Chapter 3 Intensity Transformations and Spatial Filtering

Image Enhancement in the Spatial Domain

- *No general theory* of image enhancement
- A certain amount of *trial and error* is usually required before a particular image enhancement approach is selected.

Outlines

3.1 Background

3.2 Some Basic Intensity Transformation Functions

3.3 Histogram Processing

3.4 Fundamentals of Spatial Filtering

3.5 Smoothing Spatial Filters

3.6 Sharpening Spatial Filters

3.7 Combining Spatial Enhancement Methods

3.4 Fundamentals of Spatial Filtering

- Filter, Mask, Kernel,
  Template, Window
- Coefficients
- Linear Filtering *vs*
  Nonlinear Filtering
  (e.g., median filtering)
- Linear Operator (Sec 2.6)
  - Additivity
  - Homogeneity

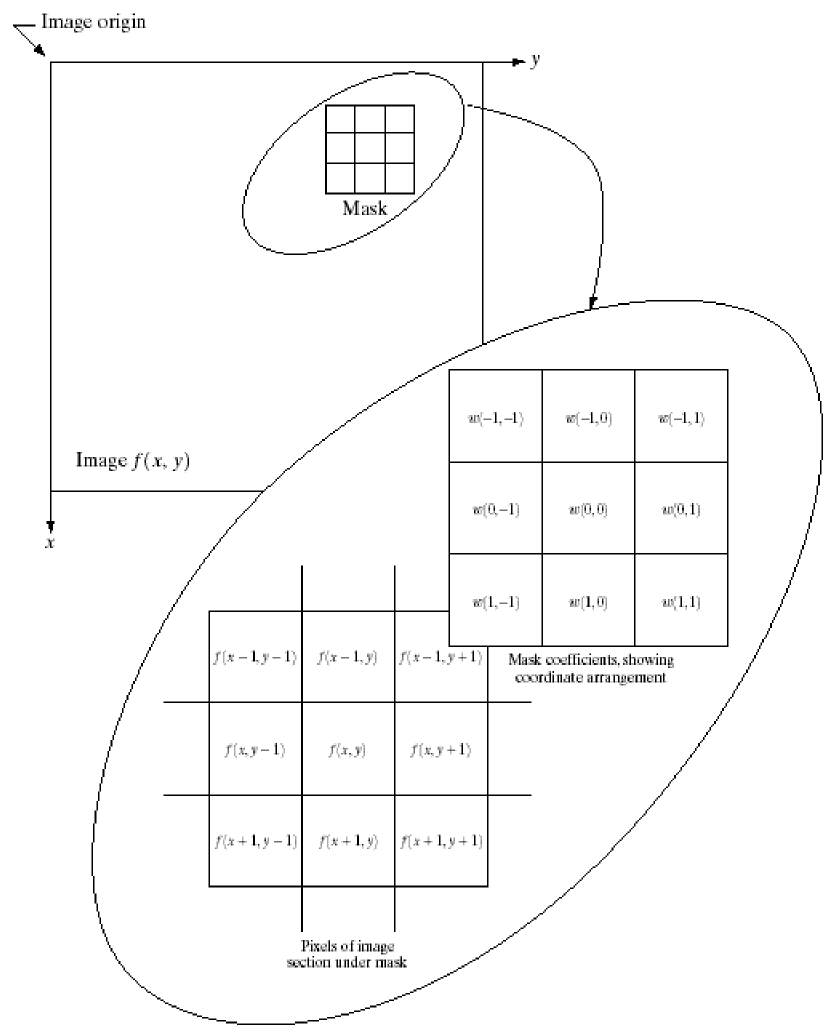


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

3 x 3 Mask, Sum of Products

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$$R = w_1 z_1 + w_2 z_2 + \ldots w_9 z_9$$
$$= \sum_{i=1}^{9} w_i z_i . = \underline{W}^T \cdot \underline{Z}$$

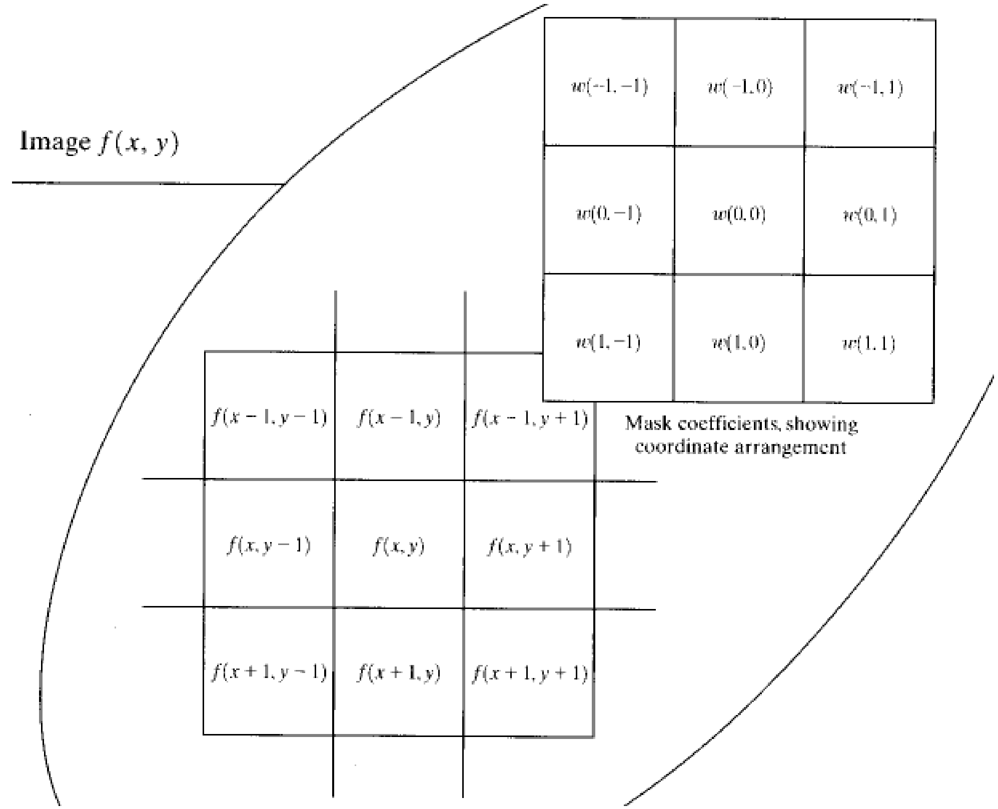


Fig 3.28

$$R = w(-1,-1)f(x-1, y-1) + w(-1,0)f(x-1, y) + \cdots$$
$$+ \underline{w(0,0)}\underline{f(x,y)} + \cdots + w(1,0)f(x+1, y) + w(1,1)f(x+1, y+1)$$

M x N Mask

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

for all pixels (*x*, *y*)

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_{mn} z_{mn} = \sum_{i=1}^{mn} w_i z_i$$

Spatial Correlation and Convolution

- Correlation

$$g(x) = f(x) \star w(x) = \sum_{s=-a}^{a} w(s) f(x+s)$$

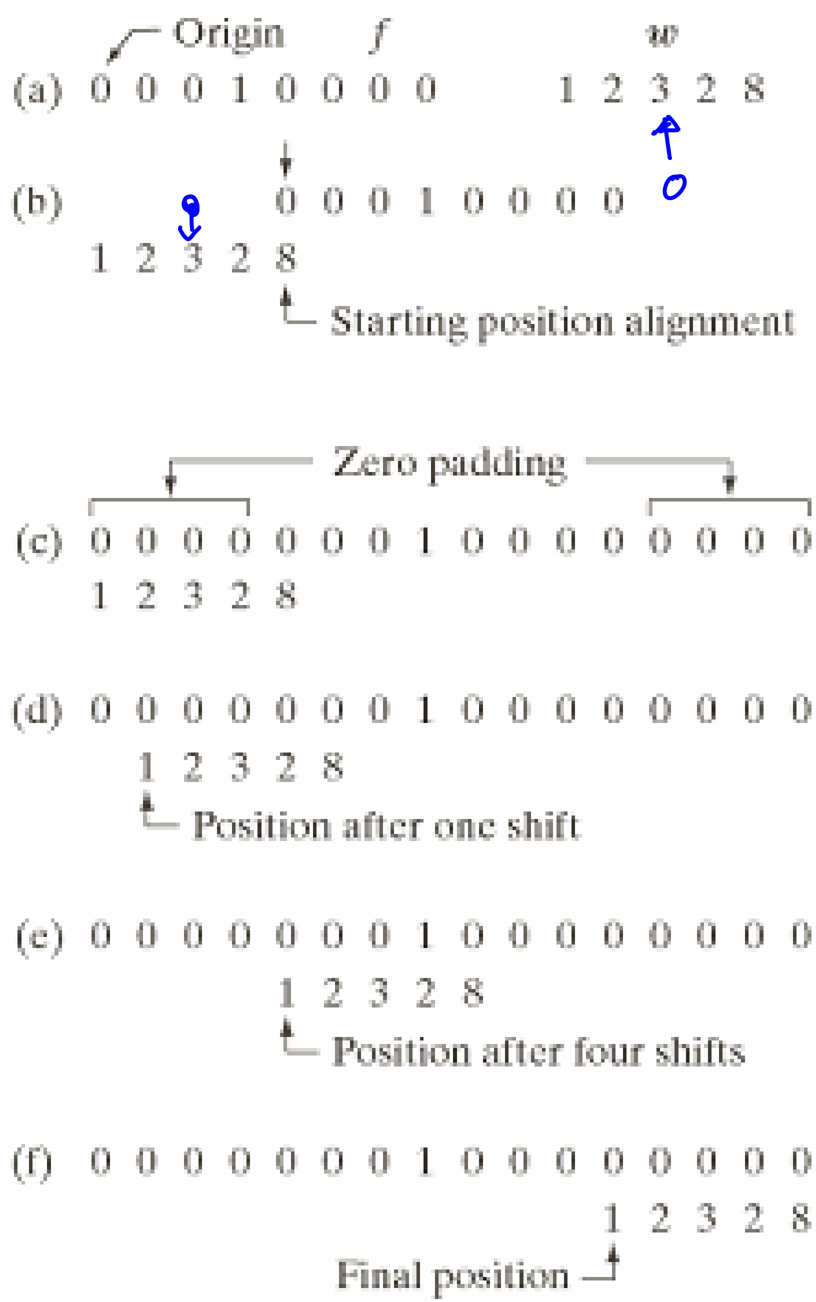
$$= w(-a) f(x-a) + \cdots + w(0) f(x) + \cdots + w(a) f(x+a)$$

- Convolution

$$g(x) = f(x) * w(x) = \sum_{s=-a}^{a} w(s) f(x-s)$$

$$= w(-a) f(x+a) + \cdots + w(0) f(x) + \cdots + w(a) f(x-a)$$

1D Correlation



$$g(x) = f(x) \star w(x)$$

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

$$\boxed{m=5 \Rightarrow a=2}$$

$$g(x) = \sum_{s=-2}^{2} w(s)\, f(x+s)$$

$$= w(-2) f(x-2) + w(-1) f(x-1)$$

$$+ w(0) f(x) + w(1) f(x+1)$$

$$+ w(2) f(x+2)$$

1D Correlation

Origin $\quad f \quad\quad w$

(a) 0 0 0 1 0 0 0 0 $\quad$ 1 2 3 2 8

(b) 0 0 0 1 0 0 0 0
1 2 3 2 8
Starting position alignment

Zero padding

(c) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 2 3 2 8

(f) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 2 3 2 8
Final position

Full correlation result

(g) 0 0 0 8 2 3 2 1 0 0 0 0

Cropped correlation result

(h) 0 8 2 3 2 1 0 0

$$g(x) = f(x) \star w(x)$$

1D Correlation

	Origin	f	w
(a)		0 0 0 1 0 0 0 0	1 2 3 2 8
(b)		0 0 0 1 0 0 0 0	
	1 2 3 2 8		

Starting position alignment

$$g(x) = f(x) \star w(x)$$

$$= \sum_{s=-a}^{a} w(s) f(x+s)$$

1D Convolution

Origin	f	w rotated 180°	
	0 0 0 1 0 0 0 0	8 2 3 2 1	(i)
	0 0 0 1 0 0 0 0		(j)
8 2 3 2 1			

$$g(x) = f(x) \bigstar w(x)$$

$$= \sum_{s=-a}^{a} w(s) f(x-s)$$

1D Convolution

$$g(x) = f(x) \star w(x)$$

$$= \sum_{s=-a}^{a} w(s) f(x-s)$$

$$\boxed{m = 5 \Rightarrow a = 2}$$

$$g(x) = w(-2) f(x+2) + w(-1) f(x+1) + w(0) f(x) + w(1) f(x-1) + w(2) f(x-2)$$

Origin f w rotated 180°

0 0 0 1 0 0 0 0 8 2 3 2 1 (i)

0 0 0 1 0 0 0 0 0 (j)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (k)
8 2 3 2 1

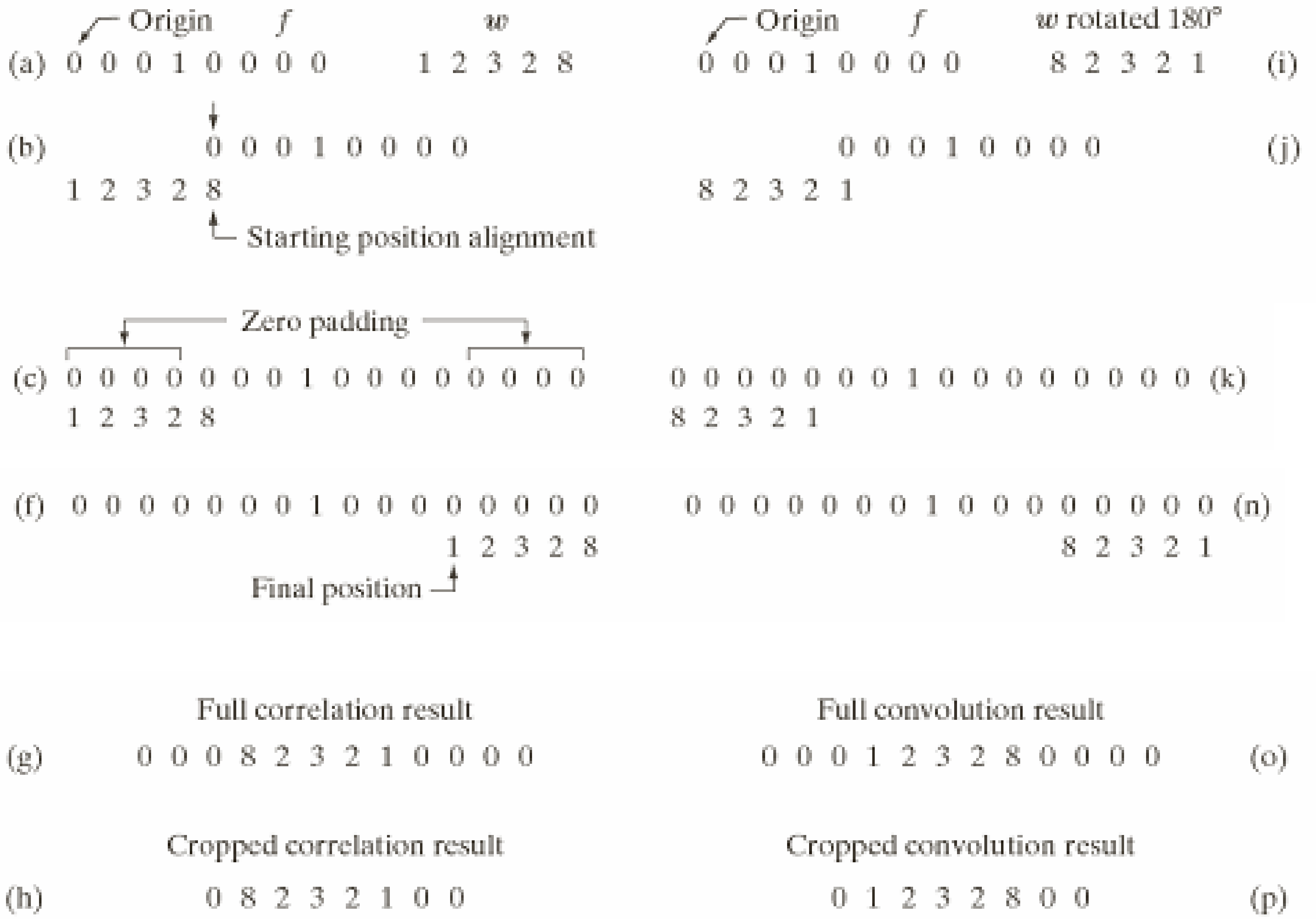
0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (l)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (m)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (n)
8 2 3 2 1

1D Correlation

1D Convolution



2D Correlation

Origin $f(x, y)$

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

$w(x, y)$

1	2	3
4	5	6
7	8	9

(a)

Padded f

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	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

(b)

Initial position for w

1	2	3	0	0	0	0	0	0
4	5	6	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	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

(c)

Full correlation result

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	9	8	7	0	0	0
0	0	0	6	5	4	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

(d)

Cropped correlation result

0	0	0	0	0
0	9	8	7	0
0	6	5	4	0
0	3	2	1	0
0	0	0	0	0

(e)

2D Convolution

Origin $f(x, y)$

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

$w(x, y)$

1	2	3
4	5	6
7	8	9

(a)

Padded f

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	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

(b)

Rotated w

9	8	7	0	0	0	0	0	0
6	5	4	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	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

(f)

Full convolution result

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	2	3	0	0	0
0	0	0	4	5	6	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

(g)

Cropped convolution result

0	0	0	0	0
0	1	2	3	0
0	4	5	6	0
0	7	8	9	0
0	0	0	0	0

(h)

Correlation vs Convolution

- To perform convolution, all we do is rotate one function by 180^o and perform correlation.
  - Notice that it makes no difference which of the two function we rotate
- Using correlation or convolution to perform spatial filtering is *a matter of preference*
  - as long as the filter mask is specified correctly
- In the IP literature, “convolving a mask with an image” may refer to “correlation” operation.
- The concept of convolution is important for Chapter 4.

Operations concerning the border of the image

1. Limit the excursions of the mask
2. Padding zero
3. Padding by replicating rows or columns

Outlines

3.1 Background

3.2 Some Basic Intensity Transformation Functions

3.3 Histogram Processing

3.4 Fundamentals of Spatial Filtering

3.5 Smoothing Spatial Filters

3.6 Sharpening Spatial Filters

3.7 Combining Spatial Enhancement Methods

3.5 Smoothing Spatial Filters

-- for blurring and for noise reduction

- Linear Smoothing Filters – averaging filters
- Nonlinear Smoothing Filters – median filters

3.5.1 Averaging Filters: *linear*

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Box Filter

$$\frac{1}{16} \times \begin{array}{|c|c|c|} \hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Weighted Average

** Hard to see the difference due to small window*

Why 16?

When to divide?

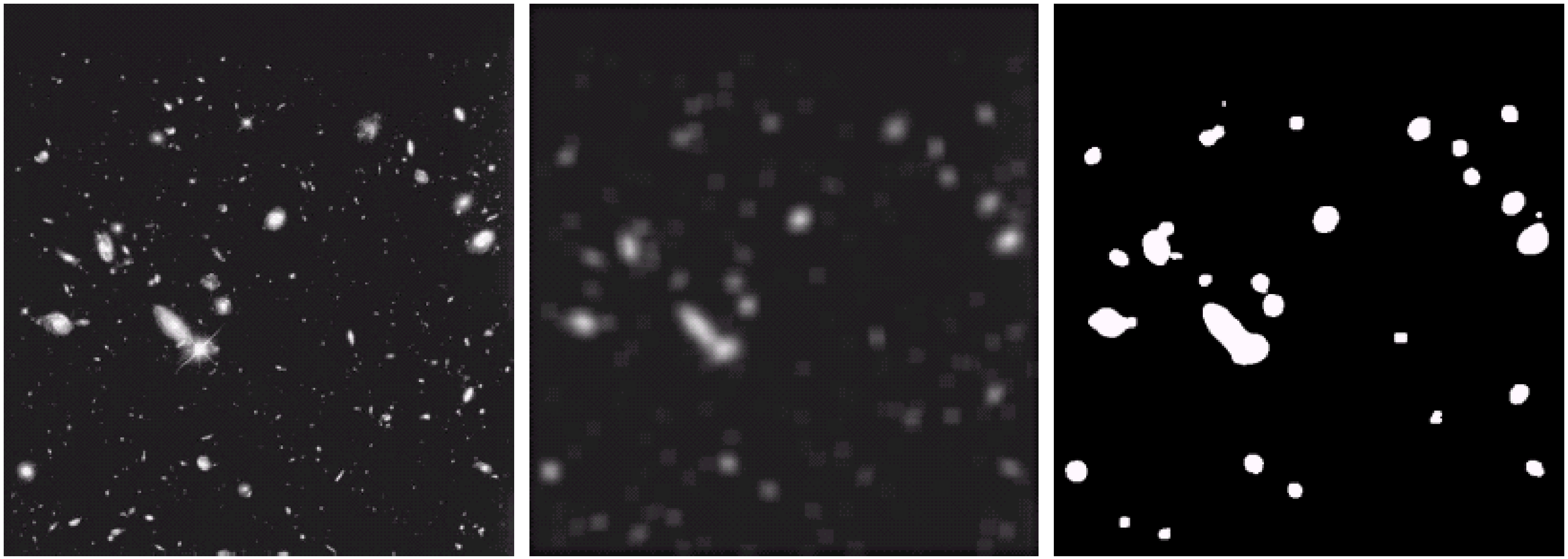
$$g(x, y) = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t) f(x + s, y + t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s, t)}$$

Averaging with different mask sizes

a b
c d
e f

FIGURE 3.35 (a) Original image, of size 500 × 500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $n = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their gray levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size 50 × 120 pixels.

An Application of Averaging: Averaging before Thresholding



a b c

FIGURE 3.36 (a) Image from the Hubble Space Telescope. (b) Image processed by a 15 × 15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

3.5.2 Order-Statistics Filters: *nonlinear*

Median Filter

-- the 50th percentile of a ranked set of numbers

-- effective for reducing *impulse noise*,

or *salt-and-pepper noise*

Max Filter

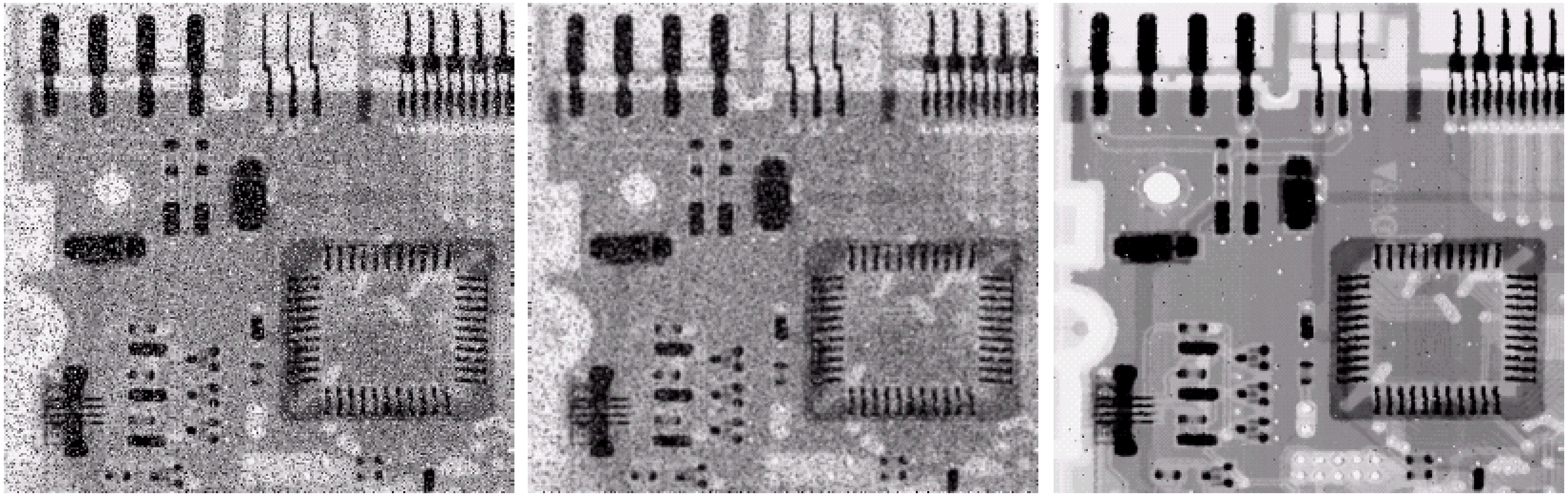
-- the 100th percentile filter

More in chapter 5

Min Filter

-- the 0th percentile filter

Comparison between Averaging Filter and Median Filter



a b c

FIGURE 3.37 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Outlines

3.1 Background

3.2 Some Basic Intensity Transformation Functions

3.3 Histogram Processing

3.4 Fundamentals of Spatial Filtering

3.5 Smoothing Spatial Filters

3.6 Sharpening Spatial Filters

3.7 Combining Spatial Enhancement Methods

3.6 Sharpening Spatial Filters

3.6.1 Foundation

3.6.2 The Laplacian

3.6.3 The Gradient

First-order derivatives

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

Second-order derivatives

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

3.6.1 Foundation -- A 1-D example

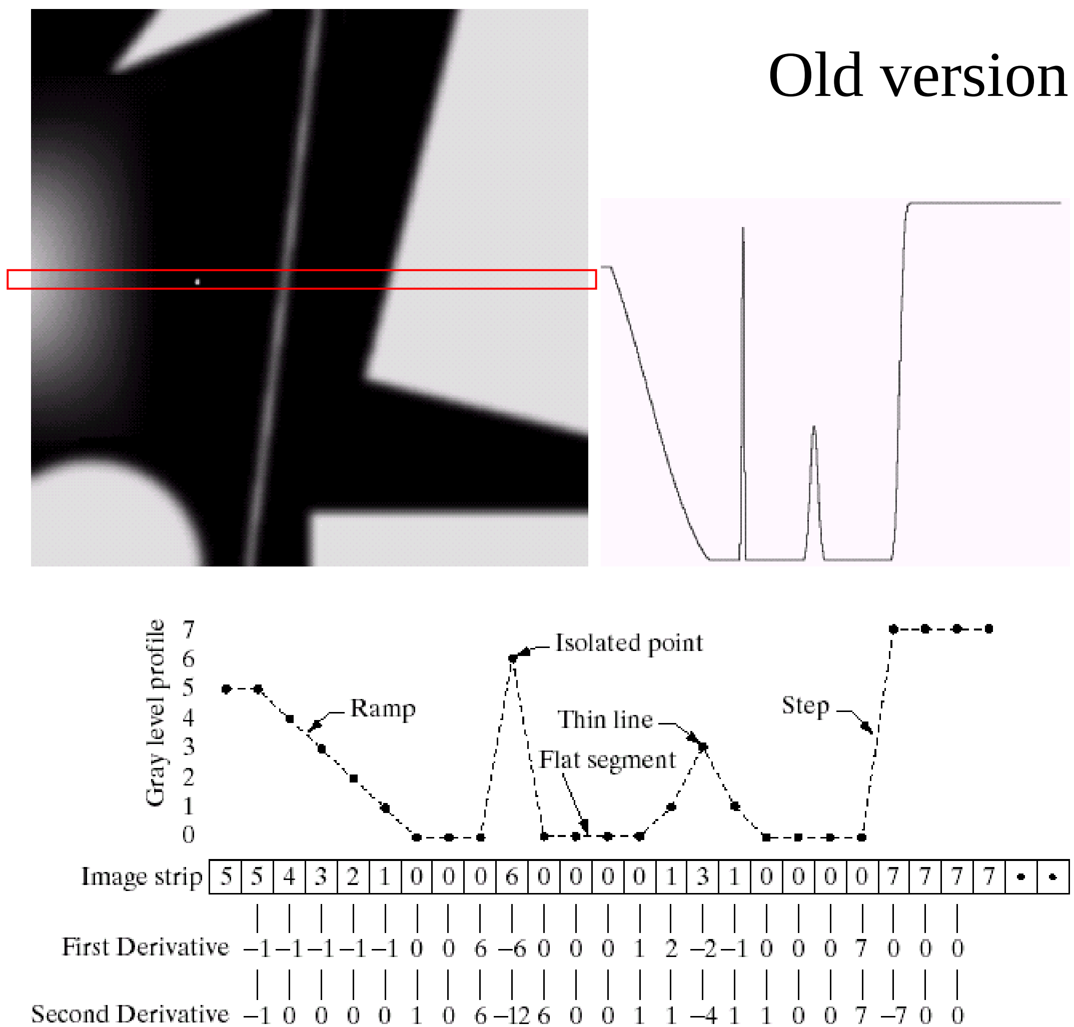
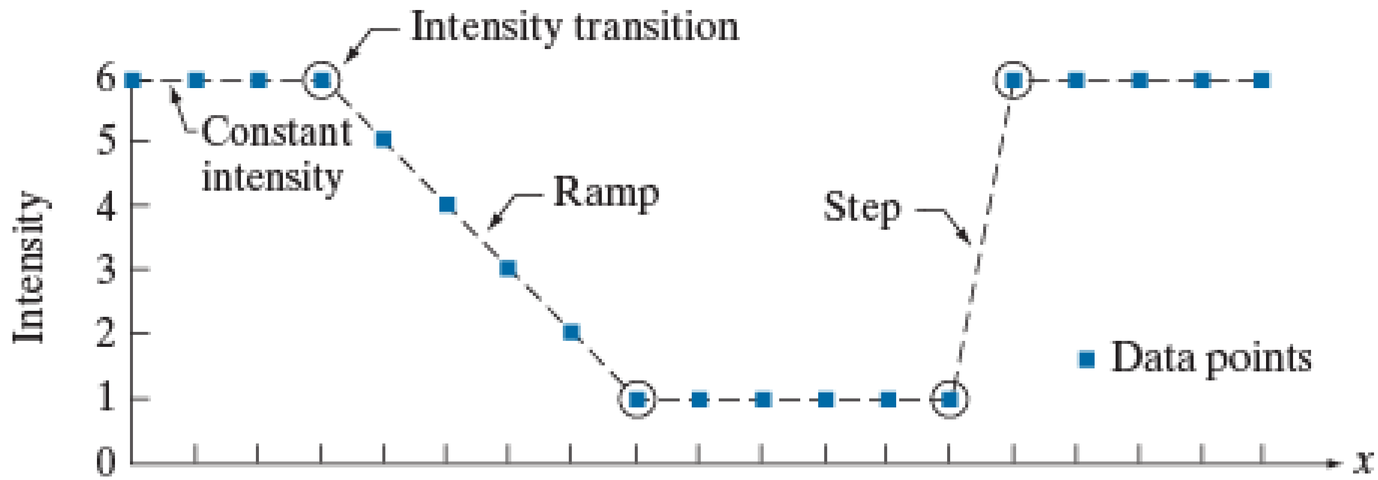
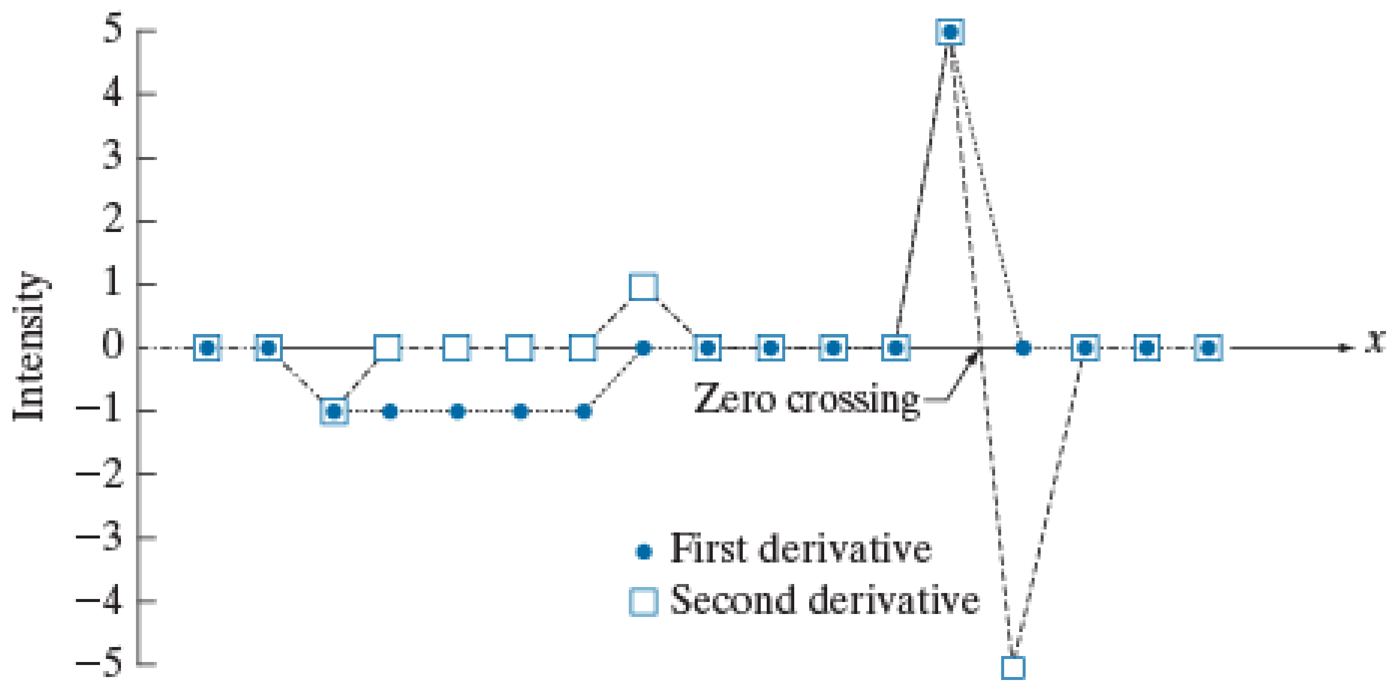


FIGURE 3.38 (a) A simple image. (b) 1-D horizontal gray-level profile along the center of the image and including the isolated noise point. (c) Simplified profile (the points are joined by dashed lines to simplify interpretation).



Values of scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6
1st derivative		0	0	−1	−1	−1	−1	−1	0	0	0	0	0	5	0	0	0	0	
2nd derivative		0	0	−1	0	0	0	0	1	0	0	0	0	5	−5	0	0	0	



3.6.2 The Laplacian
-- isotropic, i.e., rotation-invariant ?

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

Discrete form

$$\begin{cases} \dfrac{\partial^2 f}{\partial^2 x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y) \\[2ex] \dfrac{\partial^2 f}{\partial^2 y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y) \end{cases}$$

$$\nabla^2 f = [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] - 4f(x, y).$$

Filter Mask for Digital Laplacian

isotropic for increments of 90^o

including diagonal neighbors

isotropic for increments of 45^o

(a)

0	1	0
1	–4	1
0	1	0

(b)

1	1	1
1	–8	1
1	1	1

(c)

0	–1	0
–1	4	–1
0	–1	0

(d)

–1	–1	–1
–1	8	–1
–1	–1	–1

a	b
c	d

FIGURE 3.39 (a) Filter mask used to implement the digital Laplacian, as defined in Eq. (3.7-4). (b) Mask used to implement an extension of this equation that includes the diagonal neighbors. (c) and (d) Two other implementations of the Laplacian.

Laplacian-Based Enhancement -- Unsharp Masking, Crispening

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y) & \text{if the center coefficient of the Laplacian mask is negative} \\ f(x, y) + \nabla^2 f(x, y) & \text{if the center coefficient of the Laplacian mask is positive.} \end{cases}$$

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

$$\begin{aligned} g(x, y) &= f(x, y) - [f(x + 1, y) + f(x - 1, y) \\ &\quad + f(x, y + 1) + f(x, y - 1)] + 4f(x, y) \\ &= 5f(x, y) - [f(x + 1, y) + f(x - 1, y) \\ &\quad + f(x, y + 1) + f(x, y - 1)]. \end{aligned}$$

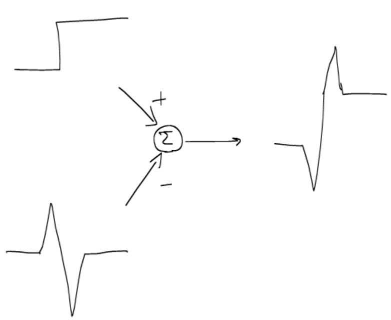
shown Black ⇒ ∵ negative values are clipped at 0 by the display

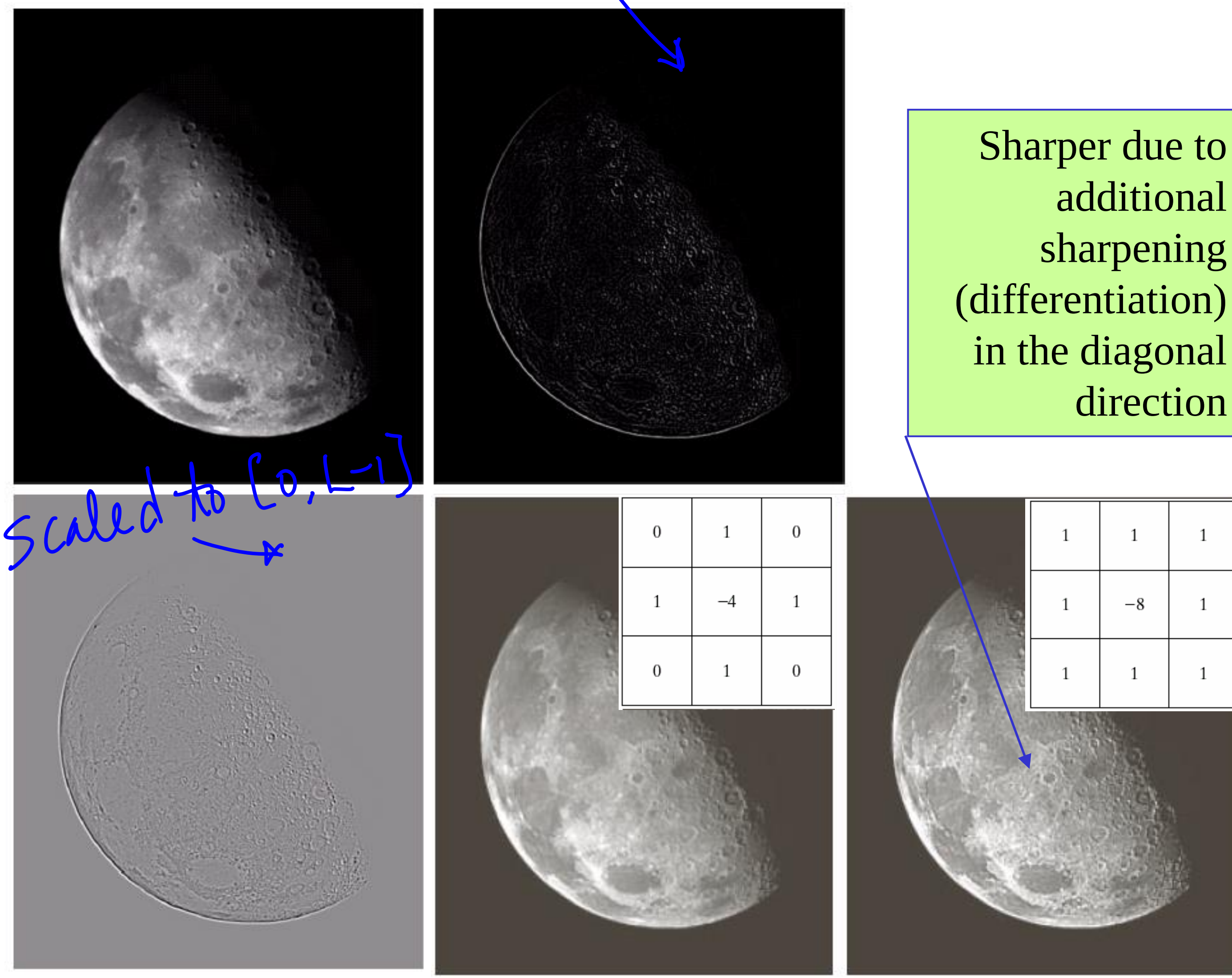
An Example of Laplacian-Based Enhancement

$$\begin{matrix} 0 & 0 & 100 & 100 \\ 0 & 0 & 100 & 100 \\ 0 & 0 & 100 & 100 \end{matrix}$$

$$\boxed{\begin{matrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{matrix}} \quad \Downarrow$$

$$\begin{matrix} 0 & 100 & -100 & 0 \\ 0 & 100 & -100 & 0 \\ 0 & 100 & -100 & 0 \end{matrix}$$





0	1	0
1	−4	1
0	1	0

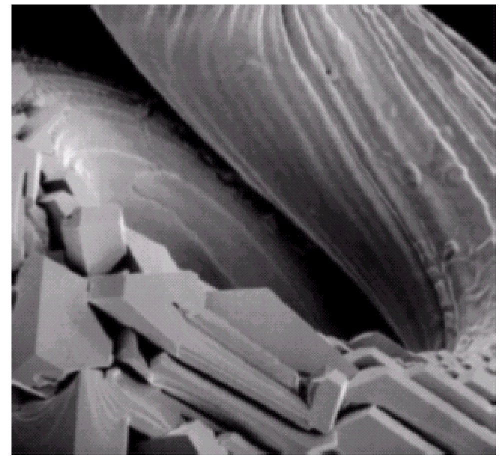
1	1	1
1	−8	1
1	1	1

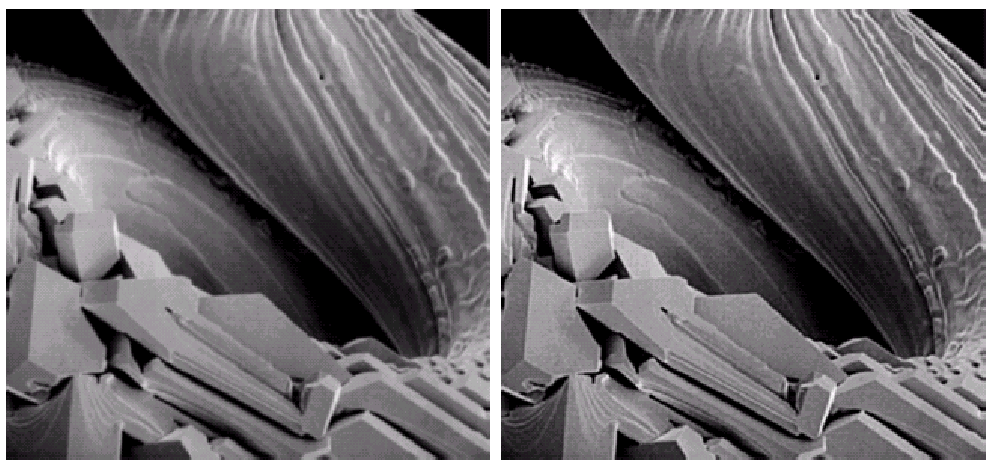
Sharper due to additional sharpening (differentiation) in the diagonal direction

Laplacian-Based Enhancement
-- implemented with one pass of a single mask

0	−1	0
−1	5	−1
0	−1	0

−1	−1	−1
−1	9	−1
−1	−1	−1





a b c
d e

FIGURE 3.41 (a) Composite Laplacian mask. (b) A second composite mask. (c) Scanning electron microscope image. (d) and (e) Results of filtering with the masks in (a) and (b), respectively. Note how much sharper (e) is than (d). (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene.)

Unsharp Masking (Crispening)

-- a common process in the publishing industry

- Enhance the edges
  - Edge enhancement means first isolating the edges in an image (high freq, e.g., Laplacian), amplifying them, and then adding them back into the image

$$
\begin{aligned}
f_s(x,y) &= f(x,y) + c\ f_{HF}(x,y) \\
&= f(x,y) + c\,\nabla^2 f(x,y)
\end{aligned}
$$

- Subtracts the “unsharp” (smoothed)
  - subtract a specified fraction of the smoothed (“unsharp”) image from the original, then add the result back to the original

$$
\begin{aligned}
f_s(x,y) &= f(x,y) + k\ g_{mask}(x,y) \\
&= f(x,y) + k\,[\,f(x,y) - \bar{f}(x,y)]
\end{aligned}
$$

Unsharp Masking

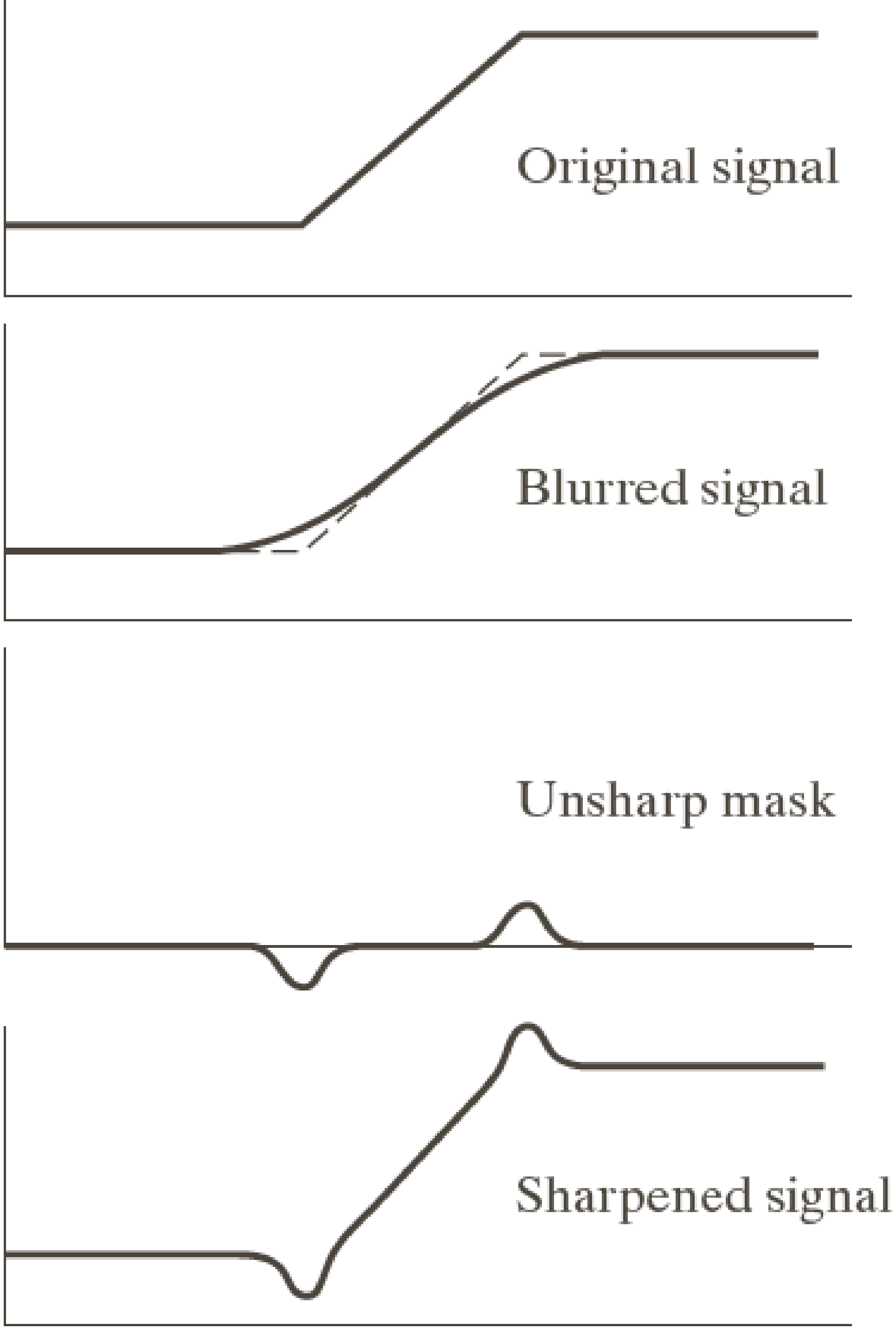
1. Blur the original image
2. Subtract the blurred image from the original
   - -> mask
3. Add the mask to the original

$$f_s(x,y) = f(x,y) + k\ g_{mask}(x,y)$$

$$= f(x,y) + k\,[\underbrace{f(x,y) - \overbrace{\bar{f}(x,y)}^{\text{blurred image}}}_{\text{mask}}]$$

k=1, ➔ unsharp masking

k>1, ➔ highboost filtering



a
b
c
d

FIGURE 3.39 1-D illustration of the mechanics of unsharp masking. (a) Original signal. (b) Blurred signal with original shown dashed for reference. (c) Unsharp mask. (d) Sharpened signal, obtained by adding (c) to (a).

k=1

k=4.5

a
b
c
d
e

FIGURE 3.40 (a) Original image. (b) Result of blurring with a Gaussian filter. (c) Unsharp mask. (d) Result of using unsharp masking. (e) Result of using highboost filtering.

3.6 Sharpening Spatial Filters

3.6.1 Foundation

3.6.2 The Laplacian

3.6.3 The Gradient (i.e., Gradient Magnitude)

First-order derivatives
Second-order derivatives

3.6.3 The Gradient

- Gradient Vector

$$\nabla \mathbf{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}.$$

- Gradient Magnitude

$$\begin{aligned} |\nabla f| &= \operatorname{mag}(\nabla \mathbf{f}) \\ &= \left[G_x^2 + G_y^2\right]^{1/2} \\ &= \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{1/2}. \end{aligned}$$

Approximation of Gradient Magnitude

$$|\nabla f| \approx |G_x| + |G_y|.$$

Digital Approximation ➔ Filter Masks

- Simplest Approximation

$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \qquad \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$G_x = (z_8 - z_5) \qquad G_y = (z_6 - z_5)$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Common Filter Masks for Computing the Gradient

a
b c
d e

FIGURE 3.44 A 3 × 3 region of an image (the z's are gray-level values) and masks used to compute the gradient at point labeled z_5. All masks coefficients sum

Roberts Cross-Gradient

derivative operator.

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

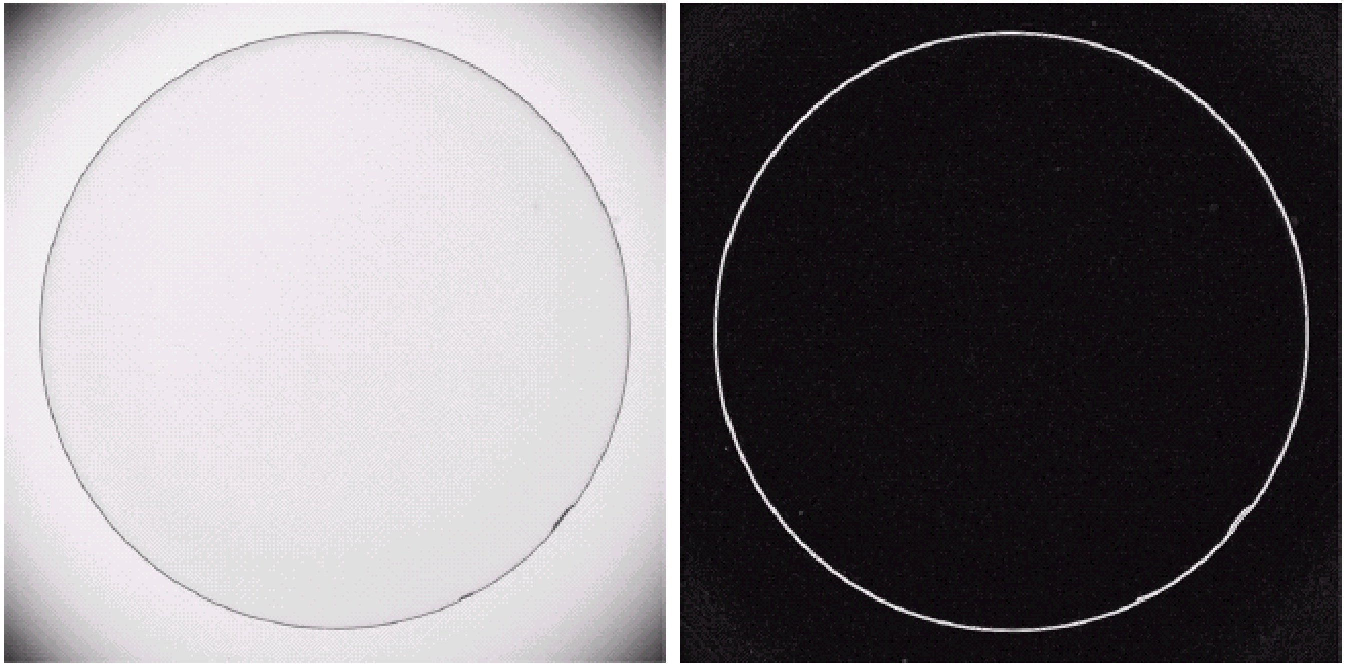
Sobel Operators

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

Gradient-based enhancement for automated inspection

-- Used as a pre-processing step for automated inspection



a b

FIGURE 3.45 Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Mr. Pete Sites, Perceptics Corporation.)

Outlines

3.1 Background

3.2 Some Basic Intensity Transformation Functions

3.3 Histogram Processing

3.4 Fundamentals of Spatial Filtering

3.5 Smoothing Spatial Filters

3.6 Sharpening Spatial Filters

3.7 Combining Spatial Enhancement Methods

3.7 Combining Spatial Enhancement Methods

a	b
c	d

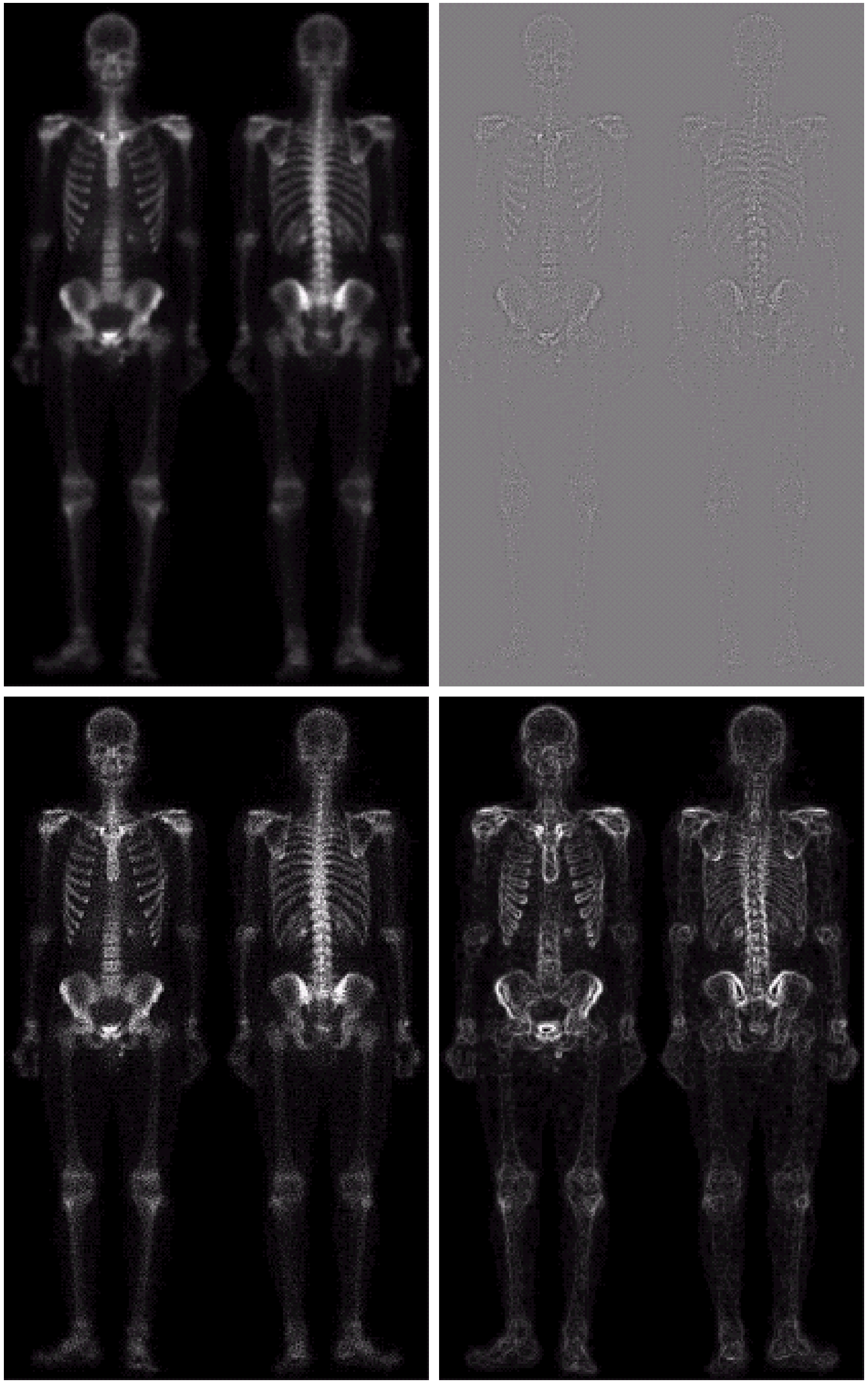
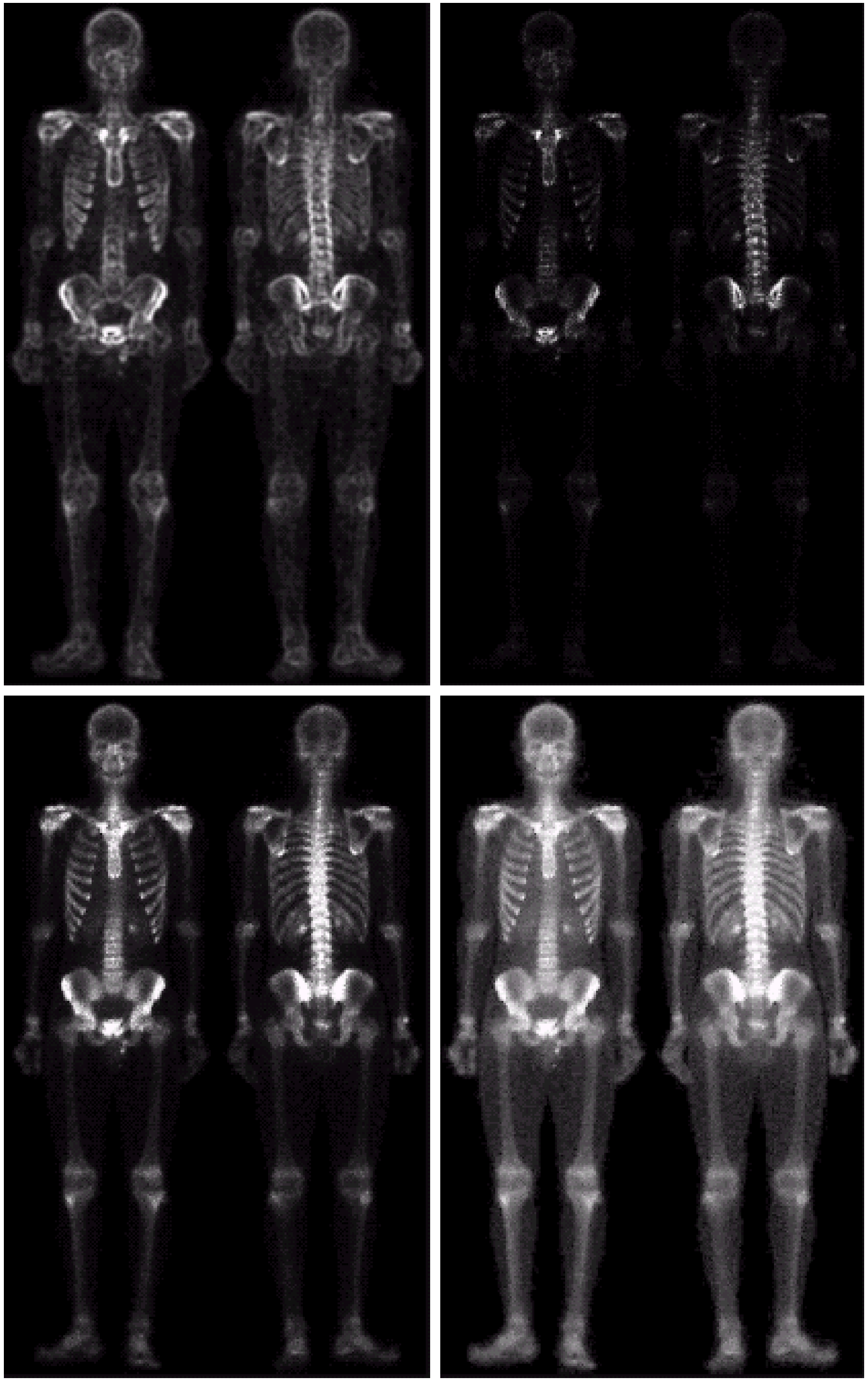


FIGURE 3.46
(a) Image of whole body bone scan. (b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel of (a).



e f
g h

FIGURE 3.46
(Continued)
(e) Sobel image smoothed with a 5 × 5 averaging filter. (f) Mask image formed by the product of (c) and (e). (g) Sharpened image obtained by the sum of (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

Homework Assignment #2
-- due: Nov 13, 10:20am

2.1 Textbook p. 240, Problem 3.8. (Explain why the discrete histogram…)

2.2 Textbook p. 244, Problem 3.44. (In the original image used to generate…)

2.3 Write a program for histogram equalization, and test it with your own selfie took in a relatively dark environment so that we can clearly see the effect of histogram equalization in image enhancement. Please show the histograms of your selfie before and after histogram equalization and explain your results. (Note: You only have to work on the gray-scale image.)