е. неопределенность появления данного символа отсутствует, и энтропия будет равна нулю. В случае же, когда вероятности появления всех символов одинаковы

$$p_i = \frac{1}{m}$$
,

энтропия достигает своего максимального значения, равного

$$H_{\max} = -\sum_{i=1}^{m} \frac{1}{m} \log_2 \frac{1}{m}.$$

Сопоставляя найденное значение энтропии с ее максимальным значением, определяют величину избыточности сигнала следующим образом:

$$R = 1 - \frac{H}{H_{\text{max}}} \,. \tag{12.2}$$

В том случае, когда вероятности появления всех символов одинаковы

$$p_i = \frac{1}{m},$$

и избыточность, как это ясно из изложенного, отсутствует.

Коэффициент, показывающий, во сколько раз можно уменьшить число двоичных единиц кода, требующихся для представления сообщений источника с энтропией *H* (в рассматриваемом случае изображения), по сравнению со случаем, когда при том же наборе символов все символы источника сообщения кодируются кодовыми словами одинаковой длины, называется *коэффициентом сжатия*

$$k_{\rm CW} = \frac{H_{\rm max}}{H}.$$
 (12.3)

До сих пор мы рассматривали случай, когда смежные растровые элементы изображения были статистически независимы, т. е. в качестве изображения был выбран белый шум. Однако в реальных изображениях значения яркостей смежных пикселов взаимно коррелированы. В этом случае, располагая значением сигнала, представляющего яркость некоторого пиксела, можно с некоторой вероятностью предсказать значения сигналов от соседних пикселов. Следовательно, информация, привносимая последующим пикселом в случае знания предшествующего, будет меньше, чем в случае, когда сигналы, представляющие значения яркости пикселов, были бы статистически независимы. Величина энтропии H в этом случае должна рассчитываться уже по другой формуле:

$$H = -\sum_{i}^{m} p(i) \sum_{j}^{m} p_{i}(j) \log_{2} p_{i}(j), \qquad (12.4)$$

где $p_i(j)$ — условная вероятность появления *j*-го символа, если предыдущим был *i*-й символ. Формула (12.4) является более общей и в частном случае, когда статистическая связь между пикселами отсутствует, она переходит в формулу (12.1). Действительно, в этом случае, поскольку $p_i(j)$ от *i* не зависит, мы можем его заменить на p(j) и записать

$$H = -\sum_{i}^{m} p(i) \sum_{j}^{m} p(j) \log_2 p(j),$$

а т. к. внутренняя сумма не зависит от *i*, то суммы можно поменять местами

$$H = -\sum_{j}^{m} p(j) \log_2 p(j) \sum_{i}^{m} p(i).$$
 (12.5)

Но поскольку $\sum_{i}^{m} p(i) = 1$, т. к. суммируются все вероятности p(i), формула (12.5)

переходит в формулу (12.1), что и требовалось доказать.

В рассматриваемом случае, когда элементы изображения взаимно коррелированы, энтропия H будет меньше, чем в случае, если бы значения сигнала были статистически независимы. Вследствие этого достижимый коэффициент сжатия возрастает. Первым шагом при сжатии данных обычно является декорреляция кодируемой последовательности, при которой устраняются статистические связи между кодируемыми отсчетами, и уже затем производится энтропийное кодирование статистически независимых отсчетов.

Психофизическая избыточность изображений обусловлена особенностями зрительной системы человека. Учет этих особенностей позволяет реализовать сжатие данных при передаче или записи изображений. Дело в том, что не все детали изображения одинаково воспринимаются зрителем. Например, детали, имеющие одновременно и малые размеры, и малый контраст, вследствие ограничений со стороны контрастной чувствительности зрения не видны зрителю на изображении, а поэтому без всякого ущерба для качества его воспроизведения могут не передаваться. Благодаря этому цифровой поток, которым передается изображение, может быть сокращен. Этот вид избыточности называется *психофизической избыточностью изображений*.

Проиллюстрируем сказанное на примере кодирования изображения звездного неба. Известно, что пороговый контраст зрения в сильной степени зависит от угловых размеров наблюдаемых объектов. Так, например, если при наблюдении объектов, имеющих большой угловой размер, пороговый контраст составляет около 0,02, то при наблюдении точечных объектов, в нашем примере — звезд, он составляет не более 0,1. Поэтому, передавая изображение пиксел за пикселом, как обычно, можно квантовать яркость звезд всего на 16 уровней, расходуя на представление яркости каждой из них по 4 двоичных единицы, а не по 8. Нетрудно видеть, что в этом случае благодаря уменьшению затрат двоичных единиц кода на представление яркости звезд имеет место сокращение кодовой последовательности и, следовательно, сжатие данных (изображения), которое в данном случае составляет 2 раза.

Описанный метод кодирования относится к группе методов *сжатия данных с потерями информации*. Смысл этого термина заключается в том, что после декодирования распределение яркости в восстановленном изображении отличается от того распределения, которое было до его кодирования, т. е. имеет место искажение изображения и соответственно потеря информации. Но следует иметь в виду, что речь здесь идет о потере информации, которой зрительная система не в состоянии воспользоваться в силу присущих ей ограничений.

На практике при сжатии изображений применяются как методы сжатия данных с потерей информации (обычно эти методы основаны на сокращении не только психофизической избыточности, но и статистической), так и методы сжатия данных без потерь информации.

12.3. Декорреляция сигнала изображения

Ранее было отмечено, что избыточность изображения обусловлена наличием сильных корреляционных связей между значениями яркости смежных пикселов и, кроме того, неравномерностью распределения плотности вероятности их значений, которая мала, что поясняется рис. 12.1, на котором приведена плотность вероятности значений яркости в исходном изображении W(L).

В связи с этим, первым шагом при использовании некоторых видов энтропийного кодирования является декорреляция кодируемой последовательности пикселов, при которой устанавливаются статистические связи между кодируемыми пикселами. Вследствиие декорреляции увеличивается неравномерность распределения плотности вероятности их интенсивностей, и уже только после этого производится кодирование статистически независимых отсчетов. Простейшим, но не оптимальным способом декорреляции, является преобразование последовательности отсчетов кодируемого сигнала, представляющего яркость пикселов изображения L(n), в последовательность отсчетов приращений этой яркости $\Delta L(n)$ при переходе с одного пиксела на другой, т. е.

$$\Delta L(n) = L(n+1) - L(n),$$

где n — номер отсчета. В результате такого преобразования статистические связи между кодируемыми отсчетами сильно ослабляются, а распределение плотности вероятности их значений становится резко неравномерным. Отмеченное поясняется рис. 12.2, на котором приведена плотность вероятности приращения яркости $W(\Delta L)$. Из сопоставления рис. 12.1 и 12.2 видно, что во втором случае плотность вероятности распределения приращений резко неравномерна, благодаря чему сигнал последовательности приращений обладает большой избыточностью, а, следовательно, может быть в большей степени сжат.



Рис. 12.1. График зависимости плотности вероятности *W*(*L*) от яркости *L*



Рис. 12.2. График зависимости плотности вероятности W(L) от приращения яркости ΔL



Рис. 12.3. Изображения: а — исходное; б — декоррелированное

Изображения, представленные на рис. 12.3, а и б, иллюстрируют сказанное.

Рисунок 12.3, *а* представляет собой исходное изображение, а рис. 12.3, δ — это же изображение, но после его декорреляции. При изготовлении рис. 12.3, δ к приращениям яркости $\Delta L(n)$, возникающим при переходе с одного пиксела на другой, добавлен серый фон для того, чтобы воспроизвести отрицательные приращения яркости, которые в противном случае были бы ограничены. Сравнивая эти изображения, мы замечаем, что в исходном изображении представлены все градации яркости примерно с одинаковой вероятностью, в то время как в декоррелированном изображении вероятность больших приращений (больших отклонений от серого фона) сравнительно мала и имеет место только на контурах. В дальнейшем, в *разд. 13.1* будет описан более совершенный вид декорреляции изображений.

12.4. Кодирование длин серий

Кодирование длин серий, или как его еще называют *RLE* (Run-Length Encoding), в настоящее время широко применяется при записи графических изображений в файлы либо как самостоятельный метод, либо в составе более сложных алгоритмов кодирования, применяемых в различных форматах графических файлов, например в JPEG [13, 50]. Этот метод применяется также в таких распространенных форматах, как PCX, TIFF и TARGA.

Известно, что многие графические изображения, например, чертежи, плакаты и т. п. включают в себя значительные однородные области, имеющие одинаковые яркость и цвет. При разложении таких изображений в растр наличие однородных областей приводит к появлению в строках последовательностей отсчетов, имеющих одни и те же значения, как показано на рис. 12.4. Эта особенность позволяет при их сжатии расходовать меньше двоичных единиц, чем при традиционном методе кодирования, записывая лишь длину серии (число повторений одинаковых отсчетов) и значение яркости, с которого начинается серия. Так при использовании метода кодирования длин серий для кодирования отсчетов яркости, показанных на рис. 12.4, получим следующую кодовую последовательность: 0,0; 2,1; 5,3; 1,2. Из изложенного следует, что при использовании этого метода в кодируемом сигнале устраняются (строго говоря, ослабляются) корреляционные связи.



Рис. 12.4. К пояснению метода кодирования длин серий

Определим величину коэффициента сжатия, которое обеспечивается при использовании этого метода. Учитывая, что для записи числа повторений одинаковых отсчетов в последовательности, максимальная протяженность которой равна N, необходимо затратить $\log_2 N$ двоичных единиц, а также необходимо затратить $\log_2 m$ двоичных единиц для записи значения самой величины, где m — число уровней квантования яркости в кодируемом изображении, найдем, что затрата двоичных единиц для записи последовательности составит

$$N_{\text{посл}} = \log_2 N + \log_2 m$$
.

Обозначая вероятность нового значения, т. е. вероятность появления последовательности, через $p_{\text{нов}}$, а число строк в изображении и число отсчетов в строке, соответственно, через $N_{\text{стр}}$ и $N_{\text{пикс}}$, найдем, что полная затрата двоичных единиц кода для записи изображения будет равна $p_{\text{нов}}N_{\text{стр}}N_{\text{пикс}}(\log_2 N + \log_2 m)$. Принимая во внимание, что при традиционном кодировании для записи такого изображения потребуется $N_{\text{стр}}N_{\text{пикс}} \times \log_2 m$ двоичных единиц, находим, что коэффициент сжатия $k_{\text{сж}}$, обеспечиваемый от применения метода кодирования длин серий, составит

$$k_{\rm cxc} = \frac{\log_2 m}{p_{\rm HOB} \left(\log_2 N + \log_2 m\right)}$$

Из этой формулы видно, что коэффициент сжатия сильно зависит от вероятности появления новых значений $p_{\text{нов}}$. При малых значениях вероятности новых значений коэффициент сжатия оказывается большим, но быстро убывает при ее увеличении. К сожалению, статистика полутоновых изображений такова, что при 256 уровнях квантования практически каждый новый отсчет (пиксел) представляет новое значение, т. е. $p_{\text{нов}} \cong 1$. Обращаясь к формуле, видим, что при $p_{\text{нов}} \cong 1$ коэффициент

сжатия оказывается меньше единицы, т. е. применение описанного метода приводит не к сокращению числа двоичных единиц, а к увеличению. Объясняется это тем, что в этом случае дополнительная затрата двоичных единиц идет на представление длительности последовательностей, хотя их протяженность почти всегда равна единице.

Недостатком этого метода является также его низкая помехоустойчивость. Даже редкие помехи приводят либо к появлению на изображениях протяженных штрихов, поскольку помеха изменяет значение яркости всей последовательности, либо, что еще хуже, к "раздергиванию" строк, если помеха искажает данные о числе повторения отсчета в последовательности. Достоинством же этого метода является простота его реализации. Отмеченные особенности определили и область его применения, а именно при записи графических изображений, в том числе цветных, содержащих большие однородные поля.

12.5. Кодирование методом LZW

Алгоритм сжатия, реализованный в этом методе, был изобретен Лемпелом (Lempel) и Зивом (Ziv) в 70-е годы прошлого века (версии LZ77 и LZ78 названы по первым буквам фамилий соавторов и по годам разработки), а доработан Терри Уэлчем (Welch) в 1984 г. В настоящее время метод LZW используется в форматах записи как графической, так и гипертекстовой информации: GIF, TIFF, PDF и ряде других. Особенностью этого метода является адаптивность и использование кодов переменной длины с максимальной длиной 12 двоичных единиц [13, 23, 50].

Рассмотрим принципы, положенные в основу метода сжатия LZW. Будем считать, что сжатию подлежит черно-белое полутоновое изображение, проквантованное по яркости на 256 уровней. Сжатие начинается с того, что строится (инициализируется) первоначальная таблица кодов, в которой каждому уровню квантования сопоставляется код, представляющий собой двоичную запись номера уровня квантования. Так, например, нулевому уровню квантования приписывается значение кода — 0, первому уровню квантования значение — 1 и т. д., 255-му уровню квантования значение — 255. Всего, как это видно из изложенного, такая таблица содержит 256 значений кода. Далее в таблицу записываются еще два кода: код очистки, которому присваивается значение 256, и код конца записи — 257. Код очистки используют для того, чтобы не произошло переполнение таблицы, которая по принятому соглашению может включать коды протяженностью не более 12 двоичных единиц (числа, не превышающие 4095). Он необходим, т. к. по мере заполнения таблицы и соответствующего увеличения длины кодового слова имеет место переход к кодам протяженностью в 10, 11 и 12 двоичных единиц. Код очистки инициализирует таблицу заново, стирая в ней все коды, начиная с 258 и выше и освобождая тем самым место для кодового представления встречающихся в изображении комбинаций символов. Код конца записи, как это видно из его названия, сигнализирует о том, что кодируемая последовательность окончилась. После завершения подготовительных операций алгоритм готов к началу сжатия данных (изображения).

Алгоритм сжатия данных можно записать следующим образом:

- 1. Инициализировать, т. е. ввести первоначальную таблицу кодов.
- 2. Очистить таблицу кодов, начиная с кода 258 и до конца.
- 3. Очистить буфер строки (String), буфер строки (Test) и буфер строки (Byte).
- 4. Далее в цикле:
 - прочитать очередной байт кодируемых данных в буфер (Byte);
 - сцепить (конкатенировать) String + Byte и поместить результат в буфер Test;
 - проверить, имеется ли в таблице кодов код, соответствующий комбинации, помещенной в буфер Test:
 - если имеется, то содержимое буфера Test переписать в буфер String и перейти в начало цикла;
 - если нет, то добавить в таблицу код, соответствующий содержимому буфера Test, присвоив ему значение, совпадающее со следующим свободным порядковым номером, вывести в выходной поток код, соответствующий содержимому буфера String, переписать содержимое буфера Byte в String и перейти в начало цикла.
- 5. Работа программы заканчивается тем, что делаются записи в выходной поток кода содержимого String и кода конца записи.

В результате применения такого алгоритма получаем коды переменной длины, причем для сочетаний из двух-трех символов, каждый из которых в отдельности описывается в таблице 8-разрядным кодом, длина полученных кодов будет составлять не 16 и не 24 бита, а существенно меньше.

Переходя к декодированию (декомпрессии) сжатых данных, отметим, что декодирующий алгоритм не нуждается в том, чтобы вместе со сжатой последовательностью ему бы передавалась и кодовая таблица. Декодирующий алгоритм, получая коды комбинаций исходных отсчетов, составляет по ним кодовую таблицу, идентичную той, которую составляет кодирующий алгоритм. Этот алгоритм несколько сложнее кодирующего и может быть записан следующим образом:

- 1. Прочитать новый код сжатых данных (Newcode).
- 2. Если Newcode представляет собой код конца записи, то завершить работу.
- 3. Если Newcode представляет собой код очистки, то необходимо:
 - проинициализировать таблицу кодов;
 - прочитать следующий код сжатых данных (если это будет код конца записи, то завершить работу);
 - найти Newcode в кодовой таблице и вывести соответствующую ему декомпрессированную последовательность отсчетов;
 - скопировать Newcode в буфер, где был записан предыдущий код (Prevcode).

- 4. Если же Newcode находится в таблице, но не является ни кодом очистки, ни кодом конца записи, то необходимо:
 - вывести соответствующую ему декомпрессированную последовательность отсчетов;
 - взять первый байт декомпрессированного кода Newcode и декомпрессированного кода Prevcode, конкатенировать их и добавить в кодовую таблицу;
 - скопировать Newcode в буфер, где хранится Prevcode.
- 5. Если Newcode в таблице отсутствует, а, кроме того, он не является кодом очистки и кодом конца записи, то необходимо:
 - конкатенировать и вывести значение декомпрессированного кода Prevcode+ первый байт того же значения;
 - добавить в таблицу элемент для вышеприведенного значения;
 - скопировать Newcode в буфер Prevcode.

Метод сжатия LZW может быть применен не только для сжатия данных, каждая единица которых имеет размер в один байт, например, отсчетов яркости черно-белого полутонового изображения, но также и для сжатия данных, имеющих произвольный размер. В этом случае кодовые последовательности этих данных объединяются в группы по 8 двоичных единиц. Так если каждый отсчет содержит 4 двоичных единицы, то объединение в группы происходит по два отсчета, а если один отсчет представлен 16 двоичными единицами кода, то такая кодовая последовательность делится пополам. Величина сжатия полутоновых и цветных изображений, обеспечиваемая при использовании этого метода, невелика и составляет примерно 1,2 раза.

12.6. Метод кодирования Хаффмена

Этот метод позволяет получить код с минимальной средней длиной при заданном распределении вероятностей значений некоррелированных отсчетов сигналов при условии, что вероятности появления отсчетов с этими значениями будут равны $1/2^n$, где n — целое число. В противном случае степень сжатия, обеспечиваемая использованием этого кода, будет ниже. Сказанное поясняется рис. 12.5, на котором тонкой линией показано количество двоичных единиц кода, требующихся на отсчет при идеальном сжатии, а жирной линией — количество двоичных единиц кода при использовании кода Хаффмена.

Особенностью этого метода кодирования является использование кодов переменной длины, при этом наиболее вероятным символам присваиваются наиболее короткие кодовые слова, а менее вероятным — длинные.

Поясним это на примере построения кодовой таблицы. На рис. 12.6 показано кодовое дерево применительно к случаю кодирования шести символов A_1 , A_2 , A_3 , A_4 ,

*A*₅, *A*₆, которые могут представлять собой значения яркости пикселов изображения, и приведены вероятности, с которыми они появляются.



Рис. 12.5. Зависимость необходимого количества двоичных единиц на отсчет: *1* — при идеальном методе сжатия; 2 — при сжатии методом Хаффмена



Рис. 12.6. Кодовое дерево

Построение кодовой таблицы начинается с того, что два символа с наименьшими вероятностями объединяются в узел кодового дерева, которому приписывается их суммарная вероятность. В нашем примере речь идет о символах A_5 и A_6 , суммарная вероятность которых равна 0,14. Далее объединяются следующие символы или узлы с наименьшей вероятностью, как это показано на рисунке. Этот процесс продолжается до тех пор, пока ветви кодового дерева не сойдутся к одному узлу, расположенному в вершине. После этого ветви дерева в зависимости от того, в какую сторону они расходятся от узла, обозначаются нулями или единицами (в нашем примере правые ветви обозначены нулями, а левые единицами). Для того чтобы найти значение кодового слова, которое следует приписать каждому символу, необходимо идти от вершины кодового дерева к данному символу, записывая нули или единицы, которыми обозначены пройденные ветви.

В случае применения кода Хаффмена для сжатия изображений необходимо вначале осуществить декорреляцию сигнала, которым представлено изображение, а уже затем применять кодирование по Хаффмену.

Кроме рассмотренного алгоритма сжатия данных, при записи изображений часто используют (например, в формате TIFF) так называемый *модифицированный алгоритм Хаффмена*, при котором применяется заранее составленная кодовая таблица. Применение метода Хаффмена для сжатия цветных изображений очевидно и пояснений не требует.

В настоящее время метод кодирования Хаффмена является компонентом целого ряда алгоритмов архивации, например, при записи изображений в формате JPEG, MPEG-2 и др.

12.7. Арифметическое кодирование

Арифметическое кодирование является методом сжатия без потерь и предназначено для кодирования последовательности статистически независимых символов [13, 33]. Степень сжатия последовательности статистически независимых символов при его использовании максимальна и приближается к теоретическому пределу. При этом количество двоичных единиц кода на символ, представляющее сжатую последовательность, стремится к величине

$$H = -\sum_{i=n}^{m} p_n \log_2 p_n , \qquad (12.6)$$

где m — количество символов в кодируемой последовательности, а p_n — вероятность появления n-го символа. Метод арифметического кодирования выгодно отличается от метода кодирования Хаффмена тем, что при его использовании обеспечивается высокая степень сжатия при любых значениях вероятности символов, а не только при значениях, кратных $1/2^i$, где i — целое число. Объясняется это тем, что при арифметическом кодировании кодируется сразу протяженная последовательность символов и код (последовательность нулей и единиц) приписывается всей кодируемой последовательности, поэтому в случае арифметического кодирования на один символ последовательности может приходиться меньше одной двоичной единицы, важно лишь то, чтобы на всю кодируемую последовательность приходилось целое число двоичных единиц. В случае же использования метода кодирования Хаффмена кодирование происходит посимвольно, т. е. каждому символу кодируемой последовательности назначается свой код, поэтому одному символу невозможно приписать код, протяженность которого была бы меньше единицы.

Поскольку арифметический метод кодирования предназначен для сжатия некоррелированных последовательностей символов, то при его использовании для сжатия черно-белых полутоновых изображений, последовательность отсчетов, представляющих значения яркости пикселов оцифрованного изображения L(n), вначале должна быть декоррелирована. В простейшем случае последовательность отсчетов L(n) заменяется последовательностью их разностей, т. е. приращений яркости $\Delta L(n)$

$$\Delta L(n) = L(n+1) - L(n),$$

где *п* — номер отсчета.

Кодирование начинается с того, что сначала путем сканирования декоррелированного изображения определяются вероятности появления каждого из символов, представляющих собой значения разности отсчетов ΔL , которые заносятся в таблицу. В качестве простого примера далее приводится табл. 12.1 для случая, когда количество символов ΔL равно восьми. Эта таблица должна быть известна как кодеру, так и декодеру. Затем каждому символу ΔL в диапазоне от 0 до 1 назначаются верхняя h и нижняя l границы интервала, который для него отводится. Протяженность этого интервала d = h - l равна вероятности появления данного символа p. Порядок расположения этих интервалов безразличен и может быть, например, таким, как это показано в табл. 12.1. Далее вся последовательность символов ΔL разбивается на цепочки (короткие последовательности), состоящие из t символов каждая. Длина цепочки также должна быть известна как кодеру, так и декодеру.

Таблица	12.1
---------	------

Символы ΔL	Вероятность <i>р</i> появления символа ΔL	Интервалы <i>l</i> — h		
ΔL_1	0,30	0,00—0,30		
ΔL_2	0,20	0,30—0,50		
ΔL_3	0,15	0,50—0,65		
ΔL_4	0,10	0,65—0,75		
ΔL_5	0,10	0,75—0,85		
ΔL_6	0,05	0,85—0,90		
ΔL_7	0,05	0,90—0,95		
ΔL_8	0,05	0,95—1,00		

Рассмотрим алгоритм арифметического кодирования на примере кодирования конкретной цепочки символов, длина которой t составляет всего три символа, а именно цепочки, включающей в себя символы ΔL_2 , ΔL_6 и ΔL_4 . В соответствии с правилами арифметического кодирования эта последовательность символов представляется вещественным числом (кодом), которое заключено в диапазоне от 0 до 1. Для простоты изложения все объяснения проведем, используя десятичную систему счисления. Поскольку первым символом кодируемой цепочки символов является символ ΔL_2 , то в диапазоне 0—1 ему отводится интервал d_2 , заключенный в границах $l_2 = 0,3$ $h_2 = l_2 + p_2 = 0,5$. Код, представляющий этот символ, будет заключен внутри указанного интервала. На рис. 12.7 показан этот интервал. При кодировании следующего значения кодируемой последовательности ΔL_6 , отводимый ему интервал, в соответствии с принципом арифметического кодирования, должен быть вложен внутрь интервала предыдущего значения ΔL_2 , т. е. внутрь интервала 0,3—0,5. Но поскольку диапазон, в который должен быть вложен новый интервал, составляет уже не 0—1, как в предыдущем случае, а только 0,3—0,5, то при определении нижней и верхней границ нового интервала значения границ, взятые из таблицы, должны быть масштабированы путем умножения на вероятность появления значения ΔL_2 .

Таким образом, нижняя граница и верхняя граница нового интервала (интервала в интервале) будут равны

$$l_{26} = l_2 + p_2 l_6$$
$$h_{26} = l_2 + p_2 h_{69}$$

а их разность

$$d_{26} = h_{26} - l_{26} = p_2 (h_6 - l_6) = p_2 d_6,$$

или, подставляя значения l_2 , l_6 , h_6 и p_2 , получим $l_{26} = 0,47$, $h_{26} = 0,48$, $d_{26} = 0,01$. Добавляя третий символ кодируемой цепочки ΔL_4 , найдем нижнюю и верхнюю границы нового интервала, представляющего символ ΔL_4 , который вкладывается в предыдущий интервал, определяемый границами l_{26} и h_{26} .

Как и прежде, при определении границ нового вкладываемого интервала следует учитывать, что границы предыдущего интервала не 0—1, как это было в самом начале кодирования, а только 0,47—0,48. Поэтому при определении нижней и верхней границ нового интервала значения границ, взятые из таблицы, должны быть масштабированы путем умножения на произведение вероятностей появления символов ΔL_2 и ΔL_6 , т. е. на $p_2 p_6 = 0,01$.

При этом нижняя граница и верхняя граница нового интервала (интервала в интервале) будут равны

$$\begin{split} l_{264} &= l_2 + p_2 l_6 + p_2 p_6 l_4, \\ h_{264} &= l_2 + p_2 l_6 + p_2 p_6 h_4, \end{split}$$

а их разность

$$d_{264} = h_{264} - l_{264} = p_2 p_6 (h_4 - l_4) = p_2 p_6 d_4,$$

подставляя значения l_2 , l_4 , l_6 , h_4 , p_2 и p_6 , получим $l_{264} = 0,4765$, $h_{264} = 0,4775$, $d_{264} = 0,001$. Рисунок 12.7 иллюстрирует последовательность описанных действий.



Рис. 12.7. К пояснению метода арифметического кодирования

Заключительным этапом описываемой процедуры является определение двоичного кода, представляющего результат сжатия цепочки символов, в нашем примере ΔL_2 , ΔL_6 и ΔL_4 . Правило определения кода, представляющего сжатую последовательность символов ΔL_2 , ΔL_6 и ΔL_4 , заключается в том, что путем дописывания разрядов в двоичную дробь

$$c = 0, a_1 a_2 a_3 \dots a_i = a_1 \frac{1}{2} + a_2 \frac{1}{4} + a_3 \frac{1}{8} + \dots + a_i \frac{1}{2^i},$$

где $a_1, a_2, a_3, ..., a_i$ могут принимать значения 0 или 1, находят значение c, удовлетворяющее сразу двум неравенствам:

$$h_{\rm KOH} > c > l_{\rm KOH} , \qquad (12.7)$$

где $l_{\text{кон}}$ и $h_{\text{кон}}$ — значения нижней и верхней границ последнего вложенного интервала. В рассматриваемом примере $l_{\text{кон}} = l_{264} = 0,4765$, $h_{\text{кон}} = h_{264} = 0,4775$. Обратим внимание на то, что дописывание разрядов в двоичную дробь должно быть сразу же прекращено, как только условие (12.7) будет выполнено, что обеспечит минимум разрядов в числе c. При этом кодом сжатой последовательности будет двоичное число, стоящее после запятой, т. е. $a_1a_2a_3...a_i$. В рассматриваемом примере сжатая цепочка символов будет представлена кодом 0111101. Это следует из того, что двоичная запись дроби 0,0111101 соответствует десятичному значению дроби c = 0,4765625.

При арифметическом кодировании, чем длиннее будет цепочка кодируемых символов, тем больше степень сжатия будет приближаться к теоретическому пределу, определяемому формулой (12.6). Декодирование кода $a_1a_2a_3...a_i$ выполняется следующим образом.

Первым шагом является сравнение дроби c с интервалами, таблица которых имеется у декодера, в нашем примере с табл. 12.1. Из сравнения следует, что дробь c заключена в интервале 0,30—0,50, из чего следует, что первым символом закодированной цепочки является символ ΔL_2 .

Вторым шагом декодирования является вычитание из дроби c значения нижней границы интервала, соответствующего декодированному символу, в нашем примере, символу ΔL_2 , и делению получившегося результата на вероятность появления

этого символа. В нашем примере это будет $\frac{0,4765621-0,3}{0,2} = 0,8828105.$

Следующим шагом будет сравнение этого результата с интервалами из таблицы, в рассматриваемом случае табл. 12.1. В результате сравнения находим, что это число попадает в интервал символа ΔL_6 . Таким образом, вторым декодируемым символом оказывается символ ΔL_6 .

Далее в цикле повторяются второй и третий шаги, в результате которых находятся последующие символы цепочки. Цикл завершается после декодирования последнего символа цепочки, в нашем примере, символа ΔL_4 . Поскольку протяженность цепочки закодированных символов декодеру заранее известна, то он знает, когда следует закончить цикл.

Аналогичным образом декодируются все цепочки, на которые разбита исходная последовательность разностей отсчетов ΔL , после чего восстанавливаются сами значения отсчетов по формуле

$$L(n) = L(n-1) + \Delta L(n).$$

Применение арифметического кодирования для сжатия цветных изображений очевидно и пояснений не требует.

12.8. Некоторые детали алгоритмов арифметического кодирования

Для арифметического кодирования известен ряд алгоритмов, использование которых до последнего времени сдерживалось наличием патентов, срок действия которых в настоящее время истек или истекает. Не останавливаясь на конкретных алгоритмах, рассмотрим лишь наиболее существенные особенности, присущие арифметическому методу кодирования.

Сжатие

Сравним по величине сжатия арифметический метод кодирования и метод кодирования Хаффмена с теоретическим пределом, определяемым величиной энтропии.

При отсутствии сжатия для представления цепочки из трех символов, в рассмотренном ранее примере символов ΔL_2 , ΔL_6 и ΔL_4 , потребовалось бы $3 \times \log_2 m$ двоичных единиц кода, где m — количество уровней квантования. Поскольку в рассматриваемом примере m = 8, то это составило бы 9 двоичных единиц.

При сжатии с использованием метода Хаффмена, используя коды, приведенные в табл. 12.2, которые легко находятся путем применения способа, описанного в *разд. 12.6*, найдем, что требуемое количество двоичных единиц для представления цепочки символов ΔL_2 , ΔL_6 и ΔL_4 составляет 9, т. е. сжатие отсутствует.

Таблица 12.2

Символы ΔL	ΔL_1	ΔL_2	ΔL_3	ΔL_4	ΔL_5	ΔL_6	ΔL_7	ΔL_8
Код	01	11	001	101	100	0001	00001	00000

Объясняется это тем, что в цепочке символов, использованной в нашем примере, оказались символы, вероятность появления которых мала и, следовательно, представляющие их коды имеют сравнительно большую протяженность.

В случае арифметического кодирования, как было найдено в предыдущем разделе, для представления цепочки символов ΔL_2 , ΔL_6 и ΔL_4 используется кодовая последовательность, состоящая из 7 символов. На самом деле, при использовании арифметического кодирования сжатие будет несколько меньше, поскольку помимо кода, представляющего последовательность символов, необходимо израсходовать некоторое количество двоичных единиц на представление таблицы вероятностей появления каждого символа, а также на представление длины цепочек и длины всей последовательность символов. Однако поскольку при сжатии изображений это дополнительное количество двоичных единиц невелико по сравнению с цифровым потоком, представляющим информацию об интенсивности пикселов, то это мало сказывается на общем снижении степени сжатия.

Количество двоичных единиц кода, необходимое для представления цепочки символов ΔL_2 , ΔL_6 и ΔL_4 , в идеальном случае, может быть рассчитано по формуле

$$N = 3 \times H = 3 \times \left(-\sum_{i=n}^{m} p_n \log_2 p_n \right).$$

Подставляя в формулу m = 3 и беря значения вероятностей p_n из табл. 12.1, получим, что для представления цепочки символов, использованной в примере, необходимо затратить около 6,3 двоичных единиц. Из сопоставления полученных результатов расчетов находим, что количество двоичных единиц кода, необходимых для представления цепочки символов ΔL_2 , ΔL_6 и ΔL_4 , в случае арифметического кодирования, действительно, близко к теоретическому пределу.

Протяженность цепочек кодируемых символов

В примере, который рассматривался в предыдущем разделе, протяженность кодируемой цепочки символов для простоты изложения была взята равной трем. Однако на практике используют неизмеримо более протяженные цепочки. Объясняется это тем, что чем более протяженные цепочки кодируются, тем ближе к теоретическому пределу оказывается получаемое при этом сжатие для стационарных процессов. При сжатии протяженных цепочек символов возникает проблема точности представления нижней и верхней границ интервала, который сокращается по мере увеличения протяженности цепочки вследствие сближения границ. В настоящее время найдено решение этой проблемы. В этом решении для представления границ используют натуральные дроби, где числитель и знаменатель являются целыми числами, а также применяют специально разработанные алгоритмы, позволяющие избежать явления переполнения разрядной сетки по мере дописывания разрядов в числителе и знаменателе дроби [13].

Адаптивное арифметическое кодирование

При рассмотрении арифметического кодирования было принято, что декодеру известны протяженности кодируемых цепочек всей последовательности символов, а также таблицы вероятностей. Это может быть достигнуто несколькими способами. Помимо непосредственной записи в файл этой информации используют также следующие способы.

В таблицу символов вводят символ END-OF-DATA, который сообщает декодеру об окончании цепочки.

Таблица вероятностей появления каждого из символов не записывается в файл при сжатии изображения для передачи декодеру, а все время обновляется в процессе упаковки и распаковки очередного символа. При использовании данного метода отпадает необходимость в записи в файл сжатого сообщения этой таблицы, поскольку она создается и все время обновляется при декодировании. Здесь важно лишь, чтобы изменения таблицы вероятностей происходили синхронно и в кодере, и в декодере, что нетрудно сделать.

Вариантом этого способа является метод, который называется *методом скользящего окна* [87, 88]. При использовании этого метода вероятность появления очередного символа определяется путем анализа содержимого окна, которое содержит Nпоследних символов кодируемой последовательности. После кодирования очередного символа (вложения очередного интервала в предыдущий) содержимое окна сдвигается на одну позицию, при этом новый символ заносится в окно, а самый ранний символ удаляется. Достоинством этого метода является то, что он позволяет кодеру адаптироваться к статистике потока символов на его входе, которая применительно к реальным изображениям является непостоянной. Такой подход к проблеме по существу приближает рассматриваемый метод кодирования к так называемому контекстному кодированию. Как правило, адаптивный метод арифметического кодирования обеспечивает большую степень сжатия, чем неадаптивный.

В заключение этого раздела отметим, что в настоящее время арифметическое кодирование нашло свое применение в архиваторе изображений JPEG-2000 и в архиваторе RAR.

12.9. Разделение кодируемого сигнала изображения на контексты

Как известно, при сжатии изображений посредством энтропийного кодирования характеристики кодера настраиваются на статистические характеристики кодируемого изображения. Однако, как правило, изображения не однородны и различные их области имеют разные статистические характеристики, точнее различные статистические свойства. Поэтому для достижения максимальной степени сжатия следовало бы кодер по-разному настраивать при кодировании этих областей. Другим вариантом решения этой проблемы является автоматическое разделение сигнала от изображения на разные цифровые потоки, представляющие эти области, с последующим кодированием этих сигналов кодерами, настроенными на их статистические характеристики.



Рис. 12.8. Изображение, "наклеенное" на паспарту

Поясним это следующими примерами. С этой целью обратимся к рис. 12.8, на котором показано изображение, "наклеенное" на паспарту. Из рисунка видно, что часть этого изображения содержит много мелких деталей, в то время как другая

часть (паспарту) представлена однородным серым фоном. При сжатии такого изображения методом арифметического кодирования кодер настраивается на среднюю статистику кодируемого изображения и, как результат, работает не оптимально, при этом сжатие составляет 2,16 раза, хотя вследствие наличия в кодируемом изображении больших однородных полей оно могло бы быть большим. Ситуация существенно улучшается, если это изображение разделить на два компонента или, как принято говорить, на два контекста: на однородную и детальную части, а затем сжимать их по отдельности. В данном случае дополнительно к информации детального компонента необходимо передать информацию о границах, разделяющих однородную и детальную части, а также значение интенсивности серого фона паспарту. Проведенный нами эксперимент показал, что при таком подходе величина сжатия увеличивается по сравнению со сжатием без разделения на контексты в 1,89 раза и становится равной 4,07.

Рассмотрим другой пример. Предположим, что передаче подлежит двухградационное изображение, показанное на рис. 12.9. Простейшим способом передачи этого изображения была бы передача значений яркости каждого пиксела путем их перебора пиксел за пикселом и строка за строкой.



Рис. 12.9. Исходное изображение

Однако для этого потребовалось бы передать $N_{\rm crp} \times N_{\rm пикс}$ значений. Обычно количество строк и количество пикселов в строке для реальных изображений измеряются сотнями, поэтому $N_{\rm crp} \times N_{\rm пикс}$ представляет собой достаточно большую величину. Вместе с тем цифровой поток можно существенно сократить, если передавать, например, значения координат начал и концов отрезков линий, рассматривая при этом точки как отрезки линий нулевой длины. Не трудно видеть, что такое представление изображения устраняет взаимную корреляцию между передаваемыми значениями сигнала. Если принять для примера, что $N_{\rm crp} = 1000$ и $N_{\rm пикс} = 1000$, то на передачу каждой координаты потребуется 10 двоичных единиц и общее число двоичных единиц составит $18 \times 2 \times 20 = 720$ вместо $N_{\rm crp} \times N_{\rm пикс} = 1000000$, т. е. сжатие составит 1389 раз. Приведенный пример демонстрирует эффективность декорреляции изображения перед его кодированием.



Рис. 12.10. Изображение, разделенное на два контекста: *а* — контекст, состоящий из прямых линий; *б* — контекст, состоящий из точек

Эффективность передачи можно еще увеличить, если изображение, приведенное на рис. 12.9, разделить на два компонента (два контекста), как это показано на рис. 12.10, *а* и *б*. При таком разделении появляется возможность для компонента (или как принято говорить контекста), показанного на рис. 12.10, *б*, ограничиться однократной передачей координат точек, в то время как при передаче отрезков линий необходимо по-прежнему сообщать координаты их начала и конца. Другими словами, становится возможным использовать признак, характерный для точек, заключающийся в том, что они, в отличие от линий, не имеют длины.

В этом случае, как не трудно видеть, получается дополнительный выигрыш. Действительно, для передачи контекста (a) необходимо затратить 160 двоичных единиц и для передачи контекста (δ) затратить 280 двоичных единиц, т. е. всего 440 двоичных единиц. Таким образом, в нашем примере уже такое простое разделение на контексты дает дополнительный выигрыш в сжатии в 1,63 раза.

В качестве еще одного примера укажем на возможность разделения сжимаемых изображений с сильно выраженной нестационарностью статистических характеристик на контексты при использовании кодов Хафмена. Простым примером, демонстрирующим эту ситуацию, является передача изображения журнальной обложки, на которой имеются крупные надписи и ряд разноплановых фотографий: портретов, пейзажей и т. д. Разделение на контексты передаваемого сообщения в этом случае дает возможность для разных контекстов использовать согласованные с ними разные кодовые таблицы и, следовательно, благодаря этому увеличить сжатие [13, 15].

На контексты может разделяться не только само изображение, но также его спектральные или вейвлет-компоненты, т. е. те компоненты, которые подлежат кодированию.

С позиций контекстного разделения изображений, перед их сжатием могут рассматриваться ряд широко применяемых при обработке изображений преобразований, например вейвлет-преобразование, на базе которого разработан архиватор JPEG 2000 [13, 16]. В этом архиваторе исходное изображение вначале рекурсивно подвергается трем вейвлет-преобразованиям, а затем сжатию полученных в результате этих преобразований компонентов (контекстов). Получающиеся при этом компоненты имеют различные статистические характеристики. Так аппроксимация (см. разд. 13.6) имеет несимметричное распределение плотности вероятности величин коэффициентов, все коэффициенты этого контекста положительны. Распределения же других контекстов являются симметричными и хорошо аппроксимируются законом Лапласа. Кроме того, дисперсии вейвлеткоэффициентов других контекстов в среднем убывают по мере перехода к более высокочастотным компонентам. Используя для сжатия каждого из полученных контекстов свой энтропийный кодер, настроенный на статистические характеристики кодируемого контекста, получают большую величину сжатия всего изображения в целом по сравнению со случаем, если бы сжатие изображения осуществлялось одним кодером для всех контекстов без его перестройки. Описанный принцип положен в основу режима сжатия без потерь, реализуемого в JPEG 2000. Величина сжатия без потерь, обеспечиваемая в этом случае, составляет около 4,8 раз при сжатии тестового изображения "Вагbara" [145].

Сжатие изображений без потерь путем их предварительного разделения на контексты в результате вейвлет-преобразования с последующим применением энтропийных кодеров, как это следует из изложенного, может найти применение для записи исходного материала, получаемого при видеосъемке.

Метод разделения на контексты при сжатии является прогрессивным методом и в настоящее время он используется в различных алгоритмах сжатия, например, в MPEG-4. При использовании этого метода необходимо помнить, что увеличение числа контекстов ведет к увеличению сжатия лишь до тех пор, пока количество символов в каждом контексте достаточно велико [13, 15].

12.10. Проблема накопления ошибок преобразования при сжатии изображений

Несмотря на то, что в настоящее время разработаны весьма эффективные методы компрессии изображений, реализованные в алгоритмах JPEG, JPEG 2000 и др., так называемые методы кодирования с потерями, обеспечивающие степени сжатия в десятки раз [13, 50], в ряде областей техники, таких как издательское дело, цифровая съемка видеофильмов и других, их применение оказалось невозможным. Объясняется это тем, что при использовании таких методов в изображения вносятся искажения, так называемый шум преобразования, которые добавляются в них при каждом новом цикле сжатия-восстановления. Поскольку в процессе подготовки репродукций к печати, а также при редактировании телевизионных программ сжатые изображения приходится неоднократно распаковывать из файла и снова записывать в файл со сжатием, результирующая помеха преобразования накапливается, и качество изображения становится недопустимо низким. Так, например, при четырех циклах сжатия/восстановления изображения, приведенного на рис. 12.11, а, посредством алгоритма JPEG в режиме высокого качества, когда сжатие составляет всего лишь 1,55 раза, отношение "сигнал/помеха" преобразования принимает следующие значения: после первого цикла — 136,33, после второго — 72,77, после третьего — 50,11 и после четвертого цикла — 38,48.



Рис. 12.11. Тестовые изображения: *а* — исходное; *б* — изображение, из которого устранены мелкие детали, размером до 7 пикселов включительно (пояснения в тексте)

В режиме сжатия с более низким качеством помеха преобразования оказывается еще более высокой. Поэтому при архивации изображений в издательском деле, при съемках цифровых видеофильмов, нелинейном монтаже телевизионных программ их либо вообще не сжимают, либо используют для этой цели энтропийное кодирование (так называемые методы сжатия без потерь), например, кодирование длин серий, код Хаффмена, арифметический метод и ряд других [13, 50]. Особенностью, присущей этим методам, как мы видели, является то, что изображение, восстановленное из сжатого файла, в точности совпадает с исходным изображением, которое было записано в этот файл. Благодаря этой особенности энтропийного кодирования при многократной перезаписи изображений с применением сжатия, к которой приходится прибегать в процессе редактирования материала, не происходит накопления ошибок в отличие от методов сжатия с потерями. Однако энтропийное кодирование при архивации реальных фотографических изображений обеспечивает существенно меньшую степень сжатия по сравнению с методами, в которых имеют место потери. Это в полной мере также относится к новому алгоритму JPEG 2000, в котором помимо режима сжатия с потерями предусмотрен режим сжатия без потерь. Малая степень сжатия обусловлена тем, что при кодировании полутоновых фотографических изображений вероятность новых значений сигнала, т. е. вероятность скачков сигнала, оказывается достаточно большой благодаря присутствию на них множества малых деталей: точек, штрихов и т. п. Эти детали, наличие которых обусловлено, в частности, присутствием шума, не несут семантической информации, но требуют для своего представления дополнительных двоичных единиц кода.

В ряде работ, относящихся к 50—70-м годам минувшего столетия, была выдвинута идея увеличения степени сжатия изображений за счет уменьшения вероятности новых значений сигнала перед его сжатием путем загрубления шкалы квантования по яркости вблизи резких световых переходов. В основе этой идеи лежит использование известного свойства зрения, заключающегося в том, что вблизи резких световых переходов его контрастная чувствительность существенно занижена, что и позволяет эти участки изображения квантовать на меньшее число уровней без опасения возникновения искажений, которые могут быть заметны зрителям [11].

В качестве другого примера можно привести исследования, направленные на уменьшение вероятности новых значений сигнала, обусловленных наличием небольшого шума на изображениях, и влияния конечных размеров анализирующей апертуры вблизи резких световых переходов [32, 90]. К сожалению, при работе с реальными полутоновыми изображениями, проквантованными на 256 уровней, исходная вероятность новых значений близка к единице и этот метод оказался неэффективным.

К современным средствам решения данной проблемы — обеспечения максимальных степеней сжатия без внесения в сжимаемые файлы изображений ненакапливаемых ошибок — следует отнести методы так называемого препроцессинга, т. е. предварительной обработки, предшествующей использованию известных методов энтропийного кодирования. Эти методы позволяют вдобавок к основному сжатию, реализуемому энтропийными кодерами, получать дополнительный выигрыш за счет такой обработки, в результате которой эффективность применения энтропийных кодеров возрастает. Как правило, такой выигрыш невелик, обычно он составляет 1—15% от основного выигрыша, но и он является важным, учитывая актуальность проблемы. Ряд примеров методов препроцессинга описан в [13].

Эффективным подходом при разработке методов препроцессинга при сжатии изображений может служить учет свойств зрения. В этом случае допустимым можно считать внесение потерь при предварительной обработке, но таких потерь, которые, во-первых, незаметны для зрителя и, во-вторых, что очень важно, не накапливаются при каждом новом цикле сжатия/восстановления. С этих позиций был разработан метод логической фильтрации изображений, в результате которой из изображения удаляются невидимые зрителю детали.

12.11. Предварительная логическая фильтрация изображений для увеличения степени их сжатия кодерами без потерь информации

Как известно, пороговый контраст зрения в сильной степени зависит от угловых размеров наблюдаемых объектов [95]. Это свойство зрительной системы использовано в описываемом далее методе предварительной обработки изображений, заключающейся в их логической фильтрации, когда из изображений устраняются детали, которые в силу ограниченности контрастной чувствительности зрения не видны [55]. В результате этой фильтрации вероятность новых значений в сигнале обработанных таким образом изображений уменьшается и, как следствие, увеличивается степень компрессии при использовании энтропийного кодирования. Проведенные экспериментальные исследования показали, что такая предварительная обработка изображений не приводит к снижению их качества. Основываясь на этом, разработан приводимый далее алгоритм логической фильтрации изображений, заключающийся в следующем. Вначале из обрабатываемого изображения исключаются однопиксельные детали путем замены их яркости, которую обозначим через L_1 , на яркость одного из примыкающих к ним пикселов L_0 , яркость которого наиболее близка к яркости исключаемой детали. При этом вычисляется абсолютное значение разности $L_1 - L_0$ и сравнивается с заранее установленным порогом T_1 , при превышении которого операция устранения отменяется. Величина порога T₁ выбирается таким образом, чтобы удаление однопиксельных деталей было бы для зрителя незаметным. Затем аналогичным образом из изображения исключаются двухпиксельные детали с подбором необходимой величины порога T₂, который, вообще говоря, отличается от порога T_1 . Продолжая эту операцию, из изображения удаляются детали, которые зритель не может видеть в силу их малого контраста. При такой предварительной обработке изображения вероятность новых значений в сигнале, которым оно представляется, уменьшается, что позволяет увеличить степень его сжатия посредством энтропийного кодирования. Критерием допустимости такого устранения малоконтрастных деталей является неразличимость для зрителя вносимых в изображение изменений.

12.12. Экспериментальное исследование метода предварительной обработки изображений

В целях иллюстрации возможностей данного направления приведем результаты экспериментальных исследований, в ходе которых был использован упрощенный вариант описанного выше алгоритма логической фильтрации изображений, включающий в себя следующие шаги.

- □ Устранение из исходного тестового изображения деталей размером в один пиксел, яркость которых отличалась от яркости любого из окружающих их пикселов не более чем на заданную в режиме диалога величину порога *T*, с записью результата в файл.
- □ Устранение из обработанного на предыдущем шаге изображения деталей, размером в два пиксела, яркость которых отличалась от яркости любого из окружающих их пикселов не более чем на заданную в режиме диалога величину порога *T*, с записью результата в файл.
- □ ...
- □ Устранение из обработанного на предыдущем шаге изображения деталей, размером в *m* пикселов, яркость которых отличалась от яркости любого из окружающих их пикселов не более чем на заданную в режиме диалога величину порога *T* с записью результата в файл.

рога T, с записью результата в файл.

□ Энтропийное кодирование обработанного изображения.

При обработке изображений в соответствии с приведенным алгоритмом каждый шаг последовательно выполнялся до тех пор, пока в обрабатываемом изображении не оставалось деталей удаляемого размера. Объясняется это тем, что удаление деталей заключается фактически в изменении их яркости, что создает условия для удаления других деталей такого же размера, которые на предыдущем пороге не были удалены, поскольку разница их яркости и яркости любого из окружающих пикселов превышала порог T.

В целях упрощения эксперимента величина порога T во всех случаях выбиралась одной и той же, а для сжатия изображения методом энтропийного кодирования использовался арифметический метод путем применения архиватора RAR.

При выполнении эксперимента в качестве тестовых изображений были взяты часто используемые при проведении исследований черно-белые полутоновые изображения Yacht и Lena.

Величина дополнительного сжатия, которое обеспечивалось вследствие описанной предварительной обработки изображений, вычислялась по формуле

$$k_{\rm cm} = \frac{N_{\Im}}{N_n} \,,$$

где N_{\Im} — число двоичных единиц в файле сжатого исходного тестового изображения, N_n — число двоичных единиц в файле сжатого предварительно обработанного изображения, из которого были устранены детали размером до m пикселов при заданном пороге T.

Среднеквадратичное значение различия между обработанным и исходным тестовым изображениями (среднеквадратичное значение помехи преобразования) вычислялось по формуле

$$\sigma = \frac{1}{255} \sqrt{\frac{\sum \sum \left[L(k,n) - L_{\text{obp}}(k,n) \right]^2}{K \times N}},$$

где L(k,n) — распределение яркости в исходном тестовом изображении, $L_{oбp}(k,n)$ — распределение яркости в обработанном изображении, k и n — номера строк и столбцов, K и N — полное число строк и столбцов в матрице, которой представлено изображение, суммирование выполняется по всему изображению.

В результате проведенных исследований было установлено следующее.

□ Зависимость величины дополнительного сжатия k_{сж}, обеспечиваемого описанной предварительной обработкой изображений, от размера устраняемых деталей *m*, как это следует из рис. 12.12, носит насыщающийся характер. Такой ход зависимости обусловлен тем, что вероятность новых значений сжимаемого сигнала *p*_{нов} при увеличении *m* вначале уменьшается довольно быстро, а затем это уменьшение замедляется, что также видно из рисунка.



Рис. 12.12. Зависимость вероятностей новых значений p_{нов} (кривые 1 — 6) и величин дополнительного сжатия k_{сж}, получаемых в результате удаления малопиксельных деталей (кривые 7 — 12) от размера удаляемых деталей m при различных значениях порога T. Кривые 1 — 6 и 7 — 12 соответствуют значениям порога 1, 2, 4, 8, 16 и 255 уровней квантования


Рис. 12.13. Зависимости вероятностей новых значений *p*_{нов} (кривые 1 — 7) и величин дополнительного сжатия *k*_{сж} (кривые 8 — 14), полученных в результате удаления из изображения малопиксельных деталей от величины порога *T* при различном размере удаляемых деталей *m*. Кривые 1 — 7 и 8 — 14 соответствуют размерам удаленных деталей 1, 2, 3, 4, 5, 6, и 7 пикселов



Рис. 12.14. Зависимость среднеквадратичной ошибки, возникающей в результате удаления из изображения малопиксельных деталей, нормированной на динамический диапазон яркостей изображения, от величины порога *T* для семи значений размеров удаляемых деталей. Кривые 1 — 7 соответствуют удаляемым деталям, размер которых равен 1, 2, 3, 4, 5, 6 и 7 пикселов соответственно

- □ Зависимость величины дополнительного сжатия k_{cx} от величины порога T, приведенная на рис. 12.13, также имеет насыщающийся характер и по тем же причинам. Из приведенного графика видно, что увеличение порога с T = 16 до T = 255 приводит лишь к незначительному увеличению дополнительного сжатия k_{cx} . Вместе с тем такое увеличение порога приводит к появлению на изображении характерных искажений, проявляющихся в том, что на границах больших яркостных перепадов появляются зазубрины, которые при меньшем значении порога незаметны.
- □ Зависимость среднеквадратичного значения помехи преобразования от величины порога, как это видно из рис. 12.14, также носит насыщающийся характер.

Однако это насыщение наступает при сравнительно больших значениях σ , имеющих место при значениях порога в диапазоне от 61 до 255, когда устранение малых деталей уже заметно и ухудшает изображение. При порогах T = 8—16 зависимость эта хотя и сильно выражена, однако сама помеха преобразования при этом невелика. Визуальное исследование показало, что при T = 8 искажения, обусловленные устранением малых деталей, невозможно обнаружить даже путем детального сравнения обработанного и исходного тестового изображений, что объясняется ограничениями со стороны контрастной чувствительности зрения. В случае порога T = 16 различия между обработанным и исходным тестовым изображением могут быть выявлены только при их детальном сравнивании.

На рис. 12.11, *а* и *б* в качестве примера приведено сюжетное полутоновое изображение Yacht до и после устранения из него деталей размером до 7 пикселов включительно при пороге равном 16 уровням квантования. В первом случае применение архиватора RAR обеспечивало компрессию в 1,37 раза, во втором — в 2,10 раза. Для зрителя оба изображения и на экране монитора, и их отпечатанные версии были не различимы. Подчеркнем, что достигнутый результат, несмотря на небольшую величину полученного дополнительного сжатия, является существенным и важным, поскольку заметно увеличивает выигрыш от использования традиционно применяемых энтропийных методов кодирования, в том числе JPEG 2000 в режиме сжатия без потерь. Аналогичные результаты были получены при обработке других изображений.

Лучшие результаты в отношении достижимой величины дополнительного сжатия могут быть получены, если пороги, используемые при устранении деталей меньших размеров, сделать большими, чем пороги, используемые при устранении деталей больших размеров, что очевидно и пояснений не требует.

Отметим также, что эксперименты по опознаванию тестовых объектов на изображениях в пороговых условиях наблюдения показали, что описанная выше обработка изображений практически не влияла на величину коэффициента эффективности используемого в психофизике зрения для оценки функционирования зрительной системы. В заключение следует отметить, что описанный способ предварительной обработки изображений, заключающийся в их логической фильтрации, может быть распространен и на цветные изображения.

Метод может быть использован и для сжатия цифровых телевизионных видеофильмов при их съемке и предварительной записи. Как известно, значительная часть мелких деталей носит случайный характер и от кадра к кадру не повторяется, т. е. время их присутствия на экране равно длительности кадра. Поскольку пороговый контраст зрительной системы в этой области длительностей велик, то из изображений без ущерба для их качества могут быть устранены детали, имеющие сравнительно высокий контраст, что увеличит степень сжатия.

Несмотря на то, что рассмотренный метод увеличения сжатия при архивации изображений в конечном итоге, по принятой терминологии, приводит к потерям, поскольку в изображения вносятся искажения при их предварительной обработке, эти искажения зрителю незаметны и не накапливаются при повторных циклах сжатия/восстановления. Эта особенность, отличающая описанный метод от известных методов сжатия с потерями, является весьма существенной.

12.13. Кодирование битовых плоскостей

Кодирование битовых плоскостей используют как составную часть алгоритмов сжатия изображений. Примером может служить алгоритм JPEG 2000 и некоторые другие.

Как известно, оцифрованное полутоновое изображение можно представить в виде полинома с основанием 2, коэффициенты (разряды) которого $a_1(k,n)$, $a_2(k,n)$, ..., $a_{n_0}(k,n)$ в зависимости от значения яркости пиксела $L_c(k,n)$, занимающего *k*-ю строку и *n*-й столбец в изображении, могут принимать значения 1 или 0:

$$L_{\rm c}(k,n) = \frac{L_{\rm cMAKC}}{m-1} \Big[a_{n_0}(k,n) 2^{n_0-1} + a_{n_0-1}(k,n) 2^{n_0-2} + \dots a_1(k,n) 2^0 \Big],$$

где m — количество уровней квантования, определяемое количеством разрядов (двоичных) единиц кода n_0 , $m = 2^{n_0}$, $L_{\text{смакс}}$ — яркость в белом.

При кодировании битовых плоскостей каждый разрядный коэффициент, образующий битовую плоскость, рассматривается как отдельное двухградационное (битовое) изображение и кодируется (сжимается) одним из методов сжатия без потерь. В качестве поясняющего примера на рис. 12.15 приведено исходное изображение, а на рис. 12.16 — набор битовых плоскостей (с указанием номера бита, соответствующего данной плоскости), полученных из исходного изображения.

Недостатком рассмотренного способа разделения изображения на битовые плоскости является то, что при сохранении в память битовых плоскостей даже в случае использования арифметического метода их сжатия, требуется большая память, чем в случае сохранения в память исходного изображения, сжатого этим же методом. Объясняется это тем, что небольшие изменения яркости в изображении могут приводить к изменениям битов во многих битовых плоскостях. Так, например, если пиксел со значением яркости 127 (01111111₂) изменит это значение на 128 (10000000₂), то во всех битовых плоскостях произойдет переход с 1 на 0 или с 0 на 1. Для того чтобы избежать этого, цифровой код $a_1(k,n)$, $a_2(k,n)$, ..., $a_{n_0}(k,n)$ предварительно преобразуют в код Грея $g_1(k,n)$, $g_2(k,n)$, ..., $g_{n_0}(k,n)$, а уже потом изображение разделяют на битовые плоскости. Преобразование выполняется следующим образом

$$g_{n_0} = a_{n_0},$$

 $g_i = a_i \oplus a_{i+1}, 1 \le i \le n_0 - 1$

где символ \oplus означает операцию исключающего ИЛИ.

Замечательной особенностью кода Грея является то, что при увеличении кодируемой величины (в рассматриваемом случае яркости пиксела) на единицу происходит изменение только в одной битовой плоскости. Например, переход яркости с уровня 127 на уровень 128 вызывает переход с 0 на 1 только в 7-й битовой плоскости, т. к. коды Грея для 127 и 128 соответственно равны 11000000₂ и 01000000₂. Благодаря этому небольшие изменения яркости в среднем будут приводить к изменениям в небольшом количестве битовых плоскостей.

На рис. 12.17 приведен набор битовых плоскостей, полученных из исходного изображения применительно к случаю, когда используется код Грея.



Рис. 12.15. Исходное изображение



Рис. 12.16. Битовые плоскости при двоичном коде



Рис. 12.17. Битовые плоскости при коде Грея

Сравнивая рис. 12.16 и 12.17, видим, что действительно, в битовых изображениях рис. 12.17, особенно в изображениях, формируемых старшими разрядами кода, меньше световых переходов, чем в соответствующих изображениях на рис. 12.16.

Алгоритм обратного преобразования, т. е. преобразования кода Грея в двоичный код, описывается рекуррентным выражением

$$a_{n_0} = g_{n_0},$$
 $a_i = a_{i+1} \oplus g_i$ для $i < n_0,$

причем преобразование осуществляется побитно, начиная со старших разрядов, и значение a_{i+1} , используемое в формуле, вычисляется на предыдущем шаге алгоритма. Действительно, если подставить в эту формулу вышеприведенное выражение для *i*-го бита кода Грея, получим

$$a_i = a_{i+1} \oplus g_i = a_{i+1} \oplus (a_i \oplus a_{i+1}) = a_i \oplus (a_{i+1} \oplus a_{i+1}) = a_i \oplus 0 = a_i.$$

Глава 13



Сжатие изображений с потерями информации

13.1. Дифференциальная кодово-импульсная модуляция

В настоящее время дифференциальная кодово-импульсная модуляция (ДКИМ), которая была предложена Катлером в 1952 г., не находит самостоятельного применения. Однако принципы, заложенные в ДКИМ, часто применяют в составе различных методов сжатия изображений как один из элементов их алгоритмов. В связи с этим рассмотрим принцип действия ДКИМ на примере кодирования полутонового черно-белого изображения [23, 50, 73]. Существо ДКИМ заключается в том, что на основе известных значений сигнала в предшествующие моменты времени t_{n-1} , t_{n-2} и т. д. делается статистическая оценка (предсказание) значения сигнала $\hat{U}_{\hat{c}}(t_n)$ в момент времени t_n . После этого предсказанное значение сигнала сравнивается с тем значением $U_c(t_n)$, которое имеется на самом деле, и находится сигнал ошибки предсказания

$$v(t_n) = U_{\hat{c}}(t_n) - \hat{U}_{\hat{c}}(t_n).$$

Сигнал ошибки предсказания квантуется, кодируется и передается в место приема или записывается, например, на магнитный диск. Восстановление исходного сигнала осуществляется путем его предсказания с применением такого же алгоритма, что и алгоритм, применявшийся при кодировании, с последующим прибавлением к предсказанному значению сигнала ошибки предсказания, т. е.

$$U_{\hat{c}}(t_n) = \hat{U}_{\hat{c}}(t_n) + v(t_n)$$

Таким образом, при использовании ДКИМ передаче (или записи) подлежит только та составляющая сигнала, которую невозможно предсказать (сигнал ошибки предсказания) и которая, следовательно, несет информацию. Обратим внимание на то, что восстановленное значение сигнала отличается от исходного на величину шума квантования сигнала ошибки предсказания, что неизбежно в цифровых системах, однако, выбирая шкалу квантования с достаточно большим числом уровней, это отличие можно сделать очень малым. Как известно, сигнал, представляющий изображение, может быть промоделирован двумерным стационарным марковским процессом 1-го порядка, наложенным на некоторое среднее значение, определяемое средней яркостью изображения $\overline{L_c}$. Известно также, что в этом случае минимальное значение среднего квадрата сигнала ошибки предсказания получается, если статистическую оценку ожидаемого значения сигнала производить следующим образом:

$$\hat{U}_{\hat{c}\sim}(t_n) = a_1 U_{\hat{c}\sim}(t_{n-1}),$$

где $U_{c\sim}(t_{n-1})$ — переменная составляющая сигнала изображения в предшествующий момент времени t_{n-1} , a_1 — весовой коэффициент. Для того чтобы при сжатии данных посредством ДКИМ вносимые в сигнал искажения, которые обусловлены процессом квантования, были минимальны, необходимо использовать такое значение весового коэффициента a_1 , при котором средний квадрат сигнала ошибки предсказания $\overline{v^2(t_n)}$ был бы также минимальным. Это следует из того хорошо известного обстоятельства, что средний квадрат шума квантования прямо пропорционален среднему квадрату квантуемого сигнала (*см. разд. 4.7*). Чтобы найти это значение весового коэффициента, запишем вначале выражение для $\overline{v^2(t_n)}$:

$$\overline{v^{2}(t_{n})} = \overline{U_{c^{\sim}}^{2}(t_{n})} - 2a_{1}\overline{U_{c^{\sim}}(t_{n})}U_{c^{\sim}}(t_{n-1}) + a_{1}^{2}\overline{U_{c^{\sim}}^{2}(t_{n-1})}, \qquad (13.1)$$

а затем продифференцируем его по a_1 и приравняем результат нулю. Решая получившееся уравнение относительно a_1 , получим:

$$a_{1} = \overline{U_{c}(t_{n})U_{c}(t_{n-1})} / \overline{U_{c}^{2}(t_{n-1})}.$$
(13.2)

Замечая, что

$$\overline{U_{c\sim}(t_n)U_{c\sim}(t_{n-1})}/\overline{U_{c\sim}^2(t_{n-1})} = \rho_c(\tau_{\mathfrak{I}}),$$

где $\rho_c(\tau_3)$ — значение коэффициента автокорреляции переменной составляющей сигнала изображения при смещении его реализаций на интервал времени, равный интервалу между двумя смежными отсчетами τ_3 , находим, что оптимальным значением весового коэффициента, при котором обеспечивается минимальный уровень шума квантования, будет

$$a_1 = \rho_c\left(\tau_{\scriptscriptstyle 9}\right). \tag{13.3}$$

Обращаясь к формуле (13.1) и заменяя в ней весовой коэффициент *a*₁ найденным значением, получим после несложных преобразований

$$\overline{v^2} = \overline{U_{c^{\sim}}^2} \left[1 - \rho_c^2(\tau_3) \right].$$
(13.4)

В этой формуле опущены скобки с аргументами t_n после значений $\overline{v^2}$ и $\overline{U_{c\sim}^2}$, поскольку сигнал считается стационарным. Из формулы (13.3) видно, что чем ближе значение коэффициента автокорреляции $\rho_c(\tau_3)$ к единице, тем ближе к единице будет значение весового коэффициента a_1 . В некоторых случаях, например, в случае портретных изображений, значение $\rho_c(\tau_3)$ мало отличается от единицы. Это позволяет в качестве предсказываемого значения сигнала $U_c^{f}(t_n)$ использовать то значение, которое он имел на предыдущем отсчете изображения, другими словами, использовать значение $U_c(t_{n-1})$. Такая замена, упрощая алгоритм, не приводит к заметному ухудшению качества декомпрессированного изображения.

Обратим внимание на то, что при формировании сигнала ошибки предсказания имеет место процесс декорреляции, т. е. сигналы $v(t_n)$ и $v(t_{n-1})$ оказываются не-коррелированными, т. е.

$$\overline{v(t_n)v(t_{n-1})}=0.$$

Другим важным обстоятельством является то, что средний квадрат сигнала ошибки предсказания $\overline{v^2}$ при значениях коэффициента автокорреляции $\rho_c(\tau_3)$ сжимаемого сигнала, близких к единице, много меньше среднего квадрата исходного сигнала, что следует из формулы (13.4). В данном случае это важно потому, что средний квадрат шума квантования, как уже на это обращалось внимание, прямо пропорционален среднему квадрату квантуемого сигнала и, следовательно, при квантовании сигнала ошибки предсказания мы будем иметь средний квадрат шума квантования $\sigma_{\rm KB, дким}^2$ много меньшим, чем если бы мы квантовали сам исходный сигнал

$$\sigma_{\text{KB},\text{дким}}^2 = \overline{\nu^2} \sigma_0^2 (n_0, W).$$

Заметим, что важно не беспредельно уменьшать шум квантования при передаче или записи изображений, а только сделать его ниже некоторого порогового уровня, при котором он еще заметен на изображении и ухудшает его качество. Поэтому, заменяя исходный сигнал сигналом ошибки предсказания, мы можем уменьшить число уровней квантования и, следовательно, уменьшить число двоичных единиц кода, которым он представляется, увеличив при этом шум квантования до прежне-го порогового уровня, т. е. осуществить сжатие данных.

В отличие от рассмотренного простейшего случая предсказания сигнала изображения, который является одномерным, т. е. зависит только от одной независимой переменной — времени, изображение можно рассматривать как двумерный сигнал, поскольку яркость каждой точки изображений является функцией двух пространственных координат. В этом случае при сжатии можно осуществлять двумерное предсказание, т. е. использовать для предсказания не только сигнал от предшествующего отсчета в данной строке, но также и сигнал от отсчета в предыдущей строке, который расположен над отсчетом, сигнал от которого предсказывается. Может показаться, что такое предсказание будет существенно более точным и, следовательно, позволит еще больше уменьшить средний квадрат сигнала ошибки предсказания, однако исследования показывают, что это не так. Переход от одномерного предсказания к двумерному приводит лишь к сравнительно небольшому

уменьшению величины v^2 , именно по этой причине двумерная ДКИМ не имеет существенных преимуществ перед одномерной.

Дифференциальная кодово-импульсная модуляция является методом сжатия данных с потерей информации, поскольку восстановленное изображение отличается от исходного наличием шума квантования.

В настоящее время ДКИМ используют как компонент достаточно сложных алгоритмов, предназначенных для сжатия данных при записи или передаче движущихся изображений, самостоятельно же, как уже было отмечено, ДКИМ для этих целей практически не применяется.

13.2. Кодирование с использованием ортогональных преобразований

Особенностью данного метода сжатия изображений является то, что при этом методе кодируется не само изображение (т. е. последовательность отсчетов, которой это изображение представлено), а значения спектральных коэффициентов, получающихся при ортогональном преобразовании изображения. Эта идея кодирования впервые была выдвинута в 1968 г. В результате ортогональных преобразований изображения $L_{c}(k, n)$, имеющего сильные корреляционные связи между смежными отсчетами (пикселами), имеет место декорреляция, в результате которой значения спектральных коэффициентов F(u,v) оказываются практически некоррелированными. В отличие от исходного изображения, для которого характерно в среднем равномерное распределение энергии между его отсчетами (пикселами), распределение энергии между спектральными коэффициентами резко неравномерно. При этом основная доля энергии приходится на спектральные коэффициенты с малыми индексами и, v, представляющие амплитуды низких пространственных частот и лишь небольшая ее часть — на прочие. В целях сжатия изображений спектральные коэффициенты, имеющие малую амплитуду, либо квантуются на малое число уровней, либо вообще отбрасываются, что позволяет для их представления использовать коды с малым числом двоичных единиц. Так как средний квадрат шума квантования пропорционален среднему квадрату квантуемого сигнала, то возникающие при этом искажения изображения невелики.

При декомпрессии (восстановлении) изображения, вначале по имеющемуся коду восстанавливаются спектральные коэффициенты, а затем путем обратного ортогонального преобразования восстанавливается само изображение. Поскольку при записи или при передаче спектральных коэффициентов, в отличие от записи или передачи значений отсчетов исходного изображения, только небольшая их часть представлена кодом с большим количеством двоичных единиц, в то время как для представления остальных расходуется значительно меньше двоичных единиц, если они вообще не отбрасываются, достигается высокая степень сжатия. Поскольку, как это следует из приведенного описания метода, восстановленное изображение отличается от исходного вследствие квантования спектральных коэффициентов с большими индексами на малое число уровней, данный метод сжатия относится к группе методов сжатия с потерей информации.

Существуют два метода отбора спектральных коэффициентов: зональный и пороговый. Первый метод заключается в том, что заранее, исходя из статистики изображений, в матрице спектральных коэффициентов выделяются зоны и все спектральные коэффициенты, входящие в одну зону, квантуются на одно и то же число уровней, как это показано на рис. 13.1.

-	-								
	v	0	1	2	3	4	5	6	7
u↓	0	12	7	5	3	2	1	0	0
	1	7	7	5	3	2	1	0	0
	2	5	5	5	3	2	1	0	0
	3	3	3	3	3	2	1	0	0
	4	2	2	2	2	2	1	0	0
	5	1	1	1	1	1	1	0	0
	6	0	0	0	0	0	0	0	0
	7	0	0	0	0	0	0	0	0

Рис. 13.1. Пример распределения двоичных единиц кода между спектральными коэффициентами при зональном методе

Второй метод состоит в том, что сохраняются только те спектральные коэффициенты, амплитуда которых превышает заранее установленный порог. Этот метод отбора сложнее зонального, поскольку кроме передачи (записи) значений спектральных коэффициентов необходимо также передавать (записывать) их индексы.

Перед тем как переходить к более детальному рассмотрению метода сжатия данных, основанного на применении ортогональных преобразований, сравним его с ДКИМ. Общим для этих двух методов является двухэтапная обработка изображений, включающая в себя декорреляцию и последующее оптимальное кодирование сигнала. Важное различие между ДКИМ и методом сжатия с использованием ортогональных преобразований состоит в том, что в первом случае имеет место декорреляция за счет предсказания, при которой используется "локальная" статистика изображения, в то время как во втором случае имеет место декорреляция за счет укрупнения и, следовательно, используется "средняя" статистика изображения. При передаче стационарных изображений эта особенность не играет роли, и оба метода сжатия дают близкие результаты. Если же изображение не стационарно, как например, при передаче мелкомасштабного объекта на фоне поля с медленно изменяющейся яркостью, это различие в способе декорреляции весьма существенно. На той части изображения, где расположен мелкомасштабный объект, "текущее" значение коэффициента автокорреляции между сигналами от соседних отсчетов невелико ($\rho_c(\tau_3) = 0.5.0,7$), поэтому его сжатие посредством ДКИМ оказывается неэффективным. В то же время значение коэффициента автокорреляции $\rho_c(\tau_{3})$, усредненное по всему изображению, может быть близким к единице, благодаря чему будет обеспечиваться высокая эффективность сжатия методом, использующим ортогональные преобразования. Рассмотрим более подробно ортогональные преобразования предварительно дискретизированных изображений, представляемых в виде массива (матрицы) чисел $\|L_{c0}(k,n)\|$ размером $N_1 \times N_1$, где k — номер строки, n — номер столбца (номер отсчета в строке). Следует обратить внимание на то, что в этой записи порядок указания координат точки отсчета яркости на изображении изменен на обратный, т. е. вместо обозначения $L_{c0}(x, y)$ мы пишем $L_{c0}(k, n)$. Это делается для согласования с формой записи, принятой в матричном анализе, где первая координата обозначает номер строки, а вторая — номер столбца. Спектральные коэффициенты F(u,v) находятся путем прямого ортогонального преобразования изображения следующим образом:

$$F(u,v) = \sum_{k=0}^{N_1-1} \sum_{n=0}^{N_1-1} L_{c0}(k,n) a(k,n,u,v),$$

где a(k, n, u, v) — ядро прямого преобразования (базисные функции, по которым происходит разложение); u, v — индексы спектральных коэффициентов, определяющие их положение в матрице спектральных коэффициентов, которая имеет тот же размер, что и преобразуемое изображение. Исходное изображение (массив чисел $||L_{c0}(k, n)||$) находится путем обратного ортогонального преобразования

$$L_{c0}(k,n) = \sum_{u=0}^{N_1-1} \sum_{v=0}^{N_1-1} F(u,v) b(k,n,u,v),$$

где b(k, n, u, v) — ядро обратного преобразования. Если преобразование разделимо, т. е. если

$$a(k, n, u, v) = a_{\rm crp}(k, u)a_{\mathfrak{I}}(n, v), \ b(k, n, u, v) = b_{\rm crp}(k, u)b_{\mathfrak{I}}(n, v),$$

а нас будут интересовать разделимые преобразования, то оно может быть выполнено в два этапа, вначале по всем строкам, а затем по всем столбцам

$$F(u,v) = \sum_{n=0}^{N_1-1} a_9(n,v) \sum_{k=0}^{N_1-1} L_{c0}(k,n) a_{crp}(k,u), \qquad (13.5)$$

и соответственно

$$L_{\rm c0}(k,n) = \sum_{\nu=0}^{N_1-1} b_{\rm s}(n,\nu) \sum_{u=0}^{N_1-1} F(u,\nu) b_{\rm crp}(k,u).$$
(13.6)

Для удобства записи и вычислений используют матричный аппарат. В матричной форме разделимые ортогональные преобразования записываются следующим образом:

$$\mathbf{F} = \mathbf{a}_{\mathfrak{I}} \mathbf{L}_{\mathfrak{c}0} \, \mathbf{a}_{\mathfrak{c}\mathfrak{r}\mathfrak{p}}^{T} \,, \qquad \mathbf{L}_{\mathfrak{c}0} = \mathbf{b}_{\mathfrak{I}} \, \mathbf{F} \mathbf{b}_{\mathfrak{c}\mathfrak{r}\mathfrak{p}}^{T} \,, \qquad (13.7)$$

где \mathbf{a}_{3} , \mathbf{a}_{crp} — ортогональные матрицы прямого преобразования по столбцам и строкам, \mathbf{b}_{3} , \mathbf{b}_{crp} — ортогональные матрицы обратного преобразования по столбцам и строкам, \mathbf{a}_{crp}^{T} и \mathbf{b}_{crp}^{T} — матрицы, полученные в результате транспонирования матриц \mathbf{a}_{crp} и \mathbf{b}_{crp} , \mathbf{L}_{c0} — матрица отсчетов изображения:

$$\mathbf{L}_{c0} = \left\| \begin{array}{cccc} L_{c0}\left(1,1\right) & L_{c0}\left(1,2\right) & \dots & L_{c0}\left(1,N_{1}\right) \\ L_{c0}\left(2,1\right) & L_{c0}\left(2,2\right) & \dots & L_{c0}\left(2,N_{1}\right) \\ \dots & \dots & \dots & \dots \\ L_{c0}\left(N_{1},1\right) & L_{c0}\left(N_{1},2\right) & \dots & L_{c0}\left(N_{1},N_{1}\right) \end{array} \right|;$$

F — матрица спектральных коэффициентов, получаемая в результате двумерного ортогонального преобразования:

$$\mathbf{F} = \begin{vmatrix} F(1,1) & F(1,2) & \dots & F(1,N_1) \\ F(2,1) & F(2,2) & \dots & F(2,N_1) \\ \dots & \dots & \dots & \dots \\ F(N_1,1) & F(N_1,2) & \dots & F(N_1,N_1) \end{vmatrix}$$

Учитывая, что $\mathbf{b}_{3} = \mathbf{a}_{3}^{-1}$, $\mathbf{b}_{crp} = \mathbf{a}_{crp}^{-1}$, а также соотношения $\mathbf{a}_{3}^{-1} = \mathbf{a}_{3}^{T}$ и $\mathbf{a}_{crp}^{-1} = \mathbf{a}_{crp}^{T}$, справедливые для ортогональных матриц, имеем

$$\mathbf{L}_{c0} = \mathbf{a}_{3}^{T} \mathbf{F} \left[\mathbf{a}_{cTp}^{T} \right]^{T} = \mathbf{a}_{3}^{T} \mathbf{F} \mathbf{a}_{cTp}, \qquad (13.8)$$

где \mathbf{a}_{9}^{-1} , \mathbf{a}_{crp}^{-1} — матрицы, полученные в результате обращения матриц \mathbf{a}_{9} , \mathbf{a}_{crp} .

Базисные функции $a_3(n,v)$, $a_{crp}(k,u)$, $b_3(n,v)$, $b_{crp}(k,u)$ в формулах (13.5) и (13.6) (или, что то же самое, ортогональные матрицы в формулах (13.7) и (13.8)) определяются применяемым ортогональным преобразованием. Так, например, в случае двумерного дискретного преобразования Фурье (ДПФ) базисные функции представляют собой комплексные экспоненты, а сами ортогональные преобразования имеют вид

$$F(u,v) = \frac{1}{N_1} \sum_{n=0}^{N_1-1} \sum_{k=0}^{N_1-1} L_{c0}(k,n) \exp\left[-\frac{2\pi \mathbf{i}}{N_1}(uk+vn)\right],$$
$$L_{c0}(k,n) = \frac{1}{N_1} \sum_{u=0}^{N_1-1} \sum_{v=0}^{N_1-1} F(u,v) \exp\left[\frac{2\pi \mathbf{i}}{N_1}(uk+vn)\right],$$

в этих формулах множитель $2\pi/N_1$ имеет смысл пространственной частоты, $\mathbf{i} = \sqrt{-1}$.

Известно, что (ДПФ) не является лучшим преобразованием для применения в целях сжатия данных, т. к. значения спектральных коэффициентов в области высоких пространственных частот при этом преобразовании имеют сравнительно высокие значения. В настоящее время при сжатии изображений широкое распространение получило дискретное косинусное преобразование (ДКП). Среди других ранее применявшихся ортогональных преобразований при сжатии изображений следует назвать: *преобразование Адамара, преобразование Хаара, наклонное преобразование (slant transform)*.

Ортогональные преобразования изображений допускают ряд следующих интерпретаций.

Во-первых, двумерное преобразование изображения можно рассматривать как его разложение в обобщенный двумерный спектр, а спектральные коэффициенты — как амплитуды соответствующих спектральных составляющих. В том случае, если применяются негармонические базисные функции, как, например, в случае преобразования Адамара, понятие частоты необходимо обобщить и пользоваться понятием секвенты. Напомним, что *секвентой* (ненормированной) называется величина, равная половине среднего числа пересечения нуля в единицу времени (на единицу длины).

Другая возможная интерпретация обусловлена тем, что матрица преобразуемого изображения и матрицы базисных изображений имеют одинаковые размеры, т. е. состоят из одного и того же числа строк и столбцов. Это дает возможность спектральные коэффициенты рассматривать как весовые коэффициенты, с которыми необходимо просуммировать базисные изображения, чтобы получить исходное изображение.

Спектральные коэффициенты можно также рассматривать и как функции взаимной корреляции между преобразуемым изображением и соответствующими базисными изображениями.

И, наконец, ортогональные преобразования можно рассматривать как поворот *N*-мерной системы координат ($N = N_1 \times N_1$), в которой преобразуемое изображение, состоящее из *N* отсчетов, представляется *N*-мерным вектором, при котором корреляция между его компонентами сводится к минимуму.

Важным свойством ортогональных преобразований является сохранение метрики, благодаря этому свойству евклидово расстояние между изображениями равно евклидову расстоянию между их образами (спектральными отображениями).

13.3. Дискретное косинусное преобразование

Двумерное дискретное косинусное преобразование является разделимым и может быть выполнено по формулам:

$$F(u,v) = \frac{2}{N_1}c(u)c(v)\sum_{n=0}^{N_1-1}\sum_{k=0}^{N_1-1}L_{c0}(k,n)\cos\left[\frac{\pi}{N_1}u\left(k+\frac{1}{2}\right)\right]\cos\left[\frac{\pi}{N_1}v\left(n+\frac{1}{2}\right)\right],$$
 (13.9)

$$L_{c0}(k,n) = \frac{2}{N_1} \sum_{u=0}^{N_1-1} \sum_{v=0}^{N_1-1} c(u) c(v) F(u,v) \cos\left[\frac{\pi}{N_1} u\left(k+\frac{1}{2}\right)\right] \cos\left[\frac{\pi}{N_1} v\left(n+\frac{1}{2}\right)\right], \quad (13.10)$$

где функции c(u) и c(v)определены следующим образом: $c(0) = 1/\sqrt{2}$, c(w) = 1 при $w = 1, 2, ..., N_1 - 1$. На рис. 13.2 приведены одномерные базисные функции ДКП для случая, когда $N_1 = 8$. Как известно, вычисление двумерного дискретного косинусного преобразования по приведенным формулам требует для его выполнения N_1^4 операций умножения и сложения, что создает серьезную проблему, поскольку значения N_1 для реальных изображений составляют несколько сотен. В связи с этим были предприняты исследования, направленные на сокращение требуемого объема вычислений. В результате этих исследований были разработаны два дополняющих друг друга метода.



Рис. 13.2. Одномерные базисные функции дискретного косинусного преобразования

Первый метод заключается в том, что кодируемое изображение размером $N_{\rm crp} \times N_{\rm пикс}$ отсчетов предварительно разбивается на отдельные блоки размером $N_{6\pi} \times N_{6\pi}$ отсчетов, а затем независимо каждый из блоков подвергается дискретному косинусному преобразованию. Поскольку каждый блок содержит в $k_0 = (N_{\rm crp} \times N_{\rm пикc})/N_{6\pi}^2$ раз меньше отсчетов, чем исходное изображение, то на его преобразование в соответствии с формулами (13.9), (13.10) потребуется уже не $(N_{\rm crp} \times N_{\rm пикc})^2$ операций

(в случае, когда $N_{\rm crp} = N_{\rm пикс} = N_1$ потребовалось бы соответственно N_1^4 операций), а только $\left(N_{\rm crp} \times N_{\rm пикс}\right)^2 / k_0^2$ вычислительных операций.

Учитывая, что все изображение содержит k₀ блоков, находим количество вычислительных операций, которые необходимо выполнить, чтобы осуществить преобразование всего изображения $\left(N_{\rm crp} \times N_{\rm пикс}\right)^2 / k_0$, т. е. в k_0 раз меньше, чем без разбиения на блоки. Поясним изложенное примером. Предположим, что исходное изображение имеет размер 576×720 отсчетов, а размер блока составляет 16×16 отсчетов. Тогда в соответствии с приведенными выше рассуждениями без разбиения изображения на блоки потребовалось бы 171 992 678 400 вычислительных операций, при разбиении же изображения на блоки потребуется всего лишь 106 168 320, т. е. в 1620 раз меньше, чем в первом случае. Из этого следует, что чем более мелкими будут блоки, тем большим будет их число k_0 и тем большим будет сокращение числа операций, необходимых для выполнения ортогонального преобразования, в данном случае ДКП. Однако, как показывает детальный анализ этой проблемы, делать блоки меньшими, чем 16×16, или, в крайнем случае, 8×8 отсчетов, не следует, т. к. корреляционные связи в изображении распространяются примерно на этот интервал и дальнейшее уменьшение размеров блоков повлечет за собой увеличение амплитуд спектральных коэффициентов F(u, v) с большими индексами и, v и, как следствие, уменьшение сжатия данных.

Второй метод сокращения требуемого объема вычислений при выполнении дискретного косинусного преобразования состоит в применении быстрого алгоритма вычисления ДКП, при котором требуемый объем вычислений (умножений и сложений) сокращается с $(N_1)^4$ до $N_1^2 \log_2(N_1)$.

Поясним эффективность этого метода на примере, полагая, что размер блока составляет $N_1 \times N_1 = 16 \times 16$ отсчетов изображения. При непосредственном вычислении спектральных коэффициентов по формулам (13.9), (13.10) потребовалось бы выполнить 65 536 операций умножения и сложения. Используя же быстрый алгоритм, потребуется выполнить всего лишь 1024 операции, т. е. в 64 раза меньше, чем в первом случае.

В настоящее время при выполнении ДКП используют оба описанных метода сокращения количества вычислительных операций, поскольку они, дополняя друг друга, позволяют существенно ускорить вычисления.

13.4. Метод оптимального распределения двоичных единиц кода между спектральными коэффициентами

Как уже отмечалось, в методе, использующем ортогональные преобразования, сжатие данных достигается за счет того, что спектральные коэффициенты, энергия которых мала, квантуются на меньшее число уровней, а, следовательно, на их представление затрачивается меньшее число двоичных единиц кода, чем на представление спектральных коэффициентов с большой энергией.

Рассмотрим задачу распределения двоичных единиц кода между спектральными коэффициентами F(u,v), при котором обеспечивается наименьший средний квадрат шума преобразования $\sigma_{\Sigma_{\rm ШКВ}}^2$, обусловленного квантованием (или отбрасыванием) спектральных коэффициентов [50]. Будем считать, что сжимаемое изображение является черно-белым полутоновым, а также, что нам заданы: размер блока $N_{6\pi} \times N_{6\pi}$, требуемый коэффициент сжатия $k_{cж}$ и значения средних квадратов спектральных коэффициентов $\overline{F^2(u,v)}$.

Решение задачи начнем с того, что определим вначале количество двоичных единиц кода N_{Σ} , которыми мы располагаем при заданном коэффициенте сжатия и которые нам предстоит распределить между спектральными коэффициентами блока. Исходя из того, что для представления яркости каждого пиксела в блоке исходного черно-белого полутонового изображения требуется один байт, т. е. 8 двоичных единиц, найдем, что для представления всего блока без сжатия расходуется $8 \times N_{6\pi}^2$ двоичных единиц кода. Отсюда следует, что при заданном значении коэффициента сжатия $k_{cж}$ мы располагаем $N_{\Sigma} = \left(8 \times N_{6\pi}^2\right)/k_{cж}$ количеством двоичных единиц кода, которые и должны распределить между спектральными коэффициентами.

Как уже указывалось, средний квадрат шума квантования прямо пропорционален среднему квадрату квантуемого сигнала. Применительно к рассматриваемому случаю это значит, что средний квадрат шума квантования спектрального коэффициента F(u, v) будет равен

$$\sigma_{F_{\rm KB}}^2(u,v) = \overline{F^2(u,v)} \sigma_0^2(n_0,W), \qquad (13.11)$$

где $\sigma_0^2(n_0, W)$ — средний квадрат шума квантования сигнала на $m = 2^{n_0}$ уровней, при условии, что средний квадрат самого квантуемого сигнала равен единице, а распределение его по яркости описывается плотностью вероятности W, n_0 — число двоичных единиц кода, которым представлен спектральный коэффициент. Средний же квадрат шума преобразования (квантования) равен сумме средних квадратов шумов, возникающих при квантовании каждого спектрального коэффициента:

$$\sigma_{\text{шкв}}^2 = \sum_{u} \sum_{v} \sigma_{F_{\text{KB}}}^2 \left(u, v \right).$$
(13.12)

Для того чтобы обеспечить минимальное значение $\sigma^2_{\Sigma_{IIIKB}}$, поступим следующим образом. Вначале выделим для представления каждого спектрального коэффици-

ента нулевое количество двоичных единиц кода и найдем значения средних квадратов шумов квантования, которые при этом возникают. Поскольку в этом случае мы совершили усечение (отбрасывание) спектральных коэффициентов, значения средних квадратов шумов квантования будут равны средним квадратам самих спектральных коэффициентов, т. е. $\sigma_{F_{KB}}^2(u, v) = F^2(u, v)$, а $\sigma_{\Sigma_{IIIKB}}^2$ — их сумме. Далее, выбираем из всех значений $\sigma_{F_{KB}}^2(u, v)$ самое большое, выделяем спектральному коэффициенту, которому соответствует это самое большое значение среднего квадрата шума квантования, одну двоичную единицу кода, уменьшаем N_{Σ} на единицу и рассчитываем для него по формуле (13.11) новое значение среднего квадрата шума квантования $\sigma_{F k B}^2(u, v)$. После этого снова сравниваем между собой все значения $\sigma_{F_{KB}}^2(u, v)$, опять находим наибольшее и снова выделяем одну двоичную единицу кода наиболее "шумящему" спектральному коэффициенту, уменьшая при этом N_{Σ} на единицу. Так повторяем до тех пор, пока не будут израсходованы все двоичные единицы кода N_{Σ} . Как не трудно видеть, при таком распределении двоичных единиц мы обеспечиваем минимальный уровень шума преобразования. Заключительным этапом описанной процедуры является объединение спектральных коэффициентов, для представления которых выделено одинаковое количество двоичных единиц кода, в зоны.

Обратим внимание, что описанный метод распределения двоичных единиц кода между спектральными коэффициентами еще не гарантирует минимальной заметности шума преобразования на изображении после его декодирования. Объясняется это тем, что разные спектральные компоненты различно воспринимаются зрительной системой. Поэтому, для того чтобы достичь минимальной заметности шума преобразования на декодированном изображении, описанную процедуру необходимо выполнять, используя для этого не средние квадраты шума квантования, а их средневзвешенные значения.

Рассмотрим, как проявляется шум квантования, а также внешний шум на декодированных изображениях. Поскольку результирующий уровень шума преобразования является результатом одновременного воздействия всех шумовых компонентов, возникающих при квантовании спектральных коэффициентов, то в силу центральной предельной теоремы его распределение будет близким к нормальному. Так как средние квадраты шума квантования всех спектральных коэффициентов близки между собой вследствие примененной стратегии распределения двоичных единиц, его спектральный состав будет близок к спектральному составу квазибелого шума. Что же касается проявления на изображении внешней помехи при его передаче по каналу связи в сжатом виде, то здесь все зависит от ее характера. Например, редкая импульсная помеха проявляется в том, что отдельные блоки изображения передаются неверно, поэтому на них пропечатываются базисные изображения, соответствующие тем спектральным коэффициентам, которые были переданы с ошибкой.

13.5. Сжатие изображений в формате JPEG

В формате записи изображений JPEG использован метод сжатия с применением дискретного косинусного преобразовании, т. е. метод сжатия с потерями информации. Аббревиатура JPEG означает название организации, разработавшей этот стандарт, — Joint Photographic Experts Group (Объединенная группа экспертов по фотографии). Этот формат предусматривает сжатие как черно-белых полутоновых изображений, так и цветных. Рассмотрим случай сжатия цветных изображений как более общий. В цветном изображении каждый пиксел представлен 3-мя байтами, по байту на красный (R), зеленый (G) и синий (B) цвета.

Сжатие изображения начинается с того, что оно разбивается на отдельные блоки размером 16×16 отсчетов каждый, которые затем сжимаются независимо друг от друга.

Далее, в каждом блоке от 3-х матриц отсчетов для красного (R), зеленого (G) и синего (B) компонентов изображения, осуществляют переход к 3-м матрицам, представляющим яркостный (Y) и два цветностных (Cb) и (Cr) компонента изображения. Компоненты Cb и Cr являются аналогами цветоразностных сигналов в хорошо известной совместимой системе цветного телевидения SECAM. В отличие от компонентов R, G, B компонент Y включает в себя только информацию о яркости пикселов, а компоненты Cb и Cr содержат информацию только об их цвете и насыщенности этого цвета. Поскольку острота зрения человека при наблюдении чисто хроматических изображений (pure chromatic) существенно ниже, чем при наблюдении изображений, имеющих только яркостный контраст (achromatic), переход к компонентам Cb и Cr выгоден, т. к. позволяет при их кодировании использовать меньшее количество отсчетов в блоке и за счет этого получить дополнительное сжатие. Этот переход (перекодирование) осуществляется следующим образом:

> Y = 0,299R + 0,587G + 0,114B, Cb = -0,169R - 0,331G + 0,5B,Cr = 0,5R - 0,419G - 0,0813B.

Далее матрица, представляющая яркостный компонент и имеющая размер 16×16 отсчетов, разделяется на 4 матрицы размером 8×8 отсчетов каждая, а две цветностных матрицы Cb и Cr путем прореживания по строкам и столбцам преобразуются в две цветностных матрицы Cb и Cr размером 8×8. При прореживании этих матриц из них исключаются каждая вторая строка и каждый второй столбец. В дальнейшем, при декодировании исключенные пикселы восстанавливаются методом интерполяции, путем использования для этой цели значений соседних сохраненных пикселов. Такое преобразование оказывается допустимым, поскольку, как уже отмечалось ранее, наше зрение имеет пониженную остроту при наблюдении чисто хроматических изображений и погрешностей, возникающих при таком способе восстановления исключенных при прореживании пикселов, не замечает.

На этом этапе сжатия, с одной стороны, в сжимаемое изображение вносятся необратимые искажения за счет прореживания, т. е. происходит потеря информации, а с другой стороны, имеет место его сжатие в два раза. Действительно, до прореживания полное количество отсчетов, которыми был представлен блок изображения, равнялось $3 \times 16 \times 16 = 768$, в то время как после прореживания только 384.

Затем каждый из отсчетов шести матриц размером 8×8 отсчетов подвергается ДКП и представляется 12-разрядным двоичным кодом. При этом получаются шесть матриц спектральных коэффициентов, 4 из которых представляют собой компонент Y, а две представляют компоненты Cb и Cr. Основное сжатие достигается на этапе квантования спектральных коэффициентов благодаря тому, что спектральные коэффициенты с большими индексами, на долю которых приходится малая доля энергии изображения, квантуются на малое число уровней (или усекаются), и, следовательно, на их представление затрачивается мало двоичных единиц кода. На этом этапе также имеет место потеря информации, т. к. в изображение вносятся необратимые искажения (шум квантования). Процесс квантования заключается в том, что матрица спектральных коэффициентов целочисленно поэлементно делится на матрицу квантования, имеющую такой же размер, что и блоки спектральных коэффициентов, т. е. 8×8. При этом значение проквантованного спектрального коэффициентов, т. е. 8×8. При этом значение проквантованного спектрального коэффициентов, т. е. 8×8. При этом значение проквантованного спектрального коэффициентов, имеющую такой же размер, что и блоки спектральных коэффициентов спектрального коэффициентов, т. е. 8×8. При этом значение проквантованного спектрального коэффициентов кы в следующим образом:

$$F_{\rm KB}\left(u,v\right) = \frac{F\left(u,v\right)}{Q\left(u,v\right)},\,$$

где F(u, v) — исходное значение спектрального коэффициента, а Q(u, v) — соответствующий ему по положению в матрице элемент матрицы квантования. Матрица квантования **Q** построена по зональному принципу, составляющие ее числа представляют собой величины, равные $2^{(12-n_0)}$, где n_0 — количество разрядов кода, которым представляется спектральный коэффициент, входящий в соответствующую зону. Эта процедура интересна тем, что процесс целочисленного деления, с одной стороны, обеспечивает приведение спектральных коэффициентов к значениям одного порядка, а с другой стороны, благодаря имеющему при этом место округлению достигается собственно квантование. После выполнения операции квантования мы получаем матрицу проквантованных спектральных коэффициентов **F**_{кв}, особенностью которой является наличие большого количества малых и нулевых спектральных коэффициентов, расположенных преимущественно в правом нижнем углу матрицы. При восстановлении сжатого изображения значения проквантованных спектральных соектральных коэффициентов на значения соответствующих коэффициентов матрицы квантования у составляются поэлементно на значения соответствующих коэффициентов матрицы квантования Q(u, v).

Следующий шаг алгоритма сжатия состоит в преобразовании полученной матрицы квантованных спектральных коэффициентов размером 8×8 пикселов в вектор из 64 элементов, в котором малые и нулевые спектральные коэффициенты должны

быть по возможности сгруппированы. Эта цель достигается путем применения так называемого зигзаг-сканирования, показанного на рис. 13.3.

Поскольку в начале зигзаг-сканирования считываются спектральные коэффициенты с большими амплитудами, а в конце — спектральные коэффициенты, величина которых мала или равна нулю, получающаяся в результате этого сканирования последовательность чисел будет в конце содержать длинные последовательности нулей. Эта особенность используется для дальнейшего сжатия данных путем энтропийного кодирования, которое состоит в последовательном применении метода кодирования длин серий и кода Хаффмена. Из ряда спектральных коэффициентов образуются пары чисел, одно из которых равно значению ненулевого спектрального коэффициента, а другое — количеству предшествующих этому спектральному коэффициенту нулей. Полученные таким образом пары сжимаются посредством применения кода Хаффмена с фиксированной таблицей. В этой таблице наиболее вероятным значениям полученных чисел, которые соответствуют последовательностям нулей и последовательностям малых ненулевых спектральных коэффициентов, ставятся в соответствие короткие коды. Поскольку код Хаффмена является префиксным, то в данном случае не требуется разделителей между кодовыми словами.



Рис. 13.3. Зигзаг-сканирование

Отдельно следует сказать о сжатии спектрального коэффициента с нулевыми индексами, т. е. постоянной составляющей или средней яркости блока (DC). При кодировании этого спектрального коэффициента используется дифференциальная кодово-импульсная модуляция, при которой его значение предсказывается на основе его уже известного значения для предыдущего блока.

Алгоритм декодирования повторяет все перечисленные операции, но в обратном порядке. При этом переход от цветового пространства Y, Cb, Cr к цветовому пространству R, G, B выполняется по формулам

$$R = Y + 1,402Cr,$$

$$G = Y - 0,344Cb - 0,714Cr,$$

$$B = Y + 1,772Cb.$$

Достоинством описанного метода является высокий коэффициент сжатия, который для цветных изображений при хорошем качестве их восстановления может дости-

гать 6—10. Величина коэффициента сжатия изображений при их записи в файл может регулироваться посредством специальной опции, которая соответствующим образом изменяет коэффициенты матрицы квантования \mathbf{Q} . С помощью этой регулировки устанавливается допустимая степень ухудшения сжимаемого изображения, как, например, это сделано в графическом редакторе Photoshop. Чем большая степень сжатия выбрана, тем большие искажения будут в восстановленном изображении. При недопустимо больших степенях сжатия, если не принято специальных мер, на восстановленном изображении будет просматриваться блочная структура, так называемый эффект забора, заклеенного объявлениями. В настоящее время этот метод сжатия широко применяется практически во всех графических редакторах.

13.6. Вейвлет-преобразование

Вейвлет-преобразования применяются при сжатии изображений [13, 16, 23, 50]. Термин Wavelet переводят обычно как "короткая волна или всплеск".

При вейвлет-преобразовании спектр исходного сигнала разделяется на низкочастотный и высокочастотный компоненты с использованием в качестве импульсных характеристик соответствующих фильтров так называемых вейвлет- и скейлингфункций [16]. Поясним сказанное на простейшем примере, когда исходная последовательность отсчетов, например значений яркости пикселов на строке изображения, разбивается на пары $L_c(2k)$ и $L_c(2k+1)$, используя которые вычисляют две последовательности — $v^{(1)}(k)$ и $v^{(2)}(k)$, одна из них представляет собой полусуммы отсчетов $L_c(2k)$ и $L_c(2k+1)$, а вторая полуразности, т. е.

$$v^{(1)}(k) = \left[L_{c}(2k) + L_{c}(2k+1) \right] / 2,$$

$$v^{(2)}(k) = \left[L_{c}(2k) - L_{c}(2k+1) \right] / 2.$$

В этих формулах значения, заключенные в скобки, представляют собой номера отсчетов. Возврат от последовательностей $v^{(1)}(k)$ и $v^{(2)}(k)$ к последовательностям $L_c(2k)$ и $L_c(2k+1)$ осуществляется следующим образом:

$$L_{c}(2k) = v^{(1)}(k) + v^{(2)}(k),$$
$$L_{c}(2k+1) = v^{(1)}(k) - v^{(2)}(k).$$

Поясним сказанное примером, когда вейвлет-преобразованию подвергается строка, состоящая из восьми пикселов, яркости которых равны: 120, 111, 112, 118, 117, 114, 110, 102. Следуя приведенному выше правилу, из исходной последовательности будут получены две последовательности отсчетов $v^{(1)}(k)$ и $v^{(2)}(k)$, равные соответственно (115,5, 115, 115,5, 106) и (4,5, -3, 1,5, 4). Характерной особенностью

второй группы отсчетов $v^{(2)}(k)$ является то, что их значения близки к нулю. Кроме того, поскольку вторая группа отсчетов содержит информацию об изменениях яркости, ее спектр сосредоточен в области высоких пространственных частот, в то время как спектр первой группы отсчетов сосредоточен в области низких пространственных частот. Таким образом, в результате вейвлет-преобразования произошло разделение сигнала по спектру частот. Рассмотренная операция может быть рекурсивно повторена по отношению к $v^{(1)}(k)$, в результате чего исходная последовательность, состоящая из восьми отсчетов, будет представлена уже тремя последовательностями: (115,25, 110,75), (0,25, 4,75), (4,5, -3, 1,5, 4). Обычно при использовании вейвлет-преобразований в алгоритмах сжатия данных они рекурсивно повторяются несколько раз.

Аналогичным образом может быть реализован алгоритм при вейвлет-преобразованиях двумерных данных, т. е. изображений. В этом случае изображение разбивается на группы размером 2×2 пиксела. Обозначая значения яркостей пикселов в группе: L(2k, 2n), L(2k+1, 2n), L(2k, 2n+1), L(2k+1, 2n+1), где k — номер строки, n — номер столбца, будем иметь после преобразования для каждой группы по четыре компонента:

$$v^{(1)}(k,n) = \left[L_{c}(2k,2n) + L_{c}(2k+1,2n) + L_{c}(2k,2n+1) + L_{c}(2k+1,2n+1) \right] / 4,$$

$$v^{(2)}(k,n) = \left[L_{c}(2k,2n) + L_{c}(2k+1,2n) - L_{c}(2k,2n+1) - L_{c}(2k+1,2n+1) \right] / 4,$$

$$v^{(3)}(k,n) = \left[L_{c}(2k,2n) - L_{c}(2k+1,2n) + L_{c}(2k,2n+1) - L_{c}(2k+1,2n+1) \right] / 4,$$

$$v^{(4)}(k,n) = \left[L_{c}(2k,2n) - L_{c}(2k+1,2n) - L_{c}(2k,2n+1) + L_{c}(2k+1,2n+1) \right] / 4.$$

Эти компоненты объединяются в четыре матрицы и обычно размещаются рядом на одной плоскости, как это показано на рис. 13.4, б. На этом рисунке показана также ориентация контуров, представляемых компонентами $v^{(2)}(k,n)$, $v^{(3)}(k,n)$ и $v^{(4)}(k,n)$. Обратим внимание на то, что суммарное количество отсчетов, которыми представлены матрицы преобразованного изображения, равно количеству отсчетов в исходном изображении, т. е. размер исходного изображения, как это видно из рис. 13.4, *a*, и суммарный размер изображений, представленных полученными четырьмя матрицами, одинаковы. В рассмотренных примерах был использован простейший случай, когда использовались вейвлеты Хаара, которые в силу ряда присущих этим вейвлетам недостатков не применяются.

На практике применяются другие вейвлеты и другие методы выполнения вейвлетпреобразований. При этом компоненты $v^{(1)}(k,n)$, $v^{(2)}(k,n)$, $v^{(3)}(k,n)$, $v^{(4)}(k,n)$ получают несколько отличным способом, который включает в себя две последовательные операции.



Рис. 13.4. Здесь: a — матрица отсчетов изображения; δ — четыре матрицы компонентов вейвлет-преобразования

Первая операция заключается в расфильтровке исходного изображения на четыре частотных диапазона посредством его свертки с импульсными характеристиками $h_L(.)$ и $h_H(.)$. Свертка с импульсной характеристикой $h_L(.)$ устраняет из изображения верхние пространственные частоты, а свертка с импульсной характеристикой $h_H(.)$ устраняет из изображения нижние пространственные частоты. При этом вначале вычисляются четыре массива расфильтрованных изображений по формулам:

$$u^{(1)}(k,n) = \sum_{k'=-\frac{N-1}{2}}^{\frac{N-1}{2}} h_L(k') \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_c(k+k',n+n')h_L(n'),$$

$$u^{(2)}(k,n) = \sum_{k'=-\frac{N-1}{2}}^{\frac{N-1}{2}} h_L(k') \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_c(k+k',n+n')h_H(n'),$$

$$u^{(3)}(k,n) = \sum_{k'=-\frac{N-1}{2}}^{\frac{N-1}{2}} h_H(k') \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_c(k+k',n+n')h_L(n'),$$

$$u^{(4)}(k,n) = \sum_{k'=-\frac{N-1}{2}}^{\frac{N-1}{2}} h_H(k') \sum_{n'=-\frac{N-1}{2}}^{\frac{N-1}{2}} L_c(k+k',n+n')h_H(n'),$$

где *N* — протяженность импульсной функции.

Вторая операция заключается в прореживании (децимации) результатов расфильтровки, при котором в окончательных матрицах, т. е. в вейвлет-компонентах $v^{(1)}(k,n), v^{(2)}(k,n), v^{(3)}(k,n)$ и $v^{(4)}(k,n)$, сохраняется каждый второй отсчет по строке и каждый второй отсчет по столбцу.



Рис. 13.5. Здесь: *а* — исходное изображение; *б* — четыре матрицы компонентов вейвлет-преобразования

На рис. 13.5, *а* приведено изображение, представленное массивом $L_c(k,n)$, а на рис. 13.5, δ — четыре изображения, соответствующие массивам $v^{(1)}(k,n)$ (верхнее левое), $v^{(2)}(k,n)$ (верхнее правое), $v^{(3)}(k,n)$ (нижнее левое) и $v^{(4)}(k,n)$ (нижнее правое). Из рис. 13.5 хорошо видно, что массив $v^{(1)}(k,n)$ представляет собой уменьшенную копию исходного изображения, которое при этом представлено с половинным разрешением. Этот компонент разложения носит название *аппроксимации*. Компоненты $v^{(2)}(k,n)$, $v^{(3)}(k,n)$ и $v^{(4)}(k,n)$ содержат резкие границы, ориентированные вертикально, горизонтально и, соответственно, по диагонали, и называются *деталями*.

Восстановление исходного изображения из компонентов, на которые оно было разложено, осуществляется в обратном порядке: вначале в каждом из компонентов разложения восстанавливаются удаленные отсчеты путем интерполяции, а затем все компоненты суммируются.

В табл. 13.1 приведены импульсные функции $h_L(k)$ и $h_H(k)$, которые используются в процедуре вычисления вейвлетов, примененных в архиваторе JPEG 2000.

Таблица 13.1

k	$h_L(k)$	$h_H(k)$
0	1,115087052456994	0,6029490182363579
±1	0,5912717631142470	-0,2668641184428723
±2	-0,05754352622849957	-0,07822326652898785
±3	-0,09127176311424948	0,01686411844287495
±4	0	0,02674875741080976
±5	0	0

13.7. Сжатие изображений на основе вейвлет-преобразования

Обратимся теперь к рассмотрению метода сжатия данных на основе вейвлетпреобразований применительно к случаю сжатия черно-белого полутонового изображения. Этот метод заключается в том, что вначале сжимаемое изображение рекурсивно подвергается обычно трехкратному вейвлет-преобразованию путем последовательного применения его сначала к исходному изображению, в результате которого получаются четыре матрицы отсчетов (коэффициентов), второй раз оно применяется к аппроксимации, т. е. к матрице, представляющей размытую версию исходного изображения (аппроксимацию), и наконец, третий раз — к полученной на втором этапе аппроксимации.



Рис. 13.6. Трехкратное вейвлет-преобразование: *а* — компоненты вейвлет-преобразования; *б* — поясняющая диаграмма Если вейвлет-преобразование осуществляется путем свертки, то после каждой операции свертки осуществляется децимация полученной матрицы. На рис. 13.6, *а* приведен результат трехкратного вейвлет-преобразования изображения, а на рис. 13.6, *б* поясняющая диаграмма.

Полученные таким образом компоненты квантуются, причем высокочастотные компоненты квантуются на меньшее число уровней, низкочастотные — на большее. Этот метод в значительной степени похож на метод сжатия данных, используемый в JPEG, и отличается от него лишь тем, что в JPEG квантованию на разное число уровней подлежат компоненты (спектральные коэффициенты), полученные в результате ДКП, в то время как в рассматриваемом методе на различное число уровней квантуются компоненты, полученные в результате вейвлет-преобразований. И в том, и в другом случае используется особенность нашего зрения, заключающаяся в том, что оно мало чувствительно к шуму квантования высокочастотных компонентов изображения. Однако благодаря тому, что при использовании вейвлет-преобразований исходное изображение не разбивается на отдельные блоки, в восстановленном после сильного сжатия изображении отсутствуют такие неприятные артефакты (искажения), как заметность блочной структуры. Благодаря тому, что высокочастотные компоненты квантуются на малое число уровней, а их большинство, как это видно из рис. 13.6, на их сохранение расходуется мало двоичных единиц кода, благодаря чему достигается сжатие цифрового потока, которым представляется изображение. При квантовании отсчетов компонентов, полученных в результате вейвлет-преобразований, на пониженное число уровней в них вносится шум квантования, т. е. имеет место потеря информации, но если сжатие не слишком велико, т. е. число используемых уровней квантования не слишком занижено, то эти шумы на восстановленном изображении будут незаметны. Квантование отсчетов компонентов осуществляется путем деления матрицы отсчетов на матрицу квантования. Затем проквантованные отсчеты подвергаются энтропийному кодированию аналогично тому, как это делается в JPEG. В результате энтропийного кодирования получается дополнительное сжатие, которое составляет около трех раз.

Результирующее сжатие данных, которое получается при использовании данного метода, достигает 30—50 раз.

В заключение отметим, что при большом сжатии данных этот метод также приводит к появлению артефактов на восстановленных изображениях в виде появления окантовок и посторонних узоров, однако они менее неприятны, чем артефакты, возникающие при использовании других методов сжатия данных. На рис. 13.7 для сравнения приведены три изображения: исходное и два сжатых изображения. Одно из них сжато с применением дискретного косинусного преобразования и другое с применением вейвлет-преобразования. В первом случае, несмотря на то, что величина сжатия составляла только 6 раз, искажения на изображении хорошо заметны. Во втором же случае сжатие составляло 9 раз, т. е. в 1,5 раз больше, чем в первом случае, и, несмотря на это, следов искажений на глаз незаметно.





а





Рис. 13.7. К сравнению двух методов сжатия изображений: *а* — исходное изображение; *б* — изображение, сжатое с использованием дискретного косинусного преобразования; *в* — изображение, сжатое с использованием вейвлет-преобразований

13.8. Сжатие изображений в формате JPEG 2000

В отличие от формата сжатия JPEG, в формате JPEG 2000 вместо дискретных косинусных преобразований используются вейвлет-преобразования, благодаря чему при больших величинах сжатия на декомпрессированных изображениях не появляются неприятные артефакты в виде блочной структуры [13, 105]. Дополнительным преимуществом, которое получается благодаря использованию вейвлетпреобразований, является возможность постепенного "проявления" изображения при передаче его по сети, поскольку вначале передается компонент $v^{(1)}(k,n)$, т. е. аппроксимация (размытая версия исходного изображения), а затем уже другие компоненты. Эта особенность позволяет просто показывать "огрубленные" изображения по заголовку.

Алгоритм JPEG 2000 включает в себя следующие вычислительные операции при сжатии цветных изображений.

Вначале осуществляется сдвиг по интенсивности каждого компонента RGBизображения перед его преобразованием в цветовое пространство YUV. Это делается для симметрирования динамического диапазона сигнала относительно нуля, благодаря чему увеличивается степень сжатия. Преобразование выполняется в соответствии с формулой

$$I_{\text{Bbix}}(k,n) = I_{\text{Bx}}(k,n) - 2^{\gamma-1}.$$

Значение степени у для каждого компонента R, G и B определяется кодером при сжатии и сообщается декодеру так, что при декомпрессии выполняется обратное преобразование

$$I_{\rm Bbix}(k,n) = I_{\rm Bx}(k,n) + 2^{\gamma-1}.$$

Следующая вычислительная операция, выполняемая кодером, заключается в преобразовании сигналов из цветового пространства RGB в цветовое пространство YUV в соответствии с формулами

$$Y = (R + 2G + B)/4,$$
$$U = R - G,$$
$$V = B - G.$$

При этом в декодере для обратного преобразования выполняются вычисления по формулам

$$R = U + G,$$

$$G = Y - (U + V)/4,$$

$$B = V + G.$$

Далее выполняется дискретное вейвлет-преобразование яркостной и двух цветоразностных матриц. В отличие от алгоритма JPEG в алгоритме JPEG 2000 не осуществляют прореживания матриц \mathbf{U} и \mathbf{V} по строкам и столбцам в целях увеличения степени сжатия.

Для достижения этой цели в отличие от матрицы яркости **Y**, для матриц **U** и **V** просто не сохраняют компоненты вейвлет-преобразования $v^{(8)}(k,n)$, $v^{(9)}(k,n)$, $v^{(10)}(k,n)$.

Следующая вычислительная операция заключается в квантовании всех компонентов (матриц), полученных в результате вейвлет-преобразований.

На заключительном этапе полученные массивы данных сжимаются посредством энтропийного кодера, который называется MQ-кодером. В этом кодере осуществляется декорреляция кодируемых отсчетов и их сжатие посредством арифметического кодирования.

Процесс декомпрессии осуществляется в обратном порядке.

13.9. Фрактальное кодирование

Фрактальный метод сжатия изображений относится к группе методов сжатия с потерей информации, другими словами, в случае применения фрактального метода восстановленное (декомпрессированное) изображение отличается от исходного на величину шума преобразования, хотя это отличие и может быть очень малым [13]. Начало разработке этого метода положили исследования Майкла Барнсли, который открыл класс теорем, позволивших эффективно сжимать изображения. При использовании фрактального метода сжатия кодируется по существу не само изображение, а алгоритм его построения. Этот метод базируется на замечательной особенности реальных изображений, заключающейся в том, что в них с небольшими вариациями многократно повторяются отдельные самоподобные фрагменты, например, листья в кроне дерева, окна в здании, пешеходы на улице, чешуя на теле рыбы, границы между темными и светлыми участками изображения, как это показано на рис. 13.8.

И хотя эти фрагменты различаются между собой в деталях, тем не менее, в них много общего. Не претендуя на детальность изложения, опишем алгоритм кодирования при фрактальном сжатии изображений.

Процесс сжатия начинается с того, что берутся два идентичных экземпляра кодируемого изображения — A и B, один из них разделяется на неперекрывающиеся блоки, называемые ранговыми областями, а на втором задается набор доменов, которые могут взаимно перекрываться, как это показано на рис. 13.9. Домены должны включать в себя характерные фрагменты, которые в дальнейшем используются для построения декодируемого изображения. После этого начинается кодирование изображения путем подбора для каждой ранговой области наиболее подходящего домена, посредством которого распределение яркости в ранговой области может быть аппроксимировано распределением яркости в домене. Для того чтобы получить наилучшую аппроксимацию, домены подвергаются аффинным преобразованиям, в результате которых происходит не только их геометрическая деформация, но и изменения контраста, а также яркости. Если таким путем не удается достичь удовлетворительной аппроксимации распределения яркости в ранговой области распределением яркости в преобразованном домене, ранговая область делится на четыре части, и процесс повторяется. Качество требуемой аппроксимации задается в виде допустимого значения среднего квадрата ошибки аппроксимации (среднего квадрата несовпадения). Номера доменов, использованных при кодировании каждой ранговой области, а также коэффициенты аффинного преобразования сжимаются путем энтропийного кодирования и записываются в файл. Файл сжатого изображения содержит заголовок с информацией о расположении ранговых областей и доменов, а также таблицу эффективно упакованных аффинных коэффициентов для каждой ранговой области.



Рис. 13.8. К пояснению принципа фрактального метода сжатия изображений





Алгоритм декодирования изображений не столь нагляден. Этот алгоритм заключается в том, что берутся два экземпляра одного и того же изображения A и B, распределение яркости в которых безразлично. На этих изображениях выделяются области, границы которых совпадают с границами ранговых областей и доменов, а затем, используя известные значения аффинных коэффициентов, по доменам, выделенным на изображении B, находятся распределения яркости в ранговых областиях изображения A. После этого изображения A и B меняются местами, и операция повторяется. Можно показать, что при многократном повторении этой операции распределение яркости в изображениях A и B будет приближаться к распределению яркости в исходном изображении.

Обратим внимание на то, что алгоритмы сжатия и декомпрессии асимметричны. Следует также отметить, что процесс сжатия требует много больше времени, чем процесс декомпрессии.

Фрактальный метод сжатия изображений обеспечивает коэффициенты сжатия, составляющие от 50 до 500 раз в зависимости от типа изображения и допустимого уровня шума преобразования. Графические файлы, при создании которых этот метод использован, легко узнать по расширению fif.

У фрактального метода традиционно называют следующие основные проблемы:

- очень медленное кодирование. В настоящее время эта проблема решена только частично [165];
- относительно медленное итерационное декодирование. В настоящее время эта проблема решена, см., например, [160].

Вследствие отмеченных недостатков в настоящее время этот метод применяется на практике сравнительно редко.

13.10. Сжатие изображений в формате MPEG-2

При разработке метода сжатия движущихся изображений к нему был предъявлен ряд весьма жестких и противоречивых требований. Во-первых, этот метод должен обеспечивать высокую степень сжатия при высоком качестве восстановленных изображений, поскольку цифровые потоки, необходимые для передачи несжатых изображений, чрезмерно велики. Для того чтобы передать цветное телевизионное изображение, при затрате 24 двоичных единиц на отсчет, разложении его на 576 строк с 720 отсчетами в строке и 25 кадрах в секунду, потребовался бы цифровой поток в 240 Мбит/с. Кроме того, метод сжатия должен обеспечить произвольный доступ к любому кадру, что необходимо при монтаже фильма. Необходимо также при передаче обеспечить синхронизацию видео- и аудиодорожек, а также помехоустойчивость самой передачи. В настоящее время для этих целей разработана целая группа методов сжатия, объединенная в стандарт MPEG-2 (Moving Picture Experts Group). В стандарте MPEG-2 использованы принципы сжатия изображений, основанные на устранении межкадровой (временной), внутрикадровой (пространственной) и психофизической избыточности.

Для устранения межкадровой избыточности применяется временная дифференциальная кодово-импульсная модуляция (ДКИМ), которая использует то обстоятельство, что последующий кадр в движущемся изображении обычно мало отличается от предыдущего, что позволяет его достаточно точно предсказать. Передавая ошибку предсказания, а по существу, разность между двумя соседними кадрами, которая содержит много нулей, можно существенно сократить цифровой поток. Точность предсказания при передаче движущихся изображений также увеличивают за счет оценки вектора движения и компенсации этого движения. Принцип формирования вектора движения состоит в том, что в передаваемом кадре выделяется движущаяся область изображения, например, автомобиль, перемещающийся в направлении, перпендикулярном оси наблюдения. Если оценить смещение этой области от кадра к кадру по горизонтали V_x и по вертикали V_y , то в качестве предсказания можно использовать эту область изображения из предыдущего кадра, смещенную по горизонтали и по вертикали на величины V_r и V_v соответственно. Набор величин V_x и V_y называется вектором движения. Компенсация движения при формировании сигнала ошибки предсказания позволяет уменьшить его значение и, как следствие, увеличить сжатие. Полностью сигнал ошибки предсказания не может быть сведен к нулю, т. к. при движении объекта обычно изменяется его

Одним из наиболее часто применяющихся методов оценки вектора движения является метод согласования блоков. Существо этого метода заключается в том, что текущий блок (а изображение разбивается на блоки, как в случае стандарта JPEG) проецируется на предыдущее (базовое) опорное изображение и, перемещаясь по нему в некоторой области поиска, попиксельно сравнивается со всеми фрагментами опорного изображения, на которые он спроецирован. Тот фрагмент опорного изображения, который меньше всего отличается от текущего блока по величине среднего квадрата ошибки, принимается за опорный блок для целей предсказания, а смещение между опорным и проецируемым блоками определяет вектор движения. Величина среднего квадрата ошибки, как не трудно видеть, вычисляется следующим образом:

ракурс и масштаб, однако он может быть значительно уменьшен.

$$\sigma^{2} = \frac{1}{KN} \sum_{k=-K/2}^{K/2} \sum_{n=-N/2}^{N/2} \left[L_{\text{c.on}} \left(k + V_{x}, n + V_{y} \right) - L_{\text{c}} \left(k, n \right) \right]^{2},$$

где K и N — соответственно высота и ширина области поиска в пикселах, $L_{\text{с.оп}}(k,n)$ — яркость пиксела, расположенного в k-й строке и n-м столбце опорного изображения, $L_{\text{с}}(k,n)$ — яркость пиксела, расположенного в k-й строке и n-м столбце текущего изображения, из которого взят блок.

Для устранения внутрикадровой и психофизической избыточности применяется метод, реализованный в стандарте JPEG.

Сжатие движущихся изображений в рассматриваемом методе происходит следующим образом. Исходная последовательность кадров движущегося изображения разбивается на макроблоки размером 16×16 отсчетов, как это сделано в стандарте JPEG, а затем разделяется на три типа кадров: *I*-кадры (Intrapictures), *P*-кадры (Predicted) и *B*-кадры (Bidirection).

I-кадры обеспечивают возможность произвольного доступа к любому кадру, являясь своеобразными точками входа в поток данных для декодера, и сжимаются независимо в соответствии с методом, используемым в стандарте JPEG. При их сжатии выполняются следующие действия:

- 1. В каждом макроблоке изображения делается переход от матриц, представляющих красный, зеленый и синий компоненты изображения, к трем матрицам, представляющим яркостный (Y) и два цветностных (Cb и Cr) компонента изображения.
- 2. Матрица, представляющая яркостный компонент размером 16×16 отсчетов, разделяется на 4 матрицы размером 8×8 отсчетов каждая.
- 3. Две матрицы, содержащие цветоразностные отсчеты Cb и Cr, путем их прореживания по строкам и столбцам преобразуются в две цветностных матрицы: Cb и Cr размером 8×8. При прореживании этих матриц из них исключаются каждая вторая строка и каждый второй столбец. На этом этапе кодирования происходит сжатие данных в два раза.
- 4. Каждая из шести матриц размером 8×8 отсчетов преобразуется посредством ДКП в матрицы спектральных коэффициентов.
- 5. Каждая из матриц спектральных коэффициентов квантуется.
- 6. Осуществляется зигзаг-сканирование каждой из шести матриц спектральных коэффициентов.
- К полученным данным применяется энтропийное кодирование, заключающееся в последовательном применении метода кодирования длин серий и кода Хаффмена.

Кодирование *P*-кадров отличается от кодирования *I*-кадров тем, что они кодируются на основе предсказания путем ссылок на блоки предыдущих *I*- или *P*-кадров. В результате этого предсказания формируется кадр сигнала ошибки предсказания, который представляет собой разность между опорным и предсказанным кадрами с учетом вектора движения. Этот кадр подвергается сжатию посредством применения той же последовательности операций, которая применяется для кодирования *I*-кадров.

При кодировании В-кадров используются ссылки на два кадра, находящихся впереди и позади них. Эти кадры сами в качестве ссылки использоваться не могут.

Кадры различных типов объединяются в группы. Каждая группа начинается с кадра типа *I*, образующего опорный сигнал для предсказания при кодировании кадров типа *P* и *B*. Для того чтобы получить высокий коэффициент сжатия, группа должна быть достаточно большой. При воспроизведении изображений последовательность кадров может быть, например, такой: *I*, *B*, *B*, *P*, *B*, *B*, *P*, *B*, *B*, *I*, ... Однако при кодировании и декодировании изображений порядок кадров различного типа в группе должен быть изменен на следующий: *I*, *P*, *B*, *B*, *P*, *B*, *B*, *I*, поскольку кадры типа *B* создаются на основании кадров *I* и *P*, которые к этому времени уже должны иметься. Группы могут быть как замкнутыми, когда последним кадром является кадр типа *P*, так и разомкнутыми, когда последним кадром является кадр типа *I*. Достоинством замкнутых групп является то, что они позволяют выполнять прямые монтажные переходы перед очередным *I*-кадром без декодирования цифрового потока, однако для этого требуется более сложный кодер.

Степень сжатия каждого из трех типов кадров различна: она меньше всего у кадров типа I, у кадров типа P она примерно в 3 раза больше, чем у кадров типа I, а у кадров типа B она примерно в 4,5 раза превышает степень сжатия кадров типа I.

Декодирование (декомпрессия) изображения при использовании описанного метода сжатия состоит в выполнении обратных преобразований в обратном порядке.

13.11. Определение векторов движения

Алгоритм определения векторов движения, как уже было сказано в предыдущем разделе, заключается в том, что путем перемещения проекции каждого блока изображения текущего кадра по базовому изображению в некоторой области поиска находится такое ее положение, при котором среднеквадратичная разность проекции блока и фрагмента опорного изображения принимает минимальное значение. Это смещение блока по горизонтали V_x и по вертикали V_y относительно его исходного положения и принимается за вектор движения. При наличии в кадре объектов, движущихся с высокой скоростью, область поиска необходимо брать достатом.

точно большой, что приводит к увеличению времени поиска и количества вычислительных операций пропорционально квадрату скорости, поскольку площадь области поиска пропорциональна квадрату скорости движения объектов.

Для сокращения времени поиска и количества вычислительных операций, при определении векторов движения используют алгоритм "иерархической оценки движения" [99, 104, 143]. В соответствии с этим алгоритмом кодируемый кадр несколько раз







Рис. 13.10. К пояснению алгоритма "иерархической оценки движения"
последовательно уменьшают, каждый раз в два раза, при этом получают последовательность изображений подобную той, которая показана на рис. 13.10.

Аналогичным образом поступают с базовым кадром. Размеры блоков при этом сохраняются неизменными, например, 16×16 пикселов. Таким образом, формируется L уровней иерархии. После этого выполняется процедура определения векторов движения для каждого блока кадра на L-м уровне иерархии. Поскольку на этом уровне блоков в 4^L раз меньше, чем в исходном кадре, а также области поиска в это же число раз меньше, то соответственно время поиска и количество вычислительных операций также оказываются небольшими. Далее переходят к предыдущему L-1 уровню иерархии и только уточняют значения векторов движения для каждого их четырех блоков, на которые разделяется каждый блок L-го уровня иерархии. Так поступают до тех пор, пока не будут определены векторы движения для каждого блока нулевого уровня, т. е. исходного кадра. Рассмотренный алгоритм позволил существенно сократить количество вычислительных операций, затрачиваемых на определение как векторов движения, так и на время их определения.

Дальнейшее развитие этого алгоритма было предложено в работах [7, 8, 110] и заключалось в том, что граничащие друг с другом блоки, с одинаковыми значениями векторов движения, объединялись. При этом изображение, разделенное на блоки, могло бы выглядеть так, как это показано на рис. 13.11. Этот алгоритм носит название "модифицированный алгоритм иерархической оценки движения".

Объединение отдельных граничащих друг с другом блоков с одинаковыми векторами движения позволяет еще больше сжать цифровой поток при передаче движущихся изображений за счет того, что векторы движения передаются один раз для каждого крупного блока, вместо того, чтобы для каждого небольшого блока, составляющего крупный объединенный блок, передавать свой вектор движения.



Рис. 13.11. К пояснению модифицированного алгоритма иерархической оценки движения

13.12. Сжатие изображений в формате MPEG-4

Стандарт компрессии MPEG (Moving Picture Experts Group) по мере развития от версии к версии все более обогащался новыми техническими решениями и обеспечивал все возрастающее возможности. Так, если стандарт MPEG-1, выпущенный в 1992 г., обеспечивал при скорости потока данных 1,5 Мбит/с запись полнометражно-го фильма в стандарте PAL: 352×288 пикселов, 25 кадров/с, стереозвуком с частотой дискретизации 44,1 кГц, то стандарт MPEG-2, выпущенный в 1995 г., обеспечивает скорость потока данных 3—15 Мбит/с и позволяет передавать 25 кадров в секунду уже с размером кадра 720×576 пикселов, существенно улучшив качество изображения. Еще одно достоинство MPEG-2 — поддержка многоканального звука.

Дальнейшая работа над совершенствованием стандарта MPEG привела к внедрению принципиально новых идей, которые были воплощены в MPEG-4, получившем международный статус стандарта в 1999 г. Его назначение формулируется в рабочих документах группы MPEG следующим образом: он задает принципы работы с контентом (цифровым представлением медиаданных) для трех областей:

- 🗖 собственно интерактивного мультимедиа;
- □ графических приложений;
- 🗖 цифрового телевидения.

Рассмотрим, в чем заключаются новые концепции, воплощенные в MPEG-4 [13, 140]. Прежде всего, следует отметить, что этот стандарт задает правила организации среды, причем среды объектно-ориентированной. Ключевым для данного стандарта является понятие "медиаобъект", т. е. MPEG-4 имеет дело не просто с потоками и массивами медиаданных, а с медиаобъектами. В качестве них могут фигурировать графические, текстовые, аудио-, видео-, аудиовизуальные объекты. Примерами объектов могут служить неподвижный фон, видеоперсонажи на прозрачном фоне, трехмерные модели, синтезированная на основе текста речь, музыкальные фрагменты, анимированный спрайт и т. д.

В отличие от предыдущих форматов, которые делили изображение на прямоугольники, кодек MPEG-4 оперирует с объектами произвольной формы. Каждый медиаобъект имеет связанный с ним набор дескрипторов, задающих его свойства, операции, необходимые для декодирования ассоциированных с ним потоковых данных, размещения в сцене, а также поведение и допустимые реакции на воздействие пользователя. Каждый объект имеет свою локальную систему координат, с помощью которой объект управляется в пространстве и времени. Из объектов строятся сцены. Сцена описывается с помощью иерархической структуры, узлами которой являются объекты. Такая структура динамически перестраивается по мере того, как узлы-объекты добавляются, удаляются или изменяются.

В MPEG-4 определен двоичный язык описания объектов, классов объектов и сцен — BIFS, который по существу является расширением языка C++. С помощью команд этого языка можно анимировать объекты, изменять их координаты, размеры,

свойства, реакции на воздействие пользователя, изменять свойства среды, обновлять сцену и т. д. Многие концепции языка BIFS позаимствованы у VRML. Среди допустимых пользовательских команд взаимодействия с контентом — изменение точки наблюдения, удаление, добавление, перемещение объектов внутри сцены и т. п.

В стандарте MPEG-4 можно выделить ряд блоков, отвечающих за решение определенного круга задач. Перечислим некоторые из них.

VLBV Core — ядро, обеспечивающее работу с видео, имеющим очень низкую скорость потока данных. Следует отметить, что за счет использования эффективных методов сжатия удается максимально повысить характеристики видео. Данное ядро направлено также на реализацию операций произвольного доступа к кадрам видеопоследовательности, что требуется в области управления медиаактивами (например, для работы с базами видеоданных).

Второй блок отвечает за работу с видео с большой скоростью потока данных, вплоть до вещательного качества. При этом предусмотрены возможности работы и с прогрессивной, и с чересстрочной развертками.

Третий блок включает в себя функции, зависящие от контента. Эти функции позволяют обрабатывать видео с произвольным силуэтом для отдельного кодирования видеообъектов и манипуляций с ними.

Посредством функций данного блока объекты кодируются таким образом, что декодер имеет возможность в случае ограничений пропускной способности сети, недостаточной вычислительной мощности, малого разрешения дисплея, огрублять изображение, декодируя и выводя лишь часть передаваемой потоковой информации (например, уменьшая частоту или разрешение кадров). Достигается "масштабируемость" видеообъектов, при этом адекватность контента сохраняется. Для кодирования неподвижных изображений и текстур применяется очень эффективный вейвлет-алгоритм. Он обеспечивает кодирование объектов произвольной формы, 11 уровней масштабируемости по разрешению и плавную масштабируемость по качеству изображения. Закодированный поток представляет собой "пирамиду" различных разрешений. В приемнике изображение постепенно "проявляется", улучшаясь настолько, насколько позволяет данная передающая среда. Межкадровое кодирование реализовано на основе предсказания и компенсации движения, а также посредством механизмов работы со спрайтами, представляющими собой неподвижные изображения, которые передаются в декодер один раз, а затем подставляются в нужное место кадра из специального спрайтового буфера всякий раз, когда это необходимо. Это позволяет существенно сократить объем передаваемых данных и обеспечивает большую гибкость в построении сцены.

В MPEG-4 предусмотрены инструменты и алгоритмы работы с синтетическими объектами, сгенерированными средствами компьютерной графики. Используются каркасные представления двумерных и трехмерных моделей, данные для анимирования этих моделей, текстуры, которые могут на эти объекты накладываться. При этом для анимации объектов достаточно передать лишь небольшой объем данных. В язык BIFS входят необходимые средства управления анимацией. Имеются набо-

ры управляющих параметров для задания лица (FDP), для его анимации (FAP), контрольные точки в полигональной сетке, отвечающие за определенные эмоции или движения, и т. п. Имеются средства синтеза речи на базе текстов. Они не только генерируют необходимые фонемы, но могут также создавать поток данных для соответствующей анимации модели лица говорящего. Кроме того, имеются средства учета точки наблюдения. Если, например, в трехмерном пространстве сцены существуют объекты переднего плана, то заслоненные ими фрагменты сцены передаваться не будут.

Для работы с аудиообъектами, которые также входят в структуру дерева сцены и описываются на языке BIFS, имеется богатый набор всевозможных средств.

13.13. Сжатие изображений в формате Н.264

H.264 является технологией сжатия видео нового поколения в рамках стандарта MPEG-4, она также известна как MPEG-4 Part 10.

При использовании стандарта H.264 обеспечивается существенное снижение скорости цифрового потока, при этом выигрыш в эффективности компрессии составляет около двух раз по сравнению со сжатием, получающимся при использовании стандарта MPEG-2. Это дает возможность передавать в одной и той же полосе частот большее число телевизионных каналов. Достигается это, однако, путем существенного усложнения схем кодеров и декодеров. Так в этом случае схема кодера содержит примерно в 8 раз больше логических вентилей, чем схема кодера применяемого в MPEG-2.

Примененные в стандарте H.264 технологии предназначены для работы с видеопоследовательностями, не разделенными предварительно на видеоплоскости. Существенно изменена по сравнению со стандартом MPEG-4 компенсация движения. В рассматриваемом стандарте реализована компенсация движения не только для блоков 16×16 , но и для блоков 16×8 , 8×16 , 8×8 , 8×4 , 4×8 и 4×4 , что заметно повышает эффективность кодирования особенно при небольших величинах сжатия, кроме того, это обеспечивает большую точность представления векторов движения. Обеспечиваемая при этом точность может составлять 1/4 или 1/8 макроблока (в стандарте MPEG-4 максимальная точность, как известно, составляет 1/4 макроблока).

По сравнению с MPEG-2 в H.264 улучшены методы устранения пространственной избыточности. Так перед тем, как выполнить ДКП для макроблока в *I*-кадре, осуществляется его пространственное предсказание (intra-prediction), при котором на основании известных уже закодированных соседних макроблоков находится оценка (предсказание) этого макроблока. Затем яркостные и цветоразностные значения отсчетов предсказанного макроблока поэлементно вычитаются из соответствующих значений кодируемого блока. После этого для "разностного" макроблока выполняется один из трех вариантов дискретного косинусного преобразования: ДКП 4×4, ДКП 8×8 и ДКП 16×16. Такой подход позволяет разделять изображение на блоки, размеры которых лучше адаптируются к содержанию конкретного изображения.

Кроме того, в стандарте H.264 улучшены методы устранения временной избыточности. Для формирования *P*- и *B*-кадра в рассматриваемом стандарте предусмотрено использование более двух предыдущих восстановленных видеокадров.

Далее, для энтропийного кодирования данных (сжатия без потерь) в стандарте предусмотрено использование контекстного адаптивного двоичного арифметического кодера (Context Adaptive Binary Arithmetic Coder, CABAC). При этом для кодирования коэффициентов ДКП и векторов движения вначале выполняется контекстное моделирование и унарная бинаризация, посредством которых входные данные разделяются на множество двоичных последовательностей. После этого каждая последовательность кодируется двоичным адаптивным арифметическим кодером (Mcoder [86, 147]).

В заключение этого раздела отметим, что нельзя назвать какое-либо одно, главное, усовершенствование, введенное в стандарт H.264, которое обеспечило его высокую эффективность по сравнению со стандартом MPEG-2. Каждое из описанных усовершенствований внесло свой сравнительно небольшой вклад, определивший высокие показатели рассмотренного стандарта в целом.

13.14. Метод сжатия, основанный на использовании 3D-моделей, для представления движущихся объектов на 2D-изображениях

Описываемый ниже метод непосредственно вписывается в стандарт MPEG-4 и является хорошей иллюстрацией возможностей этого стандарта в отношении сжатия видеопоследовательностей. Этот метод сжатия видеопоследовательностей описан в работах [74, 177]. Он включает в себя разделение изображения кадра на неподвижный задний план (фоновая составляющая) и движущиеся на его фоне объекты. Важным фактором, благодаря которому обеспечивается высокая эффективность этого метода, является использование априорной информации о структуре кадра, т. е. об объектах, которые перемещаются и, возможно, изменяются на фоне неподвижного заднего плана. В цитируемой работе изложение принципа действия метода сжатия проводится на примере передачи видеоизображений в видеотелефонии, когда передаваемое изображение представляет собой говорящего человека на фоне неподвижного заднего плана. В этом случае движение объекта ограничивается перемещением в пределах кадра, а его изменение определяется только изменением мимики лица.

Как уже было сказано, при реализации описываемого метода исходное изображение кадра вначале разделяется на изображение заднего плана и изображение объекта. Объекту (говорящему человеку) сопоставляется оболочка стандартной 3D векторной модели (например, поясного 3D-портрета, хранящейся в памяти кодирующего устройства), которая затем адаптируется к изображаемому объекту, т. е. проекция 3D-модели на плоскость совмещается с изображением объекта путем модификации координат ее вершин. Кроме того, находятся координаты, определяющие начальное положение объекта в кадре. Отдельно также выделяется текстура объекта.

Далее на изображении лица выделяются антропометрические точки, которыми определяется мимика говорящего человека и которые переносятся на модель, а также параметры карт влияния контрольных вершин модели, определяемых антропометрическими точками.

Задний план, оболочка, адаптированная к изображению объекта, координаты, определяющие начальное положение объекта в кадре, начальные значения координат антропометрических точек, параметры карт влияния и текстура записываются в память на передающей стороне, сжимаются посредством известных методов и один раз, в начале сеанса связи передаются на приемную сторону, где также записываются в память. По этим данным на приемной стороне восстанавливается 3D-изображение объекта в первом кадре, проекция которого на плоскость затем совмещается с изображением заднего плана и таким образом на приемной стороне создается изображение первого кадра передаваемой последовательности.¹

Далее при передаче изображений последующих кадров передается только изменение координат, определяющих положение объекта в кадре, которые прибавляются к координатам, определяющим его начальное положение, а также изменения координат антропометрических точек. При этом используются известные методы сжатия. По этим данным в месте приема восстанавливаются последующие кадры передаваемой видеопоследовательности.

Поскольку количество антропометрических точек невелико, в экспериментах их количество не превышало 15, величина сжатия получается внушительной. Автором работы [74] приводятся следующие данные. При передаче видео в течение одной минуты и при использовании 3D-модели объекта (поясного портрета), состоящей из 2831 вершины, в случае применения формата OBJ фирмы Autodesk потребовалось 727 500 Кбайт, в то время как при использовании описываемого метода потребовалось только 905,31 Кбайт!

Описанный метод может быть использован не только в видеотелефонии, но и в других случаях. Условием его применения является наличие априорной информации об объектах, которые составляют передаваемую сцену с тем, чтобы можно было заранее включить в память кодирующего устройства набор 3D-моделей этих объектов. Мы здесь не останавливаемся на методах разделения объектов и фона,

¹ Читатель, вероятно, уже обратил внимание на то, что при адаптации оболочки 3D-объекта к исходному изображению используется только одно изображение, чего явно недостаточно для адаптации модели по третьей координате — глубине. В данном случае потребовались бы как минимум два изображения, полученных с двух камер, расположенных в разных точках пространства. В рассматриваемом случае адаптации модели по глубине не происходит. Однако поскольку, в конечном итоге, в описываемом методе модель используется для получения плоского изображения объекта, это не проявляется на восстановленном изображении, если только объект не вращается, или если имеют место небольшие повороты (в рассматриваемом примере повороты головы). Проведенный эксперимент подтвердил это.

а также на методах адаптации 3D-моделей к изображениям этих объектов в первом кадре передаваемой видеопоследовательности, которые представляют собой отдельную специальную тему.

13.15. Обзор форматов записи изображений

В настоящее время имеется большое разнообразие форматов записи графических изображений. Одни из них предназначены для записи изображений в растровом представлении, другие — в векторном. Растровое изображение представляет собой двумерный массив с определенным числом строк и столбцов, каждый элемент характеризуется своим значением яркости или цвета. Так как задается вся область графического изображения, то такие изображения имеют фон. Что касается изображений в векторном представлении, то у них фон может быть не определен, т. к. описание изображения сводится к заданию координат точек и набора инструкций относительно того, например, где и какой толщины должна быть проведена линия. Поскольку в этом разделе основное внимание уделяется проблеме сжатия растровых изображений, то остановимся на рассмотрении форматов записи, предназначенных именно для них. Тем более что одна из наиболее бурно развивающихся областей использования эффективно сжатых изображений — это крупнейшее информационное хранилище — World Wide Web, которое поддерживает только растровые изображения.

Многочисленные используемые форматы записи растровых изображений различаются между собой такими важными характеристиками, как количество двоичных единиц, расходуемых на пиксел, наличием или отсутствием сжатия, методом сжатия, а также рядом особенностей, которые играют существенную роль, когда речь идет о выборе формата при решении конкретной практической задачи, например, размещения графики в World Wide Web. К таким особенностям относятся, например, доступность прозрачности фона, наличие режима чередования строк, возможность хранения нескольких изображений в одном файле и т. д. Само понятие формата подразумевает определенную структуру построения графических файлов.

Типичная структура графического файла включает в себя ряд обязательных элементов. Рассмотрим их кратко на примере формата GIF (Graphics Interchange Format), разработанного CompuServe Inc. В структуру этого формата входят заголовок, блок данных (собственно данные изображения), признак конца файла. В заголовке приводится информация, которая позволяет правильно прочитать записанное в файл изображение: формат файла и номер версии (например, GIF89a), количество двоичных единиц на пиксел, глобальная таблица цветов, если она присутствует (это необязательный элемент) и т. д. В блоке данных определены размеры изображения в пикселах, использование режима чередования строк, глобальной либо локальной таблицы цветов, приводится локальная таблица цветов (если она используется), а также собственно данные, закодированные с применением метода LZW. Если в одном файле содержится несколько изображений (а данный формат допускает такую возможность, что удобно для создания анимированных изображений), то независимые друг от друга блоки данных размещаются последовательно. Рассмотрим кратко некоторые характеристики наиболее часто используемых форматов.

- Windows BitMap. Этот формат файлов растровых изображений был разработан Microsoft. Файлы этого формата имеют расширение bmp. Формат поддерживает 256 цветов, а также 16- и 24-битные (True Color) цвета при неограниченном размере изображений. В этом формате изображения могут быть сохранены как без сжатия, так и с использованием метода сжатия без потерь RLE. При использовании метода сжатия без потерь, как уже отмечалось, величина сжатия оказывается небольшой, а поэтому получаются файлы больших размеров. Данный формат не рекомендуется использовать в мультимедийных приложениях, но он очень удобен для обмена данными между различными приложениями Windows. Формат Windows BitMap используется во всех графических редакторах.
- РСХ. Это один из самых старых и широко применяемых форматов хранения растровых изображений. Файлы этого формата имеют расширение рсх. Современные версии этого формата поддерживают 256 цветов, а также 24-битные цвета при максимальном размере изображений 64000×64000 пикселов. В этом формате используется метод сжатия изображений без потерь RLE. Формат применяется практически во всех графических редакторах, предназначенных для работы с растровыми изображениями.
- Аdobe Photoshop. Этот формат файлов растровых изображений специально разработан для программы Adobe Photoshop. Файлы этого формата имеют расширение psd. Формат поддерживается приложениями для Macintosh и Windows. Он позволяет сохранять изображения с многослойной структурой. Максимально допустимый размер изображения составляет 30 000×30 000 пикселов. Формат поддерживает 24-битные цвета. В этом формате применяется метод сжатия без потерь RLE. Данный формат в настоящее время широко применяется в коммерческой графике.
- Коdak Photo CD. Фактически в настоящее время имеется пять различных типов форматов изображений Photo CD: Master Photo CD, Pro Photo CD, Print Photo CD, Catalog Photo CD, Portfolio Photo CD, каждый из которых удобен для различного набора приложений. Файлы записи растровых изображений в этих форматах, разработанных Eastamn Kodak, имеют расширение сd. Форматы поддерживаются всеми операционными системами. Данные форматы позволяют сохранять изображения, максимальный размер которых составляет 4096×6144 пиксела при 24-битной глубине цвета. Форматы используются для сохранения фотографических изображений на компакт-дисках.
- Graphics Interchange Format. Этот формат, известный как формат GIF и используемый для записи растровых изображений, разработан CompuServe Inc. Файлы этого формата имеют расширение gif. Данный формат поддерживается приложениями, работающими в операционных системах MS-DOS, Macintosh, UNIX, Windows, Amiga и др. Максимальный размер изображений, который может быть сохранен в этом формате, составляет 64 000×64 000 пикселов при 256-цветовой палитре. В этом формате применяется метод сжатия без потерь LZW, который

обеспечивает сжатие изображения около двух раз и достаточно высокое быстродействие, приемлемое при просмотре сжатых файлов. Этот формат позволяет в одном файле сохранять несколько изображений, что удобно при сохранении анимированных изображений, а также поддерживает использование так называемой прозрачности фона (transparency) и режим чередования строк, что удобно при работе с графикой в Интернете.

- Joint Photographic Experts Group. Этот формат, известный как JPEG, разработан для записи фотографических изображений. Файлы этого формата имеют расширение jpg и поддерживаются приложениями для всех операционных систем. Максимальный размер изображений, который может быть сохранен в этом формате, составляет 64 000×64 000 пикселов при 24-битовой глубине цвета. В этом формате использован метод сжатия JPEG, который обеспечивает высокую степень сжатия ценой некоторой потери качества восстанавливаемых изображений. При сохранении изображений в этом формате имеется возможность выбирать степень сжатия в зависимости от требований к качеству восстановленного изображения. Файлы этого формата характеризуются невысокой скоростью просмотра. Данный формат применяется для сохранения и обмена данными, а также в Интернете.
- Fractal Image Format. Файлы этого формата имеют расширение fif и позволяют хранить изображения в 24-битных цветах в независимом от разрешающей способности виде. В данном формате использован фрактальный метод сжатия изображений. Этот метод сжатия, как уже отмечалось, обеспечивает очень высокую степень сжатия изображений, однако, как и во всех методах сжатия с потерями, достигается это за счет некоторого снижения их качества при восстановлении. Особенностью этого формата является также очень низкая скорость сжатия. Формат Fractal Image Format сравнительно мало распространен в настоящее время.
- □ Wavelet Image Files. Файлы этого формата имеют расширение wi и позволяют хранить изображения в 24-битных цветах. В этом формате использован метод сжатия изображений, основанный на применении вейвлет-преобразований, который по сравнению с методом сжатия JPEG обеспечивает более высокую степень сжатия изображений. Как и во всех методах сжатия с потерями, в этом методе высокая степень сжатия обеспечивается ценой некоторой потери качества восстанавливаемых изображений, однако в этом метода JPEG. При сохранении изображений выше, чем при использовании метода JPEG. При сохранении изображений в этом формате также имеется возможность выбирать степень сжатия в зависимости от требований к качеству восстановленного изображения. Файлы этого формата характеризуются более высокой скоростью просмотра, чем при использовании метода зписи изображения в графическом редакторе Corel PHOTO-PAINT.
- □ Portable Network Graphics. Этот формат файлов растровых изображений был разработан для обмена графическими данными. Файлы этого формата имеют расширение png. Формат поддерживает до 48 бит на пиксел. В этом формате изображения сохраняются с использованием метода сжатия без потерь LZW.

При использовании метода сжатия без потерь, как уже отмечалось, величина сжатия оказывается небольшой, а поэтому получаются файлы сравнительно больших размеров. В этом формате можно хранить изображения с прозрачным фоном. В последнее время этот формат поддерживается все большим количеством приложений и используется для размещения графики в Интернете.

- Тarga. Этот формат файлов растровых изображений был разработан фирмой True Vision. Файлы этого формата имеют расширение tga. Данный формат поддерживается приложениями, работающими в операционных системах MS-DOS, Windows, UNIX, Atari, Amiga и др. Формат поддерживает 256 цветов, а также 16- и 24-битные цвета при неограниченном размере изображений. В этом формате применен метод сжатия изображений без потерь RLE, а, следовательно, как уже отмечалось, величина сжатия оказывается небольшой, и поэтому получаются файлы больших размеров. Формат применяется в приложениях для рисования, графики и создания изображений. Этот формат является популярным при обмене файлами между различными платформами.
- ТІҒҒ (Tagged Image File Format). Это универсальный формат, важным достоинством которого является переносимость на разные платформы. Файлы, записанные в данном формате, имеют расширение tif. Формат поддерживает 24-битные (RGB) и 32-битные (CMYK) цвета. Формат использует методы сжатия без потерь: RLE и LZW. Формат позволяет сохранять сопроводительные подписи для идентификации изображений. В настоящее время он широко используется в издательских системах.

Глава 14



Виртуальная реальность

14.1. Виртуальная реальность и области ее применения

Под виртуальной реальностью понимают модель трехмерной окружающей среды, реализуемую компьютерными средствами, которая реалистично реагирует на взаимодействие с пользователем. Главной особенностью этой среды является создаваемая для пользователя иллюзия его присутствие в ней, возможность как путешествия в этой трехмерной среде (виртуальном мире), так и возможность взаимодействия с присутствующими в ней объектами, например, путем их захвата и перемещения. Практика показала, что ощущение присутствия в меньшей степени зависит от того, насколько естественно выглядят трехмерные изображения объектов виртуальной сцены, чем от того, насколько реалистично воспроизведение движений, и особенно от того, насколько реалистично моделируется взаимодействие пользователя с этой средой.

Минимум аппаратных средств, необходимых для реализации виртуальной реальности, включает в себя компьютер, монитор и мышь или джойстик, последние необходимы для передачи компьютеру сообщений о действиях пользователя, таких как его поступательное движение, поворот, касание объекта или его перемещение и т. д. При использовании мыши наше перемещение и взаимодействие с виртуальной средой ограничено в силу возможностей самой мыши.

Более совершенные системы виртуальной реальности могут включать в себя: виртуальные очки, виртуальные шлемы со стереоскопическими дисплеями и телефонами, устройства 3D-ввода, например мышь с пространственно управляемым курсором, цифровые перчатки, которые обеспечивают тактильную обратную связь с пользователем, бегущую дорожку и т. д.

В зависимости от имеющихся аппаратных средств и программного обеспечения оказывается доступной та или иная степень погружения в виртуальную реальность, начиная от наблюдения виртуального мира через окно экрана при взаимодействии с ним, и до практически полного эффекта присутствия при наличии виртуального шлема, цифровых перчаток, бегущей дорожки и т. д.

Касаясь принципа реализации виртуальной среды, нужно сказать, что само существо дела довольно просто. При перемещении пользователя в трехмерном виртуальном мире посредством команд, подаваемых пользователем с помощью мыши, джойстика или другого периферийного оборудования, просто изменяются координаты точки наблюдения, а следовательно, изменяется и рассматриваемая проекция. При перемещении же объекта изменяются его координаты.

В настоящее время системы виртуальной реальности применяются при создании тренажеров, компьютерных игр, обучающих программ, аттракционов, виртуальных галерей, в Интернете, в маркетинге и рекламе, в центрах подготовки и поддержки принятия решений, в ситуационных комнатах, в управлении технологическими процессами, в медицине и т. д.

14.2. Объектно-ориентированный подход

При создании виртуальной сцены используется объектно-ориентированный подход. Сцена состоит из объектов, каждый из них характеризуется набором свойств и событий, на которые он реагирует. Примером таких свойств являются: координаты, представляющие положение объекта в пространстве, его размеры, цвет, элементы формы, масса и т. д. Примером событий, на которые объект реагирует, могут быть: приложение силы к объекту, приводящее к изменению его положения в пространстве, разрушение объекта при его столкновении с другим объектом, изменение цвета при изменении спектрального состава источника освещения и т. д. В свою очередь, объекты сами могут генерировать события. Из объектов строится сцена (рис. 14.1).



Рис. 14.1. Пример сцены

Объекты могут быть также звуковыми, при этом они имеют и могут менять во времени свои координаты, благодаря чему достигаются стерео- и "окружающие" (surround) эффекты. Объекты могут быть как элементарными, так и составными, представляющими ту или иную композицию элементарных объектов. Примером составного объекта может служить сгенерированный трехмерный телевизор с размещенной на его экране живой видеотрансляцией.

Сцена описывается с помощью иерархической структуры (дерева) (см., например, рис. 14.2). Узлами этой структуры являются объекты, и она динамически перестраивается по мере того, как узлы-объекты добавляются, удаляются или заменяются. Трехмерные миры могут содержать сотни и тысячи объектов.



Рис. 14.2. Иерархическая структура сцены

Узлы делятся на группировочные, конечные и вспомогательные.

Группировочные узлы служат для объединения вложенных узлов в ветви "дерева сцены". Группировочные узлы называются родительскими узлами, а вложенные дочерними узлами (потомками).

Конечные узлы представляют собой геометрические примитивы, звуки, камеры и т. д. Эти узлы не могут иметь вложенных дочерних узлов, за исключением специфических вспомогательных.

Вспомогательные узлы — это специфические узлы, располагающиеся только в определенных местах. К ним относятся узлы-интерполяторы и некоторые другие узлы.

Для описания трехмерных сцен применяются специально разработанные для этой цели языки, например, язык VRML, в котором реализован объектно-ориентированный подход к представлению виртуальных сцен, а также BIFS и ряд других. Алгоритмы работы и функции BIFS тесно интегрированы с VRML.

Для большей конкретности дальнейшего изложения в качестве примера рассмотрим описание виртуальных сцен при создании 3D-сайтов в Интернете посредством алгоритмического языка VRML, придерживаясь терминологии, принятой в этом языке. В пользу такого выбора говорит и то, что VRML можно рассматривать как своего рода язык-посредник, поскольку описание виртуальной сцены, представленное на другом языке, обычно легко преобразуется в представление на VRML и обратно.

Файлы формата VRML, содержащие описание трехмерных сцен, имеют расширение wrl и представляют собой обычные текстовые файлы, которые можно создавать и редактировать посредством приложения Notepad (Блокнот) Windows.

14.3. Простейшие сцены в виртуальном мире

Рассмотрим на простейшем примере принцип построения кода, описывающего геометрический объект, например сферический примитив Sphere01, расположенный в начале координат (рис. 14.3).

```
#VRML V2.0 utf8
# Produced by 3D Studio MAX VRML97 exporter, Version 7, Revision 0,65
# Date: Mon Aug 14 19:56:56 2006
DEF Sphere01 Transform {
translation 0 0 0
children [
Shape {
  appearance Appearance {
  material Material {
    diffuseColor 0 1 0
  }
  }
  geometry Sphere { radius 28 }
}
]
                                                x
```

Рис. 14.3. Примитив Sphere01

В этом примере строки, следующие после символа #, являются комментариями, причем первая строка — это заголовок, одинаковый для всех файлов VRML.

Текст VRML-кода состоит из операторов (statements). К ним относятся операторы-узлы (node statements), представляющие собой объекты. Операторы-узлы начинаются с необязательного слова DEF (от англ. *definition* — определение), после которого следует имя узла (объекта), в нашем примере Sphere01. К обязательным элементам узла относятся название его типа, в нашем примере Transform (Преобразование), и тело узла, которое заключается в фигурные скобки. Тело узла состоит из полей (свойств). Каждое *поле (fields)* имеет свое имя и значение. В том случае, если поле представляет собой массив, т. е. имеет несколько значений, то эти значения заключаются в квадратные скобки. В качестве значений полей могут быть новые узлы, имеющие свои поля.

В рассматриваемом примере в теле узла типа Transform (Преобразование), определяющем параметры преобразования геометрического объекта Sphere01, имеется поле translation, которое задает координаты положения центра сферы, а также поле children (потомки), которое, по определению, может иметь, но в данном примере не имеет, набор значений, заключаемых в квадратные скобки. В нашем примере единственным значением поля children (потомки) является узел Shape (Форма).

Узел Shape определяет внешний вид объекта и имеет два поля: appearance (вид) и geometry (геометрия).

Значением поля appearance является узел с именем Appearance, у которого имеется единственное поле material (материал), определяющее материал геометрического объекта.

Значением поля "материал" является узел Material, имеющий единственное поле diffuseColor (диффузный цвет), которое определяет цвет диффузного рассеяния света на геометрическом объекте, в нашем примере на сфере. Значение цвета задается тремя компонентами вектора в колориметрической системе RGB. При этом имеется одна особенность. Интенсивность каждого цвета задается вещественной величиной, которая может изменяться не в привычных пределах от 0 до 255, а от 0 до 1. Например, красному цвету максимальной интенсивности будет соответствовать код (1, 0, 0), серому цвету средней интенсивности — (0,5, 0,5, 0,5), а белому цвету максимальной интенсивности — (1, 1, 1). В нашем примере цвет задан числами 0 1 0, что соответствует зеленому цвету.

И, наконец, поле geometry узла Shape определяет форму объекта. В рассматриваемом примере значением поля geometry является узел Sphere, который определяет геометрический примитив в виде сферы, имеющий единственный параметр radius (радиус), равный 28 метрам.

Аналогичным образом могут быть описаны другие геометрические примитивы: параллелепипед, цилиндр, конус. В этом случае вместо узла sphere, описывающего этот объект, следует записать код другого примитива.

Так в случае параллелепипеда код его узла вох01 имеет вид:

```
DEF Box01 Transform {
translation 8.591 0 2.749
children [
Transform {
translation 0 18.56 0
```

```
children [
Shape {
   appearance Appearance {
     material Material {
        diffuseColor 0.3451 0.3451 0.8824
     }
   }
   geometry Box { size 60.48 37.11 54.3 }
}
]
```

Здесь параметр size представляет собой геометрические размеры граней параллелепипеда в метрах: 60.48 — размер грани, параллельной оси x; 37.11 — размер грани, параллельной оси y; и 11 54.3 — размер грани, параллельной оси z.

В случае цилиндра код узла Sphere { radius 28 } должен быть заменен на код узла цилиндра.

Так в случае цилиндра код его узла Cylinder01 имеет вид:

```
DEF Cylinder01 Transform {
translation -15.46 0 1.718
children [
Transform {
  translation 0 29.21 0
  children [
    Shape {
      appearance Appearance {
        material Material {
          diffuseColor 0.1098 0.1098 0.6941
        }
      }
      geometry Cylinder { radius 30.65 height 58.42 }
    }
] }
]
}
```

Здесь параметр radius представляет собой радиус, равный 30.65 метра, а параметр height — высоту цилиндра, равную в рассматриваемом примере 58.42 метра.

В случае конуса код его узла Cone01 имеет вид:

```
DEF Cone01 Transform {
  translation -7.904 0 -1.718
  children [
  Transform {
   translation 0 54.64 0
```

```
children [
  Shape {
    appearance Appearance {
       material Material {
         diffuseColor 0.3451 0.5608 0.8824
        }
        geometry Cone { bottomRadius 71.62 height 109.3 }
    }
] }
```

Параметр bottomRadius представляет собой радиус основания, в данном примере равный 71.6 метра, а параметр height — высоту, равную 109.3 метра.

Обратим внимание на то, что при описании объектов типа параллелепипед, цилиндр и конус поля translation и children, в соответствии со стандартом VRML, повторяются дважды, при этом смещения, задаваемые каждым из полей, суммируются. В данном случае это особого смысла не имеет. Эта особенность используется при описании более сложных сцен, чем сцены, приведенные в примерах.

14.4. Более сложные сцены в виртуальном мире

Обычно сцены бывают более сложными, чем рассмотренные выше. Сложность сцен может быть обусловлена большей сложностью поверхности трехмерных объектов, чем те примитивы, которые были рассмотрены в предыдущем разделе; наличием в сцене не одного, а большого количества объектов, а также рядом других факторов.

Рассмотрим вначале способ описания сложных объектов. В соответствии с принятым стандартом в VRML для описания поверхности трехмерного геометрического объекта применяют векторную полигональную модель, т. е. используют набор граней, аппроксимирующих ее с той или иной степенью точности. На рис. 14.4 приведен пример такой модели.

Напомним, что для описания поверхностей объектов создаются два массива. В первом массиве записываются координаты вершин многоугольников, формирующих эти грани, а во втором массиве для каждой грани записываются номера вершин, из которых она сформирована. Далее приводится код, описывающий геометрический объект Pyramid01 (Пирамида), расположенный в начале координат и окрашенный в красный цвет.

```
#VRML V2.0 utf8
# Produced by 3D Studio MAX VRML97 exporter, Version 7, Revision 0,65
# Date: Tue Aug 15 14:35:51 2006
DEF Pyramid01 Transform {
  translation 0 0 0
  children [
```

```
Shape {
appearance Appearance {
material Material {
diffuseColor 1 0 0
}
}
# Начало поля geometry (геометрия) узла Shape, которое определяет
# форму объекта.
geometry DEF Pyramid01-FACES IndexedFaceSet {
coord DEF Pyramid01-COORD Coordinate { point [
0 42.61 0, -31.62 0 27.84, 31.62 0 27.84, 31.62 0 -27.84, -31.62 0 -27.84,
0 0 01
}
coordIndex [
0, 1, 2, -1, 0, 2, 3, -1, 0, 3, 4, -1, 0, 4, 1, -1, 1, 5, 2, -1,
2, 5, 3, -1, 3, 5, 4, -1, 4, 5, 1, -1]
}
# Конец поля geometry узла Shape, которое определяет форму объекта.
}
]
}
                                                             б
                   а
```

Рис. 14.4. Векторная полигональная модель представления трехмерных изображений: *а* — оболочка объекта; *б* — полигональная сетка

Сравнивая коды, описывающие сферу, которые приведены в начале предыдущего раздела, с кодами, описывающими пирамиду, видим, что они отличаются только строением поля geometry узла shape. В первом случае браузеру указывается, что нужно построить сферу с заданным радиусом. Все остальное, т. е. расчет полигональной сетки, он делает самостоятельно. Во втором случае браузеру задаются координаты вершин полигональной сетки полем point, и, кроме того, полем coordIndex указываются индексы (номера) вершин, которыми формируется каждая грань. При записи координат вершин, как это видно из примера, каждая из координат отделяется от соседней координаты пробелом, а группы, состоящие из трех координат, отделяются друг от друга запятыми. При записи индексов (номеров) координат в поле coordIndex каждый индекс отделяется от соседнего индекса запятой с пробелом, причем группа индексов, описывающих грань (полигон), завершается числом (меткой) –1.

При внимательном рассмотрении приведенного кода читатель обнаружит, что поле geometry включает в себя два узла.

Рассмотрим теперь описание сцен виртуального мира, которые включают в себя не один, а несколько объектов. В этом случае, в отличие от предыдущего, в текст кодов включаются узлы, описывающие эти объекты. Если эти объекты между собой никак не связаны, то коды соответствующих узлов включаются независимо друг за другом, как это показано в приводимом далее примере, представляющем сцену, включающую в себя зеленый куб и красную сферу.

```
#VRML V2.0 utf8
# Produced by 3D Studio MAX VRML97 exporter, Version 7, Revision 0,65
# Date: Tue Aug 15 16:44:15 2006
# Описание куба
DEF Box01 Transform {
translation 66.32 0 0.3436
rotation 0 1 0 -0.7808
children [
Transform {
translation 0 0 0
children [
Shape {
appearance Appearance {
material Material {
diffuseColor 0 1 0
}
}
geometry Box { size 40.7 40.7 40.7 }
}
] }
```

]

```
}
# Описание сферы
DEF SphereO1 Transform {
translation -40.89 0 -2.405
children [
Shape {
appearance Appearance {
material Material {
diffuseColor 1 0 0
}
}
geometry Sphere { radius 23.83 }
]
]
```

В этом примере следует обратить внимание на то, что при описании куба появилось новое поле rotation со значением 0 1 0 -0.7808. Это поле показывает, как повернут куб. Первые три значения поля представляют собой вещественные числа, которые задают вектор нормали, относительно которого происходит вращение, последнее число представляет собой угол поворота в радианах.

Коды, описывающие другие рисованные геометрические объекты, могут быть легко получены следующим образом. Сначала в редакторе 3ds Max создается объект, далее этот объект экспортируется в формате VRML, а затем его код читается с помощью Windows-приложения Notepad (Блокнот).

14.5. Объекты, обеспечивающие путешествие в виртуальном мире

Для того чтобы видеть виртуальный мир и перемещаться в нем, необходимо использовать дополнительные объекты, обеспечивающие освещение геометрических объектов (источники света), их наблюдение (всевозможные камеры) и, наконец, возможность перемещения в виртуальном пространстве (виртуальный путешественник).

В VRML предусмотрен ряд объектов — источников света, например, spot (Точечный), Direct (Направленный), Omni (Всенаправленный). Каждый из типов источников обладает своими свойствами, определяющими область его применения.

Рассмотрим в качестве примера кодовое представление источника света (объекта) omni (Всенаправленный):

```
DEF Omni01 PointLight {
    intensity 2.1
```

}

```
color 1 1 0
location -78.69 0 -96.56
on TRUE
radius 337.6
```

В приведенном листинге код оmni01 является именем узла, PointLiht (Tovevный источник света) — типом узла, а затем следуют пять полей, данные которых представляют характеристики (свойства) источника света. Значение поля intensity (интенсивность) задает интенсивность источника, которая в данном случае равна 2.1. Значение поля color (цвет) задает цвет источника света в колориметрической системе RGB, который в приведенном примере является желтым (смесь красного и зеленого). Еще раз обратим внимание на то, что в VRML интервал интенсивностей красного, зеленого и синего задается как 0—1, а не как 0—255. Значение поля location (местоположение) задает декартовы координаты источника света. Значение поля оп (включено) определяет, включен (TRUE) или выключен (FALSE) источника света. И, наконец, значение поля radius (радиус) задает область действия источника света. По традиции этот объект помещают в начале кода, описывающего сцену, хотя он может быть размещен в любом другом месте текста.

При перемещениях (путешествиях) в виртуальном мире удобно использовать несколько камер, что дает возможность путем простого переключения мгновенно изменять точку, с которой ведется наблюдение, а также угол зрения. Объект Camera (Камера) имеет следующее кодовое представление:

```
DEF Camera01 Viewpoint {
   position 3.093 131 -0.2396
   orientation 0.8146 -0.09662 0.572 -1.491
   fieldOfView 0.6024
   description "Camera01"
}
```

В приведенном коде camera01 является именем узла, описывающего объект, a Viewpoint (Точка зрения) — типом узла. Затем следуют четыре поля, описывающие свойства камеры. Значение поля position (позиция) задает координаты камеры. Значение поля orientation (ориентация) — ориентацию камеры относительно системы координат. Первые три значения поля представляют собой вещественные числа, которые задают вектор, относительно которого происходит вращение, последнее число представляет собой угол поворота в радианах. Значение поля fieldofview (поле зрения) задает поле зрения в радианах. Так значению 0.6024 соответствует поле зрения в 34,5°. В поле description (описание) записывается имя камеры. Имя камеры необходимо при переключениях с камеры на камеру для того, чтобы указывать браузеру, на какую камеру он должен переключиться.

Рассмотрим теперь вспомогательный объект NavInfo, который регламентирует навигацию в виртуальном мире. По существу, этот объект описывает свойства путешественника, перемещающегося в виртуальном мире. Объект NavInfo имеет следующее кодовое представление:

```
DEF NavInfo01 NavigationInfo {
   avatarSize [0.25, 1.6, 0.75]
   headlight FALSE
   speed 1
   type "WALK, ANY"
   visibilityLimit 0
}
```

В приведенном коде NavInfo01 является именем узла, описывающего вспомогательный объект, а NavigationInfo (Информация о навигаторе) — типом узла. Затем следуют пять полей, описывающие свойства путешественника. Значение поля avatarSize задает размеры путешественника (рис. 14.5). Первое число (A) представляет допустимое расстояние, на которое путешественник может приближаться к объектам, т. е. нечто вроде радиуса его корпуса. Второе число (B) — это его рост. Третье число (C) определяет максимальный размер объектов, через которые путешественник может перешагивать. Задание этих значений необходимо для того, чтобы перемещение по виртуальному миру приближалось бы к реальному (например, чтобы путешественник не мог бы пролезать через щели).



Рис. 14.5. Схематичное представление путешественника в виртуальном мире

Поскольку в некоторых частях виртуального мира свет может отсутствовать, то путешественник снабжен собственным источником света, который закрепляется у него на голове подобно тому, как закрепляется лампа на каске у шахтера. Состояние этого источника задается полем headlight, которое может принимать два значения: TRUE — включен, FALSE — выключен.

Поле speed задает скорость перемещения в виртуальном мире в метрах в секунду.

Поле type задает тип перемещения в виртуальном мире. Предусмотрено несколько типов перемещений:

- "WALK" движение пешком по земле. В этом случае высота точки наблюдения, из которой смотрит камера, определяется ростом посетителя виртуального мира;
- "FLY" полет. При полете на посетителя не действует сила притяжения Земли, и он может совершать любые перемещения;
- "ЕХАМІΝЕ" ЭТОТ ТИП перемещения в виртуальном пространстве предназначен для обзора отдельных объектов. При этом виде движения камера, используемая для наблюдения, вращается вокруг выбранного объекта под управлением пользователя, что дает возможность осмотреть объект со всех сторон;
- "АNY" позволяет пользователю использовать любой тип движения;
- "NONE" запрещает любой вид движения.

По умолчанию полю type присваивается значение "WALK, ANY", при этом все виды движения разрешены.

Поле visibilityLimit устанавливает предел видимости. В том случае, если значение этого поля установлено равным нулю, видимость не ограничивается.

Объект NavInfo принято помещать в самое начало файла.

14.6. Объекты, обеспечивающие взаимодействие пользователя с объектами виртуального мира

В реальной жизни помимо перемещения в пространстве мы постоянно взаимодействуем с окружающими нас предметами: открываем и закрываем двери, передвигаем стулья, перекладываем книги на письменном столе и т. д. Для осуществления аналогичного взаимодействия с объектами виртуального мира используют специально для этого разработанные вспомогательные объекты. Рассмотрим некоторые из них.

Объект PlaneSensor (Датчик плоского перемещения) обеспечивает перемещение геометрического объекта в плоскости *ху* посредством наведения на него курсора мыши и последующего перетаскивания в нужную точку виртуального пространства. Этот объект имеет следующее кодовое представление:

```
DEF PlaneSensor01-SENSOR PlaneSensor {
  enabled TRUE
  minPosition -150 -33.82
  maxPosition 100 100
  offset -4.467 0 9.966
}
```

В приведенном коде PlaneSensor01 является именем узла, описывающего объект, а SENSOR PlaneSensor — типом узла. Поле enabled (разрешено) в зависимости от присвоенного ему значения либо разрешает (TRUE), либо запрещает (FALSE) перемещение геометрического объекта. Поля minPosition (минимальная позиция) и maxPosition (максимальная позиция) устанавливают разрешенный диапазон перемещения геометрического объекта. Поле offset задает исходное положение геометрического объекта по координатам X и Y, которое должно быть установлено равным тому значению, которое установлено для перемещаемого геометрического объекта в поле translation. Узел PlaneSensor01 должен быть помещен в поле children перемещаемого геометрического объекта. И, наконец, в конец файла VRML необходимо поместить код

```
ROUTE PlaneSensor01-SENSOR.translation changed TO Name.translation
```

где Name — имя узла перемещаемого геометрического объекта, например, Cylinder01, если узел перемещаемого объекта носит это имя. Далее приводится пример кода для рассмотренного случая. В целях сокращения текста при описании объекта вох01 пропущенные строки кода заменены многоточием.

#VRML V2.0 utf8 # Produced by 3D Studio MAX VRML97 exporter, Version 7, Revision 0,65 # Date: Wed Aug 16 17:57:25 2006

```
# Описание геометрического объекта Cylinder01
DEF Cylinder01 Transform {
translation -4.467 0 9.966
children [
Transform {
translation 0 27.49 0
children [
Shape {
appearance Appearance {
material Material {
diffuseColor 0.5529 0.02745 0.2275
}
}
geometry Cylinder { radius 14.87 height 54.98 }
}
# Описание объекта PlaneSensor
DEF PlaneSensor01-SENSOR PlaneSensor {
```

enabled TRUE minPosition -150 -33.82 maxPosition 100 100 offset -4.467 0 9.966

```
]
}
# Описание геометрического объекта Box01
DEF Box01 Transform {
...
}
ROUTE PlaneSensor01-SENSOR.translation changed TO Cylinder01.translation
```

Объект CylinderSensor (Датчик цилиндрического поворота) обеспечивает поворот геометрического объекта вокруг геометрической оси у посредством наведения на него курсора мыши и его последующего перетаскивания в горизонтальном направлении. Этот объект имеет следующее кодовое представление:

```
DEF CylinderSensor01-SENSOR CylinderSensor {
  enabled TRUE
  maxAngle 0
  minAngle -6.28
  offset -0.4861
}
```

В приведенном коде cylinderSensor01 является именем узла, описывающего объект, а SENSOR CylinderSensor — типом узла. Поле enabled (разрешено) в зависимости от присвоенного ему значения либо разрешает (TRUE), либо запрещает (FALSE) поворот геометрического объекта. Поля maxAngle (максимальный угол) и minAngle (минимальный угол) устанавливают разрешенный диапазон поворота геометрического объекта. Поле offset задает исходное значение угла, на который повернут геометрический объект вокруг оси у. Это значение должно быть установлено равным тому значению, которое установлено для поворачиваемого геометрического объекта. Узел CylinderSensor01 должен быть помещен в поле children поворачиваемого объекта. И, наконец, в конец файла VRML необходимо поместить код

```
ROUTE CylinderSensor01-SENSOR.rotation_changed TO Name.rotation
```

Здесь **Name** — имя узла поворачиваемого геометрического объекта вокруг вертикальной оси его локальной системы координат, например, Box01. Далее приводится пример кода для рассмотренного случая. В целях сокращения текста при описании объекта Box02 пропущенные строки кода заменены многоточием.

```
#VRML V2.0 utf8
# Produced by 3D Studio MAX VRML97 exporter, Version 7, Revision 0,65
# Date: Thu Aug 17 12:16:14 2006
```

```
# Описание геометрического объекта Box01
DEF Box01 Transform {
translation 50.64 0 -0.7045
```

```
rotation 0 1 0 -0.4861
children [
Transform {
translation 0 15.46 0
children [
Shape {
appearance Appearance {
material Material {
diffuseColor 0.6039 0.8431 0.898
}
}
geometry Box { size 30.93 30.93 32.99 }
}
1 }
# Описание объекта CylinderSensor01
DEF CylinderSensor01-SENSOR CylinderSensor {
    enabled TRUE
    maxAngle 0
    minAngle -6.28
    offset -0.4861
    }
1
}
# Описание геометрического объекта Вох02
DEF Box02 Transform {
  . . .
}
```

```
ROUTE CylinderSensor01-SENSOR.rotation_changed TO Box01.rotation
```

Объект spheresensor (Датчик сферического поворота) обеспечивает поворот геометрического объекта вокруг центра его локальной системы координат на произвольный угол путем наведения на него курсора мыши и его последующего перетаскивания в горизонтальном и вертикальном направлениях. Этот объект имеет следующее кодовое представление:

```
DEF SphereSensor01-SENSOR SphereSensor {
   enabled TRUE
   offset 0 1 0 -0.4861
}
```

В приведенном коде SphereSensor01 является именем узла, описывающего объект, а SENSOR SphereSensor — типом узла. Поле enabled (разрешено) в зависимости

от присвоенного ему значения либо разрешает (TRUE), либо запрещает (FALSE) поворот геометрического объекта. Поле offset задает исходное значение углов, на который повернут геометрический объект вокруг оси у. Это значение должно быть установлено равным тому значению, которое установлено для поворачиваемого геометрического объекта в поле rotation. Узел SphereSensor01 должен быть помещен в поле children поворачиваемого объекта. И, наконец, в конец файла VRML необходимо поместить код

```
ROUTE SphereSensor01-SENSOR.rotation changed TO Name.rotation
```

Здесь **Name** — имя узла поворачиваемого геометрического объекта вокруг центра его локальной системы координат, например, Box01. Далее приводится пример кода для рассмотренного случая. В целях сокращения текста, как и в предыдущем случае, при описании объекта Box02 пропущенные строки кода заменены многоточием.

```
#VRML V2.0 utf8
# Produced by 3D Studio MAX VRML97 exporter, Version 7, Revision 0,65
# Date: Thu Aug 17 12:16:14 2006
# Описание геометрического объекта Вох01
DEF Box01 Transform {
translation 50.64 0 -0.7045
rotation 0 1 0 -0.4861
children [
Transform {
translation 0 15.46 0
children [
Shape {
appearance Appearance {
material Material {
diffuseColor 0.6039 0.8431 0.898
}
}
geometry Box { size 30.93 30.93 32.99 }
}
1 }
# Описание объекта SphereSensor01
DEF SphereSensor01-SENSOR SphereSensor {
enabled TRUE
#maxAngle 0
#minAngle -6.28
offset 0 1 0 -0.4861
}
```

```
]
}
# Описание геометрического объекта Box02
DEF Box02 Transform {
...
}
```

ROUTE SphereSensor01-SENSOR.rotation_changed TO Box01.rotation

14.7. Алгоритмические языки, применяемые в системах виртуальной реальности

Язык моделирования виртуальной реальности (Virtual Reality Modeling Language, VRML) представляет собой файловый формат для описания интерактивных трехмерных объектов и миров. Как уже было отмечено, алгоритмический язык VRML был разработан для использования в Интернете, корпоративных сетях и на локальных клиентских системах. Этот язык также является универсальным форматом обмена для интегрированной трехмерной графики и мультимедиа. В настоящее время VRML используется в образовательном процессе, исследованиях и прочих областях, где особенно ценится высокая детализация.

Его можно применять в различных прикладных областях, например, в инженерной и научной визуализации, в образовании, мультимедийных презентациях, для развлечений, а также в Web-страницах и разделяемых виртуальных мирах. Ранее на примере алгоритмического языка VRML было показано, каким образом может быть описана сцена в виртуальном мире. В последнее время язык VRML вытесняется форматом X3D, который является его усовершенствованным вариантом.

Однако, как уже было отмечено, в настоящее время разработан целый ряд алгоритмических языков для этой цели, которые отличаются от него теми или иными достоинствами. Перечислим некоторые из них.

Алгоритмический язык Python является простым и, в то же время, мощным интерпретируемым объектно-ориентированным языком программирования. Он предоставляет собой структуры данных высокого уровня и использует динамический контроль типов, что делает его идеальным языком для быстрого написания различных приложений, в том числе приложений виртуальной реальности, работающих на большинстве платформ.

Алгоритмический язык BIFS — это двоичный язык, предназначенный для описания объектов, классов объектов и сцен, который можно характеризовать как расширение языка C++. BIFS используется для описания виртуальных сцен. С помощью его команд можно анимировать объекты, менять их координаты, размеры, свойства, задавать поведение, реакции на воздействия пользователя, менять свойства среды,

изменять и обновлять сцену, выполнять 2D-геометрические построения и т. п. Поскольку язык двоичный, он весьма компактен и быстр в интерпретации. Многие концепции BIFS, как уже отмечалось, позаимствованы у VRML, и сейчас MPEG и Web 3D Consortium продолжают работу по сближению MPEG-4 и VRML

Назовем еще следующие языки программирования и программные средства виртуальной реальности: 3DMLW (3D Markup Language for Web), O3D, TurnTool и т. д. Так, например, достаточно популярный в настоящее время инструмент TurnTool разработан для совместной работы с такими приложениями трехмерного моделирования, как 3ds Max, Autodesk Viz, а также с рядом других приложений аналогичного назначения и устанавливается в них в качестве плагина. Одной из отличительных особенностей файлов, создаваемых TurnTool, является их сравнительно небольшая величина по сравнению с файлами, которые создает VRML, и большая гибкость. Приложение TurnTool состоит из двух компонентов: компонента TurnToolBox и компонента TurnTool Viewer. Компонент TurnToolBox предназначен для генерации файлов с расширением tnt при экспорте 3D-сцены из трехмерного редактора, например, 3ds Max в формат TNT. Компонент TurnTool Viewer предназначен для визуализации сгенерированных файлов в браузере.

14.8. Имитация искусственного интеллекта

Реалистичность виртуального мира в значительной мере возрастает в том случае, если в нем присутствуют "разумные" существа, которые пытаются взаимодействовать с нами, проявляя при этом свой "разум", а также управляемые ими объекты. Этими существами могут быть: аватары, 3D-модели животных и насекомых. Простейшим примером, иллюстрирующим эту ситуацию, может служить компьютерная игра, в которой имеют место всевозможные сражения, поиски, преследования и т. д.

Проблема, с которой при этом встречаются разработчики виртуального мира, это имитация интеллекта этих существ. Для этой цели используют конечные автоматы, управляемые окружающей средой [70]. Рассмотрим применение конечных автоматов на примере преследования жертвы (игрока) виртуальным комаром, который стремится уклониться от укуса и, в свою очередь, прибить комара. С этой целью обратимся к рис. 14.6, на котором показан конечный автомат.

Рассмотрим отдельные состояния этого автомата:

- □ *состояние преследования* заключается в том, что приложение сравнивает текущие координаты жертвы и комара и в зависимости от этого изменяет координаты комара таким образом, чтобы расстояние между ними уменьшить;
- □ *состояние атаки* заключается в том, что приложение сравнивает текущие координаты жертвы и нападающего комара и при их совпадении совершает "укус" жертвы;
- □ *состояние уклонения* состоит в том, что приложение сравнивает текущие координаты жертвы и комара и в зависимости от этого изменяет координаты комара таким образом, чтобы расстояние между ними увеличить;



Рис. 14.6. Конечный автомат

состояние шаблона заключается в том, что при этом состоянии комар совершает движения по одной из нейтральных траекторий в стороне от жертвы, выбор которой осуществляется по случайному закону.

Состояний может быть и больше, но мы ограничимся этими. Переходом из одного состояния в другое управляет блок управления состоянием на основании информации, получаемой в результате измерения координат нападающего комара и жертвы. В том случае, если расстояние между жертвой и комаром велико и жертва удаляется от него, включается состояние преследования. Если жертва не реагирует на нападение, то при сближении комара и жертвы автомат переходит в состояние атаки, которое завершается "укусом". В противном случае, если жертва начинает движение навстречу комару, автомат переходит в состояние уклонения и комар стремится избежать встречи с жертвой. Состояние шаблона предназначено для того, чтобы разнообразить поведение комара, придать ему некоторую индивидуальность, и время от времени команда на его использование поступает от датчика случайных чисел.

Несмотря на примитивность описанного алгоритма, создается полная иллюзия разумного поведения комара, нападающего на жертву, в чем читатель легко убедится, реализовав этот алгоритм в несложной программе.

Перейдем теперь к краткому описанию некоторых практических применений систем виртуальной реальности.

14.9. Тренажеры

Приведем краткие описания некоторых существующих тренажеров.

Тренажер NASA, имитирующий работу аэропортов

Тренажер предназначен как для использования при проектировании новых аэропортов, так и для тренинга персонала, обслуживающего аэропорты, уже существующие. Тренажер моделирует практически все аспекты работы большого аэропорта: прилеты и вылеты, обработку грузов, управление всем хозяйством аэропорта. Специалисты считают, что только посредством применения тренажера можно выяснить все слабые места любого аэропорта.

Разработанный тренажер имитирует виртуальную круговую 3D-панораму, открывающуюся с главной башни управления, а также оборудование, которое устанавливается в таких башнях. Особенность тренажера заключается в том, что он показывает проектировщикам или же авиадиспетчерам вид либо на проектируемый аэропорт, либо на один из существующих. Достигается это путем введения в компьютер в первом случае чертежей проектируемых аэропортов, во втором случае фотографий и чертежей реальных аэропортов. Виртуальные самолеты, которые прилетают и улетают из аэропорта, реализуются посредством самостоятельных тренажеров, каждым из которых управляет реальный пилот с отдельного компьютера или из самостоятельного тренажера, имитирующего кабину лайнера. Тренажер самолета отрабатывает множество различных параметров его движения в воздухе и на земле. Виртуальный самолет может включать навигационные огни, заправляться и принимать виртуальных пассажиров.

В свою очередь, диспетчеры, находящиеся в башне управления, получают всю информацию от виртуальных локаторов и видеокамер наблюдения, ведут по радио переговоры с пилотами, техниками на поле, отслеживают перемещение грузов по аэродрому, работу тягачей и заправщиков.

Тренажер, имитирующий взлет шаттла

В США разработан тренажер, имитирующий взлет шаттла. Разработка обошлась 60 млн долларов. В настоящее время он используется как аттракцион в космическом центре Кеннеди. Астронавты NASA, опробовавшие на себе новый тренажер, отмечают реалистичность испытываемых ощущений.

Виртуальный тренажер бензопилы, основанный на принципе смешанной реальности

Этот тренажер создан специалистами германского института компьютерной графики Фраунгофера (Fraunhofer IGD) по заказу компании Dolmar, крупнейшего в мире производителя бензопил. Особенность этого тренажера состоит в том, что в нем объединена виртуальная реальность и физический тренажер, моделирующий саму пилу, а также распиливаемое бревно и козлы.

Тренажер ориентирован на использование не только зрения, но и чувства осязания, поскольку он сделан из настоящей бензопилы, из которой удалили двигатель и цепь. Вместо этого в состав кибернетической пилы вошли электроника, динамики для имитации звука работающей пилы и вибропривод, имитирующий ее вибрации. Распиливаемое бревно, выполненное из прозрачного пластика, устанавливается на специальных козлах. Начиная работу, ученик берет в руки бензопилу и дергает за рукоятку стартера, при этом пила начинает вибрировать. На цепи работающей кибернетической бензопилы зажигаются светодиоды. Видеокамеры, смонтированные в бревне и козлах, фиксируют их, при этом электроника определяет точное положение инструмента. Обучающийся, держащий в руках тренажер, видит на экране цепь виртуальной пилы и нарисованное бревно. Лесоруб нажимает пилой на пластиковое бревно, и оно, имитируя сопротивление дерева, медленно опускается вниз. Вибрация инструмента и издаваемый при этом звук усиливают иллюзию действительности. Обучающийся при этом смотрит на большой экран, где пила строго в соответствии с его движениями уходит все глубже в бревно, в котором появляется разрез, и из которого сыпятся стружка и щепки. После того как часть сосны отпилена, система показывает ее крупное изображение на экране и вес спиленной части. Так как тренажер внешне и по весу не отличается от настоящей бензопилы, а имитатор бревна сопротивляется распиливанию, как настоящее бревно, то переход обучающегося от тренажера к настоящей работе оказывается максимально плавным и легким, считают германские инженеры.

Тренажер "комната виртуальной реальности"

В "Центре приложений виртуальной реальности" (Virtual Reality Applications Center) университета Айовы (Iowa State University) создана комната виртуальной реальности. Изображения виртуального мира этой системы, которую разработчики назвали C6, состоят из 100 млн пикселов, окружающих пользователя со всех сторон, в том числе и на полу, создавая трехмерное изображение. Кроме того, в системе обеспечено создание многоканального звука. Внутренний размер помещения составляет $3\times3\times3$ метра, стены которого представляют собой экраны. На эти экраны выводятся 3D-изображения, которые наблюдаются через специальные очки. Система предназначена для использования при научных исследованиях. C6, например, позволяет ученым наблюдать вблизи гигантские белковые молекулы, а при необходимости перемещать их. Комната позволяет моделировать внутренности сложных машин и наблюдать их работу. С ее помощью была смоделирована функционирующая трехмерная клетка, которая была использована в учебном процессе при обучении биологов.

14.10. Компьютерные игры

Для примера рассмотрим игровое виртуальное приложение Nintendogs, разработанное сравнительно недавно компанией Nintendo для ее игровой консоли, которое вызвало настоящий ажиотаж в магазинах Японии.

Это приложение имитирует жизнь и поведение виртуального 3D-щенка, представленного в 3D-среде, с которым может общаться пользователь. Это общение включает в себя кормление и дрессировку виртуального щенка, уборку за ним, игру с ним. Чем больше вы общаетесь с ним, тем больше он к вам привыкает. Щенок слышит своего хозяина благодаря микрофону, встроенному в консоль, он реагирует на его поглаживания стилусом, например, переворачиваясь на спинку и подставляя живот. С помощью стилуса ему можно бросить мячик или летающую тарелку, а он побежит ее ловить, и потом принесет вам назад. Чем больше хозяин общается со щенком и чем больше играет с ним, тем преданней будет собачка. Если же хозяин будет лениться, то щенок останется в дальнем углу своего виртуального мирка и не подойдет на зов.

Виртуальные щенки способны к обучению. Щенок может запомнить множество команд и со временем наращивает свои навыки, становится как бы умнее. Однако хозяину виртуального щенка нужно учесть, что виртуальный персонаж чувствует время, и потому его нельзя заставить заниматься обучением слишком часто, он устает.

Консоли Nintendo DS обладают беспроводной связью и чувствуют друг друга за 30 метров, поэтому если два обладателя этой виртуальной игры встречаются вместе, то между консолями устанавливается связь, и щенки двух хозяев начинают бегать и резвиться вместе. Игра предоставляет на выбор 18 различных пород веселых щенков, которые никогда не вырастут и никогда не умрут, даже если их не кормить. В этом случае они просто останутся голодными и недовольными.

14.11. Периферийные устройства

Шлем виртуальной реальности

Шлем виртуальной реальности (HMD — Head Mounted Displays) представляет собой устройство, предназначенное для отображения виртуального окружения пользователя, которое, как правило, включает в себя:

- два дисплея, для правого и левого глаза, выполненных на специальных жидкокристаллических панелях и обеспечивающих вывод стереоизображения;
- систему виртуальной ориентации (трекинг), которая отслеживает и передает в компьютер линейные и угловые координаты положения головы пользователя;
- 🗖 стереотелефоны.

Некоторые типы шлемов не включают в свой состав систему виртуальной ориентации, и тогда она дополняет их как отдельное устройство. Длительное время серьезную проблему составляло невысокое разрешение дисплеев, используемых в этих устройствах, которое равнялось 240×320 пикселов, что сильно снижало качество выводимых на экраны изображений. Однако недавно в связи с разработкой американской компанией Kopin технологии Kopin CyberDisplay VGA эта проблема была решена, и разрешение микродисплеев, используемых в шлемах виртуальной реальности, было доведено до 640×480 пикселов при глубине цвета 8 бит на пиксел. Мощность, потребляемая такими дисплеями, составляет 70 мВт. В качестве примера приведем технические характеристики шлема GVD510-3D:

- □ разрешение каждого экрана: 640×480 пикселов;
- размер матрицы микроэкрана: 11 мм;
- □ диагональный угол обзора: 28°;

- □ входной видеосигнал: NTSC/PAL;
- потребление энергии: менее 1,2 Вт;
- вес: 200 грамм;
- □ встроенные стереонаушники.

Существуют также варианты шлемов с возможностью использования технологии Augmented Reality (наложенной виртуальной реальности), при использовании которых зрительно совмещаются реальные и виртуальные объекты.

Системы трекинга

К трекинговым системам (tracking systems) относятся системы отслеживания положения самого пользователя и его головы (трекеры движения). Эти системы необходимы для правильной генерации изображения для глаз пользователя системой виртуальной реальности в зависимости от положения его головы. Использование таких систем позволяет интерактивно взаимодействовать с виртуальной графической сценой. Трекинговые системы используют различные датчики, такие как, например: гироскопические, лазерные, оптические, ультразвуковые или электромагнитные.

Джойстик и руль автомобиля

Игровые манипуляторы, руль и джойстик, применяют в компьютерных играх и тренажерах, предназначенных для отработки техники управления автомобилем. В наиболее продвинутых моделях предусмотрено наличие обратной связи (Force Feedback), благодаря которой достигаются эффекты, имитирующие все детали автомобильной гонки. Они позволяют не только управлять самим объектом, но также и скоростью изменения его положения. Связь с компьютером осуществляется через порт USB. В настоящее время джойстики производятся многими широко известными в мире фирмами-производителями: Logitech, Thrustmaster, Saitek, Sega, Sony PlayStation, Genius, DVTech и др.

Аналогичные периферийные устройства выпускаются для использования в компьютерных играх, имитирующих управление самолетами и вертолетами.

Перчатки виртуальной реальности (VR Gloves)

В качестве примера рассмотрим игровую перчатку P5 Glove от компании Essential Reality. Перчатка Essential Reality P5 Glove была разработана для компьютерных игр. Она имеет следующие характеристики: интерфейс USB 1.1, питание по шине USB, шесть степеней свободы, пять независимых датчиков (по одному датчику на каждый палец), погрешность определения координат — 6,4 мм, погрешность определения угла поворота — 4°.

Перчатка P5 Glove выполнена в виде пластикового корпуса, который закрепляется на запястье и имеет пять резиновых пальчиков в виде полосок. Каждая из полосок посредством пластиковых колец закрепляется на пальце. Отметим, что виртуальная перчатка P5 Glove рассчитана только на правую руку.

Каждая полоска представляет собой датчик изгиба и передает компьютеру информацию об относительном положении пальца, с которым она связана.

На корпусе перчатки P5 Glove установлены 8 инфракрасных светодиодов, которые передают сигнал приемнику, установленному на базовой станции.

Базовая станция представляет собой 35-сантиметровую башенку, в которой размещаются контроллер перчатки, инфракрасный приемник и процессор, вычисляющий положение элементов перчатки P5 Glove в пространстве. Базовая станция подключается к USB-порту компьютера и получает от него питание.

В базовой станции размещаются 4 фотодиода. Одна пара фотодиодов контролирует горизонтальные перемещения, другая — вертикальные. Опрашивая датчики с частотой 45 раз в секунду, P5 Glove создает впечатление одновременной слежки за всей перчаткой.

Всенаправленная беговая дорожка

Всенаправленная беговая дорожка (omni-directional treadmill, ODT) — разработка компании Virtual Space Devices, которая продвигается ее разработчиками под торговой маркой Plane.

Сочетание этого устройства с другими периферийными устройствами системы виртуальной реальности позволяет человеку ходить по виртуальному ландшафту. У дорожки практически отсутствует инерционность — изменение направления движения человека по ней сразу же приводит к изменению направления движения дорожки, благодаря чему человек все время остается в ее пределах.

Систему предполагается применить для тренировок пожарных, полицейских, солдат и строительных рабочих. Кроме того, эту дорожку можно использовать в развлекательно-образовательных программах. Ведь используя ее, можно создать полную иллюзию прогулки около Великих пирамид, по Великой китайской стене и т. д. Разработка этого проекта, как пишет компания, спонсировалась Министерством обороны США.

Устроена беговая дорожка следующим образом. Широкая дорожка, бегущая между двумя барабанами, состоит из тонких дорожек, бегущих в поперечном направлении. Описываемое устройство включает в себя систему контроля за перемещением человека, находящегося на дорожке. Система контроля, отслеживая положение человека на дорожке, управляет движением дорожки таким образом, чтобы человек, идущий по ней, все время оставался бы в ее пределах.

Аналогичные разработки имеются в Японии.

14.12. Методы записи движений человеческого тела

В настоящее время известен ряд систем захвата движения, которые можно разделить на две группы — активные и пассивные.

В активных системах применяют датчики, которые передают информацию о своем состоянии. Для анализа движения человека используют специальный костюм, в котором смонтированы датчики, позволяющие определять их пространственные и угловые координаты. Эти системы отличаются простотой и точностью передаваемой информации, однако они не всегда удобны, поскольку затрудняют движение объекта. Одним из преимуществ систем с активными датчиками является их простота.

Системы с пассивными датчиками являются более сложными, хотя сами пассивные датчики устроены проще, чем активные датчики. В этих системах основная нагрузка ложится на программную обработку получаемой от них информации. Применение систем с активными датчиками не всегда возможно, например, при распознавании речи по движению губ. В этих случаях используют пассивные системы, которые по изображениям, получаемым от видеокамер, определяют координаты интересующих точек движущегося объекта, например, координаты отдельных точек при движении губ.

При работе с пассивными системами вводится понятие ключевых областей, под которыми понимается совокупность элементов изображения объекта, по которым возможно восстановить картину его движения. По методу использования ключевых областей пассивные системы можно разделить на три группы.

К первой группе относятся системы, в которых ключевые области помечаются маркерами (пассивными датчиками), различающимися между собой по форме или по цвету.

Ко второй группе относятся системы, в которых ключевые области также помечаются маркерами, но маркерами, которые по внешнему виду между собой ничем не различаются. Это позволяет существенно увеличить количество маркеров и тем самым расширить возможности системы, однако усложняет задачу различения между собой маркеров, установленных в различных ключевых областях.

И, наконец, к третьей группе относятся системы, вообще не использующие маркеры. В этих системах распознавание ключевых областей производится посредством анализа взаимного расположение неоднородных частей изображения на следующих друг за другом кадрах. Естественно, в этих системах используются более сложные алгоритмы, требующие большего количества вычислений.

Как правило, применяются трехмерные системы захвата движения.

Процесс обработки данных системой захвата движения состоит из ряда этапов [4, 26].
Вначале производится инициализация системы, которая включает в себя:

- настройку и калибровку датчиков, если они имеются;
- □ установку системных данных о начальном положении исследуемого объекта;
- инициализацию начальной позы объекта и, возможно, формирование графа свобод и ограничений для каждого участка скелета, если он предусмотрен, по которому система анализирует данные на завершающем этапе работы.

Формирование начальной позы может выполняться различными способами. В некоторых системах предусмотрена определенная инициализационная поза объекта, при этом система начинает сбор и обработку данных только после того, как подсистема инициализации сообщает, что объект занял заданную начальную позу. В других системах начальная поза объекта может быть любой, но в этом случае требуется проанализировать несколько кадров данных для того, чтобы убедиться, что данные о начальной позе получены правильно.

Граф свобод и ограничений может либо с самого начала быть заданным, либо формироваться на начальном отрезке времени работы системы в результате анализа характера движения объекта.

После того как система проинициализирована, выполняется отслеживание ключевых областей (трекинг), которое включает в себя их выделение и описание отслеживания изменений. Выделение и активных, и пассивных датчиков не создает проблемы. В пассивных системах, в которых отсутствуют маркеры, возникает проблема разделения объекта и фона, которая решается либо путем сравнения ряда последовательных кадров, либо путем анализа пространственной структуры изображения. Следующим этапом работы системы является распознавание положения и формы. Если объект является жестким, т. е. не изменяет свою форму, то его форма и положение находятся путем сопоставления положения ключевых областей модели объекта. В случае объектов, которые могут изменять свою форму, как, например, кисть руки, задача усложняется. Так, например, при анализе мимики лица человека используют детальную модель. Обработка такого рода движений в настоящее время вызывает особенно большой интерес у разработчиков аппаратуры для крупных студий, занимающихся компьютерной графикой.

Распознавание действия является завершающим этапом анализа, при этом распознавание может производиться как путем использования статических, так и динамических данных.

При использовании для распознавания действия объекта статических данных сравнивают позу объекта с хранящимися в системе шаблонами. Системы такого рода обладают очень ограниченными возможностями. Они могут, например, определить позу человека: стоит он или сидит, но не могут определить, подпрыгнул ли он или падает с какой-либо высоты.

Системы, в которых для распознавания действия используют динамические данные, обладают большими возможностями. Эти системы могут распознавать: идет ли человек, стоит, бежит или танцует. И статические, и динамические системы на конечном этапе своей работы сравнивают действие объекта с шаблонами, хранящимися в их памяти. При этом динамические системы распознавания действия могут использовать такие данные, как, например, состояние отдельных мышц тела человека.

Заключительным этапом работы является реакция системы, которой может быть, например, перенос данных на компьютерную модель и ее визуализация.

Наиболее продвинутыми в данном направлении являются фирмы Vicon, Acty, Advanced Mechanical Technology, DKH, Gsport, Motion Analysis, Peak Performance, PhaseSpace, Phoenix Technologies, Polhemus, Gypsy и Qualisys.

Примером одной из лучших разработок в данной области является система Vicon 8. В этой системе использованы пассивные датчики. Система предназначена для работы с большими объемами данных и может осуществлять захват движения сразу нескольких актеров. Она не накладывает ограничений на количество маркеров, размещаемых на модели. Как правило, при использовании этой системы на объекте размещается около 120 маркеров, но при необходимости их количество может быть увеличено до 400 при работе в автономном режиме. Система поддерживает работу с 24 разработанными фирмой видеокамерами Vicon Mcam2, которые обеспечивают разложение изображения в растр с 1 300 000 пикселов при 1000 кадрах в секунду. Такие параметры позволяют различать соприкасающиеся маркеры даже на расстоянии 25 метров от камер и регистрировать быстрые движения. Имеется возможность работы в режиме реального времени. Оборудование с такими характеристиками позволяет добиваться высокого качества обработки таких видов движения, как взмах клинка, приемы каратэ и т. д.

Вместе с оборудованием фирма поставляет пакет программного обеспечения, который включает в себя следующие программы:

- Vicon IQ использует элементы искусственного интеллекта, благодаря чему решается проблема "исчезнувших" маркеров;
- Vicon Workstation выполняет полный цикл обработки захвата движения, начиная от выделения маркеров и заканчивая 3D-моделированием;
- Vicon BodyBuilder позволяет создавать и анализировать кинематическую модель скелета.

Другим примером системы захвата движения является электромеханическая система Gypsy Motion Capture System. Она включает в себя скелет, сделанный из легких алюминиевых стержней, следующих за движением актера, на котором расположены датчики. При движении за счет изменения сопротивления датчиков генерируются соответствующие сигналы. Для определения углов поворота в этой системе применены гироскопы. Данные преобразуются в цифровую форму и вводятся в компьютер, где и обрабатываются посредством специализированного программного обеспечения.

Заключение

Подводя итог написанному, автор хотел бы еще раз подчеркнуть, что в настоящее время технологии обработки изображений быстро развиваются. В последние годы эта область чрезвычайно расширилась, включив в себя такие направления, как получение путем сканирования и обработку 3D-изображений, технологии фотографирования и, главным образом, отображения на экране изображений сцен с большим динамическим диапазоном, новые адаптивные методы фильтрации зашумленных изображений, новые методы интерполяции изображений, элементы виртуальной реальности и ряд других. Речь идет уже о регулярном 3D цифровом телевизионном вещании. Оживляются исследования в области создания систем виртуальной реальности. Публикуется много статей в области этих технологий, в основном на английском языке. В имеющихся книгах, посвященных цифровой обработке изображений, главным образом, переводных, перечисленные проблемы не рассматриваются уже хотя бы потому, что в момент их написания эти технологии были еще неизвестны. Кроме того, отсутствуют книги, в которых рассматривался бы широкий круг вопросов цифровой обработки не только 2D-изображений, но и 3D-изображений, и были бы освещены последние достижения в этой области, а также приведены ссылки на последние публикации.

Вместе с тем автор этой книги хотел бы обратить внимание читателей еще на одно важное обстоятельство. Если в настоящее время исследователи, работающие в ряде других областей науки, постоянно сталкиваются с трудно разрешимой проблемой отсутствия дорогостоящего оборудования, на котором можно было бы проводить научные эксперименты, что заставляет опускать руки, или же искать такую возможность для себя в зарубежных фирмах и лабораториях, то здесь все обстоит иначе. Для работы в этой области ученому не требуется какого-либо специального оборудования, а необходим лишь вполне доступный компьютер, оснащенный сравнительно недорогой периферией, и, главное, *новые пробивные идеи*.

В этом отношении по существу каждая глава, лучше сказать, каждое из направлений цифровой обработки изображений, изложенных в главе, содержит как бы затравку и для сугубо прикладных научных исследований ученых, и для диссертационных исследований аспирантов и магистрантов.

При подборе материала для этой книги автор стремился охватить возможно более широкий круг проблем, важных для обработки изображений, предназначенных, в первую очередь, для зрительного наблюдения.

Литература

- 1. Агостон Ж. Теория цвета и ее применение в искусстве и дизайне. М.: Мир, 1982. 181 с.
- Айриг С., Айриг Э. Подготовка цифровых изображений для печати / Перев. с англ. Минск: Попурри, 1997. — 176 с.
- Айриг С., Айриг Э. Сканирование, профессиональный подход / Перев. с англ. Минск: Попурри, 1997. — 169 с.
- Андреев В. А. Разработка первой отечественной системы видеозахвата движения человека: Труды конференции "Новые информационные технологии", г. Судак (Крым), 22—29 мая, 2005. — С. 317—318.
- 5. Анисимов Б. В., Курганов В. Д., Злобин В. К. Распознавание и цифровая обработка изображений. — М.: Высшая школа, 1983. — 295 с.
- Антипин М. В. Интегральная оценка качества телевизионного изображения. Л.: Наука, 1970. — 154 с.
- Беляев Е. А., Тюрликов А. М. Алгоритмы оценки движения в задачах сжатия видеоинформации на низких битовых скоростях // Компьютерная оптика. — Т. 32 (2008). — № 4. — С. 69—76.
- Беляев Е. А., Чуйков А. В. Модифицированный алгоритм иерархической оценки движения в задачах сжатия видеоинформации: Научная сессия ГУАП, 2007. Сб. докл. / ГУАП. СПб. — С. 56—60.
- Бондарко В. М., Данилова М. В., Красильников Н. Н., Леушина Л. И., Невская А. А., Шелепин Ю. Е. Пространственное зрение. СПб.: Наука, 1999. 218 с.
- Брацлавец П. Ф., Росселевич И. А., Хромов Л. И. Космическое телевидение. М.: Связь, 1973. — 248 с.
- 11. Будрикис З. Л. Критерий верности воспроизведения изображения и его моделирование // ТИИЭР. — Т. 60 (1972) — № 7. — С. 48—59.
- 12. Василенко Г. И., Тараторин А.М. Восстановление изображений. М.: Радио и связь, 1986. 302 с.
- 13. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. М.: Диалог-МИФИ, 2002. — 381 с.

- Володин А. Сыграйте мне эту формулу! // Советский Союз. 1977. № 3. С. 32.
- 15. Володин А. Б. Адаптивная контекстная компрессия изображений // Вестник молодых ученых. 2002. № 9. С. 88 96.
- Воробьев В. И., Грибунин В. Г. Теория и практика вейвлет-преобразования. СПб.: ВУС, 1999. — 204 с.
- 17. Ворошилов А. Телевизионный мираж // Информационно-технический журнал "625". 2009. № 10. С. 2—25.
- Гласман К. Конференция IBC2005: Мобильное телевидение // Информационнотехнический журнал "625". — 2005. — № 10. — С. 66—73.
- 19. Гласман К. Ф., Логунов А. Н. Оценка заметности артефактов видеокомпрессии // Техника кино и телевидения. — 2003. — № 4. — С. 29—32.
- 20. Гласман К., Логунов А., Бабаян А. Корреляционно-энергетический критерий качества изображения, кодированного по стандарту MPEG-2 // Информационно-технический журнал "625". 2006. № 2. С. 85—89.
- 21. Глезер В. Д. Зрение и мышление. Л.: Наука, 1985. 246 с.
- 22. Глезер В. Д., Цуккерман И. И., Цыкунова Т. М. О пропускной способности зрения // Техника кино и телевидения. — 1961. — № 3. — С. 27—32.
- Гонсалес Р., Вудс Р. Цифровая обработка изображений / Перев. с англ. М.: Техносфера, 2006. — 1070 с.
- 24. Гриненко Е. Мобильное телевидение // Информационно-технический журнал "625". 2009. № 9. С. 54—63.
- Губанов П. В. Автоматическая сегментация текстурированных изображений на основе локальных распределений характеристик // Вестник Томского государственного университета. — Т. 271. — 2000, июнь. — С. 74—77.
- Гуленко И. Е. Система видеозахвата и анализа движения распознавание трансформаций и движения объекта: Труды конференции "Новые информационные технологии", г. Судак (Крым), 15—25 мая, 2004. — С. 141—142.
- Гунько С., Карпов С. Видеомагнитофоны формата D-9 // Информационнотехнический журнал "625". — 2000. — № 6. — С. 18—20.
- Гурский Ю., Корабельникова Г. Photoshop 7. Трюки и эффекты. М.: Питер, 2005. — 464 с.
- 29. Дуда Р., Харт П. Распознавание образов и анализ сцен / Перев. с англ. Мир, 1976. 511 с.
- 30. Ивенс Р. М. Введение в теорию цвета / Перев. с англ. М.: Мир, 1964. 442 с.
- Ильясова Н. Ю. Классификация кристаллограмм с использованием методов статистического анализа текстурных изображений / Н. Ю. Ильясова, А. В. Куприянов, А. Г. Храмов // Компьютерная оптика. — 2000. — № 20. — С. 122—127.

- Кириллин Ю. П., Сорин В. Я. Повышение эффективности цифровой системы передачи видеосигнала // Телевидение (Межвузовский сборник). — 1978. — С. 80—82.
- Климов А. С. Форматы графических файлов. К.: НИПФ "Диасофт Лтд.", 1995. — 480 с.
- 34. Комар В. Г., Сон Д. Ю., Сабо С. А., Майоров В. П., Семин М. С., Беляев С. В., Балясный Л. М., Крутик М. И., Любич О. А., Лапотенко В. Е., Котляр В. Л. Трехмерная цветная телевизионная многоракурсная безочковая система с голографическим экраном // Техника кино и телевидения. — 1998. — № 4. — С. 29—40.
- Комолова Н. В. Adobe Photoshop CS4 для всех. СПб.: БХВ-Петербург, 2009. 992 с.
- 36. Конушин А., Баринова О., Конушин В., Якубенко А., Велижев А. Введение в компьютерное зрение // МГУ ВМК, Graphics&Media Lab. — 2008 (http://courses.graphicon.ru).
- 37. Красильников Н. Н. Математическая модель темновой адаптации в зрительной системе человека // Оптический журнал. Т. 64 (1997). № 11. С. 38—44.
- Красильников Н. Н. Обобщенная функциональная модель зрения и ее применение в системах обработки и передачи изображений // Автометрия. — 1990. — № 6. — С. 7—14.
- Красильников Н. Н. Принципы обработки изображений, основанные на учете их семантической структуры // Информационно-управляющие системы. — 2008. — № 1. — С. 2—6.
- Красильников Н. Н. Помехоустойчивость телевизионных устройств. М. Л.: Госэнергоиздат, 1961. — 268 с.
- 41. Красильников Н. Н., Красильникова О. И. Определение координаты глубины по 2D-изображению // Оптический журнал. Т. 78 (2011). № 10.
- Красильников Н. Н. Влияние шумов на контрастную чувствительность и разрешающую способность приемной телевизионной трубки // Техника телевидения. — 1958. — Вып. 25. — С. 26—43.
- 43. Красильников Н. Н. Исследование статистических характеристик оболочек 3Dобъектов // Оптический журнал. — Т. 73 (2006). — № 5. — С. 18—23.
- 44. Красильников Н. Н. Метод получения 3D–изображений, основанный на диффузном отражении света сканируемыми объектами // Информационно-управляющие системы. — 2009. — № 6 (43). — С. 7—11.
- 45. Красильников Н. Н. Определение координаты глубины в методах 3Dсканирования: Тезисы докладов IX Международной конференции "Прикладная оптика — 2010", г. Санкт-Петербург, октябрь 2010 г.
- 46. Красильников Н. Н. Принципы обработки изображений, основанные на учете их семантической структуры // Информационно-управляющие системы. — 2008. — № 1. — С. 2—6.

- 47. Красильников Н. Н. Реставрация изображений с учетом их структуры // Оптический журнал. — Т. 76 (2009). — № 2. — С. 7—12.
- 48. Красильников Н. Н. Статистическая теория передачи изображений. М.: Связь, 1976. 184 с.
- Красильников Н. Н. Теория передачи и восприятия изображений. М.: Радио и связь, 1986. — 247 с.
- 50. Красильников Н. Н. Цифровая обработка изображений. М.: Вузовская книга, 2001. 320 с.
- Красильников Н. Н., Красильникова О. И. Получение трехмерного изображения объекта путем измерения интенсивности диффузного отражения света различными точками его поверхности // Оптический журнал. Т. 77 (2010). № 6. С. 19—24.
- 52. Красильников Н. Н., Астратов О. С., Любимов В. А. Исследование возможности уменьшения вероятности новых значений в кодированном видеосигнале // Межвузовский сборник "Вопросы теории и проектирования телевизионных систем передачи, приема и отображения информации". — 1977. — Вып. 115. — С. 3—7.
- 53. Красильников Н. Н., Байдаков М. П., Пастухов О. В. О различении человеком изображений в гауссовом шуме // Автометрия. 1973. № 1. С. 7—14.
- 54. Красильников Н. Н., Байдаков М. П., Пастухов О. В., Потоцкий В. К. К вопросу о различении наблюдателем зашумленных изображений // Автометрия. — 1970. — № 3. — С. 20—25.
- 55. Красильников Н. Н., Волошина Н. В. Предварительная логическая фильтрация изображений для увеличения степени их сжатия энтропийными кодерами // Оптический журнал. — Т. 70 (2003). — № 5. — С. 39—43.
- Красильников Н. Н., Горшков А. С. О цифровой магнитной видеозаписи сигналов вещательного телевидения // Межвузовский сборник "Телевидение". — 1978. — Вып. 126. — С. 11—25.
- 57. Красильников Н. Н., Красильникова О. И., Шелепин Ю. Е. Эффект кажущегося размытия резких и обострения размытых границ при наблюдении движущихся объектов // Оптический журнал. Т. 71 (2004). № 11. С. 61—68.
- Красильников Н. Н., Красильникова О. И. Эффективный метод повышения качества изображений // Техника кино и телевидения. — 1989. — № 1. — С. 23—27.
- 59. Красильников Н. Н., Мироненко Е. П. Исследование погрешностей восприятия формы при наблюдении 3D-объектов // Оптический журнал. Т. 73 (2006). № 5. С. 18—23.
- Красильников Н. Н., Мироненко Е. П., Красильникова О. И. Коэффициент эффективности зрительной системы человека при произвольных ракурсах наблюдения трехмерных объектов // Оптический журнал. — Т. 73 (2006). — № 10. — С. 63—68.

- Красильников Н. Н., Посохова О. И. Способ двумерной апертурной коррекции видеосигнала и устройство для его реализации / Авт. свид. № 1008926 (СССР). БИ, 1983, № 12.
- 62. Красильников Н. Н., Росселевич И. А. Перспективы использования цифрового метода в ТВ вещании // Техника кино и телевидения. 1976. № 5. С. 32—35.
- Красильников Н. Н., Филатов А. И. О выборе числа уровней квантования при кодово-импульсной модуляции // Техника кино и телевидения. — 1964. — № 70. — С. 51—54.
- 64. Красильников Н. Н., Шелепин Ю. Е., Красильникова О. И. Фильтрация в зрительной системе человека в условиях порогового наблюдения // Оптический журнал. — Т. 66 (1999). — № 1. — С. 5—14.
- 65. Красильникова О. И. Метод повышения четкости изображений в системе с предварительной фильтрацией // Техника кино и телевидения. — 1989. — № 12. — С. 30—32.
- 66. Кривошеев М. И., Кустарев А. К. Цветовые измерения. М.: Энергоатомиздат, 1990. — 240с.
- 67. Кривошеев М. И., Кустарев А. К. Цветовые измерения. М.: Энергоатомиздат, 1990. — 240с.
- Крылов В. Н., Полякова М. Морфологический метод контурной сегментации изображений на основе репагулярного вейвлет-преобразования: Труды Одесского политехнического университета, г. Одесса, 2006. — С. 98—103.
- 69. Кустов В. Н., Федчук А. А Методы встраивания скрытых сообщений // Защита информации. Конфидент. 2000. № 3. С. 34—37.
- Ла Мот А., Ратклифф Д., Семинаторе М., Тайлер Д. Секреты программирования игр / Перев. с англ. — СПб.: Питер, 1995. — 720 с.
- 71. Людаев М. Ю. Применение алгоритма адаптивной фильтрации для обработки аэрофотоснимков // Электросвязь. 2009. № 7. С. 1—3.
- Марр Д. Зрение. Информационный подход к изучению представления и обработки зрительных образов / Перев. с англ. М.: Радио и связь, 1987. — 400 с.
- 73. Методы компьютерной обработки изображений / Под ред. В. А. Сойфера. М.: Физматлит, 2001.— 784 с.
- 74. Мироненко Е. П. Алгоритм дифференциальной кодово-импульсной модуляции в задачах компрессии цифрового потока данных, описывающих движение 3Dмодели // Известия вузов. Приборостроение. — 2007. — С. 104—108.
- 75. Никулин Е. А. Компьютерная геометрия и алгоритмы машинной графики. СПб.: БХВ-Петербург, 2005. — 560 с.
- Новаковский С. В. Цвет в цветном телевидении. М.: Радио и связь, 1988. 288 с.

- 77. О'Нейл Э. Введение в статистическую оптику / Перев. с англ. М.: Мир, 1966. 254 с.
- Обработка изображений и цифровая фильтрация / Под ред. Т. Хуанга. Перев. с англ. — М.: Мир, 1979. — 318 с.
- 79. Оппенхейм А. В., Лим Дж. С. Важность фазы при обработке сигналов // ТИИЭР. — Т. 69 (1981). — № 5. — С. 39—54.
- Певзнер Б. М. Качество цветных телевизионных изображений. М.: Радио и связь, 1988. — 224 с.
- Пескин А. Е., Смирнов А. В. Цифровое телевидение. От теории к практике. М.: Горячая линия-Телеком, 2005. — 349 с.
- 82. Порев В. Н. Компьютерная графика. СПб.: БХВ-Петербург, 2004. 432 с.
- Прэтт У. К., Фожра О. Д., Гагалович А. Зрительное различение стохастических текстурных полей // ТИИЭР. — Т. 26 (1978). — № 11.
- Прэтт У. Цифровая обработка изображений / Перев. с англ. М.: Мир, 1982. Т. 1. — 310 с. // М.: Мир, 1982. — Т. 2. — 790 с.
- Прэтт У. К., Фожра О. Д., Гагалович А. Применение моделей стохастических текстур для обработки изображений // ТИИЭР. — Т. 69 (1981). — № 5. — С. 54—64.
- Ричардсон Я. Видеокодирование. Н/264 и MPEG-4 стандарты нового поколения. — М.: Техносфера, 2005. — 366 с.
- 87. Рябко Б. Я. Сжатие данных при помощи "мнимого скользящего окна" // Проблемы передачи информации. — Т. 32 (1996). — № 2. — С. 22—30.
- Рябко Б. Я. Эффективный метод арифметического кодирования для источников с большими алфавитами // Проблемы передачи информации. — Т. 35 (1999). — № 4. — С. 95—108.
- Смирнов А. В. Основы цифрового телевидения. М.: Горячая линия-Телеком, 2001. — 224 с.
- Сорин В. Я., Федоренко В. Н. Уменьшение вероятности новых значений при квантовании фототелеграфного сигнала изображения: Труды Ленинградского института авиационного приборостроения, вып. 64, 1971 г. — С. 22—25.
- 91. Стокхем Т. Обработка изображений в контексте моделей зрения // ТИИЭР. Т. 60 (1972). — № 7. — С. 93—107.
- Таненбаум Э. Компьютерные сети / Перев. с англ. СПб.: Питер, 2002. 848 с.
- Телевидение: учебник для вузов / Под ред. В. Е. Джаконии. 4-е изд. М.: Горячая линия-Телеком, 2007. — 616 с.
- 94. Федотов С. Нелинейные и гибридные системы видеомонтажа // Информационно-технический журнал "625". — 1997. — № 3. — С. 5—35.
- 95. Физиология сенсорных систем. Ч. первая. Физиология зрения / Под ред. Г. В. Гершуни. — Л.: Наука, 1971. — 416 с.

- Фисенко В. Т., Фисенко Т. Ю. Компьютерная обработка и распознавание изображений: Учеб. пособие. — СПб.: СПбГУ ИТМО, 2008. — 192 с.
- Фурман Я. А., Кревецкий А. В., Передреев А. К. и др. Введение в контурный анализ и его приложения к обработке изображений и сигналов. — М.: Физматлит, 2003.
- Цифровое кодирование телевизионных изображений / Под ред. И. И. Цуккермана. — М.: Радио и связь, 1981. — 239 с.
- 99. Цифровое преобразование изображений: Учеб. пособие для вузов / Р. Е. Быков, Р. Фрайер, К. В. Иванов, А. А. Манцветов; под ред. Р. Е. Быкова. — М.: Горячая линия — Телеком, 2003. — 228с.
- 100. Чирков Л. Видеозапись: форматы и логика // Информационно-технический журнал "625". 1998. № 1. С. 5—19.
- 101. Чочиа П. А. Пирамидальный алгоритм сегментации изображений // Информационные процессы. — Т. 10 (2010). — № 1. — С. 23—35.
- 102. Яне Б. Цифровая обработка изображений / Перев. с англ. М.: Техносфера, 2007. 583 с.
- 103. Ярославский Л. П. Цифровая обработка сигналов в оптике и голографии. М.: Радио и связь, 1987. — 296 с.
- 104. Accame M., De Natale F.G.B., Giusto D.D. Hierarchical motion estimator (HME) for block-based video coders // IEEE Transactions on Consumer Electronics. 1997. № 43. P. 1320—1330.
- 105. Acharya T. and Tsai P.-S. JPEG2000 Standard for Image Compression. New York: Wiley, 2005.
- 106. Allebach J. and Wong P.W. Edge-Directed Interpolation // IEEE Int'l Conf Image Processing (ICIP 97), IEEE Press, Piscataway, N.J. — 1996. — P. 707—710.
- 107. Ashikhmin M. A Tone Mapping Algorithm for High Contrast Images // Eurographics Workshop on Rendering. 2002. P. 1—11.
- 108. Barnard T. W. An Image Evaluation Method // A Symposium on Sampled Images, Perkin-Elmer Corporation, Norwalk, Connecticut, 1971.
- 109. Baroncini V. A. Automatic Visual Quality Control in Digital TV Servises // Proceedings of 1998 International Broadcasting Convention. — P. 425—430.
- 110. Belyaev E., Koski T., Paavola J. Turlikov A. Adaptive power saving on the receiver side in digital video broadcasting systems based on progressive video codes // The 11-th International Symposium on Wireless Personal Multimedia Communications (WPMC'08). 2008, Lapland, Finland.
- 111. Bender Walter, Gruhl Daniel, Morimoto Norishige and Lu Anthony. Techniques for data hiding // IBM Systems Journal. — 1996. — 35(3/4). — P. 313—336.
- 112. Bovik A., Liu S., DCT-domain blind measurement of blocking artifacts in DCTcoded images // Proc. IEEE Int. Conf. Acoust., Speech, and Signal Processing. — May 2001. — V. 3. — P. 1725—1728.

- Boyer K. L., Kak A. C. Color-encoded structured light for rapid active ranging. // IEEE Trans. on Pattern Analysis and Machine Intelligence. 1987. № 9 (1). P. 14—28.
- 114. Boykov Y. and Jolly M.-P. Interactive organ segmentation using graph cuts // In Medical Image Computing and Computer-Assisted Intervention. — 2000. — P. 276—286.
- 115. Boykov Y. and Jolly M. P. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images // In Proc. Of the International Conference on Computer Vision. 2001. Vol. 1. P. 105—112.
- 116. Boykov Y. and Kolmogorov V. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. // IEEE Trans. Pattern Anal. 2004. Mach. Intell. 26, 9. P. 1124—1137.
- 117. Burt P. J., Hong T. H. and Rosenfeld A. Segmentation and estimation of image region properties through cooperative hierarchical computation. // IEEE Trans. SMC. — 1981. — № 11. — P. 802—809.
- 118. Burt P. J. The pyramid as a structure for efficient computation. In Rosenfeld, ed., Multiresolution image processing and analysis, vol. 12 of Springer Series in Information Sciences. — New York: Springer, 1984 — P. 6—35.
- 119. Canny J.E. A computational approach to edge detection // IEEE Trans Pattern Analysis and Machine Intelligence. 1986. № 8. P. 679—698.
- 120. Ce Liu, Richard Szeliski, Sing Bing Kang, C. Lawrence Zitnick, William T. Freeman Automatic Estimation and Removal of Noise from a Single Image // IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008. — Vol. 30. — № 2 (February). — P. 299—314.
- 121. Cok D. R. Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal // U.S. Patent 4, 642, 678, (1987).
- 122. Dowling J.E. The retina. (An approachable part of the brain) Cambridge, Massachusetts, London, The Belknap press of Harvard university press. 1987. — 281 p.
- 123. Durand F. and Dorsey J. Fast Bilateral Filtering for the Display of High-Dynamic-Range Images. In John Hughes, editor, SIGGRAPH 2002 Conference Graphics Proceedings, Annual Conference Series // ACM Press/ACM SIGGRAPH 2002 — P. 257—265.
- 124. Dyn N., Levin D., and Rippa S. Data Dependent Triangulations for Piecewise Linear Interpolation. // IMA J. Numerical Analysis. — 1990. — Vol. 10. — P. 137—154.
- 125. Faugeras O.D. Digital color image processing within the framework of a human visual model // IEEE Tran. Acoust., Speech., Signal Processing, Aug. — 1979. — Vol. ASSP-27. — P. 380—393.
- 126. Fechter F. Objective Beurteilung der Qualitaet komprimierter Bildfolgen Ein heuristisch optimiertes Modell. // Fernseh-und Kino-Technic. — 1998. — № 7. — P. 417—421.

- 127. Franz E., Jerichow A., Moller S., Pfitzmann A., Stierand I. Computer Based Steganography: How it works and why therefore any restrictions on cryptography are nonsense, at best, In Information hiding: first international workshop, Cambridge, UK. Lecture Notes in Computer Science, vol. 1174. — Berlin Heidelberg New York: Springer-Verlag, 1996.
- 128. Grady, L. Multilabel random walker image segmentation using prior models. // In CVPR. — 2005. — № 1. — P. 763—770.
- Grady L., Funka-Lea G. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. In ECCV Workshops CVAMIA and MMBIA, 2004. — P. 230—245.
- 130. Grady L., Schiwietz T., Aharon S., Westermann, R. Random walks for interactive organ segmentation in two and three dimensions: Implementation and validation. In Proceedings of MICCAI 2005, Springer, Palm Springs, CA, J. Duncan and G. Gerig, Eds., no. 2 in LNCS 3750, MICCAI Society, P. 773—780.
- 131. Hall C.F., Hall E.L. A nonlinear model for the spatial characteristics of the human visual system // IEEE Tran. Syst., Man., Cybern. Vol. SMC-7. Mar. 1977. P. 161—170.
- 132. Hamada T. Satoshi H. Matsumoto S. Picture Quality assessment by three-layered bottom-up noise weighting considering human visual perception. // SMPTE, January, 1999.
- 133. Hammett S. T., Georgson M. A., Gorea A. Motion blur and motion sharpening: temporal smear and local contrast non-linearity // Vision Res. — 1998. — V. 38. — P. 2099—2108.
- 134. Johnson N. F., Jajodia S. Exploring Steganography: Seeing the Unseen // IEEE Computer. — February 1998. — Vol. 31. — No. 2. — P. 26—34.
- 135. Kang S. B., Webb J. A., Zitnick C. L., Kanade T. A multibaseline stereo system with active illumination and real-time image acquisition. // Proc. Int. Conf. on Computer Vision. — June 1995. — P. 88—93.
- 136. Kimmel R. Demosaicing: Image Reconstruction from Color CCD Samples // IEEE Transactions on Image Processing. — Vol. 8—9. (September 1999). — P. 1221— 1228.
- 137. Koenderink J. J., Van De Grind W. A., Bouman M. A. Opponent color coding: A mechanistic model and a new metric for color space // Kybernetik. — Vol. 10 (1972). — № 2. — P. 78—98.
- 138. Krasilnikov N. N., Krasilnikova O. I., Shelepin Y. E. Perception of achromatic, monochromatic, pure chromatic and chromatic noisy images by real human-observer under threshold conditions // Proceedings of SPIE. International Symposium on Medical Imaging 2000. San Diego, California. — February, 2000. — V. 3981 — P. 78—85.
- 139. Kretzmer E. R. Statistics of Television Signals // BSTJ. 1952. July. V. 31. P. 751—763.

- 140. Kuhn P. Algorithms, Complexity Analysis and Vlsi Architectures for Mpeg-4 Motion Estimation // Boston Hardbound: Kluwer Academic Publishers. — 1999. — June. — P. 248.
- 141. Kunt M., Ikonomopoulos A., Kocher M. Second-Generation Image-Coding Techniques // Proceedings IEEE. V. 73 (1985). № 4. P. 549—574.
- 142. Lawson C. L. Software for c1 Surface Interpolations // Mathematical Software III, J.R. Rice, ed., Academic Press, New York. — 1977. — July. — P. 161—194.
- 143. Lopes F., Ghanbari M. Hierarchical motion estimation with spatial transforms // 2000 International Conference on Image Processing. — 2000. — V. 2. — P. 558—561.
- 144. Lowry E.M. The luminance Discrimination of the Human Eye. // J. SMPTE. 1951. Sept. V. 57 P. 187—196.
- 145. Marcellin M. W., Gormish M. J., Bilgin A., Boliek M. P. An Overview of JPEG-2000 // Proceedings of IEEE. Data Compression Conference. — 2000. — P. 523— 541.
- 146. Marpe D., Cycon H. L., Zander G., Barthel Kai-Uwe.Context-based Denoising of Images Using Iterative Wavelet Thresholding. // Proc. SPIE. — 2002. — Jan. — Vol. 4671. — P. 907—914.
- 147. Marpe D., Schwarz H., Wiegand T. Context-based adaptive binary arithmetic coding in the H.264/AVC video compression standard // IEEE Transactions on Circuits and Systems for Video Technology. — 2003. — V. 7. — P. 620—636.
- 148. Martin D., Fowlkes C., Tal D., Malik J. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. // Proceedings of 8th International Conference on Computer Vision. — 2001. — № 1. — P. 416—423.
- 149. Monobe Y. Yamashita H., Kurosawa T., Kotera H. Dynamic Range Compression Preserving Local Image Contrast for Digital Video Camera // IEEE Transactions on Consumer Electronics. — 2005. — February. — Vol. 51. — № 1. — P. 1—10.
- 150. Montanari U. A method for obtaining skeletons using a quasi-Euclidean distance // J. ACM. — 1968. — 15, October. — P. 600—624.
- 151. Montanari U. Continuous skeletons from digitized images // J. ACM. 1969. 16, October. C. 534—549.
- 152. Morita H., Yajima K., Sakata S. Reconstruction of surfaces of 3d objects by m-array pattern projection method. // Proc. Int. Conf. on Computer Vision. — 1988. — P. 468—473.
- Mortensen E., Barrett W. Intelligent scissors for image composition. // Proc. ACM Siggraph. — 1995. — P. 191—198.
- 154. Mortensen E., Barrett, W. Tobogan-based intelligent scissors with a four parameter edge model. // In Proc. IEEE Conf. Computer Vision and Pattern Recog. — 1999. — Vol. 2. — P. 452—458.

- 155. Nayar S. K., Watanabe M., Noguchi M. Real-time focus range sensor. // IEEE Transactions on Pattern Analysis and Machine Intelligence. 1996. V. 18. № 12. P. 1186—1198.
- 156. Perona P., Malik J. Scale-space and edge detection using anisotropic diffusion // IEEE Trans. on Pat. Annual and Mach. Intel. 1990. 12 (7). P. 629—639.
- 157. Pietikainen M., Rosenfeld A. Image segmentation by texture using pyramid node linking // SMC. 1981. № 11. P. 822—825.
- 158. Pizurica A., Philips W. Lemahieu Ignace and Acheroy Marc A Versatile Wavelet Domain Noise Filtration Technique for Medical Imaging // IEEE Trans Med Imaging. — 2003. — Mar., 22 (3) — P. 323—331.
- Pizurica A., Zlokolica V., Philips W. Noise Reduction in Video Sequences Using Wavelet-Domain and Temporal Filtering: SPIE Conference on Wavelet Applications in Industrial Processing 27—31 October 2003.
- 160. Ponomarenko N. N., Egiazarian K., Lukin V. V., Astola J. T. Cascade Fractal Image Compression and its Modification // Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Philadelphia, USA. — 2005. — March. — Vol. 2. — P. 361—364.
- 161. Portilla J., Strelay V., Wainwrightz M. J. Simoncelli Eero P.Adaptive Wiener Denoising Using A Gaussian Scale Mixture Model In The Wavelet Domain: Proceedings of the 8th IEEE Int'l Conf on Image Processing Thessaloniki, Greece. 7—10 October, 2001.
- 162. Raymond B. Wolfgang and Edward J. Delp. A watermark for digital images: In International Conference on Images Processing, Lausanne, Switzer-land // IEEE. 1996. September. P. 219—222.
- 163. Reinhard E., Ward G., Pattanaik S., Debevec P. High Dynamic Range Imaging / ELSEVIER Morgan Kaufmann Publisher, 2006. — 495 P.
- 164. Reinhard E., Stark M., Shirley P., Ferwerda J. Photographic Tone Reproduction for Digital Images. In Proceedings of SIGGRAPH 2002, Computer Graphics Proceedings, Annual Conference Series. ACM Press /ACM SIGGRAPH, July 2002.
- 165. Saupe D., Hartenstein H. Lossless Acceleration of Fractal Image Compression by Fast Convolution: Proceedings of IEEE International Conference on Image Processing. — 1996. — Sep. — Vol. 1. — P. 185—188.
- Serra J. Image analyses and mathematical morphology. London: Academic Press, 1982.
- Soiled P. Morphological Image Analysis. Principles and Applications. 2nd edn. Berlin: Springer, 2002.
- 168. Stockham T. G. Image Processing in the Context of a Visual Model // Proceedings of the EEE. — 1972. — V. 60. — No. 7. — P. 828—842.
- 169. Sutherland N. The representation of three-dimensional objects // Nature. 1979. № 278. — P. 395—398.

- 170. Sziklai G. Some studies in the speed of visual perception // Trans. IRE Inform Theory. — 1956. — IT-2. — P. 125—135.
- 171. Terzopoulos D., Regularization of Inverse Visual Problems Involving Discontinuities: IEEE Trans. Pattern Analysis and Machine Intelligence, // IEEE. — 1986. — July. — Vol. 8. — No. 4. — P. 413—424.
- 172. Tomasi C., Manduchi R. Bilateral filtering for gray and color images // Proc. IEEE Int. Conf. on Computer Vision. 1998. P. 836—846.
- 173. Tschumperle D. Fast anisotropic smoothing of multi-valued images using curvaturepreserving // PDE's Init'l Journal on Computer Vision. — 2006. — № 68 (1). — P. 65—82.
- 174. Tumblin J., Turk G. LCIS: A Boundary Hierarchy for Detail-Preserving Contrast Reduction. In Proceedings of SIGGRAPH 99, Computer Graphics Proceedings, Annual Conference Series. Los Angeles, California, August 1999. ACM SIGGRAPH / Addison Wesl. — P. 83—90.
- 175. Vincent L., Soille P. Watersheds in digital spaces: an efficient algorithm based on immersion simulations // IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-13. 1991. № 6. P. 583—598.
- 176. Wang Z., Bovik A., Evans B. Blind measurement of blocking artifacts in images // Proc. IEEE Int. Conf. Image Proc. — 2000. — Sept. — V. 3. — P. 981—984.
- 177. Welsh W. J., Searby S., Waite J. B. Model-based image coding // British Telecom Technology. 1990. Vol. 8. No 3. P. 94—106.
- 178. Yu X., Morse B. S., Sederberg T.W. Image Reconstruction Using Data-Dependent Triangulation // IEEE Computer Graphics and Application. — 2001. — № 5/6. — P. 62—68.
- Zitnick C. L., Jojic N., Kang S. B. Consistent segmentation for optical flow estimation: In Proc. IEEE Int'l Conf. Computer Vision, 2005.

Предметный указатель

C, D

Corel PHOTO-PAINT 277, 279, 545 Data-dependent triangulation (DDT) 305

Η

HiFiColor 74 High Dynamic Range (HDR) 236

L, P

Low Dynamic Range (LDR) 238 Photoshop 75, 211, 277, 380, 544 фильтры 458

Α

Аксиомы: Грассмана 49 уравнивания цветов 51 Альфа-канал 80 Анимация 455 форматы файлов 457 Аффинные преобразования изображений 282

Б

Баланс белого 77 Бинаризация изображения 316

В

Варпинг 454 Вейвлет-преобразование 522 Вектор движения 534, 536 Вилео: мобильное 466 объемное 466 цифровое 441 Видеосигнал: композитный 443 компонентный 443 Виртуальная реальность 547 Виртуальная студия 468 Виртуальная сцена 548 Виртуальные взаимодействия с объектами 559 Виртуальные путешествия 556 Воляной знак 47 Воксел 396 Восприятие движения 104

Г

Гамма-корректор 217, 230 Гамма-коррекция 238 Гипотеза Юлеша 15 Гистограмма распределения яркости изображения 27 Гранулометрия 383 Графика: векторная 8 кривые Безье 10 объектно-ориентированная 9 пиксел 6 растровая 6 растровый элемент 8

Д

Диспарантность 109 Дисторсия 282, 285

3

Закон: Блоха — Шарпантье 103 вычитания цветов 51 сложения цветов 51 смешения цветов: второй 52 первый 51 третий 51 Тальбота 103, 440 транзитивности цветов 52 Запись на магнитную ленту 447, 461, 462 Зрительная система: биполярные клетки 91 ганглиозные клетки 91 зрительный нерв 91 колбочки 89 оптика глаза 88 палочки 89 сетчатка 89

И

Избыточность изображения: декорреляция 477 психофизическая 476 статистическая 474 Изображение: 2,5D 395 анаглифное 391 бинарное 315 векторное 8, 543 геометрическое искажение 281 движущееся, сжатие 534 двумерное 426 дефекты 205 динамическое 439 маркирование 47 модель 15 нейронное 89 неподвижное ахроматическое 10 оцифровка 134 проекция на плоскость 410 разрешение 86, 395, 397, 400, 441, 444 растровое 6, 543 реставрация 183 с прозрачным фоном 81 сбалансированное 233 семантическое 11 спои 80

спектр 17 спектральная интенсивность 18 стереоскопическое 387 текстурное 11 трехмерные изображения 398 увеличение 296 уменьшение 306 цветное 8, 10 вывод на экран 70 печать 73 черно-белое 37 Имитация искусственного интеллекта 565 Импульсная характеристика системы 18 Интерполяция: бикубическая 300 билинейная 299 нулевого порядка 296 при использовании матриц Байера 308 функцией $\sin x/x$ 302 Искажение: апертурное 171 геометрическое 281 перспективное 287 Искажения: амплитудно-частотные 15, 85, 158 геометрические 85, 289 градаций яркости 215 дисторсия 282 линейные 16, 85, 112 нелинейные 15, 85, 120 ослабление 256 фазочастотные 85 цветовые 85

К

Кадр, ключевой 456 Калибровка 75 монитора 75 печатающего устройства 76 сканера 76 Канал 80 альфа-канал 80 цветовой 80 Квантование 150, 154 оптимальное 152 по яркости 150 разрежение шкалы квантования 230 тоновая коррекция 233 Квантователь Ллойла — Макса 152 Кинескоп 217 Код Грея 503 Кодирование битовых плоскостей 502 Колориметрическая система 55 CMY 69 CMYK 69 HLS 69 HSB 69 L*a*b* 68 **XYZ 58** координаты единичных цветов 55 локус 57 Мак-Алама 67 Манселла 66.69 смешение цветов 51, 54, 56, 63, 71, 73 удельные координаты 54 цветовой модуль 57, 60, 62 Компонент: градиентный 122 контурный 120 фактурный 122 Корректор, апертурный 248 Коррекция: апертурных искажений 248 воспроизведения градаций яркости 218 гамма-корректор 217 геометрических искажений 85, 289 методом видоизменения гистограмм 220 неравномерности освещенности 256 Коэффициент автокорреляции: изображения 24, 27 квазибелого шума 41 оболочек трехмерных объектов 30 помехи 40 Коэффициент сжатия 475 Кривая: Безье, третьего порядка 10 второго порядка, уравнение 10 третьего порядка, уравнение 10 Кривая смешения 54 Критерий Рэлея 86, 157

Л

Логическая апертура 320 Логические операции 317 Ложный контур 122, 154, 231, 233, 270, 279 заметность 126 ослабление заметности 154 устранение 245 Локус 57

Μ

Маркирование изображений 47 Маска 81. 276 Метол: брекетинг экспозиции 235 взвешенного суммирования недодержанного, нормально экспонированного и передержанного изображений 238 Гуро 407 закраски граней 406 записи движений тела человека 572 контрастирующей лупы 226 логической фильтрации изображений 497 нелинейной обработки изображений 277 нерезкого маскирования 250 обратной трассировки лучей 409 обращения контраста 229 подчеркивания световых границ 164, 197, 198, 202 прямой трассировки лучей 408 согласования динамических диапазонов изображения и дисплея 222 степного пожара 340 структурированного света 416 триангуляции 304 уравнивания цветов 56 Фонга 408 экспертных оценок 127 Метод выделения контуров: градиентный 355 Канни 273, 363 Кирша 362 пересечения нулевого уровня 362 Превитта 358, 361 Робертса 357, 361 с использованием лапласиана 356, 360 Собела 358, 361 Уописа 362

Метод нерезкого маскирования 198 Метод сжатия изображения: H.264 540 JPEG 471, 519 JPEG 2000 494, 530 LZW 480 MPEG-2 448. 533 MPEG-4 538 RLE 478 арифметический 484 без потери информации 474 вейвлет-преобразование 526 ДКИМ 507 на основе вейвлет-преобразований 494 препроцессинга 496 разделением на контексты 494 с использованием 3D-моделей 541 с использованием ортогональных преобразований 510 с потерями информации 476 фрактальный 531 Хаффмена 482 Множество, структурообразующее 325 Мода 154, 156 Модель освешения 401 Модель изображения 15 векторная 8 полигональная 398 воксельная 396 двумерная 6,8 Монтаж цифрового видео 446 варпинг 454 морфинг 453 нелинейный 446 разбивкой на отдельные сцены 449 создание переходов 450 спецэффекты 457 Морфинг 453 Морфологическая операция 315 с бинарным изображением: верх шляпы (tophat) 342 выделение периметра (remove) 341 дилатация (наращивание) 322, 327 закрытие (close) 329 заливка (fill) 339 мост (bridge) 336 низ шляпы (bothat) 342 открытие (open) 329

скелет (skel) 340 удаление (clean) 321 удаление внутренних пикселов объектов (remove) 321 удаление центрального пиксела в конфигурациях с Н-связностью 338 усечение (spur) 335 утолщение (thicken)) 334 утончение (thin) 333 эрозии (erode) 323, 328 с полутоновым изображением: верх шляпы (tophat) 347 дилатация 344 закрытие 346 морфологический градиент 349 низ шляпы (bothat) 347 открытие 346 фильтрация 350 эрозия 345

Η

Накопление ошибок при сжатии изображений 494 Насыщенность цвета 38

0

Обнаружение и различение на изображении объектов известной формы 385 Оболочка объекта 31 обработка 434 Обработка изображения: 3D 434 в ограниченной пространственной области 195, 276 с учетом восприятия зрительной системой 189 увеличение 296 уменьшение 306 Обращение контраста 229 Объект, оболочка 31 Окно: Бартлетта 167 Блэкмана 167 Кайзера 167 квадратное 240, 260

крестообразное 261, 263 прямоугольное 267 стереоскопическое 390 треугольное 167 Ханна 167 Хэмминга 167 Определение координаты глубины 426 Острота зрения 98 Отрезок, уравнение 10 Оценка качества: контраст 87 корреляционно-энергетический критерий 129 отношение сигнала к шуму 111 разрешающая способность 87, 111 частные критерии 87 экспертная 111

П

Периферийные устройства 548, 569 Печать: 3D-изображений 431 выборочное лазерное спекание 433 разрешение принтера 157 Пирамида Гаусса 365 Подчеркивание границ 197 методом нерезкого маскирования 198 Полоски Маха 99 Помеха: аддитивная 39 биполярная импульсная 43 импульсная 259, 262, 266 коэффициент автокорреляции 40 мультипликативная 39, 253 начальный момент распределения 39 плотность вероятности распределения 39, 40 пространственной дискретизации 143 изображений 137 стационарная 39 структурная 44 флуктуационная 40 центральный момент распределения 40 Пороговая обработка 368 Принтер, разрешение 157

Ρ

Разделение кодируемого сигнала изображения на контексты 491 Растр 6 Рельефность 109 Реперная точка 292

С

Сверхразрешение 210 Светлота 37 Свойства зрения: адаптация к освещенности 90 восприятие: движения 104 объема 108 ивета 104 инерционность 101 контрастная чувствительность 92 острота зрения 98 пороговые условия наблюдения 100, 109 Связность пикселов 318 Сегментация изображений методом: К-методом 378 векторных признаков 376 выращивания областей 370 лвижения 372 интерактивным 380 пороговой обработки 368 разделения и слияния областей 371 связывания пирамиды 365 статистических характеристик текстуры 378 эрозии и наращивания 374 Секвента 514 Сжатие видеозаписи 448 Сжатие изображений 473 без потери информации 474 с использованием: вейвлет-преобразования 526 дискретного косинусного преобразования 514 ортогональных преобразований 510 распределения двоичных единиц кода между спектральными коэффициентами 516 с потерями информации 476

Сканер: 3D 411 барабанный 159 контактный 412 лазерный 413 на основе эффекта диффузного отражения света 420 на осное структурированного света 416 на основе метода фокусного ряда 414 планшетный 160 со стереоэффектом 415 Слой 80 Спектр: амплитудный 13, 15 изображения 17 пространственный 16 разности двух изображений 17, 18 фазовый 13, 15 Спектральная интенсивность: изображения 18 помехи 41 Спецэффекты 457 Спрайт 455 Среда 455 Стереолитография 431 Стереоэффект 108, 387 Структура расположения отсчетов: диагональная 148 ортогональная 136 треугольная 148 шахматная 148 Структурообразующий элемент 325 Сцена с большим динамическим диапазоном: методы отображения 237 методы фотографирования 234

Т

Текстура: выделение из изображения 247 гипотеза Юлеша 15, 122 наложение на оболочку объекта 293 Телевидение: мобильное 464 монтаж программы 447 объемное 466 Температура, цветовая 65 Теорема Котельникова 134 Тоновая коррекция 233, 277 Трассировка лучей 408 Тренажеры 566

У

Удельная координата 54 Узел 549 Уровни квантования 230

Φ

Файл графический, структура 543 Фильтр: адаптивный 181 Байера 161, 308 Бакуса — Гильберта 188, 190 Винера 180, 188, 189 инверсный 183 контргармонический 268 линейный 181 низкочастотный 189 среднегармонический 267 среднегеометрический 267 Фильтрация: адаптивная 180 медианная 263 анизотпропная 191 билатеральная 241 в спектральной области 169 гомоморфная 253 зашумленного изображения 181 инверсная 183 линейная 163. 183 логическая 497 логической апертурой 350 медианная 259 методом свертки 164 ранговая 263 распределение прозрачности в апертуре 176 цветного изображения 171 Формат: записи изображений 543 растров цветных изображений 443 цифровой видеозаписи 464 Формат файлов анимации 457

Функциональная модель зрения: обобщенная 113 Холла 111 Функция: автоковариации изображений 22 гашения 340 рассеяния точки 114, 117 частотная передаточная 99, 118 частотно-передаточная 18

Ц

Цвет: доминирующая длина волны 38 калибровка 75 насыщенность 38 светлота 37 управление цветом 74 цветовая температура 65 цветовой канал 80 иветовой тон 38 чистота 38 Цветовой тон 38 Цифровая камера: видео 161 фото 161 Цифровая обработка изображений 133

Ч

Чистота цвета 38

ш

Шаблон 326 интегральный 343 Шкала квантования: неравномерная 150 равномерная 150, 152 Шум 109 белый 41, 179 гауссов, ослабление 246 квазибелый 41 квантования 151 преобразования 494 считывания 44, 45 темновой 44 флуктуационный 14, 40, 373 матрицы ПЗС 46 фотонный 44

Э

Эффект: уменьшения угловых размеров объекта 108 экранирования 109

Я

Язык VRML 549 комментарий 550 оператор 550 поле узла 551 тело узла 551 Яркость 38



Цв. 1. Цветовое преобразования изображений в области, ограниченной произвольным контуром: *a* — исходное изображение; *б* — изображение с наложенной на него маской; *e* — преобразованное изображение



Цв. 2. Феномен окрашенной тени

а	КИНО	КИНО	КИНО	КИНО	КИНО
б	КИНО	КИНО	КИНО	КИНО	КИНО
в	КИНО	KRINO	КИНО	КИНО	КИНО
г	КИНО	КИНО	КИНО	КИНО	КИНО

Цв. 3. Здесь: *а* — исходное тестовое изображение; *б* — то же изображение, но с уменьшенной в три раза шириной спектров цветоразностных компонентов; *в* — то же изображение, но с уменьшенной в три раза шириной спектра яркостного компонента; *е* — то же изображение, но с уменьшенной в три раза шириной спектров яркостного и цветоразностных компонентов



Цв. 4. Здесь: *а* — исходное сюжетное изображение; *б* — то же изображение, но с уменьшенной в три раза шириной спектров цветоразностных компонентов; *в* — то же изображение, но с уменьшенной в три раза шириной спектра яркостного компонента; *г* — то же изображение, но с уменьшенной в три раза шириной спектров яркостного и цветоразностных компонентов



Цв. 5. Изображения: *а* — недодержанное; *б* — нормально экспонированное; *в* — передержанное; *г* — полученное описанным выше методом. Изображения (*a*), (*б*), (*в*) взяты из [163]





Цв. 6. Иллюстрация действия билатерального фильтра: *а* — исходное изображение; б — изображение, полученное в результате билатеральной фильтрации



Цв. 7. Фильтр Байера



Цв. 8. Анаглифное изображение