

ГЛАВА 3

ЯРКОСТНЫЕ ПРЕОБРАЗОВАНИЯ И ПРОСТРАНСТВЕННАЯ ФИЛЬТРАЦИЯ

*Вся разница в том, что именно мы хотим увидеть:
темное на светлом или светлое на темном.*

Дэвид Линдсей

Введение

Термин *пространственная область* относится к плоскости изображения как таковой, и методы обработки изображений, относящиеся к данной категории, основаны на прямых операциях с пикселями изображения. Это составляет противоположность обработке изображений в *трансформационной области*, которая, как введено в разделе 2.6.7 и более подробно обсуждается в главе 4, означает первоначально преобразование изображения в трансформационную область, выполнение обработки в ней и осуществление обратного преобразования для перевода результата снова в пространственную область. Двумя главными классами пространственной обработки являются яркостные преобразования и пространственная фильтрация. Как будет показано в этой главе, яркостные преобразования оперируют отдельными пикселями изображения, главным образом с целью управления контрастом и пороговыми операциями над изображением. Пространственная фильтрация осуществляет выполнение таких операций, как повышение резкости изображения, оперируя над окрестностью каждой точки изображения. В последующих разделах будет рассмотрен ряд «классических» методов яркостного преобразования и пространственной фильтрации. Также будут обсуждены некоторые детали нечетких методов, которые позволяют объединить неточную, взятую из базы знаний информацию в формулировке алгоритмов яркостных преобразований и пространственной фильтрации.

3.1. Предпосылки

3.1.1. Основы яркостных преобразований и пространственной фильтрации

Все методы обработки изображений, обсуждаемые в настоящем разделе, реализованы в *пространственной области*, которая, как мы знаем из раздела 2.4.2, есть просто плоскость, содержащая пиксели изображения. Как отмечено в раз-

деле 2.6.7, пространственные методы оперируют непосредственно пикселями изображения, в противоположность, например, частотным методам (что является предметом обсуждения главы 4), в которых операции выполняются над результатами Фурье-преобразования изображения, а не над самим изображением. Как вы узнаете из дальнейшего прочтения книги, одни задачи обработки изображений легче интерпретируются при реализации в пространственной области, тогда как для других лучше годятся иные подходы. Как правило, пространственные методы в вычислительном отношении являются более эффективными и требуют меньших вычислительных ресурсов при реализации.

Процессы пространственной обработки, рассматриваемые в настоящей главе, описываются уравнением

$$g(x,y)=T[f(x,y)], \quad (3.1-1)$$

где $f(x,y)$ — входное изображение, $g(x,y)$ — обработанное изображение, а T — оператор над f , определенный в некоторой окрестности точки (x,y) . Оператор T может быть применим к одному изображению (именно этому уделено основное внимание в данной главе) или же к набору изображений, например для выполнения поэлементного суммирования последовательности изображений с целью подавления шума, как рассматривалось в разделе 2.6.3. На рис. 3.1 показан основной вариант реализации уравнения (3.1-1) при обработке одного изображения. Отмеченная точка (x,y) может находиться в любом месте изображения, а окружающая ее небольшая отмеченная область является окрестностью точки (x,y) , как было указано в разделе 2.6.5. Обычно окрестность является прямоугольной, центрированной относительно (x,y) и значительно меньшей в размерах, чем само изображение.

Иногда используются окрестности другой формы, такие как дискретное приближение к кругу, однако почти всегда превалирует прямоугольная форма, потому что она значительно проще при реализации вычислений.

Рис. 3.1 иллюстрирует процесс обработки, при котором осуществляется переход начала координат окрестности от точки к точке и применения оператора T к пикселям окрестности для получения выходного значения в текущей точке. Таким образом, для любой выбранной точки (x,y) исходного изображения f выходное значение g вычисляется в результате применения T к элементам окрестности точки (x,y) . Предположим, например, что окрестность является квадратом размерами 3×3 элементов и оператор T определен как «вычислить среднюю яркость по окрестности». Рассмотрим произвольную точку изображения, например имеющую координаты $(100, 150)$. Полагая, что начало координат окрестности расположено в ее центре, результат $g(100, 150)$ вычисляется как сумма значений самой точки $f(100, 150)$ и 8 ее соседей, деленная на 9 (т. е. средняя яркость точек, попадающих в окрестность). Затем начало координат окрестности передвигается на соседнюю точку и процедура повторяется для получения значения следующего элемента выходного изображения g . Как правило, обработка начинается с верхнего левого угла исходного изображения и продолжается последовательно для всех пикселей строки по горизонтали, по одной строке за проход. Когда начало координат окрестности попадает на границу изображения, часть

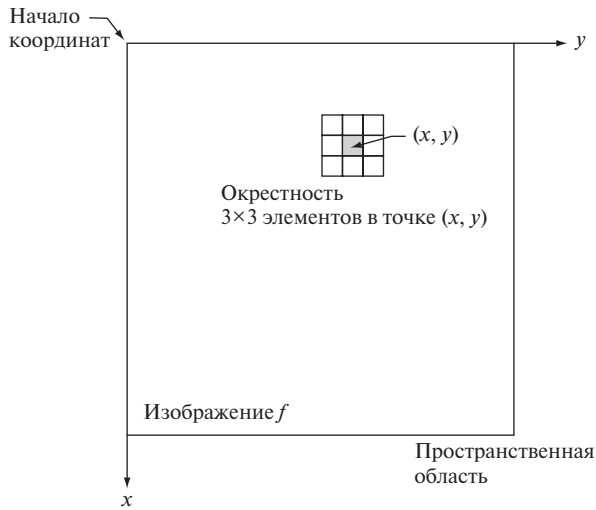


Рис. 3.1. Окрестность 3×3 вокруг точки (x, y) изображения в пространственной области. При формировании выходного изображения окрестность передвигается от точки к точке изображения

окрестности оказывается вне изображения. В таком случае при выполнении вычислений, заданных оператором T , приходится либо игнорировать элементы, выходящие за границы изображения, либо присваивать им нулевые или какие-то другие значения. Ширина такой полосы расширения изображения зависит от размера окрестности. Мы еще вернемся к этому вопросу в разделе 3.4.1.

Только что описанный процесс обработки подробно рассмотрен в разделе 3.4 и называется *пространственной фильтрацией*, в которой окрестность вместе с заданной операцией называется *пространственным фильтром* (также используются названия *маска*, *ядро* или *окно*)¹.

Наименьшая возможная окрестность имеет размеры 1×1 . В этом случае g зависит только от значения f в точке (x, y) , и T в уравнении (3.1-2) становится *функцией градационного преобразования*, также называемой *функцией преобразования яркости* или *функцией отображения*, имеющей вид

$$s = T(r), \quad (3.1-2)$$

где для простоты обозначения r и s суть переменные, обозначающие соответственно значения яркостей изображений f и g в каждой точке (x, y) . Например, если $T(r)$ имеет вид, показанный на рис. 3.2(a), то эффект от применения этого преобразования к каждому пикселю изображения f для получения соответствующего пикселя изображения g выразится в формировании изображения более высокого контраста по сравнению с оригиналом посредством затемнения пикселей со значениями, меньшими k , и повышении яркостей пикселей со значениями большими k . В этом методе, иногда называемом *усилением контраста* (см. раздел 3.2.4), значения r , меньшие k , при приближении к уровню черного сжимаются с помощью функции преобразования во все более узкий диапазон s .

¹ Понятие «окно» обычно означает лишь пространственную совокупность элементов исходного изображения без учета оператора T . — *Прим. перев.*

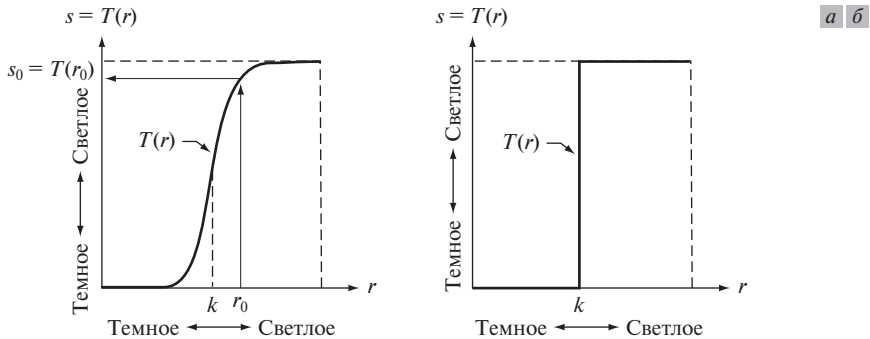


Рис. 3.2. Функция градационного преобразования. (а) Функция повышения контраста. (б) Пороговая функция

Обратный эффект имеет место для значений r , больших k . Обратим внимание, как значение яркости r_0 отображается в соответствующее значение s_0 . В предельном случае, показанном на рис. 3.2(б), $T(r)$ дает в результате двухградиционное (бинарное) изображение. Отображение такой формы называют *пороговой функцией*. С помощью градационных преобразований могут быть построены некоторые довольно простые, но действенные методы обработки изображений. В данной главе мы используем яркостные преобразования главным образом для улучшения изображений. В главе 10 они будут использованы для сегментации изображений. Подходы, в которых результирующее значение элемента зависит лишь от яркости соответствующего исходного элемента, называются методами *поэлементной обработки* в противоположность методам обработки по окрестности, обсуждаемым далее в данном разделе.

3.1.2. О примерах, приводимых в данной главе

Хотя яркостные преобразования и пространственная фильтрация охватывают весьма широкий диапазон приложений, большинство примеров в данной главе связаны с их применениями в области улучшения изображений. *Улучшение* означает такую процедуру преобразования изображения, что результат оказывается более применимым для конкретного приложения, чем оригинал. Слово *конкретный* является здесь важным, поскольку с самого начала постулирует, что методы улучшения являются проблемно-ориентированными. Так, например, метод, весьма полезный для улучшения рентгеновских изображений, может не оказаться наилучшим для улучшения изображений со спутника, полученных в инфракрасном участке спектра. Не существует общей «теории» улучшения изображений. Когда изображение обрабатывается с целью визуальной интерпретации, то конечным судьей, решающим, насколько хорошо работает тот или иной метод, является человек. Дать количественную оценку конкретной процедуре обработки проще в случае, когда мы имеем дело с машинным восприятием. Например, в системе автоматического распознавания символов наиболее подходящим методом улучшения является тот, который приводит к наилучшему уровню распознавания, оставляя в стороне другие критерии, такие как вычислительные требования того или иного метода.

внимания тот факт, что яркие области в левой части остались почти неизменными, что являлось одним из первоначальных требований. ■

3.4. Основы пространственной фильтрации

В настоящем разделе выносятся на рассмотрение несколько основных концепций, лежащих в основе пространственной фильтрации в обработке изображений. Пространственная фильтрация является одним из основных инструментов, используемых в данной области для широкого спектра приложений, поэтому крайне рекомендуется выработать твердое понимание этих концепций. Как отмечалось в начале этой главы, приводимые здесь примеры в основном имеют отношение к применению пространственных фильтров для улучшения изображений. Другие приложения пространственной фильтрации рассматриваются в дальнейших главах.

Термин *фильтр* заимствован из области частотной фильтрации, которая есть тема следующей главы, где «фильтрация» относится к принятию (пропусканию) или подавлению определенных частотных компонент. Например, фильтр, который пропускает нижние частоты, называется *низкочастотным* фильтром. Цель, на которую направлен эффект, производимый низкочастотным фильтром, заключается в размывании (сглаживании) изображения. Подобное сглаживание можно выполнить прямо по изображению путем использования пространственных фильтров (также называемых пространственными *масками*, *ядрами*, *шаблонами* или *окнами*)⁸. Фактически, как будет показано в главе 4, имеется взаимно однозначное соответствие между линейными пространственными фильтрами и фильтрами в частотной области. Однако пространственные фильтры предлагают значительно большую гибкость, поскольку, как будет видно далее, они могут быть использованы также для нелинейной фильтрации, что не может быть реализовано в частотной области.

См. раздел 2.6.2 касательно линейности.

3.4.1. Механизмы пространственной фильтрации

На рис. 3.1 мы кратко обрисовали, что пространственный фильтр состоит из (1) *окрестности* (обычно небольшого прямоугольника) и (2) *заданной операции*, которая выполняется над пикселями изображения, попадающими в окрестность. Фильтрация создает новый пиксель, значение которого зависит от оператора фильтра, а координаты совпадают с координатами центра окрестности.⁹

⁸ Также часто используется термин *оператор*. — Прим. перев.

⁹ Полученное на выходе фильтра значение обычно присваивается пикселю в соответствующем месте нового изображения, создаваемого как выход результата фильтрации изображения. В редких случаях значение на выходе фильтра замещает первоначальное значение пикселя в исходном изображении; в такой ситуации новое значение будет менять содержание изображения, все еще находящегося в процессе обработки.

Обработанное (фильтрованное) изображение возникает в процессе сканирования исходного изображения фильтром. Если оператор, выполняемый над пикселями исходного изображения, является линейным, то фильтр называют *линейным пространственным фильтром*. В противном случае фильтр является *нелинейным*. Первоначально внимание будет сосредоточено на линейных фильтрах, а затем будут проиллюстрированы некоторые простые нелинейные фильтры. Раздел 5.3 содержит более полный список нелинейных фильтров и их приложений.

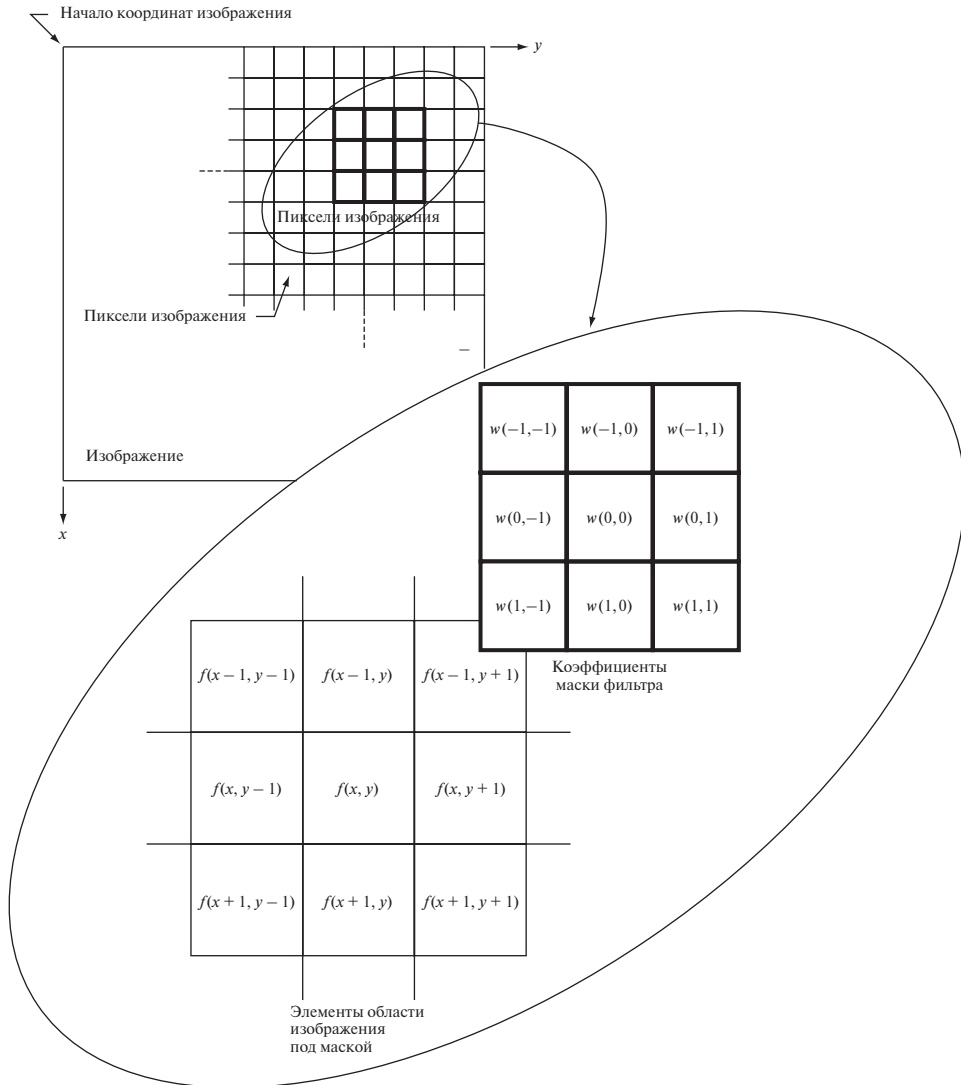


Рис. 3.28. Схема линейной пространственной фильтрации при использовании маски фильтра 3×3 . Выбранная форма обозначения координат коэффициентов маски фильтра упрощает запись выражений для линейной фильтрации

На рис. 3.28 иллюстрируется схема линейной пространственной фильтрации с использованием окрестности 3×3 пикселей. В каждой точке (x, y) изображения отклик $g(x, y)$ задается суммой произведений коэффициентов фильтра на соответствующие значения пикселей в области, покрытой маской фильтра:

$$g(x, y) = w(-1, -1)f(x-1, y-1) + w(-1, 0)f(x-1, y) + \dots + w(0, 0)f(x, y) + \dots + w(1, 0)f(x+1, y) + w(1, 1)f(x+1, y+1).$$

Заметим, что центральный коэффициент $w(0, 0)$ стоит при значении пикселя в точке (x, y) . В случае маски размерами $m \times n$ будем полагать, что $m = 2a + 1$ и $n = 2b + 1$, где a и b суть неотрицательные целые. Это означает, что в дальнейшем будут рассматриваться маски нечетных размеров, причем наименьшей будет маска 3×3 элемента. В общем виде фильтрация изображения размерами $M \times N$ с помощью фильтра размерами $m \times n$ задается выражением

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t)f(x+s, y+t),$$

где x и y изменяются так, что каждая точка оператора w в какой-то момент попадает на каждый пиксель изображения f .

В принципе, возможно работать и с окрестностями четного размера и даже смесью четного и нечетного размеров. Однако работа с нечетными размерами упрощает индексирование и является интуитивно более естественной, поскольку центр в таких фильтрах имеет целочисленные координаты.

3.4.2. Пространственная корреляция и свертка

Существуют две близкородственные концепции, которые при выполнении линейной пространственной фильтрации необходимо четко понимать. Одна из них — *корреляция*, а другая — *свертка*. Корреляция есть процесс движения маски фильтра по изображению и вычисление суммы произведений значений элементов маски и значений пикселей, на которые попадают соответствующие элементы маски, для всех точек изображения; в точности, как объяснялось в предыдущем разделе. Механизмы свертки такие же, но за исключением того, что предварительно маска фильтра поворачивается на 180° . Лучшим способом объяснить различия между двумя концепциями является пример. Начнем с одномерной иллюстрации.

На рис. 3.29(а) показана одномерная функция f вместе с некоторым фильтром w , а рис. 3.29(б) показывает начальную позицию для выполнения корреляции. Первое, что мы замечаем, — это что имеются участки функций, которые не перекрываются. Решение этой проблемы состоит в расширении f добавлением нулей с обеих сторон в количестве, достаточном для перекрытия с w . Если фильтр имеет размер m , то необходимо добавить $m - 1$ нулей с каждой стороны f . На рис. 3.29(в) показана корректно расширенная функция. Первым значением корреляции является сумма произведений элементов f и w для начальной позиции, показанной на том же рисунке (сумма произведений равна 0). Этот результат соответствует смещению $x = 0$. Чтобы получить второе значе-

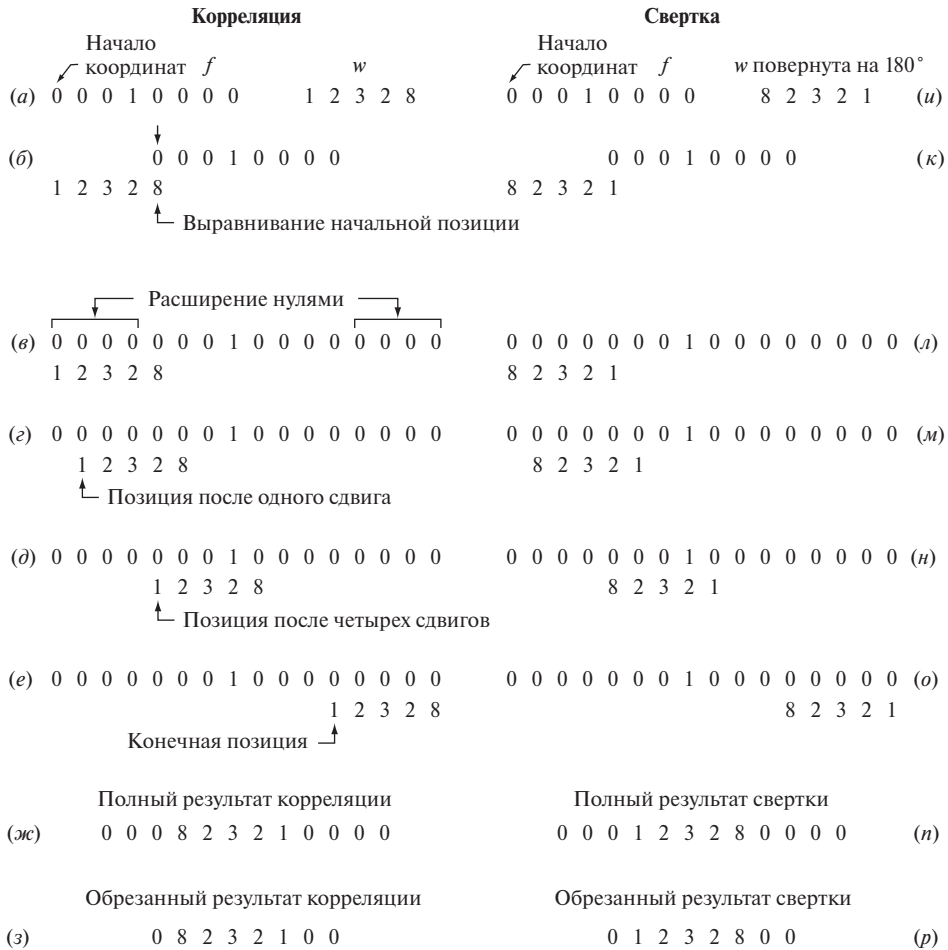


Рис. 3.29. Иллюстрация одномерной корреляции и свертки фильтра с дискретным единичным импульсом. Заметим, что и корреляция, и свертка являются функциями *смещения*

ние корреляции, w сдвигается вправо на один пиксель (смещение $x = 1$), и опять вычисляется сумма произведений. Результат опять равен 0. В действительности первый ненулевой результат появится при $x = 3$, когда восьмерка в w наползет на единичку в f и результат корреляции будет равен 8. Продолжая в таком духе, мы получим полный результат корреляции, показанный на рис. 3.29(ж). Заметим, что в итоге получилось 12 значений x (т.е. $x = 0, 1, 2, \dots, 11$) для полного сдвига w по f , так что каждый элемент w соответствовал каждому пикселю f . Часто удобно работать с корреляционными массивами, имеющими тот же размер, что и f , и в таком случае приходится обрезать полную корреляцию до размеров исходной функции, как это показано на рис. 3.29(з).

Из предыдущего параграфа следует отметить два важных факта. Во-первых, корреляция есть функция *смещения* фильтра. Другими словами, значение корреляции в нулевой позиции соответствует нулевому смещению, значение в первой позиции — единичному смещению и так далее. Второй факт, который сле-

дует отметить, — это то, что корреляция фильтра w с функцией, содержащей единицу в одной точке и нули во всех остальных, приводит к результату, являющемуся копией w , но повернутому на 180° . Такую функцию называют *дискретным единичным импульсом*. Итак, можно заключить, что корреляция некоторой функции с дискретным единичным импульсом приводит к зеркальному повороту функции на 180° в месте положения импульса.

Расширение нулями не является единственным возможным вариантом. Так, например, можно $m - 1$ раз дублировать значения первого и последнего элементов f , или зеркально перевернуть $m - 1$ первых и последних элементов f и использовать их для расширения.

Концепция *свертки* является краеугольным камнем теории линейных систем. Как будет обсуждаться в главе 4, фундаментальным свойством свертки является то, что свертка функции с единичным импульсом дает в результате копию функции в месте импульса. В предыдущем параграфе было показано, что корреляция также дает в результате копию функции, однако повернутую на 180° . Таким образом, если предварительно повернуть фильтр и затем выполнить скользящую сумму операций умножения, то в результате мы должны получить желаемый результат. Как видно из правого столбца на рис. 3.29, это и в самом деле так. Таким образом, видно, что для выполнения свертки достаточно перевернуть одну из функций на 180° и выполнить те же операции, что и в случае корреляции. Оказывается, не имеет значения, которую из двух функций переворачивать.

Заметим, что поворот функции на 180° эквивалентен ее горизонтальному перевороту.

Рассмотренные выше принципы легко расширяются на изображения, что иллюстрируется на рис. 3.30. В случае фильтра размерами $m \times n$ изображение расширяется минимум на $m - 1$ строку нулей сверху и снизу, а также на $n - 1$ столбец нулей слева и справа. В данном случае m и n равны 3, так что f расширяется на две строки нулей сверху и снизу и на два столбца нулей справа и слева, как это видно на рис. 3.30(б). На рис. 3.30(в) показано начальное положение маски фильтра для выполнения корреляции. На рис. 3.30(г) представлен полный результат корреляции, а на рис. 3.30(д) — соответствующий обрезанный результат. Еще раз заметим, что в результате получили повернутые на 180° значения элементов фильтра. В случае свертки требуется предварительно, как и ранее, повернуть маску и затем выполнить скользящее суммирование произведений. Результат представлен на рис. 3.30(е)—(з). Еще раз видно, что свертка функции с единичным импульсом копирует функцию в месте импульса. Ясно, что если маска фильтра является симметричной, то и корреляция, и свертка дадут одинаковый результат.

В двумерном случае поворот на 180° эквивалентен перевороту маски сначала по одной оси, а затем по другой.

Начало ↙ координат		Расширенное f	
$f(x, y)$			
0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 1 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0		0 0 0 0 0 0 0 0 0	
	$w(x, y)$		
	1 2 3		
	4 5 6		
	7 8 9		
(a)		(б)	
Начальное ↙ положение w		Полный результат корреляции	Обрезанный результат корреляции
1 2 3	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 0 0 0 0
4 5 6	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 9 8 7 0
7 8 9	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 6 5 4 0
0 0 0 0 0 0 0 0 0		0 0 0 9 8 7 0 0 0	0 3 2 1 0
0 0 0 0 0 1 0 0 0 0		0 0 0 6 5 4 0 0 0	0 0 0 0 0
0 0 0 0 0 0 0 0 0 0		0 0 0 3 2 1 0 0 0	
0 0 0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0 0	
(в)		(г)	(д)
↙ Повернутая w		Полный результат свертки	Обрезанный результат свертки
9 8 7	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 0 0 0 0
6 5 4	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 1 2 3 0
3 2 1	0 0 0 0 0 0 0	0 0 0 0 0 0 0 0 0	0 4 5 6 0
0 0 0 0 0 0 0 0 0		0 0 0 1 2 3 0 0 0	0 7 8 9 0
0 0 0 0 0 1 0 0 0 0		0 0 0 4 5 6 0 0 0	0 0 0 0 0
0 0 0 0 0 0 0 0 0 0		0 0 0 7 8 9 0 0 0	
0 0 0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0		0 0 0 0 0 0 0 0 0	
(е)		(ж)	(з)

Рис. 3.30. Корреляция (средний ряд) и свертка (нижний ряд) двумерного фильтра с двумерным единичным импульсом. Нули показаны серым для удобства визуального анализа

Подводя итог обсуждению, сформулируем сказанное в аналитическом виде. Корреляция фильтра $w(x, y)$ размерами $m \times n$ с изображением $f(x, y)$, обозначаемая как $w(x, y) \star f(x, y)$, задается уравнением, записанным в конце предыдущего раздела, которое повторим здесь для удобства:

$$w(x, y) \star f(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x+s, y+t). \quad (3.4-1)$$

Это уравнение вычисляется для всех значений переменных смещения x и y , так что все элементы из w попадают на каждый из пикселей f , при этом предполагается, что f расширено соответствующим образом. Как объяснено ранее, $a = (m-1)/2$, $b = (n-1)/2$ и для удобства обозначений предполагается, что m и n — целые нечетные.

Подобным же образом свертка $w(x, y) \star f(x, y)$, обозначаемая как $w(x, y) \star f(x, y)$ ¹⁰, задается уравнением

$$w(x, y) \star f(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x-s, y-t), \quad (3.4-2)$$

где знак минус действует как смена порядка на обратный (т.е. поворот на 180°). Переворот и сдвиг f вместо w сделан для упрощения обозначений, а также следуя общепринятым правилам. Результат тот же самый. Как и в случае корреляции, это уравнение вычисляется для всех значений переменных смещения x и y , так что все элементы из w попадают на каждый из пикселей f , расширенного соответствующим образом. Вам следует раскрыть уравнение (3.4-2) для случая маски 3×3 и убедиться, что результат, получаемый с помощью данного уравнения, идентичен примеру на рис. 3.30. На практике часто приходится работать с алгоритмом, который реализует уравнение (3.4-1). Если необходимо выполнить операцию корреляции, следует на вход алгоритма подавать w ; если же необходима свертка, то предварительно следует повернуть w на 180°. Обратный вариант будет в случае, если реализован алгоритм по уравнению (3.4-2).

Достаточно часто, когда не возникает неясности, результат корреляции или свертки обозначают через $g(x, y)$, а не $w(x, y) \star f(x, y)$ или $w(x, y) \star f(x, y)$. Например, см. уравнение в конце предыдущего раздела или уравнение (3.5-1).

Как уже упоминалось ранее, свертка является краеугольным камнем теории линейных систем. Как будет показано в главе 4, свойство свертки функции с единичным импульсом копировать функцию в место расположения импульса играет центральную роль во многих важных выводах. Мы вернемся к свертке в главе 4 в контексте преобразования Фурье и теоремы о свертке. Однако, в отличие от уравнения (3.4-2), мы будем иметь дело со сверткой функций одинаковых размеров. Форма уравнения остается той же самой, но пределы суммирования становятся другими.

Использовать корреляцию или свертку для выполнения пространственной фильтрации — это вопрос предпочтения. Поскольку в реальности как формула (3.4-1), так и (3.4-2) могут быть использованы для выполнения функции другой, достаточно лишь перевернуть фильтр, важным становится то, что маска фильтра, используемая в данной задаче фильтрации, должна быть задана способом, соответствующим планируемой операции. В настоящей главе все результаты линейной пространственной фильтрации основываются на уравнении (3.4-1).

Наконец, мы обращаем внимание, что в литературе по обработке изображений встречаются термины *фильтр свертки*, *маска свертки* или *ядро свертки*. Как правило, эти термины используются для обозначения пространственного фильтра и не обязательно, чтобы этот фильтр использовался именно для свертки. Аналогично фраза «свертка маски с изображением» часто используется для обозначения скользящего процесса суммирования произведений, который мы только что рассмотрели, а не обязательно для дифференциации между корреля-

¹⁰ Поскольку свертка является коммутативной операцией, имеем $w(x, y) \star f(x, y) = f(x, y) \star w(x, y)$. Это свойство не выполняется для корреляции, как можно видеть, например, поменяв местами функции на рис. 3.29(a).

цией и сверткой. Вернее, она используется для общего указания любой из этих двух операций. Эта неточность терминологии является частым источником путаницы.

3.4.3. Векторное представление линейной фильтрации

Когда интерес представляет характеристика отклика R некоторой маски (либо корреляции, либо свертки), иногда удобно записать сумму произведений в виде

$$R = w_1 z_1 + w_2 z_2 + \dots + w_{mn} z_{mn} = \sum_{k=1}^{mn} w_k z_k = \mathbf{w}^T \mathbf{z}, \quad (3.4-3)$$

где множество w — коэффициенты фильтра размерами mn , а z — соответствующие значения яркостей пикселей изображения, покрываемых фильтром. Чтобы использовать это уравнение для свертки, мы просто поворачиваем маску на 180° , как объяснено в предыдущем разделе. Подразумевается, что уравнение (3.4-3) справедливо для конкретной пары координат (x, y) . В следующем разделе будет видно, почему такая форма записи удобна для объяснения характеристик заданного линейного фильтра.

Обратитесь к разделу обучающих материалов на интернет-сайте книги для краткого обзора векторов и матриц.

В качестве примера на рис. 3.31 показана общая маска 3×3 с коэффициентами, обозначенными так же, как и выше. В этом случае уравнение (3.4-3) принимает вид

$$R = w_1 z_1 + w_2 z_2 + \dots + w_9 z_9 = \sum_{k=1}^9 w_k z_k = \mathbf{w}^T \mathbf{z}, \quad (3.4-4)$$

где \mathbf{w} и \mathbf{z} — 9-мерные вектора, сформированные из коэффициентов маски и, соответственно, значений яркостей пикселей изображения, покрываемых фильтром.

3.4.4. Формирование масок пространственных фильтров

Для формирования *линейного* пространственного фильтра размерами $m \times n$ требуется задать mn коэффициентов маски. И наоборот, выбор этих коэффициен-

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Рис. 3.31. Другой вариант представления часто используемой маски фильтра по окрестности 3×3 элемента

тов базируется на том, какие действия ожидаются от фильтра при условии, что в случае линейной фильтрации мы можем реализовать лишь сумму произведений. Предположим, например, что требуется заменить значение пикселя на изображении значением средней яркости по окрестности 3×3 с центром в данном пикселе. Среднее значение в любой точке (x, y) изображения есть сумма девяти значений яркости в окрестности 3×3 с центром в (x, y) , деленная на 9. Полагая, что z_i , $i = 1, 2, \dots, 9$, обозначает эти яркости, среднее будет равно

$$R = \frac{1}{9} \sum_{i=1}^9 z_i.$$

Но эта формула совпадает с (3.4-4) при значениях коэффициентов $w_i = 1/9$. Другими словами, операция линейной фильтрации по маске 3×3 , коэффициенты которой равны $1/9$, выполняет желаемое усреднение. Как будет обсуждаться в следующем разделе, эта операция приводит к сглаживанию изображения. Далее будут обсуждаться многие другие маски фильтров, основанных на этом фундаментальном подходе.

В некоторых приложениях встречаются непрерывные функции двух переменных, и целью является получение маски пространственного фильтра, основанного на таких функциях. Например, гауссова функция двух переменных имеет основной вид

$$h(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}},$$

где σ означает стандартное отклонение и, как обычно, предполагается, что координаты x и y являются целыми. Чтобы сформировать из этой функции, скажем, маску фильтра 3×3 , берутся значения этой функции вокруг ее центра. Так, $w_1 = h(-1, -1)$, $w_2 = h(-1, 0)$, ..., $w_9 = h(1, 1)$. Маска фильтра размерами $m \times n$ формируется аналогичным образом. Напомним, что двумерная гауссова функция имеет колоколообразную форму и что стандартное отклонение задает степень «растянутости» колокола.

Формирование *нелинейного* фильтра требует, чтобы был задан размер окрестности и операция (операции), выполняющаяся над значениями пикселей изображения, попадающими в окрестность. Например, напоминая, что операция выбора максимума является нелинейной (см. раздел 2.6.2), фильтр максимума по окрестности 5×5 , центрированный в любой точке (x, y) изображения, дает в результате значение максимума из 25 пикселей и присваивает его пикселю обрабатываемого изображения в точке (x, y) . Нелинейные фильтры являются весьма мощными средствами и в некоторых приложениях могут осуществлять операции, выходящие далеко за возможности линейных фильтров, как будет видно из последующих обсуждений в настоящей главе, а также в главе 5.

3.5. Сглаживающие пространственные фильтры

Сглаживающие фильтры применяются для расфокусировки изображения и подавления шума. Расфокусировка может применяться в задачах предварительной обработки изображения, например для удаления мелких деталей перед обнаружением больших объектов или же для устранения разрывов в линиях или

деталей. Для подавления шумов может использоваться расфокусировка с применением как линейной, так и нелинейной фильтрации.

3.5.1. Линейные сглаживающие фильтры

Выход (отклик) простейшего линейного сглаживающего пространственного фильтра есть среднее значение элементов по окрестности, покрытой маской фильтра. Такие фильтры иногда называют *усредняющими* или *сглаживающими фильтрами*. Как упоминалось в предыдущем разделе, их также называют *низкочастотными фильтрами*.

Идея применения сглаживающих фильтров достаточно ясна. Заменой исходных значений элементов изображения на средние значения по маске фильтра достигается уменьшение «резких» переходов уровней яркости. Поскольку случайный шум как раз характеризуется резкими скачками яркости, наиболее очевидным применением сглаживания является подавление шума. Однако контуры (которые обычно представляют интерес на изображении) также характеризуются резкими перепадами яркостей, поэтому негативной стороной применения сглаживающих фильтров является расфокусировка контуров. Другим применением такой процедуры может быть сглаживание ложных контуров, которые возникают при преобразованиях с недостаточным числом уровней яркости, как это обсуждалось в разделе 2.4.3. Главное использование сглаживающих фильтров состоит в подавлении «несущественных» деталей на изображении. Под «несущественными» здесь понимаются совокупности пикселей, которые малы по сравнению с размерами маски фильтра. Это последнее применение будет проиллюстрировано ниже.

На рис. 3.32 показаны два сглаживающих фильтра по окрестности 3×3 . Первый из них дает обычное среднее значение по маске. Подстановкой коэффициентов маски в уравнение (3.4-4) получим

$$R = \frac{1}{9} \sum_{i=1}^9 z_i,$$

что как раз и дает среднее значение яркостей по окрестности 3×3 , как отмечалось ранее. Заметим, что коэффициенты фильтра указаны как единицы вместо $1/9$. Причина в том, что такой вариант является более эффективным при ком-

а б

$\frac{1}{9} \times$	1	1	1
	1	1	1
	1	1	1

$\frac{1}{16} \times$	1	2	1
	2	4	2
	1	2	1

Рис. 3.32. Две маски сглаживающих (усредняющих) фильтров по окрестности 3×3 . Постоянный множитель перед каждой из масок равен единице, деленной на сумму значений коэффициентов, как это необходимо для нормировки

пьютерных вычислениях. По окончании процесса суммирования полученное значение делится на 9. Маска размерами $m \times n$ будет иметь нормировочный коэффициент, равный $1/mn$. Такой пространственный фильтр, все коэффициенты которого одинаковы, иногда называют *однородным усредняющим фильтром*.

Вторая маска на рис. 3.32 несколько более интересна. Эта маска дает так называемое *взвешенное среднее*; этот термин применяется, чтобы показать, что значения элементов суммируются с разными коэффициентами, что позволяет присвоить им как бы разные «важности» (веса) по сравнению с другими. В маске на рис. 3.32(б) коэффициент в центре маски имеет самое большее значение (вес), тем самым давая соответствующему элементу большую важность при вычислении среднего. Значения остальных коэффициентов в маске уменьшаются по мере удаления от центра маски. Диагональные члены, по сравнению с ортогональными, расположены от центра дальше и, таким образом, «вешат» меньше, чем ближайшие соседи центрального элемента. Основная стратегия присвоения центральному пикселю наибольшего веса, а остальным — обратно пропорционально их расстоянию, имеет целью уменьшение расфокусировки при сглаживании. Можно было бы выбрать и другие значения коэффициентов маски для достижения поставленной цели, но сумма коэффициентов, приведенных на рис. 3.32(б), равна 16, что удобно при компьютерной реализации, поскольку это степень двойки. Следует отметить, что на практике достаточно трудно увидеть разницу между изображениями, сглаженными фильтрами по одной из масок на рис. 3.32 или какими-то другими, аналогичными по конструкции, поскольку размеры области, покрываемые маской при фильтрации одного элемента, очень малы.

Как следует из уравнения (3.4-1), общая формула фильтрации изображения размерами $M \times N$ фильтром взвешенного среднего по окрестности $m \times n$ (m и n — нечетные) задается выражением

$$g(x, y) = \frac{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x+s, y+t)}{\sum_{s=-a}^a \sum_{t=-b}^b w(s, t)}. \quad (3.5-1)$$

Параметры в этом уравнении такие же, как были определены для уравнения (3.4-1). Как и ранее, подразумевается, что полная фильтрация изображения достигается применением формулы (3.5-1) ко всем парам $x = 0, 1, 2, \dots, M-1$ и $y = 0, 1, 2, \dots, N-1$. Знаменатель в (3.5-1) есть просто сумма всех коэффициентов маски, следовательно, является константой и требует вычисления лишь один раз.

Пример 3.13. Сглаживание изображения по маскам различных размеров.

■ Зависимость эффекта сглаживания от размера фильтра проиллюстрирована на рис. 3.33, где представлено исходное изображение и результаты сглаживания усредняющим фильтром с квадратной окрестностью размерами $m = 3, 5, 9, 15$, и 35 элементов. Главные результаты этого эксперимента таковы. При $m = 3$ можно заметить легкую расфокусировку на всем изображении, однако, как и следовало ожидать, более всего искажены мелкие детали, имеющие размеры, близкие к размерам фильтра. На изображениях, сглаженных по окрестностям 3×3 и 5×5 , уже видна значительная расфокусировка на маленькой букве «а» и на мелкозернистом шуме. Заметно заглажены выступающие края букв, а также серые кру-



Рис. 3.33. (а) Исходное изображение размерами 500×500 элементов. (б)—(е) Результаты сглаживания усредняющим фильтром с квадратной маской размерами $m = 3, 5, 9, 15$ и 35 . Черные квадратики в верхней части имеют размеры 3, 5, 9, 15, 25, 35, 45 и 55 пикселей соответственно и расположены на расстоянии в 25 пикселей друг от друга. Буквы внизу изменяются в размерах от 10 до 24 пунктов, с увеличением каждый раз на 2 пункта; размер самой большой буквы — 60 пунктов. Вертикальные полосы имеют 5 пикселей в ширину и 100 пикселей в высоту; расстояния между ними — 20 пикселей. Диаметр кружков — 25 пикселей, и расположены они на расстоянии 15 пикселей друг от друга; их уровни яркостей изменяются от 0 % до 100 % черного с шагом 20 %. Общий фон изображения — 10 % черного. Прямоугольники с шумом имеют размеры 50×120 пикселей

жочки. Заметим, что шум стал значительно менее выраженным, а зубцы букв стали сглаженными.

Небольшое увеличение размеров окрестности сглаживания (больше $m = 5$) не приводит к сильным изменениям результатов. Но при $m = 9$ можно видеть уже значительно более сильную расфокусировку и 20-процентный черный кружочек уже не так сильно отличается от фона, как на предыдущих трех изображениях, тем самым иллюстрируя эффект смешивания, являющийся результатом расфокусировки. Также видно значительно более сильное сглаживание зашумленных прямоугольников. Результаты с $m = 15$ и 35 следует считать предельными по отношению к размерам объектов на изображении. Такое чрезмерное сглаживание используется для удаления с изображения мелких объектов. Например, три маленьких квадрата, два круга и большинство из зашумленных прямоугольников на рис. 3.33(е) неотличимы от фона. На этом изображении стал сильно заметен темный бордюр, который является следствием расширения изображения перед операцией сглаживания нулями (т.е. черным фоном) и последующего обрезания краев после фильтрации. Этот эффект проявляется на всех изображениях, но наиболее заметен на изображении, сглаженном фильтром самого большого размера. ■

Как отмечалось выше, важным применением пространственного сглаживания является расфокусировка изображения, позволяющая создать грубый образ объектов, которые могут представлять интерес. При этом яркость мелких объектов смешивается с фоном, в то время как большие объекты остаются в виде пятен и могут быть легко обнаружены. Размеры объектов, которые будут смешиваться с фоном, приблизительно совпадают с размерами маски сглаживающего фильтра. В качестве примера рассмотрим изображение на рис. 3.34(а), полученное телескопом «Хаббл» с орбиты Земли. Результат применения усредняющего фильтра с маской 15×15 элементов показан на рис. 3.34(б). Можно видеть, что многие из объектов или оказались смешаны с фоном, или же их яркость значительно уменьшилась. Обычно после такого сглаживания следует опера-

а б в

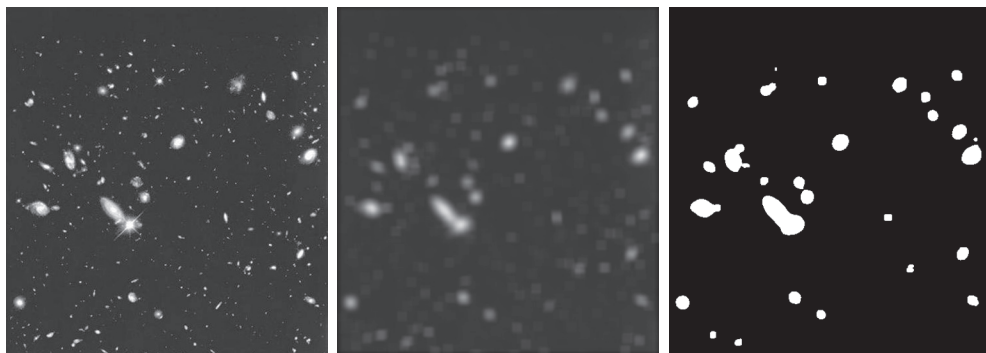


Рис. 3.34. (а) Изображение размерами 528×485 пикселей, полученное космическим телескопом «Хаббл». (б) Изображение, обработанное сглаживающей маской размерами 15×15 элементов. (в) Результат применения порогового обнаружения к изображению (б). (Исходное изображение предоставлено агентством NASA)

ция разделения по порогу, позволяющая убрать объекты малой интенсивности. Результат применения пороговой операции к изображению на рис. 3.2(б) с уровнем порога в 25 % от наибольшей яркости показан на рис. 3.34(в). Сравнивая это изображение с исходным, можно сделать вывод, что полученный результат был бы приемлемым с позиции поиска самых больших и ярких объектов.

3.5.2. Фильтры, основанные на порядковых статистиках (нелинейные фильтры)

Фильтры, основанные на порядковых статистиках, относятся к классу нелинейных пространственных фильтров. Отклик такого фильтра определяется предварительным упорядочиванием (ранжированием) значений пикселей, покрываемых маской фильтра, и последующим выбором значения, находящегося на определенной позиции упорядоченной последовательности (т.е. имеющего определенный ранг). Собственно фильтрация сводится к замещению исходного значения пикселя (в центре маски) на полученное значение отклика фильтра. Наиболее известен *медианный фильтр*, который, как следует из названия, заменяет значение пикселя на значение медианы распределения яркостей всех пикселей в окрестности (включая и исходный). Медианные фильтры весьма популярны потому, что для определенных типов случайных шумов они демонстрируют отличные возможности подавления шума при значительно меньшем эффекте расфокусировки, чем у линейных сглаживающих фильтров с аналогичными размерами. В частности, медианные фильтры эффективны при фильтрации *импульсных шумов*, иногда называемых шумами «соль и перец», которые выглядят как наложение на изображение случайных черных и белых точек.

Медиана набора чисел есть такое число ξ , что половина чисел из набора меньше или равны ξ , а другая половина — больше или равны ξ . Чтобы выполнить медианную фильтрацию для элемента изображения, необходимо сначала упорядочить по возрастанию значения пикселей внутри окрестности, затем найти значение медианы и, наконец, присвоить полученное значение обрабатываемому элементу. Так, для окрестности 3×3 элементов медианой будет пятое значение по величине, для окрестности 5×5 — тринадцатое значение и так далее. Если несколько элементов в окрестности имеют одинаковые значения, эти значения будут сгруппированы. Например, пусть в окрестности 3×3 элементы имеют следующие значения: (10, 20, 20, 20, 15, 20, 20, 25, 100). После упорядочивания они будут расположены следующим образом: (10, 15, 20, 20, 20, 20, 20, 25, 100), а, следовательно, медианой будет значение 20. Можно сказать, что основная функция медианного фильтра заключается в замене отличающегося от фона значения пикселя на другое, более близкое к его соседям. На самом деле изолированные темные или светлые (по сравнению с окружающим фоном) кластеры, имеющие площадь не более чем $m^2/2$ (половина площади маски фильтра), будут удалены медианным фильтром с маской размерами $m \times m$. В данном случае «удалены» означает, что значения пикселей в соответствующих точках будут заменены на значения медиан по окрестностям. Кластеры больших размеров искажаются значительно меньше.

Хотя медианный фильтр значительно более распространен в обработке изображений, чем остальные виды фильтров, основанные на порядковых статистиках, тем не менее он не является единственным. Медиана представляет собой

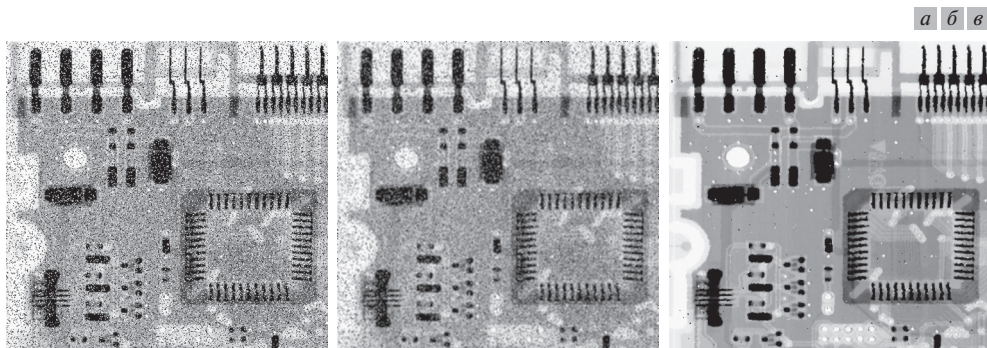


Рис. 3.35. (а) Рентгеновский снимок монтажной платы, искаженный импульсным шумом. (б) Подавление шума усредняющим фильтром по окрестности 3×3 . (в) Подавление шума медианным фильтром по окрестности 3×3 . (Исходное изображение предоставил Джозеф Пасенте, компания Lixi, Inc.)

квантиль уровня 0,5 упорядоченного набора чисел, но, как следует из основ статистики, упорядочивание предоставляет много других возможностей. Например, использование квантиля уровня 1 приводит к так называемому *фильтру максимума*, который полезен при поиске на изображении наиболее ярких точек по отношению к окружающему фону. Отклик фильтра максимума по окрестности 3×3 задается выражением $R = \max\{z_k | k = 1, 2, \dots, 9\}$. Квантиль уровня 0 является *фильтром минимума*, используемым для поиска противоположных значений. Медианный фильтр, фильтры максимума и минимума, а также некоторые другие фильтры будут более детально рассмотрены в главе 5.

Касательно квантилей см. раздел 10.3.5.

Пример 3.14. Использование медианной фильтрации для подавления шума.

■ На рис. 3.35(а) представлен рентгеновский снимок монтажной платы, сильно искаженный импульсным шумом. Чтобы проиллюстрировать преимущество медианной фильтрации по сравнению с усредняющим фильтром, на рис. 3.35(б) показан результат обработки зашумленного изображения усредняющим фильтром по окрестности 3×3 , а на рис. 3.35(в) — результат медианной фильтрации по окрестности 3×3 . Усредняющий фильтр делает изображение нерезким, и его способность снижения шума невысока. При этом совершенно очевидно преимущество медианного фильтра во всех отношениях. Вообще медианная фильтрация намного больше подходит для удаления импульсного шума, чем усредняющая фильтрация. ■

3.6. Пространственные фильтры повышения резкости

Главная цель повышения резкости заключается в том, чтобы подчеркнуть яркостные переходы. Повышение резкости изображений используется достаточно ши-



роко — от цифровой печати и медицинской интроскопии до технического контроля в промышленности и систем автоматического наведения в военной сфере. В предыдущем разделе мы видели, что расфокусировка изображения может быть достигнута пространственной операцией усреднения значений точек по окрестности. Поскольку усреднение аналогично интегрированию, то логично прийти к выводу, что повышение резкости, будучи явлением, обратным по отношению к расфокусировке, может быть достигнуто пространственным дифференцированием. Это действительно так, и в настоящем разделе будут обсуждаться различные способы задания и использования операторов повышения резкости путем численного дифференцирования. С принципиальной точки зрения величина отклика оператора производной в точке изображения пропорциональна степени разрывности изображения в данной точке. Таким образом, дифференцирование изображения позволяет усилить перепады и другие разрывы (например шумы) и не подчеркивать области с медленными изменениями уровней яркостей.

3.6.1. Основы

В последующих двух разделах будут рассмотрены фильтры повышения резкости, основанные на первой и второй производных. Однако прежде чем перейти к непосредственному обсуждению, необходимо остановиться на некоторых фундаментальных свойствах этих производных в контексте цифровых методов. Для простоты изложения первоначально остановимся на одномерных производных. В частности, представляет интерес поведение этих производных на областях постоянной яркости, в начале и в конце разрывов (разрывы в виде ступенек и участков изменения яркости — склонов), а также на протяжении самих склонов. Как будет показано в главе 10, эти типы разрывов могут использоваться для описания шумовых всплесков, линий и контуров на изображении. Также важным является поведение производной на протяжении перехода от начала до окончания указанных особенностей.

Производные дискретной функции определяются в терминах разностей. Эти разности можно задать различными способами, однако мы будем руководствоваться следующим. *Первая производная* должна быть: (1) равной нулю на областях с постоянным уровнем яркости; (2) ненулевой в начале и в конце ступеньки или склона яркости; (3) ненулевой на склонах яркости. Аналогично *вторая производная* должна быть: (1) равной нулю на плоских участках; (2) ненулевой в начале и в конце ступеньки или склона яркости; (3) равной нулю на склонах постоянной крутизны. Так как мы оперируем ограниченными численными значениями, максимальное значение изменения яркости также конечно, а кратчайшее расстояние, на котором это изменение может происходить, есть расстояние между соседними пикселями.

Первая производная одномерной функции $f(x)$ определяется как разность значений соседних элементов:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x). \quad (3.6-1)$$

Мы вернемся к уравнению (3.6-1) в разделе 10.2.1 и покажем, как оно следует из разложения в ряд Тейлора. На данный момент мы принимаем его как определение.

Здесь использована запись в виде частной производной для того, чтобы сохранить те же обозначения в случае двух переменных $f(x, y)$, где придется иметь дело с частными производными по двум пространственным осям. Использование частной производной не меняет существа рассмотрения. Ясно, что $\frac{df}{dx} = \frac{df}{dx}$, когда у функции имеется только одна переменная; то же верно в случае второй производной.

Вторая производная определяется как разность соседних значений первой производной:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x). \quad (3.6-2)$$

Легко проверить, что оба данных определения удовлетворяют сформулированным ранее условиям касательно производных первого и второго порядков. Чтобы проиллюстрировать это, а также рассмотреть основные сходства и различия между производными первого и второго порядков в контексте обработки изображений, рассмотрим пример на рис. 3.36.

На рис. 3.36(б) (в центре рисунка) показан участок строки изображения (профиль яркости). Числа в квадратах суть значения яркостей элементов строки изображения, которые отмечены черными точками на рис. 3.36(а). Для визуаль-

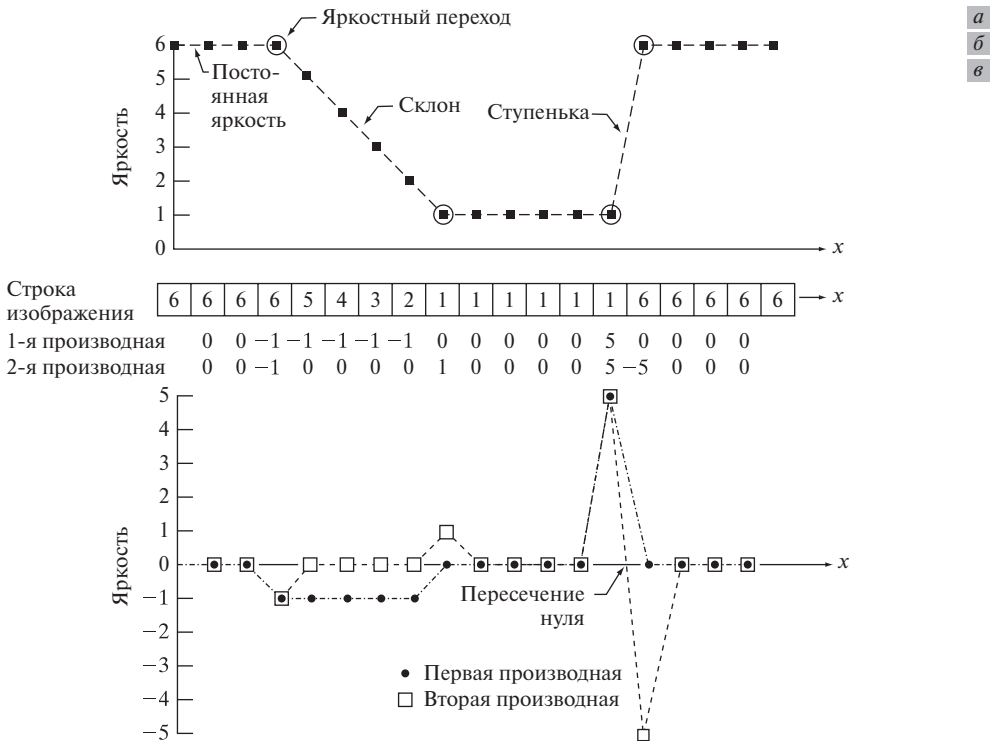


Рис. 3.36. Иллюстрация первой и второй производных одномерной дискретной функции, представляющей собой участок горизонтального профиля яркости некоторого изображения. На (а) и (в) точки отсчетов соединены пунктирными линиями для визуального удобства



ного удобства добавлена пунктирная линия, соединяющая эти точки. Как видно, на участке строки содержатся три участка постоянной яркости, участок склона и яркостная ступенька. Кружки показывают начало или конец яркостных переходов. Первая и вторая производные, вычисленные на основе данных выше двух определений, приведены ниже строки изображения на рис. 3.36(б), а также в виде графиков на рис. 3.36(в). При вычислении первой производной в точке x значение в данной точке вычиталось из значения в следующей точке, так что это операция с «предварительным просмотром». Аналогично для вычисления второй производной в точке x использовались значения в предыдущей и следующей точках. Чтобы отвлечься от вопроса выхода предыдущей или следующей точки за границы участка строки, значения производных на рис. 3.36 приведены лишь от второго до предпоследнего элемента.

Рассмотрим поведение первой и второй производных при движении вдоль профиля слева направо. В начале находится область с постоянной яркостью, на которой, как видно на рис. 3.36(б) и (в), обе производные равны нулю, значит, для обеих выполняется условие (1). Затем идет склон яркости, за которым следуют площадка и ступенька. Отметим, что первая производная отлична от нуля на склоне и на ступеньке; аналогично вторая производная отлична от нуля в начале и в конце как склона, так и ступеньки; таким образом, свойство (2) удовлетворяется для обеих производных. Наконец, свойство (3) также выполняется для обеих производных, поскольку на склоне первая производная не равна нулю, а вторая равна нулю. Заметим, что знак второй производной меняется в начале и в конце ступеньки или склона. Действительно, на рис. 3.36(в) видно, что линия, соединяющая два значения производных на участке ступенчатого перехода, в середине между экстремумами пересекает горизонтальную ось. Это свойство пересечения нулевого уровня весьма полезно для локализации контуров, как будет показано в главе 10.

На цифровых изображениях контурные переходы зачастую выглядят как склоны яркостей, на которых первая производная будет диагностировать в результате широкие контуры, поскольку она отлична от нуля на всем протяжении склона. Напротив, вторая производная даст двойной контур толщиной в одну линию и разделенный нулями. Отсюда можно сделать вывод, что вторая производная усиливает мелкие детали значительно лучше, чем первая, — свойство, идеально подходящее для повышения резкости изображений. Также, как будет показано далее в настоящем разделе, вторую производную значительно легче реализовать, чем первую; так что вначале наше внимание будет сконцентрировано на вторых производных.

3.6.2. Повышение резкости изображений с использованием вторых производных: лапласиан

В данном разделе мы рассмотрим применение двумерной второй производной в задачах повышения резкости изображений. Мы вернемся к рассмотрению второй производной в главе 10, где она будет использоваться для сегментации изображений. Подход сводится к выбору дискретной формулировки второй производной и к последующему построению маски фильтра, основанной на данной формулировке. Рассматриваться будут *изотропные* фильтры, отклик которых

не зависит от направления неоднородностей на обрабатываемом изображении. Другими словами, изотропные фильтры являются *инвариантными к повороту* в том смысле, что поворот изображения и последующее применение фильтра дает тот же результат, что и первоначальное применение фильтра с последующим поворотом результата¹¹.

Можно показать ([Rosenfeld и Kak, 1982]), что простейшим изотропным оператором, основанным на производных, является *лапласиан* (оператор Лапласа), который в случае функции двух переменных $f(x, y)$ определяется как

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}. \quad (3.6-3)$$

Поскольку производные любого порядка являются линейными операторами, то, значит, и лапласиан является линейным оператором. Чтобы выразить это уравнение в дискретной форме, используем определение, выраженное уравнением (3.6-2), принимая во внимание, что теперь имеются две переменные. Для частной второй производной по x будет использоваться следующая формула:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y), \quad (3.6-4)$$

и аналогично для производной по y :

$$\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y). \quad (3.6-5)$$

Таким образом, из предыдущих трех уравнений следует, что дискретная формулировка двумерного лапласиана двух переменных будет

$$\nabla^2 f(x, y) = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y). \quad (3.6-6)$$

Это уравнение может быть реализовано с помощью маски фильтра, представленной на рис. 3.37(а), которая дает изотропный результат для поворотов на углы, кратные 90° . Способы реализации такие же, как в разделе (3.5.1) для линейных сглаживающих фильтров. Здесь попросту используются другие значения коэффициентов.

Диагональные направления могут быть включены в формулу дискретного лапласиана (3.6-6) добавлением еще двух членов — по одному для каждого из диагональных направлений. Вид каждого из них такой же, как в уравнении (3.6-4) или (3.6-5), но указываются координаты точек, расположенных по диагоналям. Поскольку каждая диагональная добавка включает член $-2f(x, y)$, то суммарный вычитаемый из суммы член составит $-8f(x, y)$. Маска фильтра, соответствующая такому новому определению, представлена на рис. 3.37(б). Такая маска является изотропной для поворотов на углы, кратные 45° . Две оставшиеся маски, показанные на рис. 3.37(в) и (г), также часто используются на практике. Они получены из определения лапласиана, являющегося «негативным» по от-

¹¹ В применении к дискретному сигналу это утверждение верно лишь настолько, насколько точными в его отношении можно считать операцию поворота и круговую симметрию масок фильтров. — *Прим. перев.*

a	b
c	d

0	1	0	1	1	1
1	-4	1	1	-8	1
0	1	0	1	1	1

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Рис. 3.37. (а) Маска фильтра, используемая для реализации уравнения (3.6-6). (б) Маска, используемая для реализации расширения этого уравнения путем добавления диагональных членов. (в) и (г) Две другие реализации лапласиана, часто встречаемые на практике

ношению к тому, который использовался в уравнениях (3.6-4) и (3.6-5). По существу, они дают идентичный результат, но различие в знаке должно учитываться при комбинации — операцией сложения или вычитания — изображения, отфильтрованного лапласианом, с другим изображением.

Поскольку оператор Лапласа по сути является второй производной, его применение подчеркивает разрывы уровней яркостей на изображении и подавляет области со слабыми изменениями яркостей. Это приводит к получению изображения, содержащего сероватые линии на месте контуров и других разрывов, наложенные на темный фон без особенностей. Но фон можно «восстановить», сохранив при этом эффект повышения резкости, достигаемый лапласианом. Для этого достаточно прибавить изображение-лапласиан к исходному изображению. Как было сказано в предыдущем абзаце, при этом необходимо помнить, какое из определений лапласиана было использовано. Если использовалось определение, использующее отрицательные центральные коэффициенты, тогда для получения эффекта повышения резкости изображение-лапласиан следует *вычитать*, а не прибавлять. Таким образом, обобщенный алгоритм использования лапласиана для повышения резкости изображений сводится к следующему:

$$g(x, y) = f(x, y) + c[\nabla^2 f(x, y)], \quad (3.6-7)$$

где $f(x, y)$ и $g(x, y)$ — исходное изображение и изображение с повышенной резкостью. Константа $c = -1$, если использована маска фильтра лапласиана на рис. 3.37(а) или (б), и $c = 1$, если использован один из оставшихся двух фильтров.

Пример 3.15. Повышение резкости изображения с помощью лапласиана.

■ На рис. 3.38(а) представлено слегка нерезкое изображение Северного полюса Луны. На рис. 3.38(б) показан результат фильтрации данного изображения лапласианом с маской на рис. 3.37(б). Большие области этого изображения являются черными, поскольку лапласиан имеет как положительные, так и отрицательные значения, а все отрицательные обрезаются до 0 при воспроизведении.

Обычный способ преобразовать изображение-лапласиан — подвергнуть его градационной коррекции: прибавить значение его минимума ко всем значениям пикселей, а затем растянуть полученный результат на весь диапазон яркостей $[0, L - 1]$ согласно уравнениям (2.6-10) и (2.6-11). Именно таким образом обработано изображение на рис. 3.38(в). Видно, что основными деталями данного изображения являются контуры и резкие перепады яркости различного уровня. Фон, ранее черный, теперь, вследствие градационной коррекции, стал серым. Такое сероватое проявление фона является типичным для правильно откорректированных изображений-лапласианов. На рис. 3.38(г) показан результат, полученный с использованием уравнения (3.6-7) с $c = -1$. Детали на этом изображении видны значительно более чистыми и резкими, чем на исходном изображении. Добавление исходного изображения к лапласиану восстанови-

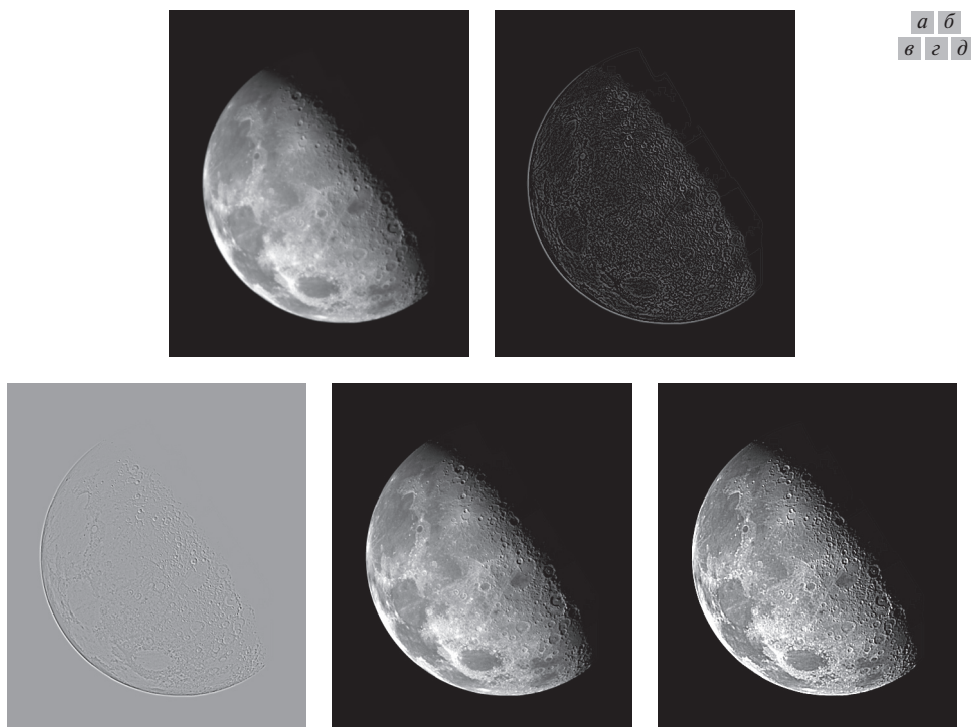


Рис. 3.38. (а) Нерезкое изображение Северного полюса Луны. (б) Изображение-лапласиан без масштабирования. (в) Изображение-лапласиан, подвергнутое градационной коррекции. (г) Изображение с повышенной резкостью при использовании маски на рис. 3.37(а). (д) Изображение с повышенной резкостью при использовании маски на рис. 3.37(б). (Исходное изображение предоставлено агентством NASA)



ло общий диапазон изменения яркостей на изображении, а лапласиан усилил контрасты в местах яркостных разрывов. Конечным результатом стало изображение, на котором мелкие детали улучшены, а фоновые полутона достаточно хорошо сохранены. Наконец, на рис. 3.38(д) показан результат, почти повторяющий предыдущую процедуру с использованием фильтра на рис. 3.37(б). Здесь заметно значительное повышение резкости по сравнению с 3.38(г). Это не является неожиданным, поскольку использование фильтра на рис. 3.37(б) добавляет дополнительное дифференцирование (повышение резкости) по диагональным направлениям. Результаты, подобные представленным на рис. 3.38(г) и (д), сделали лапласиан часто используемым инструментом для повышения резкости цифровых изображений. ■

3.6.3. Нерезкое маскирование и фильтрация с подъемом высоких частот

Процедура, много лет использующаяся в полиграфии и в издательском деле для повышения резкости изображений, заключается в вычитании из изображения его нерезкой (сглаженной) копии. Эта процедура, называемая *нерезким маскированием*, состоит из следующих шагов.

1. Создание нерезкого изображения.
2. Вычитание нерезкого изображения из исходного изображения (результатирующая разность называется *маска*).
3. Прибавление маски к исходному изображению.

Обозначая через $\bar{f}(x, y)$ сглаженное изображение, нерезкое маскирование выражается в формульном виде следующим образом. Первоначально формируется маска

$$g_{\text{mask}}(x, y) = f(x, y) - \bar{f}(x, y). \quad (3.6-8)$$

Затем маска с заданным весом прибавляется к исходному изображению:

$$g(x, y) = f(x, y) + k g_{\text{mask}}(x, y), \quad (3.6-9)$$

где для общности введен весовой коэффициент k ($k \geq 0$). При $k = 1$ мы получаем нерезкое маскирование, как определено выше. При $k > 1$ процесс называется *фильтрацией с подъемом высоких частот*. Выбор $k < 1$ снижает вклад нерезкой маски.

На рис. 3.39 объяснено, как действует нерезкое маскирование. График на рис. 3.39(а) можно интерпретировать как профиль яркости строки, пересекающей вертикальный контур на переходе от темной к светлой области на изображении. На рис. 3.39(б) показан результат сглаживания, наложенный на график исходного сигнала (показанный пунктиром) для ориентира. График на рис. 3.39(в) — нерезкая маска, полученная вычитанием сглаженного сигнала из исходного¹². Сравнивая этот результат с участком на рис. 3.36(в), соответствующим склону на рис. 3.36(а), можно отметить, что нерезкая маска

¹² На рис. 3.39(в) и (г) представлены не вполне корректные рисунки. Вершинки пиков должны быть острыми. — *Прим. перев.*

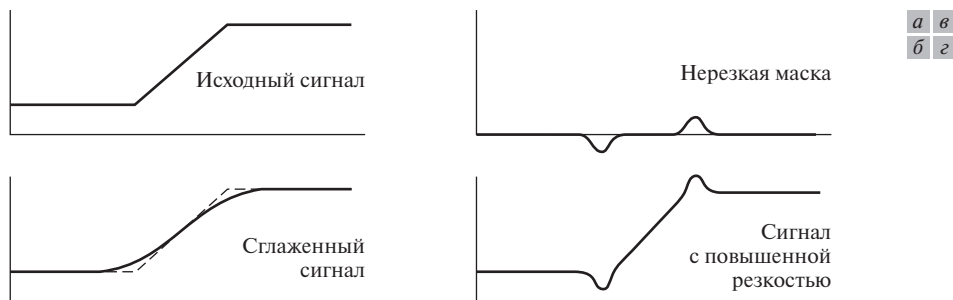


Рис. 3.39. Одномерная иллюстрация механизма нерезкого маскирования. (а) Исходный сигнал. (б) Сглаженный сигнал; исходный для сравнения показан пунктиром. (в) Нерезкая маска. (г) Сигнал с повышенной резкостью, полученный прибавлением (в) к (а)

на рис. 3.39(г) очень похожа на то, что можно было бы получить при помощи второй производной. Конечный результат с повышенной резкостью, полученный прибавлением маски к исходному сигналу, показан на рис. 3.39(г). Точки, в которых были изменения наклона графика яркости, теперь оказались усиленными (повышена резкость). Заметим, что отрицательные значения также прибавляются к исходному сигналу. Тем самым в результате суммирования могут появиться отрицательные значения, если на исходном изображении были близкие к нулю значения или если значение коэффициента k выбрано излишне большим, в результате чего пики маски могут выйти за допустимый диапазон яркостей. Отрицательные значения могут вызвать появление темного ореола вокруг контуров, что при достаточно большом k может привести к нежелательному результату.

Пример 3.16. Повышение резкости изображения с помощью нерезкого маскирования.

- На рис. 3.40 (а) представлено слегка сглаженное изображение белого текста на темно-сером фоне. Изображение на рис. 3.40(б) получено с помощью гауссова сглаживающего фильтра (см. раздел 3.4.4) размерами 5×5 с $\sigma = 3$. Рис. 3.40(в) — нерезкая маска, полученная преобразованием согласно уравнению (3.6-8). Изображение на рис. 3.40(г) получено использованием нерезкого маскирования (уравнение (3.6-9) с $k = 1$). Это изображение несколько лучше оригинала, но можно достичь большего. На рис. 3.40(д) показано применение уравнения (3.6-9) с $k = 4,5$, наибольшим значением, которое можно использовать, чтобы



Рис. 3.40. (а) Исходное изображение. (б) Результат сглаживания гауссовым фильтром. (в) Нерезкая маска. (г) Результат нерезкого маскирования. (д) Результат фильтрации с подъемом высоких частот

значения всех пикселей конечного результата оставались положительными. Улучшение на этом изображении по сравнению с исходным является значительным. ■

3.6.4. Использование производных первого порядка для (нелинейного) повышения резкости изображений: градиент

В обработке изображений первые производные реализуются через модуль градиента. Для функции $f(x, y)$ градиент f в точке (x, y) определяется как двумерный вектор-столбец:

$$\nabla f \equiv \text{grad}(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}. \quad (3.6-10)$$

Этот вектор имеет важное геометрическое свойство — он указывает на направление наибольшего возрастания f в точке (x, y) .

Детальное обсуждение градиента дано в разделе 10.2.5. Здесь же нас интересует лишь использование модуля градиента для повышения резкости изображений.

Модуль (длина) вектора ∇f обозначается как $M(x, y)$, где

$$M(x, y) = \text{mag}(\nabla f) = \sqrt{g_x^2 + g_y^2} \quad (3.6-11)$$

есть величина скорости изменения f в направлении вектора градиента. Заметим, что сформированное изображение $M(x, y)$ имеет тот же размер, что и исходное изображение f , поскольку x и y имеют тот же диапазон изменения. Обычной практикой является называть $M(x, y)$ *градиентным изображением* или просто *градиентом*, если смысл ясен.

Поскольку компоненты вектора градиента являются производными, они представляют собой линейные операторы. Однако модуль вектора таковым не является, поскольку он выражается через операции возведения в квадрат и извлечения квадратного корня. С другой стороны, частные производные в уравнении (3.6-10) не являются инвариантными к повороту (изотропными), но модуль вектора градиента таковым является. В некоторых приложениях бывает более выгодно с позиций вычислений вместо квадратов и квадратных корней использовать приближение значения модуля градиента суммой модулей отдельных компонентов:

$$M(x, y) \approx |g_x| + |g_y|. \quad (3.6-12)$$

Это выражение все еще сохраняет относительные изменения в уровнях яркостей, но свойство изотропности, вообще говоря, пропадает. Однако, как и в случае лапласиана, для определяемого в следующем абзаце дискретного градиента свойства изотропности сохраняются только для ограниченного набора угловых приращений, которые зависят от масок фильтров, используемых для прибли-

жения производных. Как оказывается, маски, наиболее часто используемые для приближения градиента, являются изотропными для углов, кратных 90° . Эти результаты не зависят от того, какое уравнение используется — (3.6-11) или (3.6-12), так что при использовании последнего уравнения ничего существенного не теряется.

Аналогично лапласиану первоначально будут определены дискретные приближения приведенных выше уравнений, а затем уже по ним будут сформированы соответствующие маски фильтров. Чтобы упростить последующие рассуждения, для указания точек в окрестности 3×3 будут использоваться обозначения, показанные на рис. 3.41(а). Так, например, центральная точка z_5 обозначает $f(x, y)$ в произвольной точке (x, y) , z_1 обозначает $f(x - 1, y - 1)$ и так далее, используя нотацию, указанную на рис. 3.28. Как отмечено в разделе 3.6.1, простейшими приближениями первой производной, которые удовлетворяют сформулированным в том же разделе условиям, являются следующие: $g_x = (z_8 - z_5)$ и $g_y = (z_6 - z_5)$. Два других определения, предложенные Робертсом [Roberts, 1965] в ранних исследованиях по цифровой обработке изображений, используют перекрестные направления

$$g_x = (z_9 - z_5) \quad \text{и} \quad g_y = (z_8 - z_6). \quad (3.6-13)$$

Если выбрать уравнения (3.6-11) и (3.6-13), то градиент будет вычисляться по следующей формуле:

$$M(x, y) = [(z_9 - z_5)^2 + (z_8 - z_6)^2]^{1/2}. \quad (3.6-14)$$

Если используются уравнения (3.6-12) и (3.6-13), то

$$M(x, y) \approx |z_9 - z_5| + |z_8 - z_6|, \quad (3.6-15)$$

где x и y изменяются в пределах всего изображения так же, как и ранее. Частные производные, использующиеся в (3.6-13), могут быть реализованы при помощи двух масок линейных фильтров, показанных на рис. 3.41(б) и (в). Эти маски называют *перекрестным градиентным оператором Робертса*.

Однако маски четного размера реализовывать неудобно, поскольку они не имеют элемента, являющегося центром симметрии. Наименьшая маска фильтра, которая нас интересует, имеет размеры 3×3 . Приближениями к g_x и g_y , использующими окрестность 3×3 , центрированную относительно z_5 , будут

$$g_x = \frac{\partial f}{\partial x} = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3) \quad (3.6-16)$$

и

$$g_y = \frac{\partial f}{\partial y} = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7). \quad (3.6-17)$$

Эти уравнения могут быть реализованы при помощи масок на рис. 3.41(г) и (д). Разность между значениями пикселей в третьей и первой строках окрестности 3×3 аппроксимирует значение производной по направлению x , а разность между значениями пикселей в третьем и первом столбцах — значение производной по направлению y . После вычисления производных при помощи этих масок амплитуда градиента вычисляется так же, как и ранее. Так, например, подставляя g_x и g_y в уравнение (3.6-12), получим



z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

-1	0	0	-1
0	1	1	0

-1	-2	-1	-1	0	1
0	0	0	-2	0	2
1	2	1	-1	0	1

Рис. 3.41. (а) Область изображения размерами 3×3 элемента (z_1, \dots, z_9 — значения яркости соответствующих пикселей под маской). (б)—(в) Перекрестный градиентный оператор Робертса. (г)—(д) Оператор Собела. Суммы коэффициентов по каждой из масок равны нулю, как и должно быть у градиентных операторов

$$M(x, y) \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)|. \quad (3.6-18)$$

Маски, представленные на рис. 3.41(г) и (д), называют *оператором Собела*. Использование значения 2 у центральных весовых коэффициентов (в строках и столбцах) основано на стремлении достичь большей гладкости за счет присвоения центральным точкам большей значимости (этот вопрос детально будет рассмотрен в главе 10). Заметим, что суммы коэффициентов каждой из масок, показанных на рис. 3.41, равны 0, и это означает, что на участках изображения с постоянным уровнем яркости отклик любого из приведенных операторов будет равен 0, как и должно быть у оператора, соответствующего первой производной.

Как отмечалось ранее, вычисления g_x и g_y являются линейными операциями, поскольку основаны на производных, и, таким образом, могут реализовываться как суммы произведений, в которых используются коэффициенты пространственных масок на рис. 3.41. Нелинейный аспект повышения резкости при помощи градиента состоит в использовании при вычислении $M(x, y)$ операций возведения в квадрат и извлечения квадратного корня либо абсолютных значений; каждая из этих операций является нелинейной. Эти операции выполняются *после* линейного процесса, формирующего g_x и g_y .

Пример 3.17. Применение градиента для улучшения контуров.

■ Градиент часто используется в техническом контроле для того, чтобы помочь человеку обнаружить дефекты, или, что является более общим, как предвари-

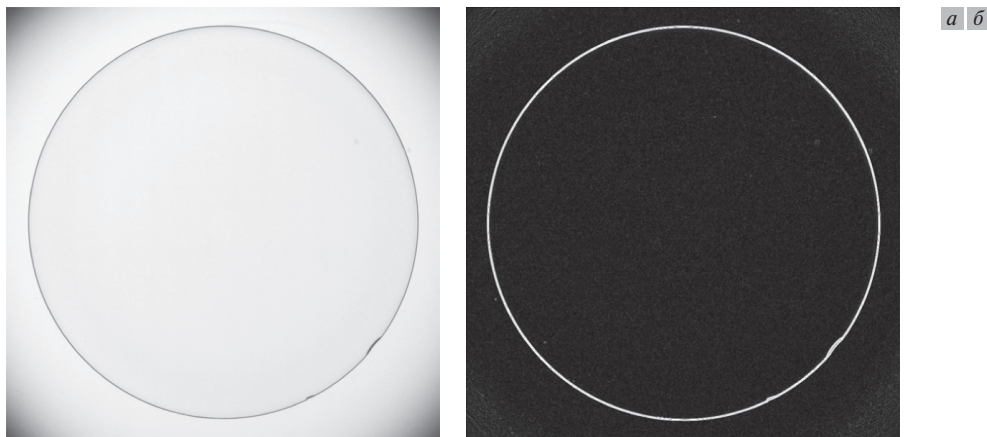


Рис. 3.42. (а) Оптическое изображение контактной линзы (дефекты видны на краях окружности в позициях «4 и 5 часов»). (б) Градиент, полученный оператором Собела. (Исходное изображение предоставил Пит Сайте, Perceptics Corporation)

тельная обработка в автоматизированном контроле. Об этом будет говориться в главах 10 и 11. Однако представляется полезным рассмотреть некоторый простой пример прямо сейчас, чтобы показать, как может быть использован градиент для улучшения видимости дефектов и для удаления слабо меняющихся характеристик фона. В данном примере улучшение используется в качестве шага предварительной обработки для последующего автоматического контроля, а не для визуального анализа.

На рис. 3.42(а) показано оптическое изображение контактной линзы, освещенной специальной световой установкой, предназначенной для подчеркивания неточностей. В данном случае можно заметить два дефекта на краях (они видны на окружности в позициях «4 и 5 часов»). На рис. 3.42(б) показан градиент, подсчитанный по формуле (3.6-12) с масками оператора Собела, показанными на рис. 3.41(е) и (д). Дефекты краев видны как на изображении 3.42(а), так и на 3.42(б), но преимуществом второго изображения является то, что устранены медленные изменения яркостей, тем самым упрощена задача автоматического контроля. Градиентная обработка может быть также использована для подчеркивания мелких дефектов, которые нелегко заметить на полутоновом изображении (такими дефектами могут быть чужеродные включения, воздушные карманы в держателе, а также неточности мениска линзы). Способность подчеркнуть мелкие неоднородности на ровном сером поле есть еще одна важная характеристика градиента. ■

3.7. Комбинирование методов пространственного улучшения

За некоторыми исключениями, как, например, в случае комбинации сглаживания с пороговым преобразованием (рис. 3.34), до настоящего момента внимание