Chapter 3
Image Enhancement in the Spatial Domain

Preview

- Objective of enhancement- process an image so that the result is more suitable than the original image for a specific application
- Two image enhancement categories
  - Spatial domain methods
    - Spatial domain refers to the image plane itself
    - Based on direct manipulation of pixels in an image
  - Frequency domain methods
    - Based on modifying the Fourier transform of an image

3.1 Background

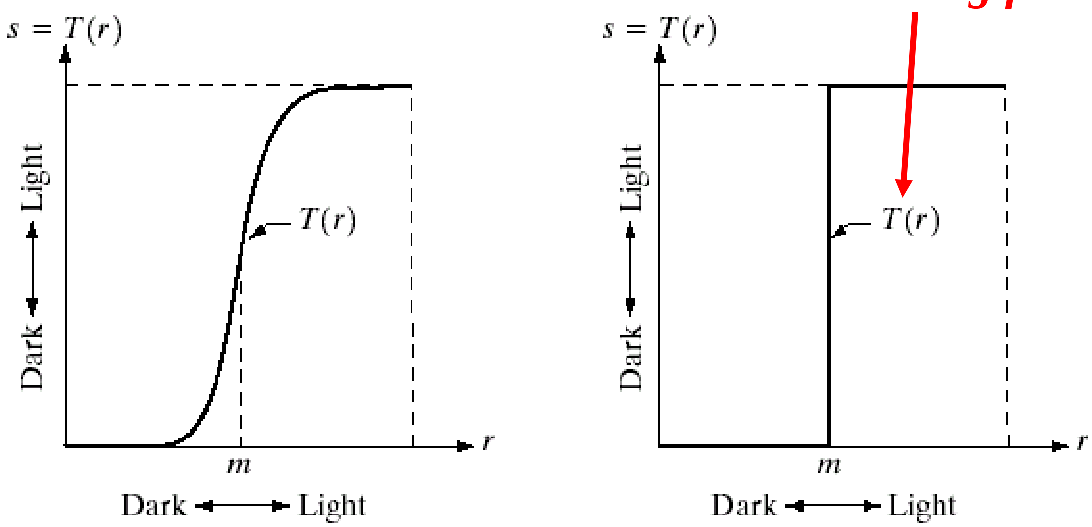
- Spatial domain processing can be denoted by
  - $g(x, y) = T[f(x, y)]$
    - $f(x, y)$: input image
    - $g(x, y)$: processed image
    - T: an operator on f, defined over some <u>neighborhood</u> of (x, y)
  - Point processing - neighborhood of size 1×1
  - Mask processing (filtering): neighborhood of size $m\times n$ (Section 3.5)

3.1 Background

- Point processing – g depends on the value of f at (x, y), T is called a gray-level (intensity, mapping) transformation function:

$$s = T(r)$$

thresholding function



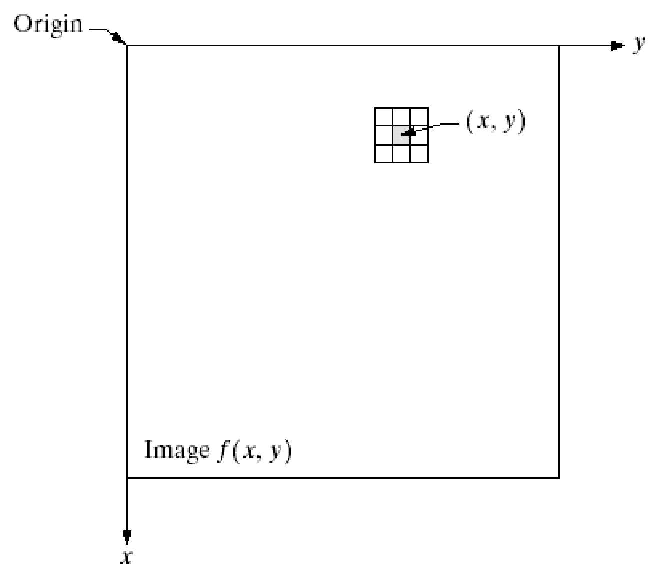
a b

FIGURE 3.2 Gray-level transformation functions for contrast enhancement.

3.1 Background

- Mask processing (filtering) – the value of g at (x, y) depends on the value of f in a predefined neighborhood of (x, y) called masks (filters, kernels, templates, windows)

FIGURE 3.1 A 3×3 neighborhood about a point (x, y) in an image.

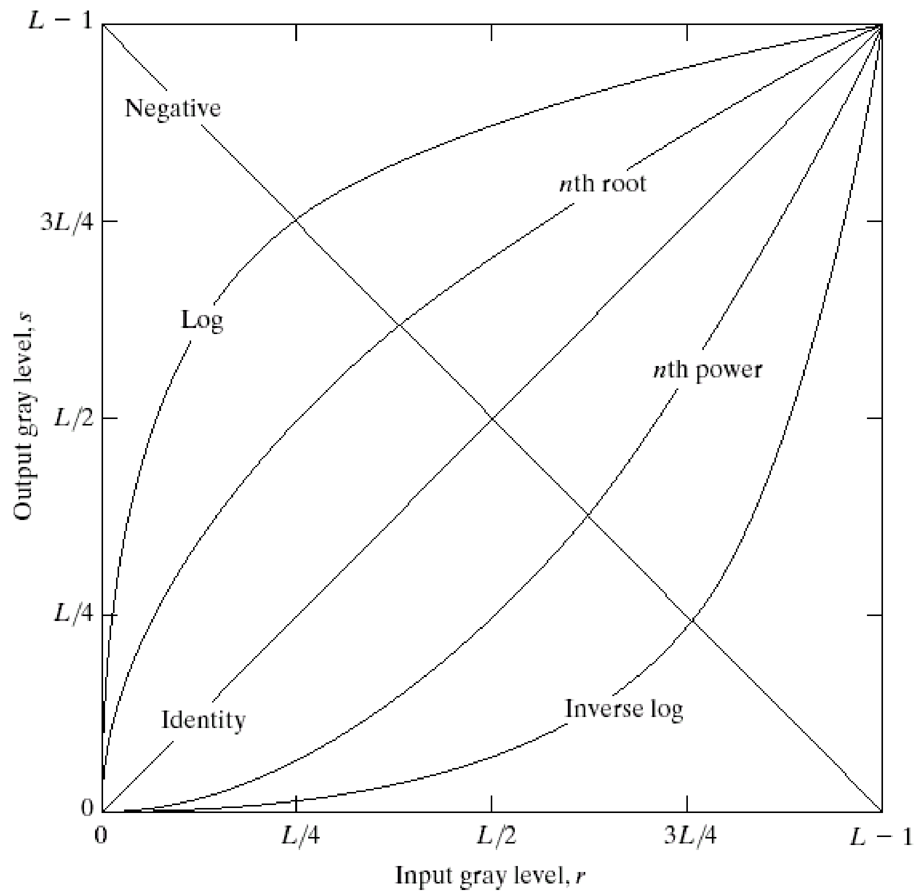


3.2 Some basic gray level transformations

- Three basic types of transformation functions used for image enhancement:
  - Linear (negative and identity transformations)
  - Logarithmic (log and inverse-log transformations)
  - Power-law (*n*-th power and *n*-th root transformations)

3.2 Some basic gray level transformations

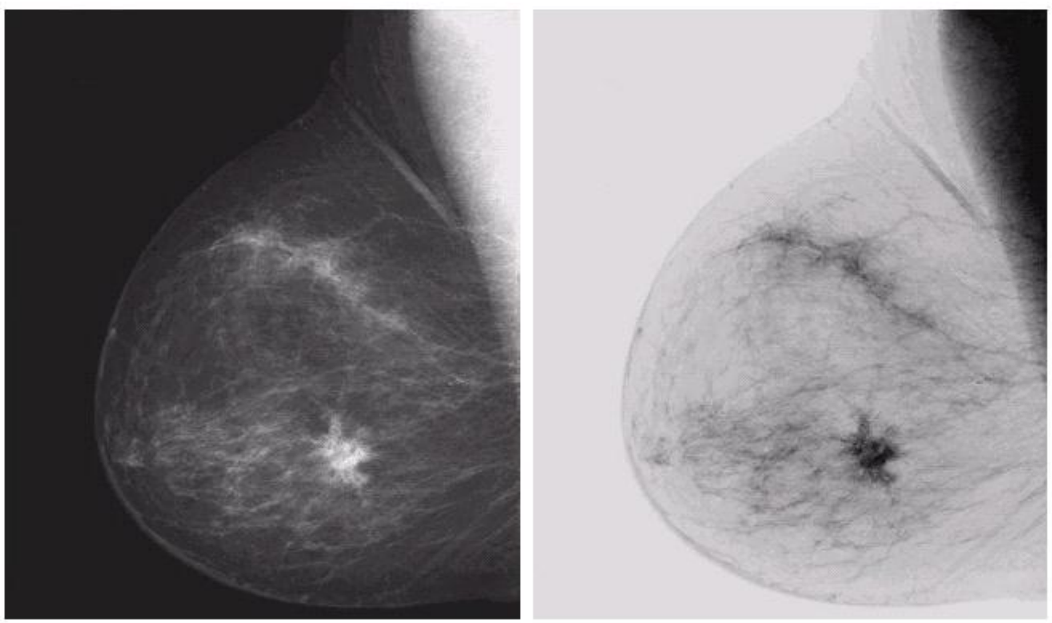
FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.



3.2.1 Image Negatives

- The negative of an image with gray levels in the range [0, L-1]:

  $s = L - 1 - r$（正片↔負片，相片↔底片）

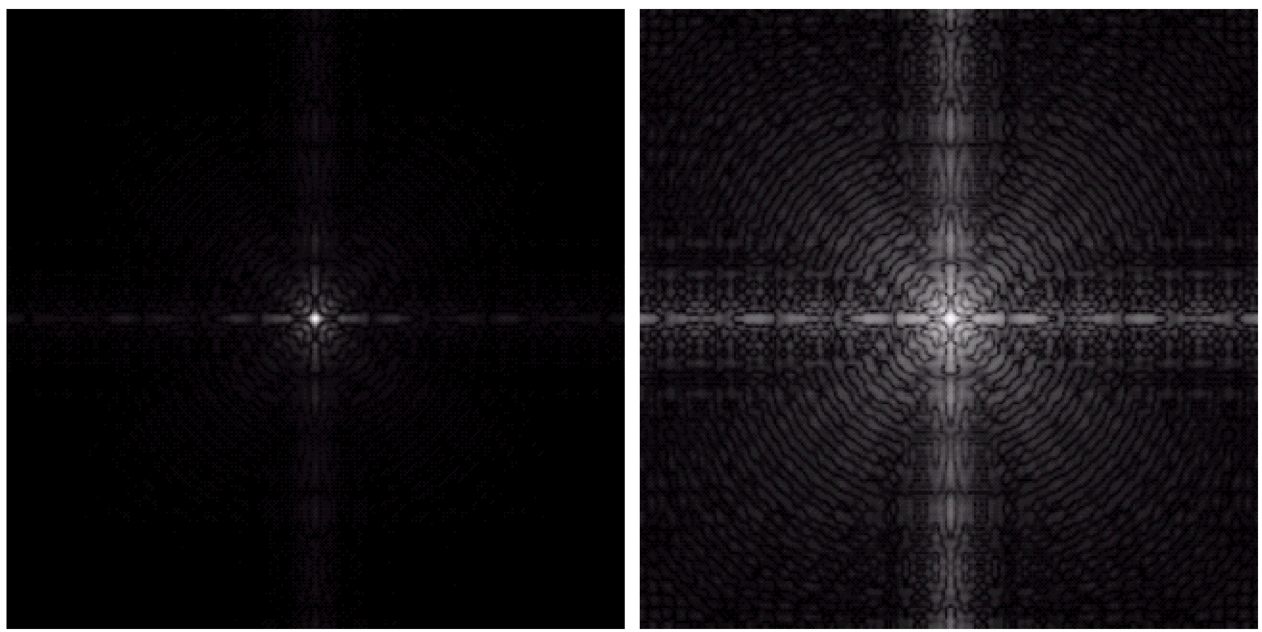


a b

FIGURE 3.4 (a) Original digital mammogram. (b) Negative image obtained using the negative transformation in Eq. (3.2-1). (Courtesy of G.E. Medical Systems.)

3.2.2 Log transformation

- $s = c\log(1+r)$（加強暗的區域）
  - Expand the values of dark pixels while compressing the higher-level values



a b

FIGURE 3.5 (a) Fourier spectrum. (b) Result of applying the log transformation given in Eq. (3.2-2) with $c = 1$.

3.2.3 Power-law transformations

- **Gamma correction**:

  $s = c\, r^{\gamma}$ or

  $s = c\, (r+\varepsilon)^{\gamma}$

  - c: positive constant
  - γ: gamma, positive constant
  - $\gamma > 1$: expand bright values while compressing dark values
  - $\gamma < 1$: expand dark values while compressing bright values

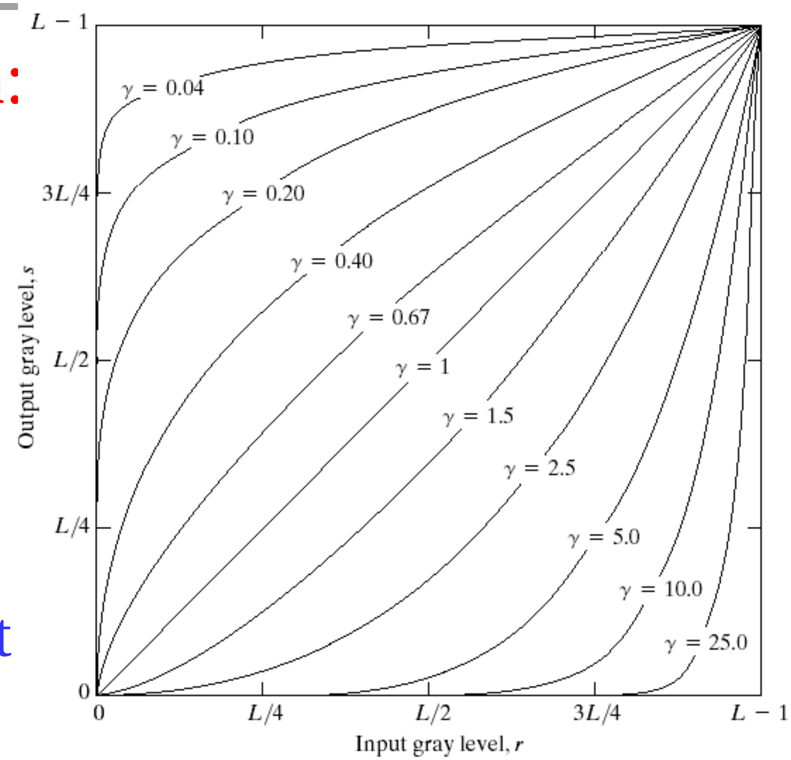


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

3.2.3 Power-law transformations

- Example (CRT):
  - The intensity-to-voltage response of CRT is a power function with γ = 1.8 ~ 2.5 (γ=2.5 is shown in Fig 3.7).
  - Gamma correction is important if displaying image accurately on a computer screen is of concern
    - Intensity
    - Color: the ratio among R,G, B.

3.2.3 Power-law transformations

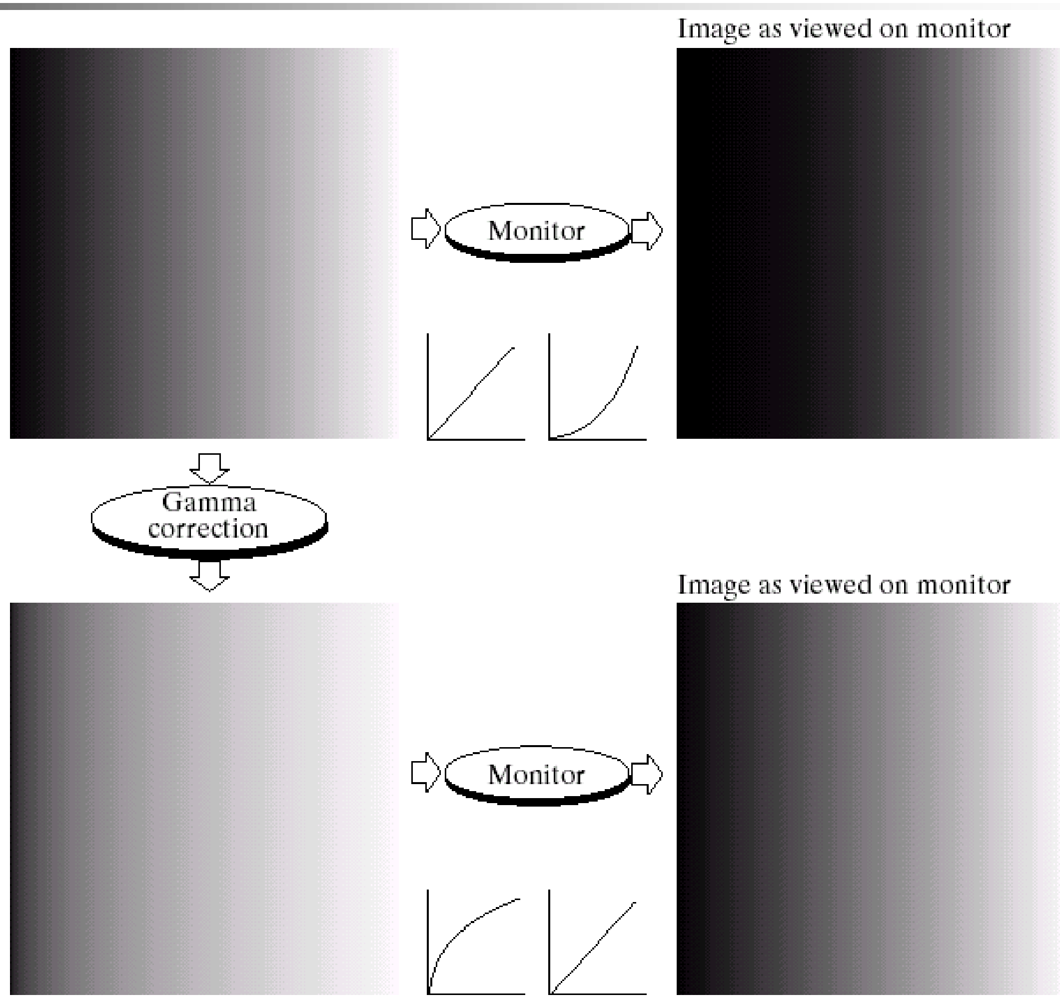
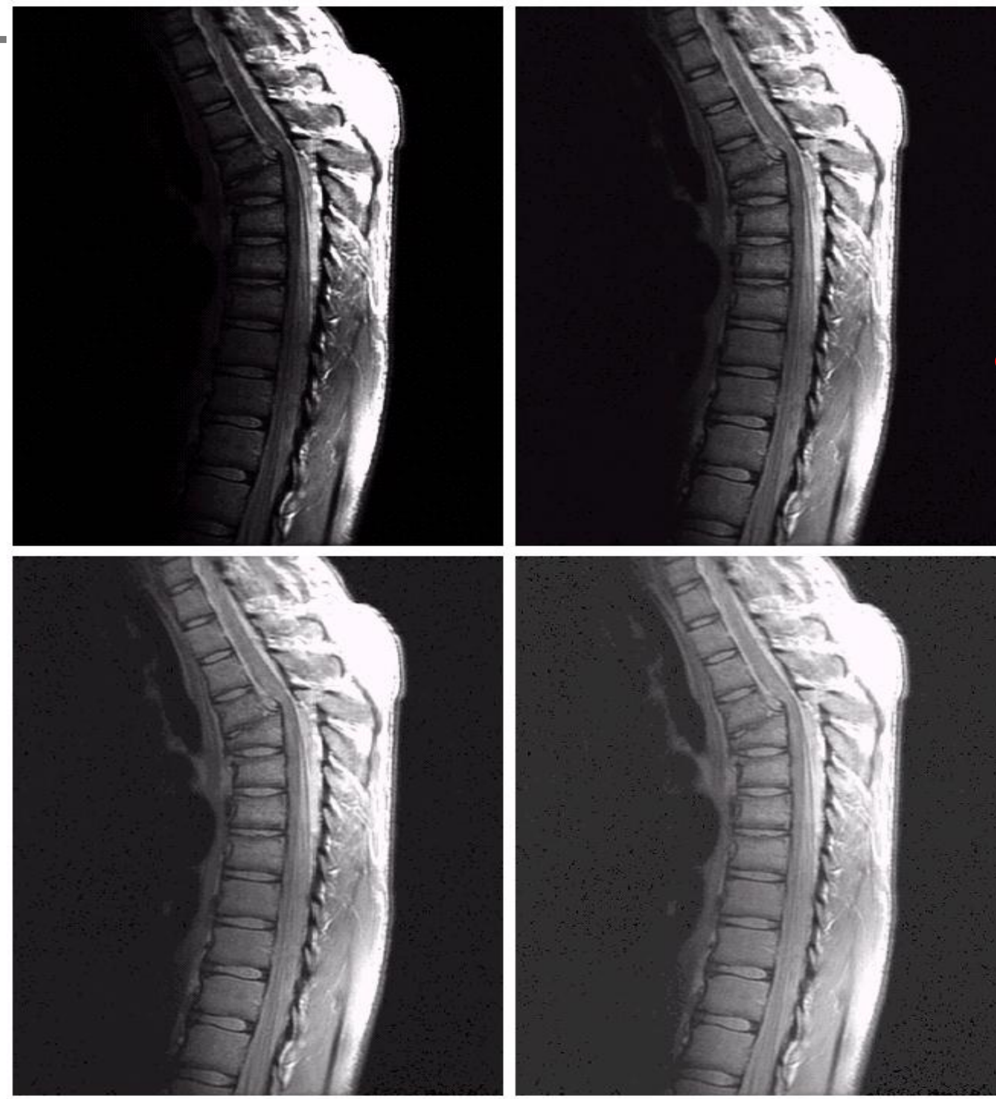


FIGURE 3.7 (a) Linear-wedge gray-scale image. (b) Response of monitor to linear wedge. (c) Gamma-corrected wedge. (d) Output of monitor.

3.2.3 Power-law transformations

- Ex. 3.1

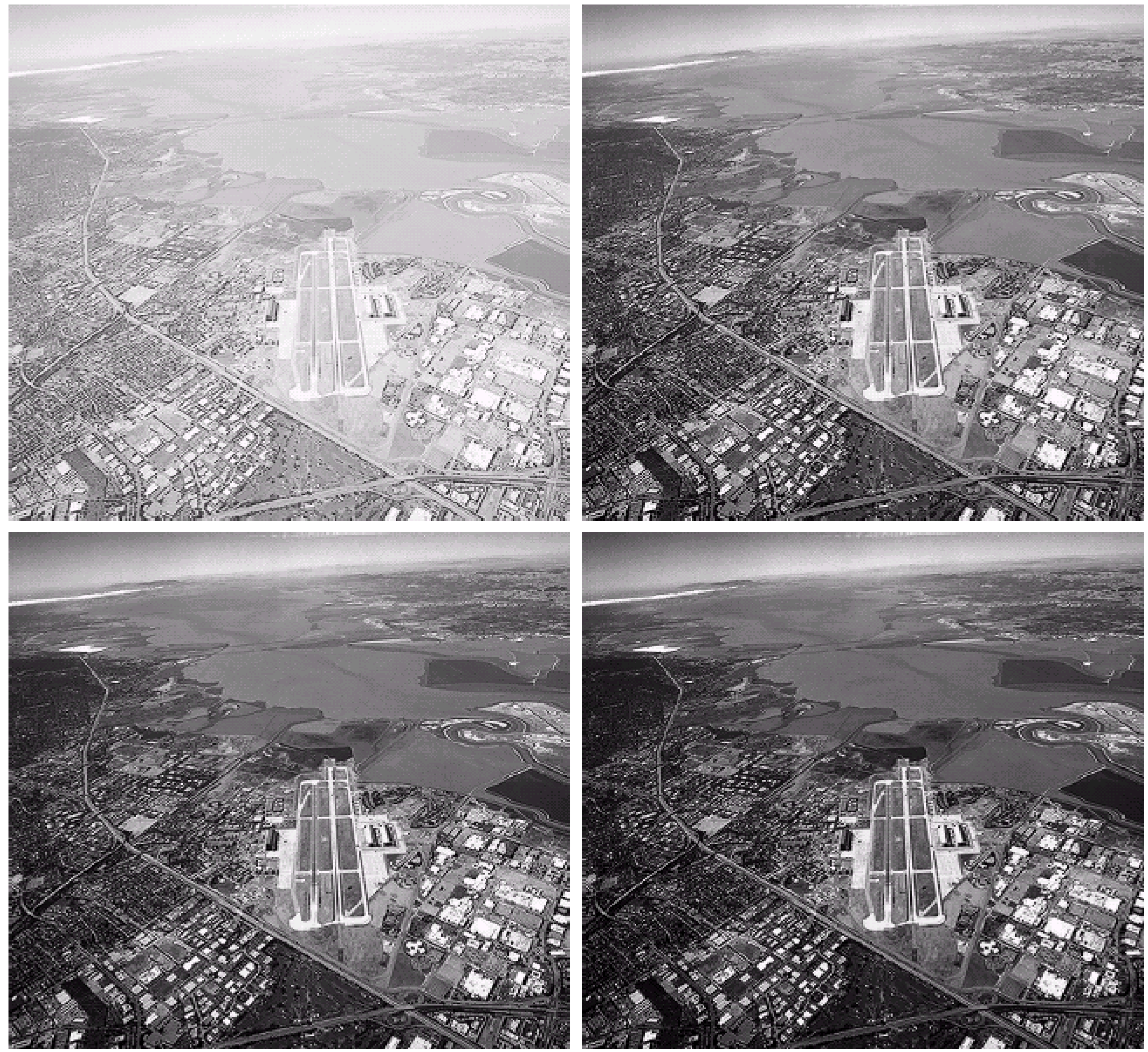
  Contrast Enhancement using power-law transformations



a b
c d

FIGURE 3.8 (a) Magnetic resonance (MR) image of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image for this example courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

3.2.3 Power-law transformations



a b
c d

FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 3.0, 4.0,$ and 5.0, respectively. (Original image for this example courtesy of NASA.)

- Ex 3.2

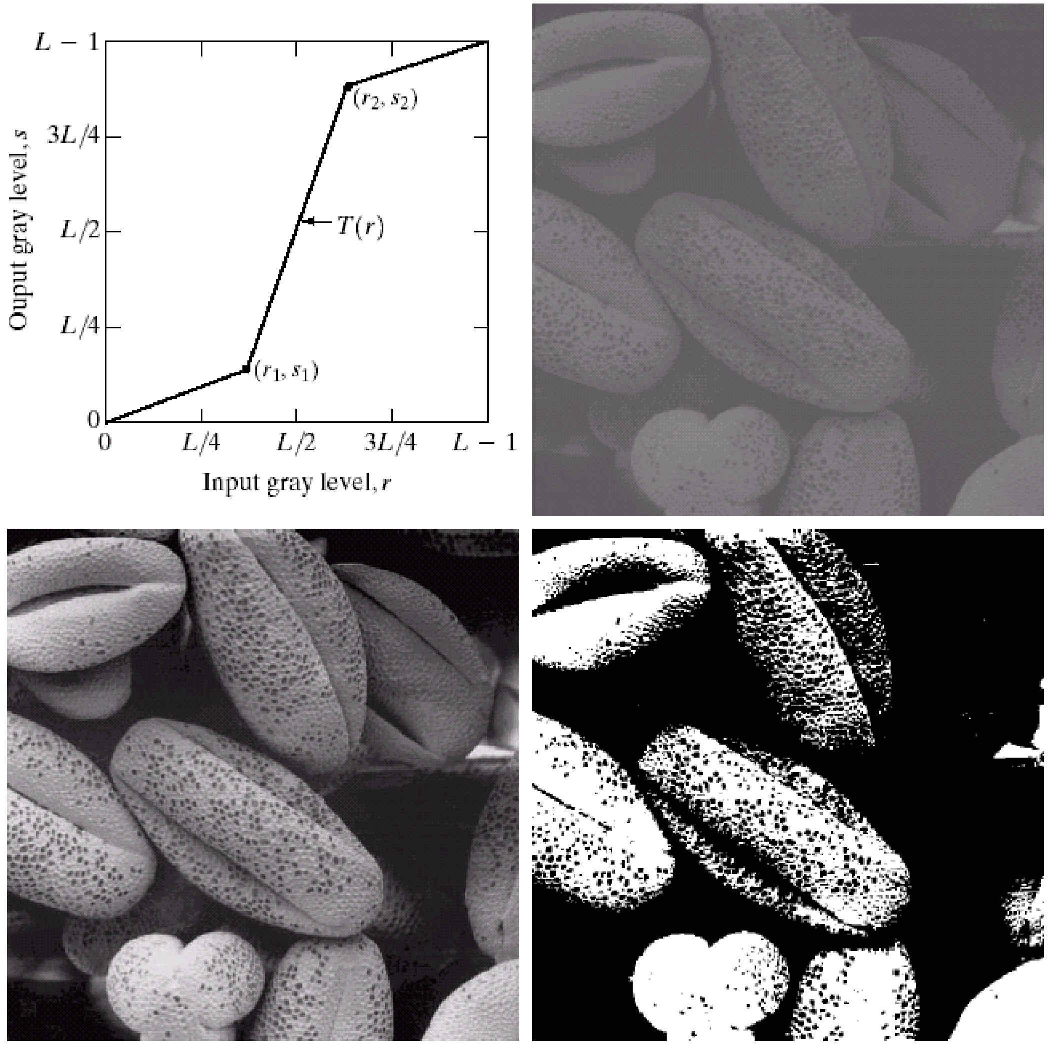
  Power-law transformations

3.2.4 Piecewise-Linear transformation functions

- Advantage – the form of piecewise functions can be arbitrarily complex
- Disadvantage – their specification <u>requires considerably more user input</u>
- Applications-
  - Contrast stretching
  - Gray-level slicing
  - Bit-plane clicing

3.2.4 Piecewise-Linear transformation functions

- Contrast stretching

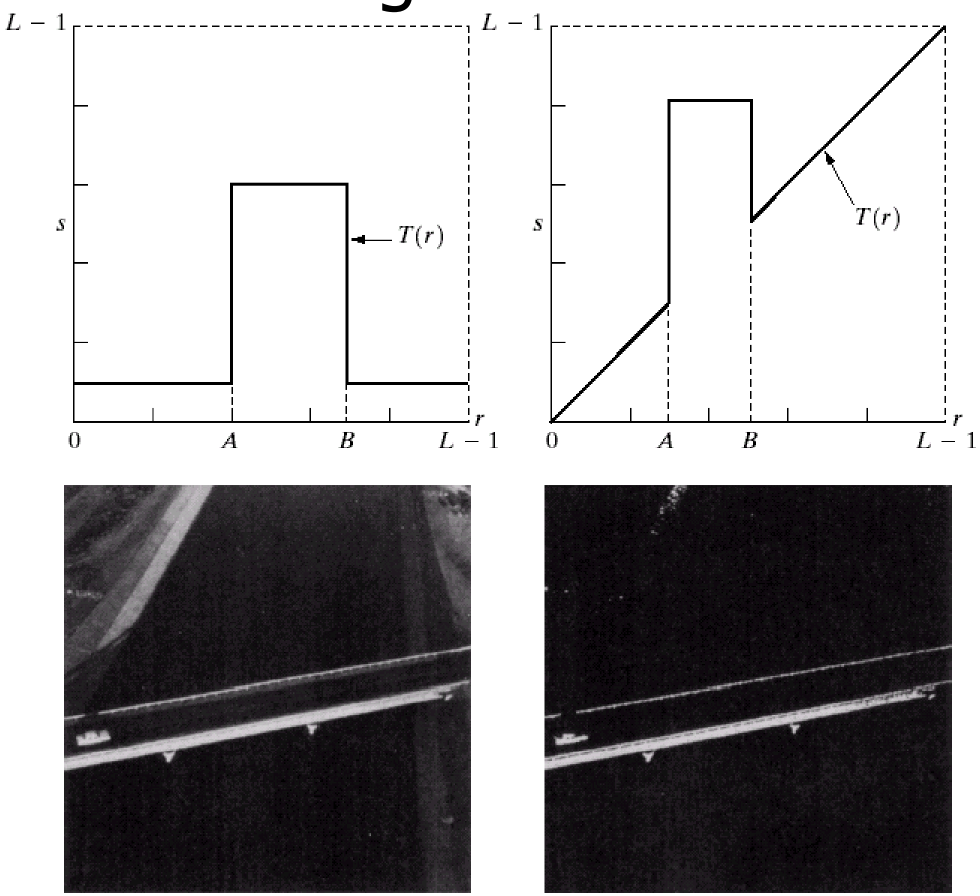


a b
c d

FIGURE 3.10 Contrast stretching. (a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

3.2.4 Piecewise-Linear transformation functions

- Gray-level slicing



a b
c d

FIGURE 3.11 (a) This transformation highlights range [A, B] of gray levels and reduces all others to a constant level. (b) This transformation highlights range [A, B] but preserves all other levels. (c) An image. (d) Result of using the transformation in (a).

3.2.4 Piecewise-Linear transformation functions

- Bit-plane slicing

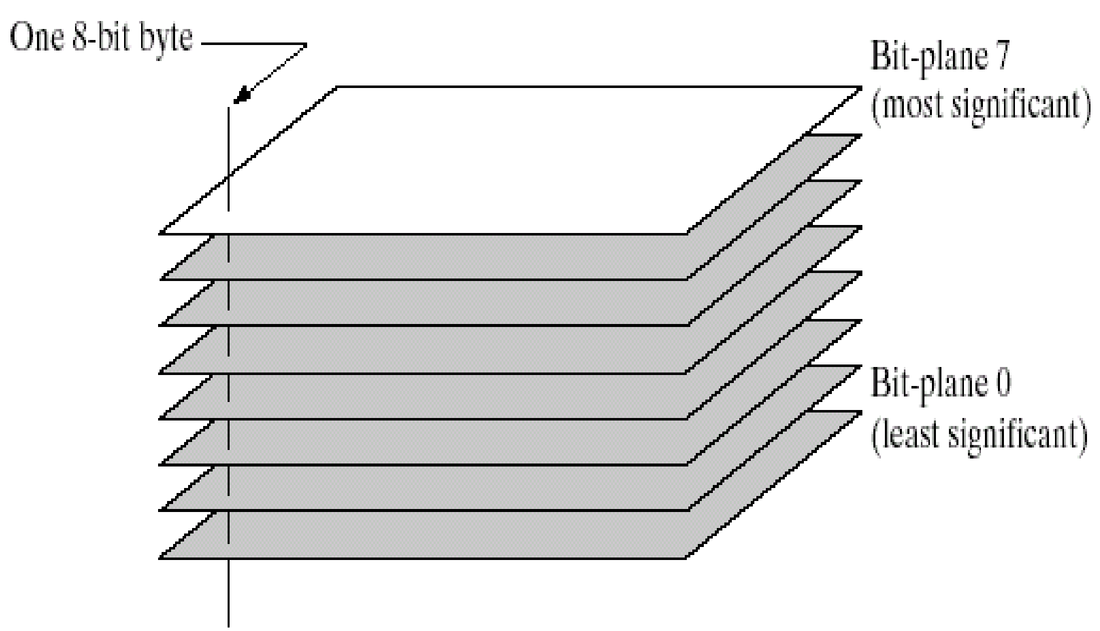


FIGURE 3.12 Bit-plane representation of an 8-bit image.

3.2.4 Piecewise-Linear transformation functions

- Bit-plane slicing
- Ex. An 8-bit fractal image

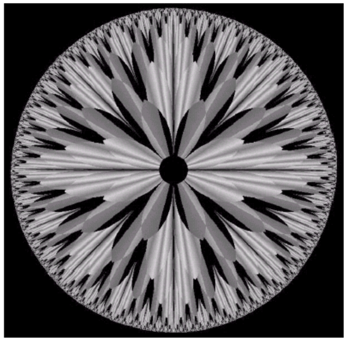


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)

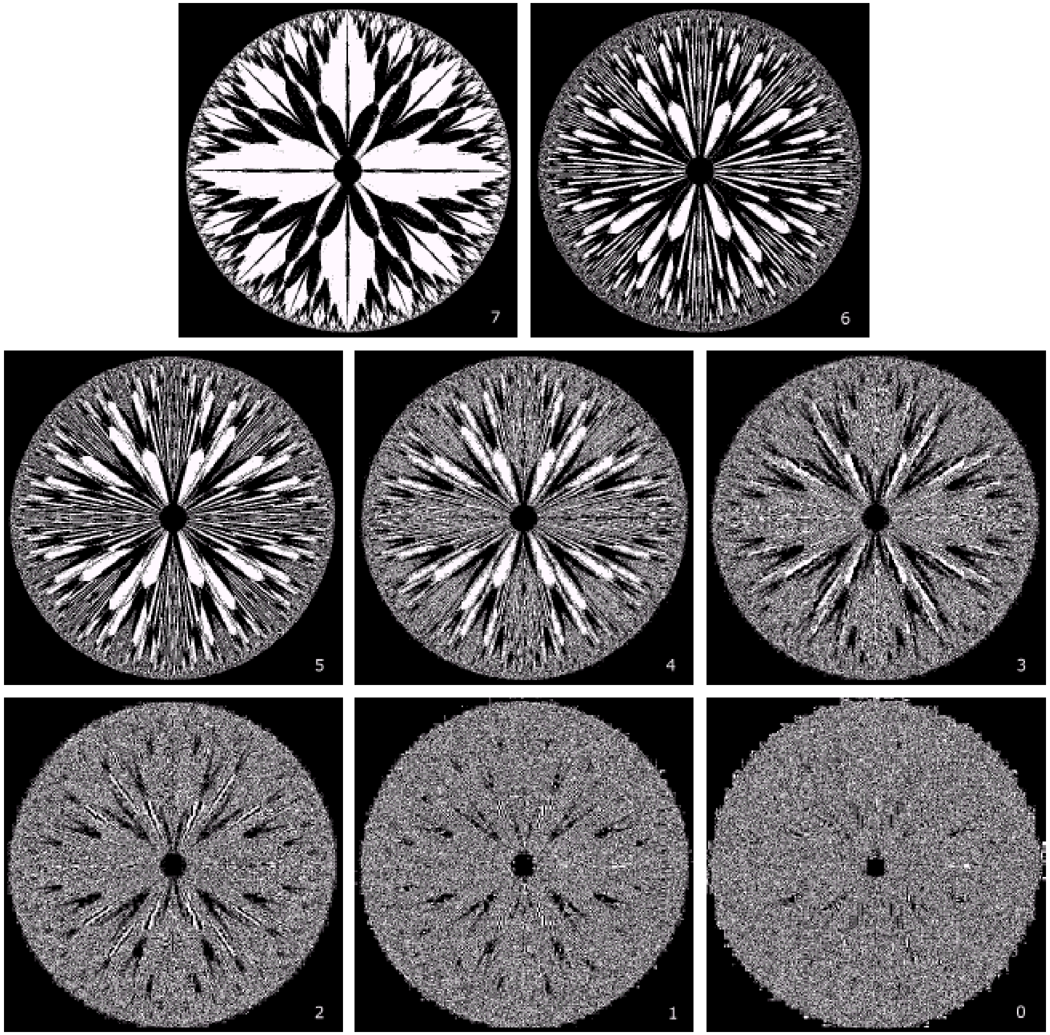


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

3.3 Histogram Processing

- Histogram processing for
  - Image enhancement
  - Image compression (Chapter 8)
  - Image/video segmentation (Chapter 10)
- Histogram- $h(r_k) = n_k$,
  - r_k is the k-th gray level, $r_k \in [0, L\text{-}1]$
  - n_k is the number of pixels in the image having gray level r_k
- Normalized histogram - $p(r_k) = n_k/n$
  - n is the total number of pixels in the image
  - An estimate of the probability of occurrence of gray level r_k

3.3 Histogram Processing

- Histogram in the high contrast image cover a broad range of the gray scale and the distribution is not far from the uniform distribution.

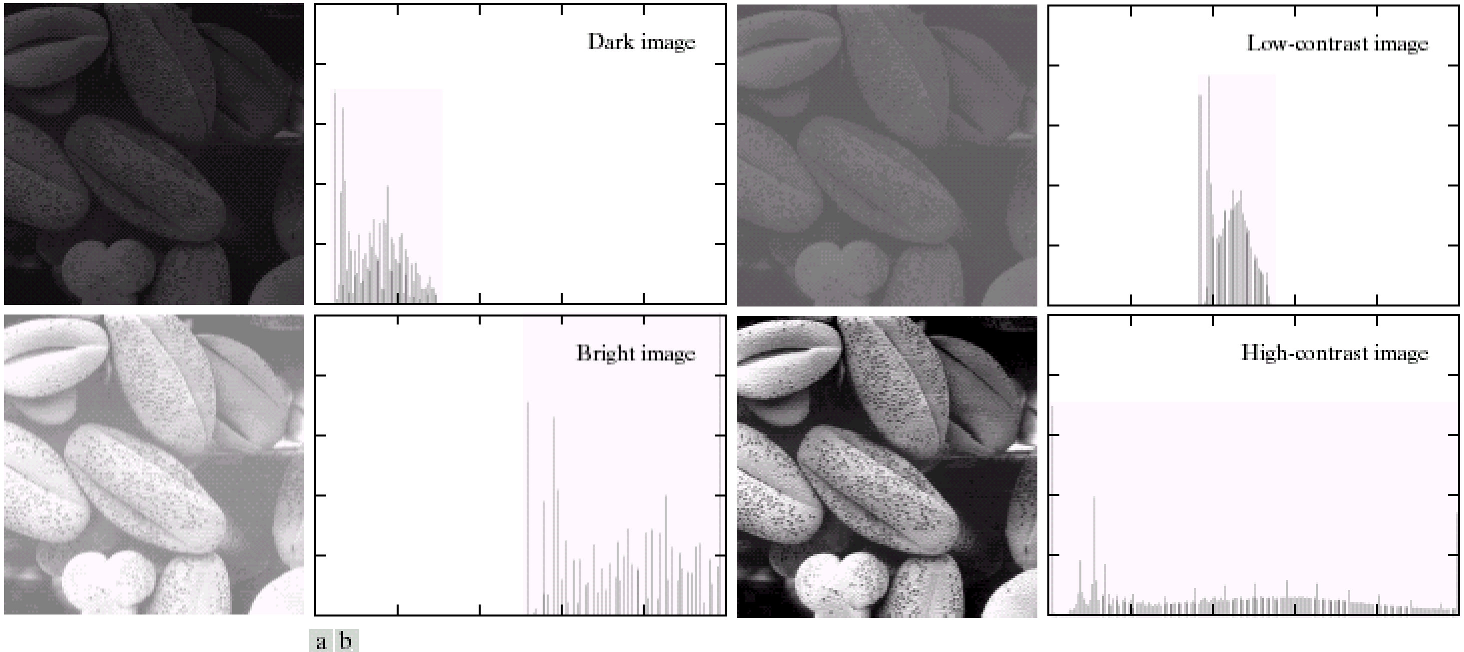


FIGURE 3.15 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

3.3.1 Histogram Equalization

- Histogram equalization :
  - $s = T(r)$, $0 \leqq r \leqq 1$
  - $T(r)$ is single-valued and monotonically increasing
  - $0 \leqq T(r) \leqq 1$
  - $T^{-1}(s)$?

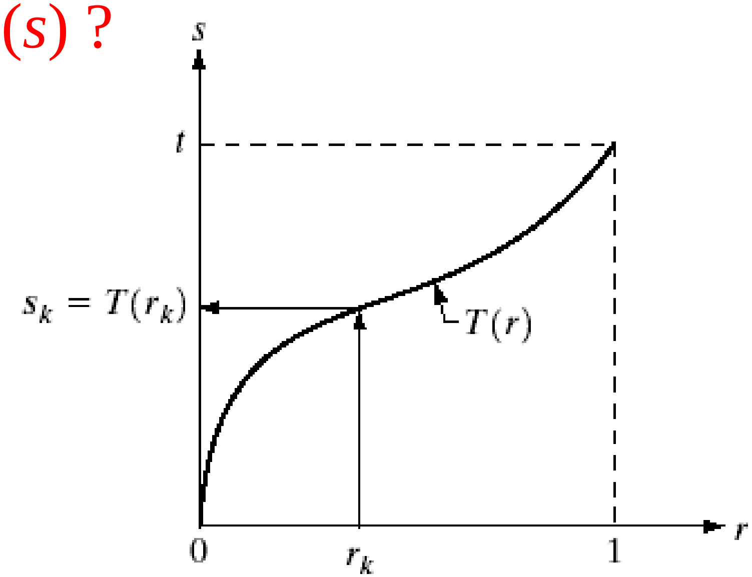


FIGURE 3.16 A gray-level transformation function that is both single valued and monotonically increasing.

3.3.1 Histogram Equalization

- Transformation function:

$$s = T(\mathrm{r}) \Rightarrow \mathrm{p}_{\mathrm{s}}(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

$$\text{If } s = T(r) = \int_0^r p_r(w)dw$$

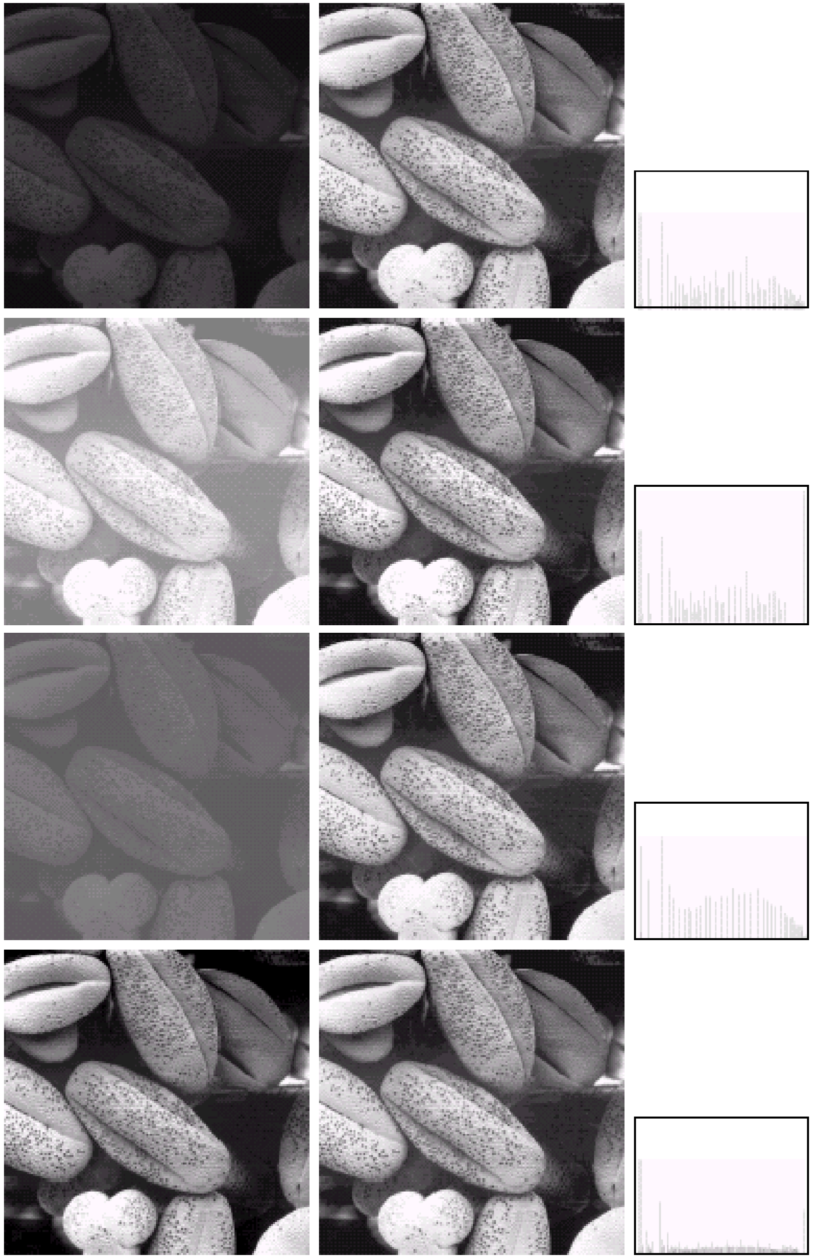
$$\text{then, } \frac{ds}{dr} = \frac{dT(r)}{dr} = p_r(r) \rightarrow \mathrm{p}_{\mathrm{s}}(s) = 1$$

- Discrete version:

$$p_r(r_k) = \frac{n_k}{n}, k = 0,1,2,\ldots,L-1$$

$$s_k = T(r_k) = \sum_{j=0}^{k} p_r(r_j) = \sum_{j=0}^{k} \frac{n_j}{n}$$

Example 3.3



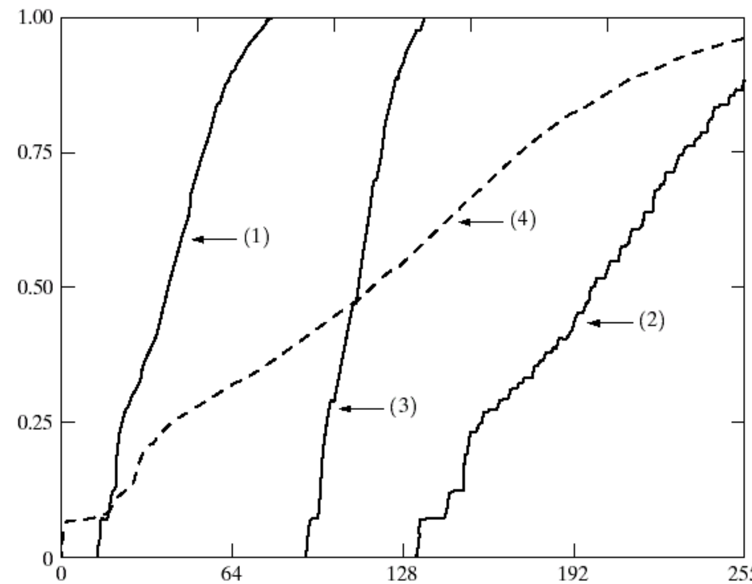


FIGURE 3.18 Transformation functions (1) through (4) were obtained from the histograms of the images in Fig.3.17(a), using Eq. (3.3-8).

a b c

FIGURE 3.17 (a) Images from Fig. 3.15. (b) Results of histogram equalization. (c) Corresponding histograms.

3.3.2 Histogram Matching (Specification)

- Definition- to process an image that has a specified histogram
- Objective: $p_r(r)$ is transformed to $p_z(z)$

$$s = T(r) = \int_0^r p_r(w)dw \quad s = G(z) = \int_0^z p_z(t)dt$$

  - $P_r(r)$: input histogram
  - $P_z(z)$: specified output histogram
- $z = G^{-1}(s) = G^{-1}[T(r)]$
- G^{-1} ?

3.3.2 Histogram Matching (Specification)

- Discrete formulation:

$$s_k = T(r_k) = \sum_{j=1}^{k} p_r(r_j) = \sum_{j=1}^{k} \frac{n_j}{n}, \; k = 0,1,2,\cdots,L-1$$

$$v_k = G(z_k) = \sum_{i=1}^{k} p_z(z_i) = s_k, \; k = 0,1,2,\cdots,L-1$$

➢ $z_k = G^{-1}[T(r_k)] = G^{-1}(s_k), \; k = 0,1,2,\cdots,L-1$

3.3.2 Histogram Matching (Specification)

- Implementation
  1. Form 1-D array: $\{r_j\}$, $\{s_j\}$, $\{z_j\}$
  2. Construct look-up tables (accumulative mapping function):
     $r_k \rightarrow s_k$
     $z_k \rightarrow v_k$
  3. For each k, find s_k such that $s_k = v_k$
  4. Inverse mapping: $s_k \rightarrow z_k$
- Details: please see pp. 99-100

3.3.2 Histogram Matching (Specification)

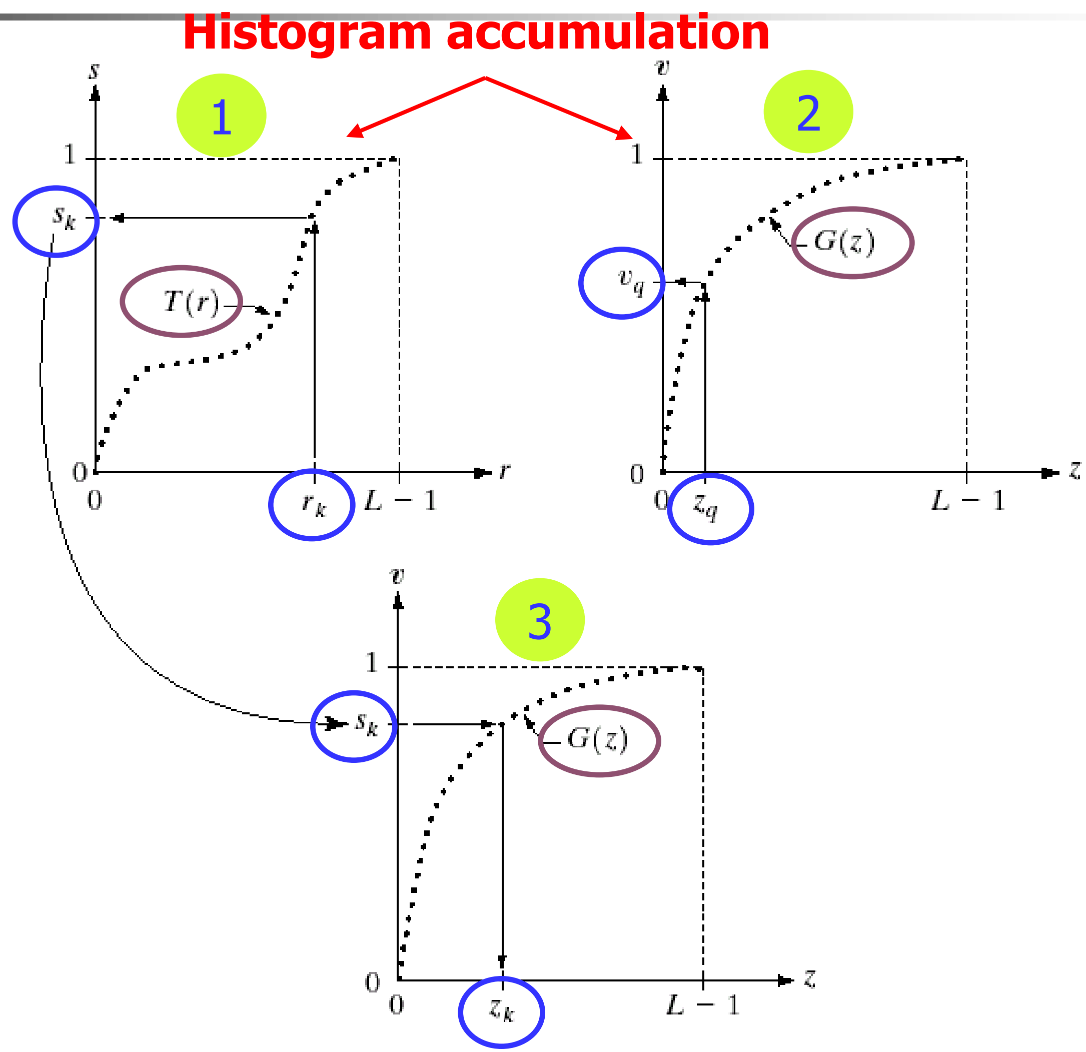
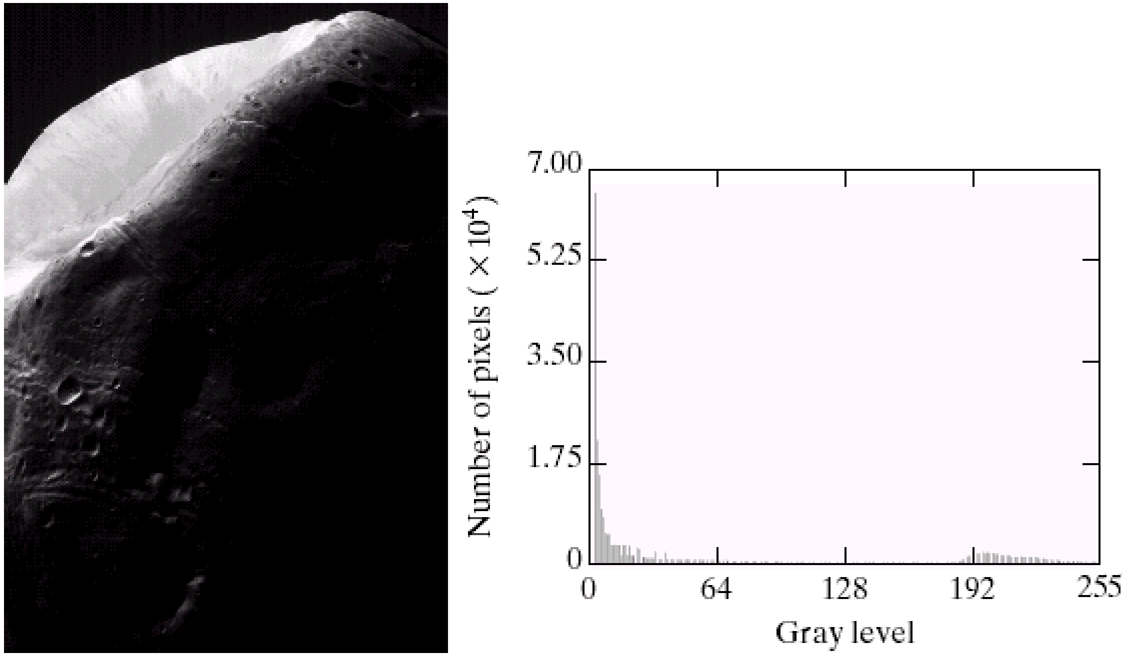


FIGURE 3.19 (a) Graphical interpretation of mapping from r_k to s_k via $T(r)$. (b) Mapping of z_q to its corresponding value v_q via $G(z)$. (c) Inverse mapping from s_k to its corresponding value of z_k.

3.3.2 Histogram Matching (Specification)

- Ex. 3.4

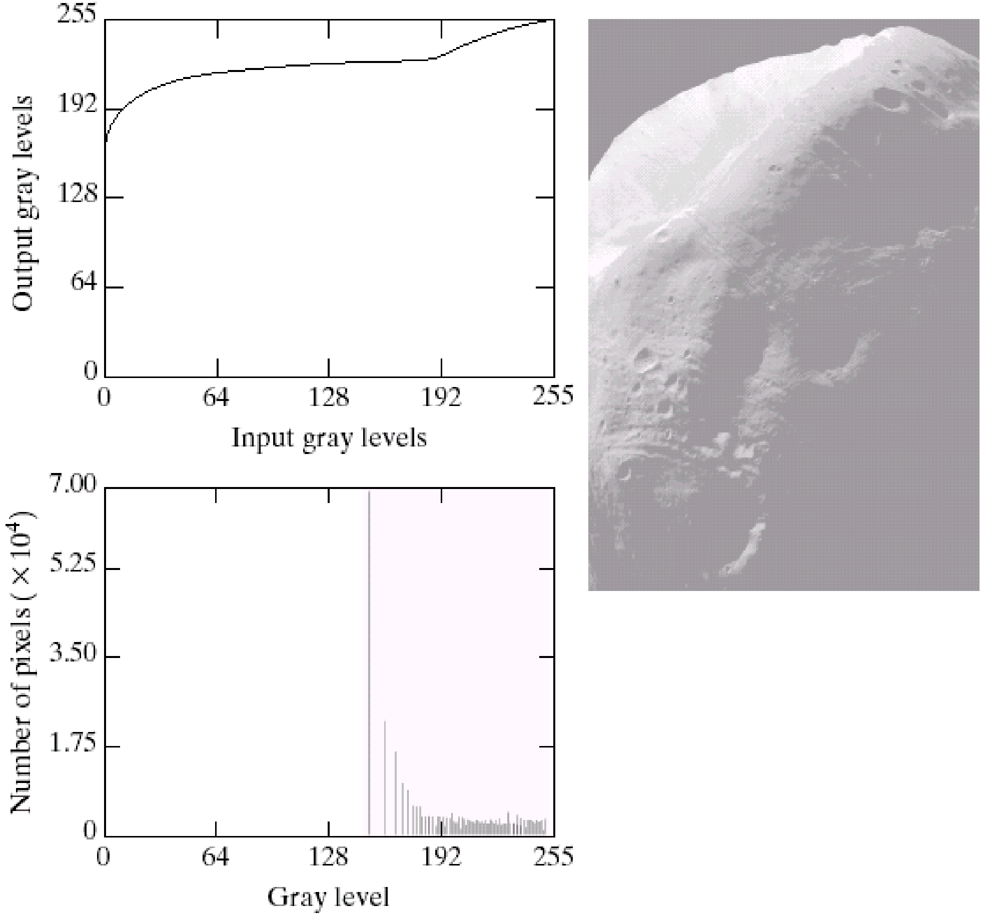


a b

FIGURE 3.20 (a) Image of the Mars moon Photos taken by NASA's *Mars Global Surveyor.* (b) Histogram. (Original image courtesy of NASA.)

3.3.2 Histogram Matching (Specification)

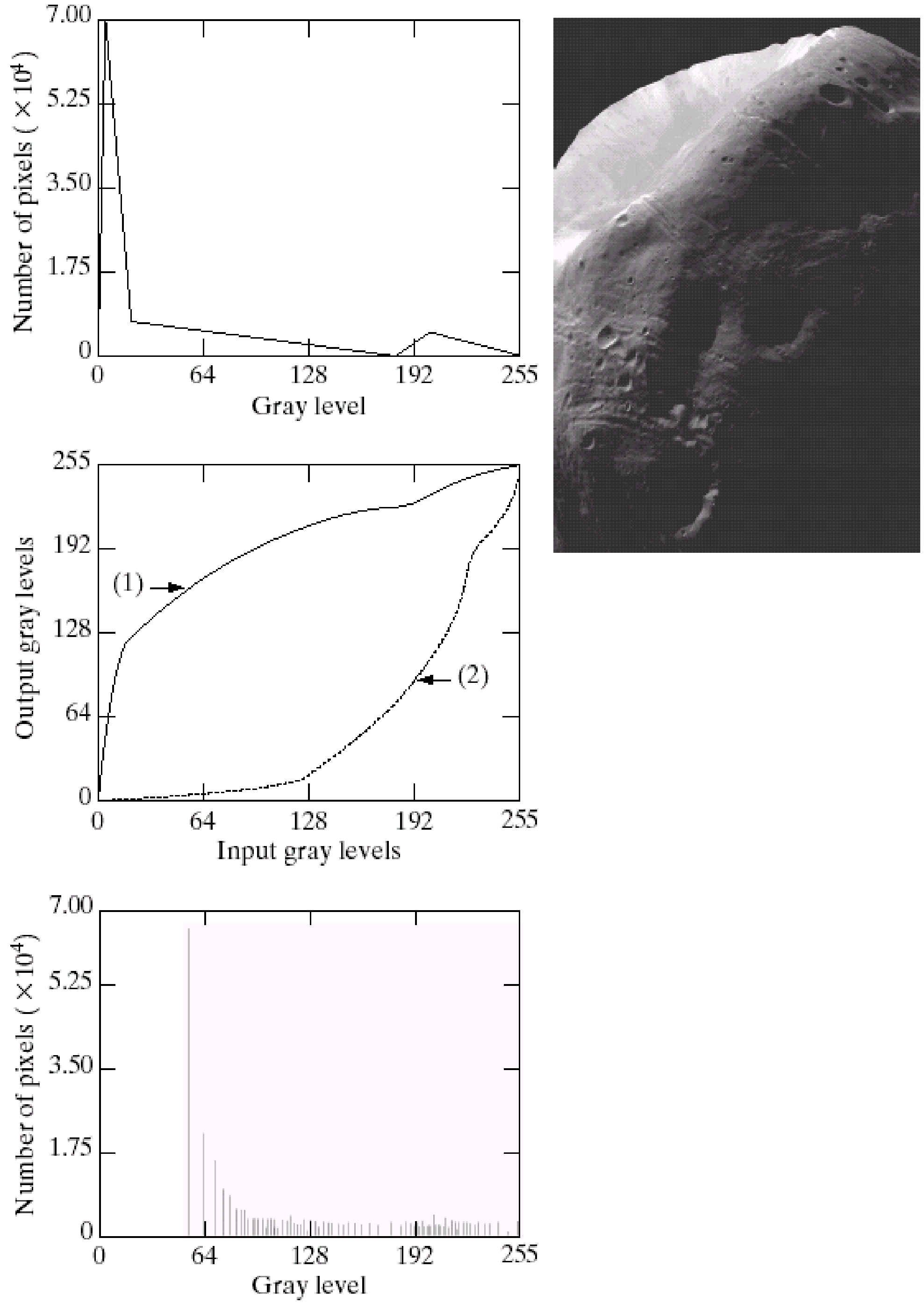
- Histogram equalized result- washed-out appearance



a b
c

FIGURE 3.21
(a) Transformation function for histogram equalization.
(b) Histogram-equalized image (note the washed-out appearance).
(c) Histogram of (b).

Histogram specification



a	c
b	
d	

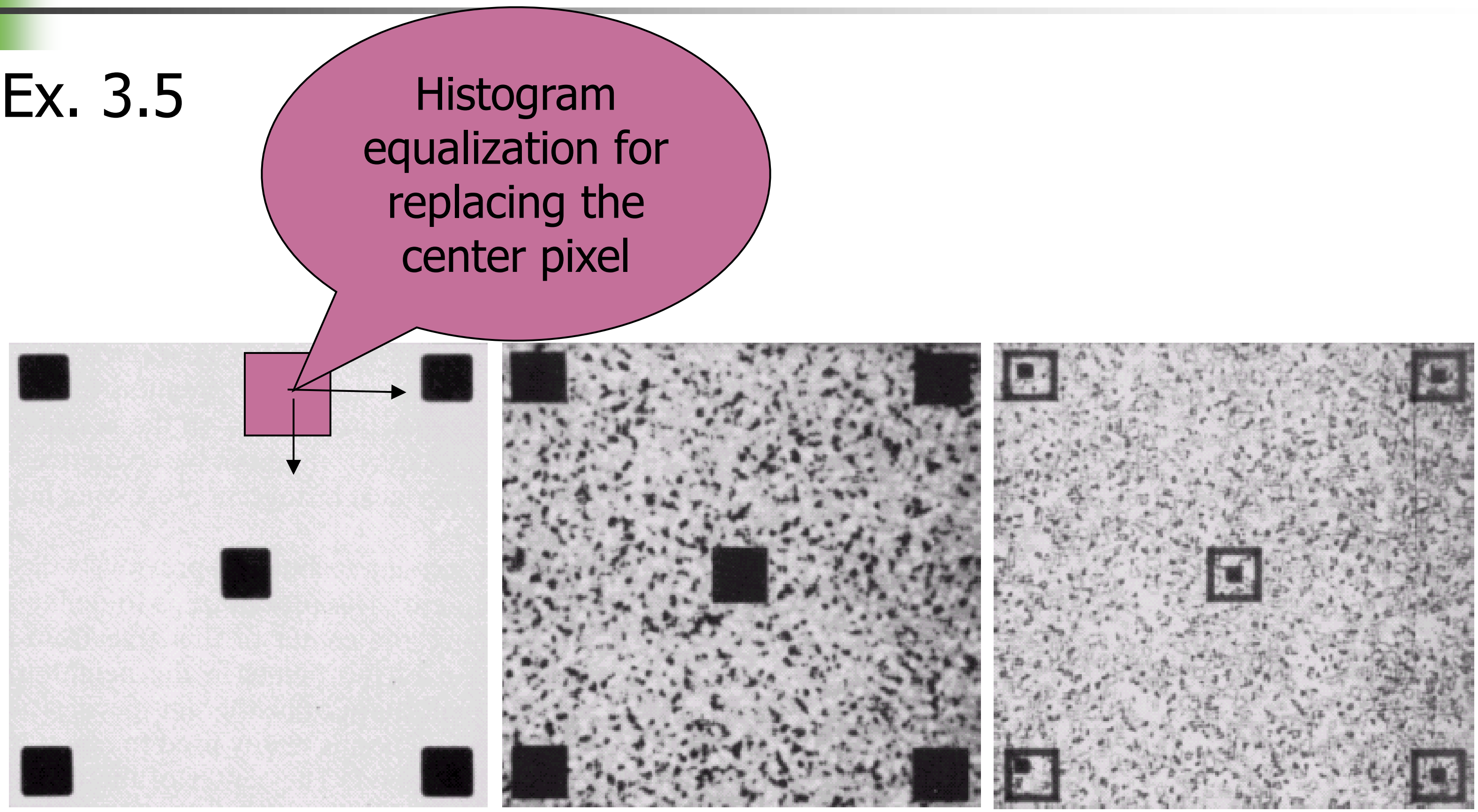
FIGURE 3.22
(a) Specified histogram.
(b) Curve (1) is from Eq. (3.3-14), using the histogram in (a); curve (2) was obtained using the iterative procedure in Eq. (3.3-17).
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).

3.3.3 Local Enhancement

- Global histogram processing- pixels are modified by a transformation function based on the gray-level content of an entire image
- Local enhancement- the transformation functions are based on the gray-level distribution (or other properties) in the neighborhood (defined as a square or rectangular) of every pixel in the image → **Mask operation**

3.3.3 Local Enhancement

- Ex. 3.5



a b c

FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7×7 neighborhood about each pixel.

3.3.4 Use of Histogram statistics for Image Enhancement

- n-th moments: $\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n p(r_i)$
  - Global mean (3.3-19): $m = \sum_{i=0}^{L-1} r_i p(r_i)$
  - Global variance (3.3.20): $\mu_2(r) = \sum_{i=0}^{L-1} (r_i - m)^2 p(r_i)$
  - Local mean (3.3-21): $m_{S_{xy}} = \sum_{(s,t)\in S_{xy}} r_{s,t} p(r_{s,t})$
  - Local variance (3.3.22): $\sigma_{S_{xy}}^2 = \sum_{(s,t)\in S_{xy}} [r_{s,t} - m_{S_{xy}}]^2 p(r_{s,t})$
    - S_{xy}: a neighborhood (subimage) of specified size, centered at (x, y)

3.3.4 Use of Histogram statistics for Image Enhancement

- Enhancement Application

$$g(x,y)=\begin{cases} E\cdot f(x,y), \text{if } m_{s_{x,y}} \le k_0 M_G \text{ and } k_1 D_G \le \sigma_{s_{x,y}} \le k_2 D_G \\ f(x,y), otherwise \end{cases}$$

  - M_G: global mean of the input image
  - D_G: global standard deviation
  - E, k_0, k_1, k_2: parameters

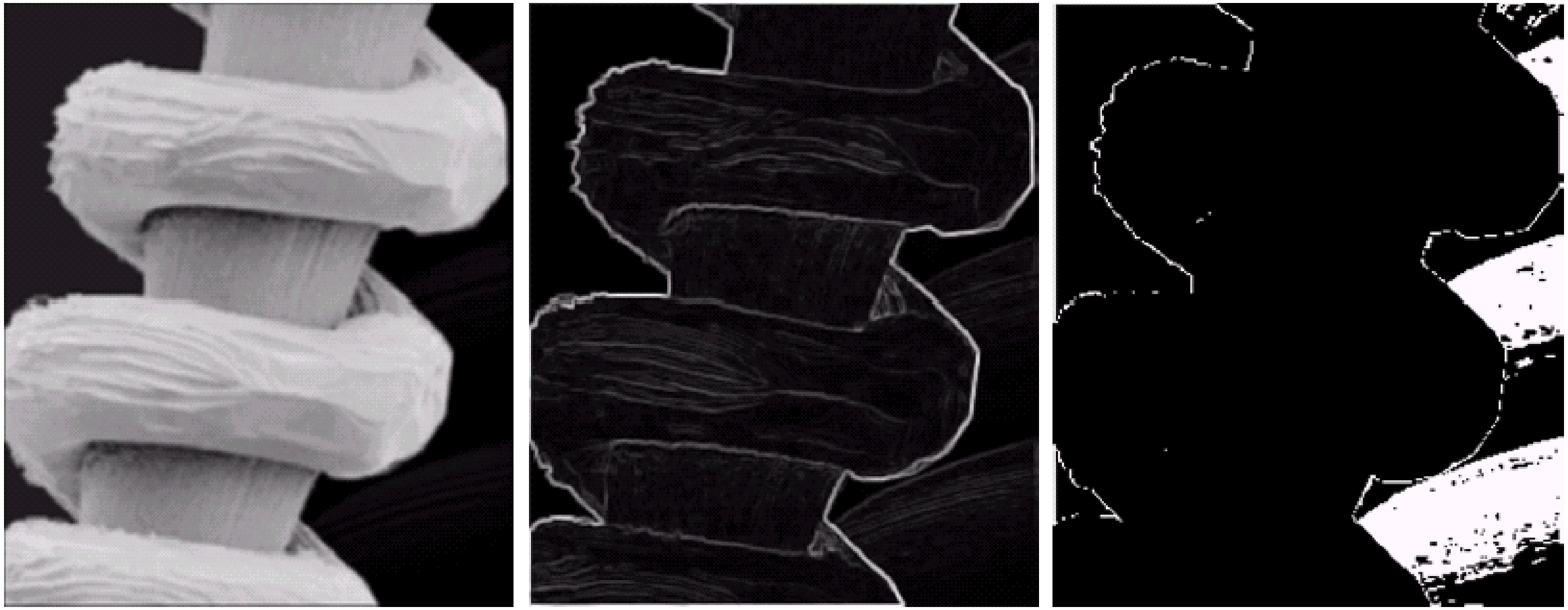
3.3.4 Use of Histogram statistics for Image Enhancement

FIGURE 3.24 SEM image of a tungsten filament and support, magnified approximately 130×. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene).

3.3.4 Use of Histogram statistics for Image Enhancement

$$g(x,y) / f(x,y) = \begin{cases} E \\ 1 \end{cases}$$

$m_{S_{xy}}$ $\mu_{S_{xy}}$



a b c

FIGURE 3.25 (a) Image formed from all local means obtained from Fig. 3.24 using Eq. (3.3-21). (b) Image formed from all local standard deviations obtained from Fig. 3.24 using Eq. (3.3-22). (c) Image formed from all multiplication constants used to produce the enhanced image shown in Fig. 3.26.

3.3.4 Use of Histogram statistics for Image Enhancement

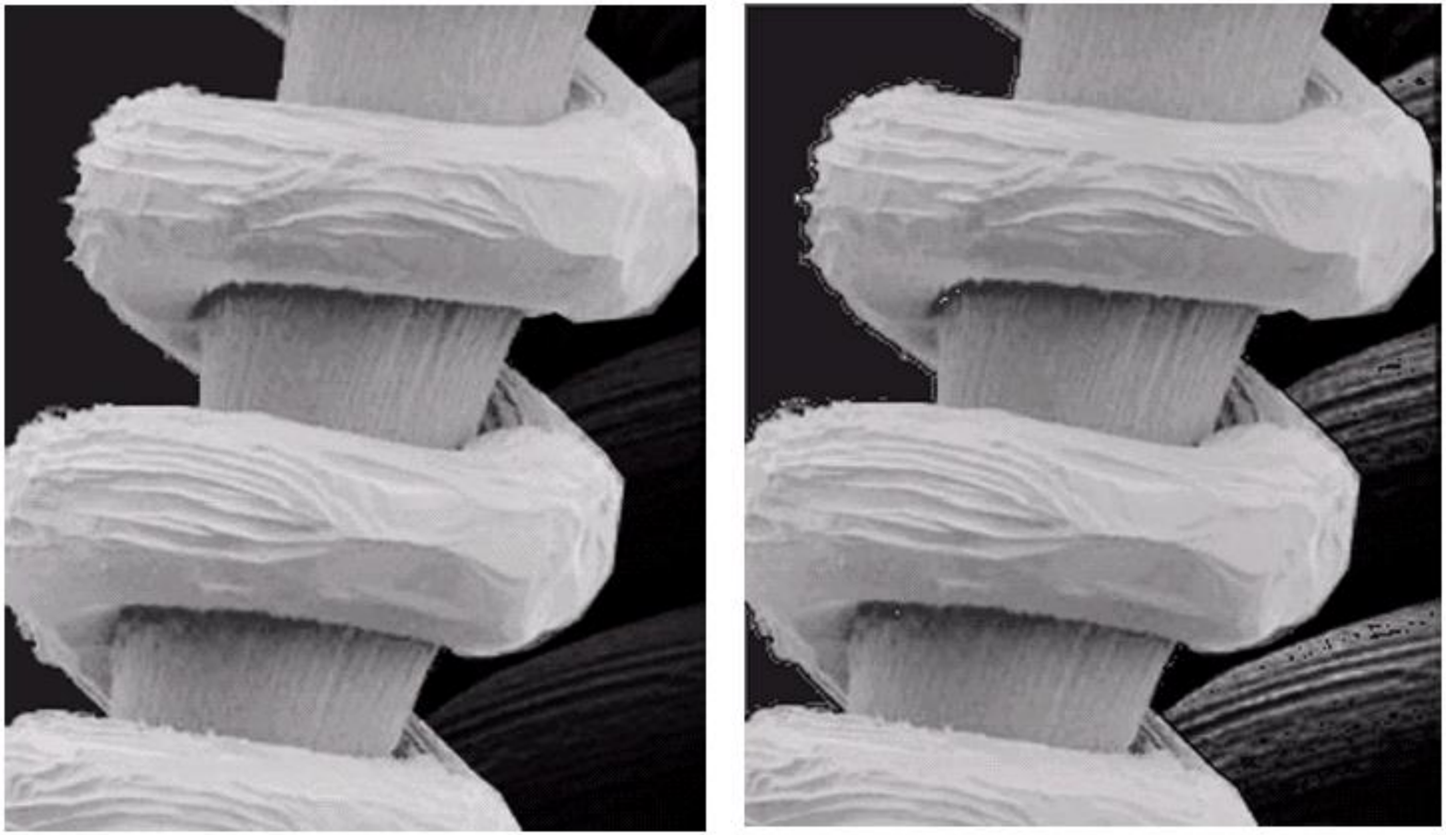
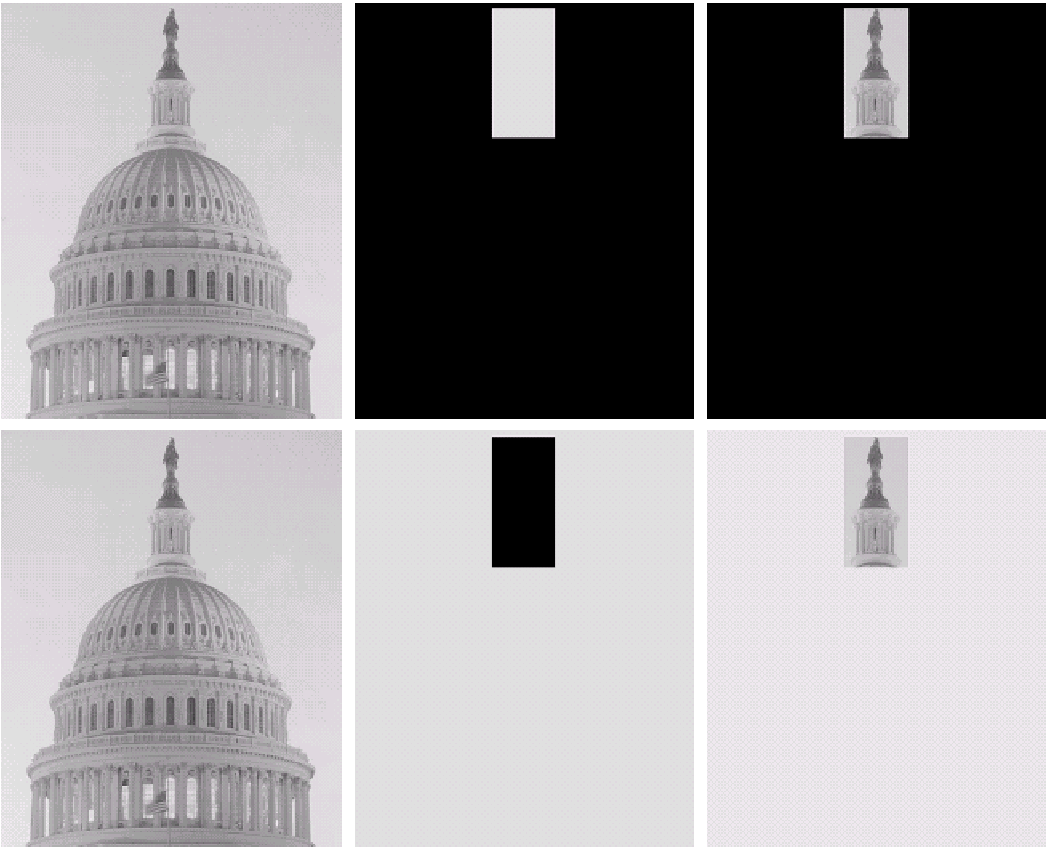


FIGURE 3.26 Enhanced SEM image. Compare with Fig. 3.24. Note in particular the enhanced area on the right side of the image.

3.4 Enhancement using arithmetic/logic operations

- AND, OR, NOT operations
- Masking: extract the region of interest (ROI)

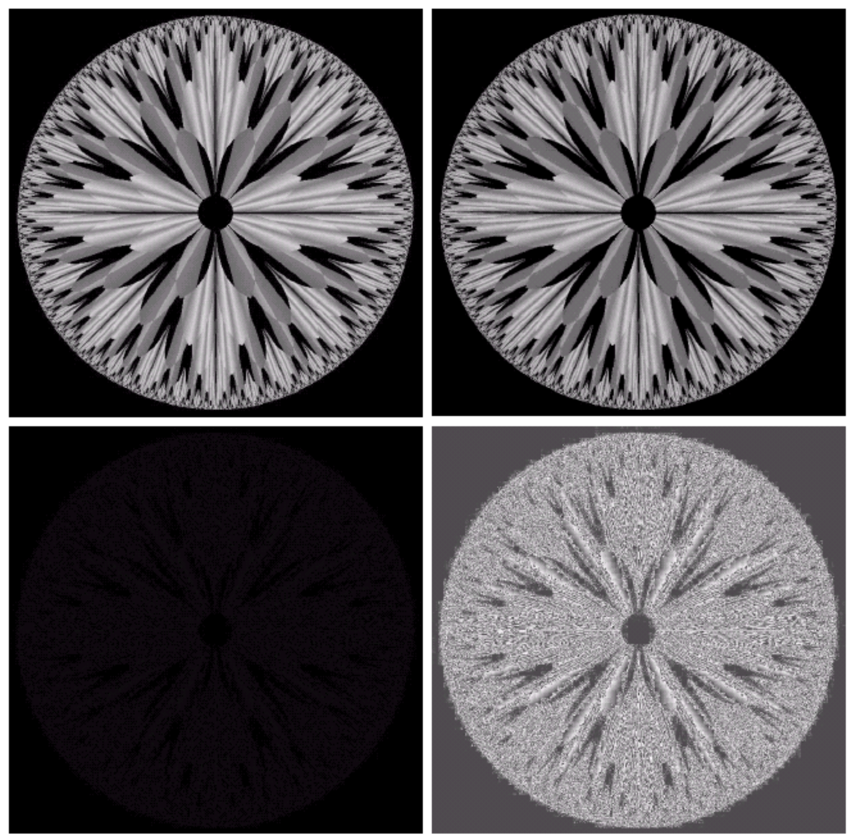


a	b	c
d	e	f

FIGURE 3.27 (a) Original image. (b) AND image mask. (c) Result of the AND operation on images (a) and (b). (d) Original image. (e) OR image mask. (f) Result of operation OR on images (d) and (e).

3.4.1 Image subtraction

- $g(x,y) = f(x,y) - h(x,y)$
- Observe the differences between two images.

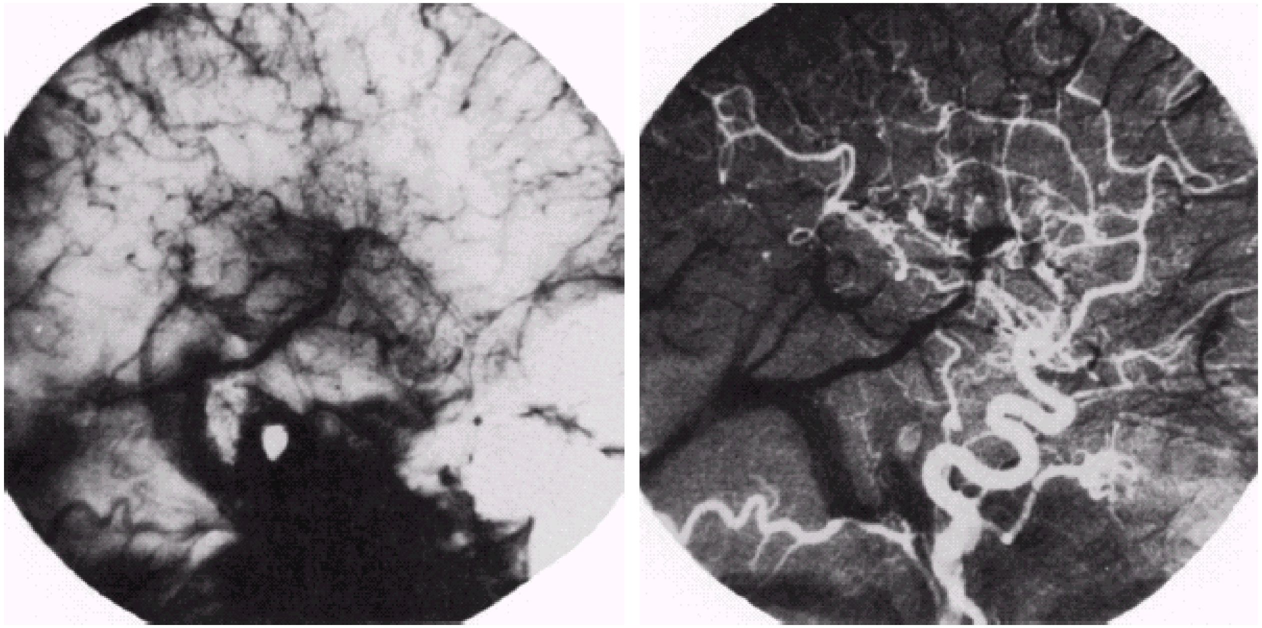


a b
c d

FIGURE 3.28 (a) Original fractal image. (b) Result of setting the four lower-order bit planes to zero. (c) Difference between (a) and (b). (d) Histogram-equalized difference image. (Original image courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA).

3.4.1 Image subtraction

- Ex. 3.7: Mask mode radiography (X光)



a b

FIGURE 3.29 Enhancement by image subtraction. (a) Mask image. (b) An image (taken after injection of a contrast medium into the bloodstream) with mask subtracted out.

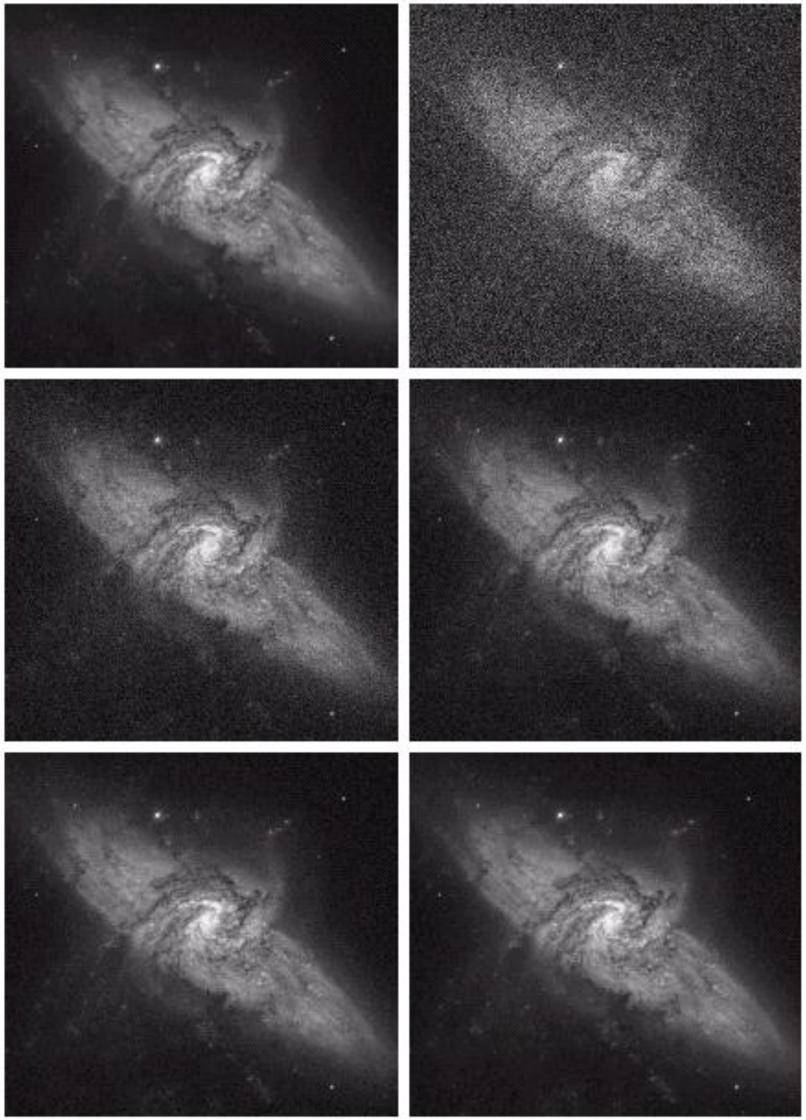
3.4.2 Image Averaging

- Noise reduction by image averaging
  - Given a noisy image $g(x, y) = f(x, y) + \eta(x, y)$,
    - $f(x, y)$ is the original image
    - $\eta(x, y)$ is a zero mean noise
  - Averaging K different noisy images:

$$\overline{g}(x, y) = \sum_{i=1}^{K} g_i(x, y)$$

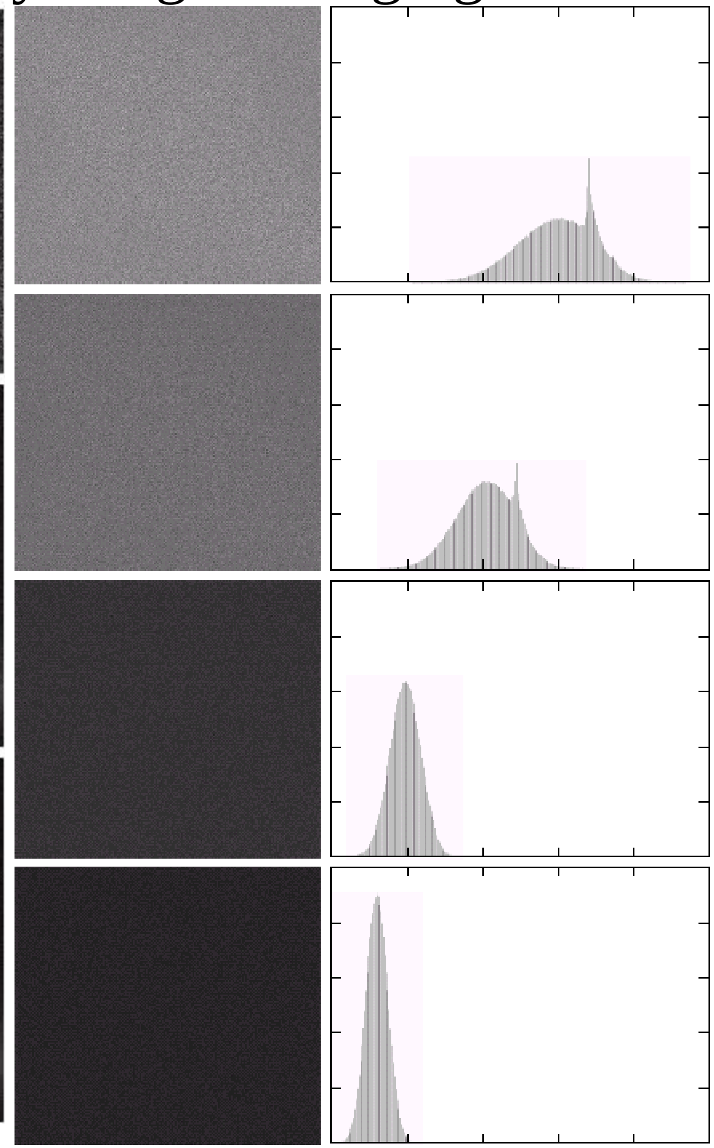
then $$E\left\{\overline{g}(x, y)\right\} = f(x, y) \text{ and } \sigma^2_{\overline{g}(x, y)} = \frac{1}{k}\sigma^2_{\eta(x, y)}$$

Ex. 3.8 Noise reduction by image averaging



a b
c d
e f

FIGURE 3.30 (a) Image of Galaxy Pair NGC 3314. (b) Image corrupted by additive Gaussian noise with zero mean and a standard deviation of 64 gray levels. (c)–(f) Results of averaging $K = 8, 16, 64$, and 128 noisy images. (Original image courtesy of NASA.)



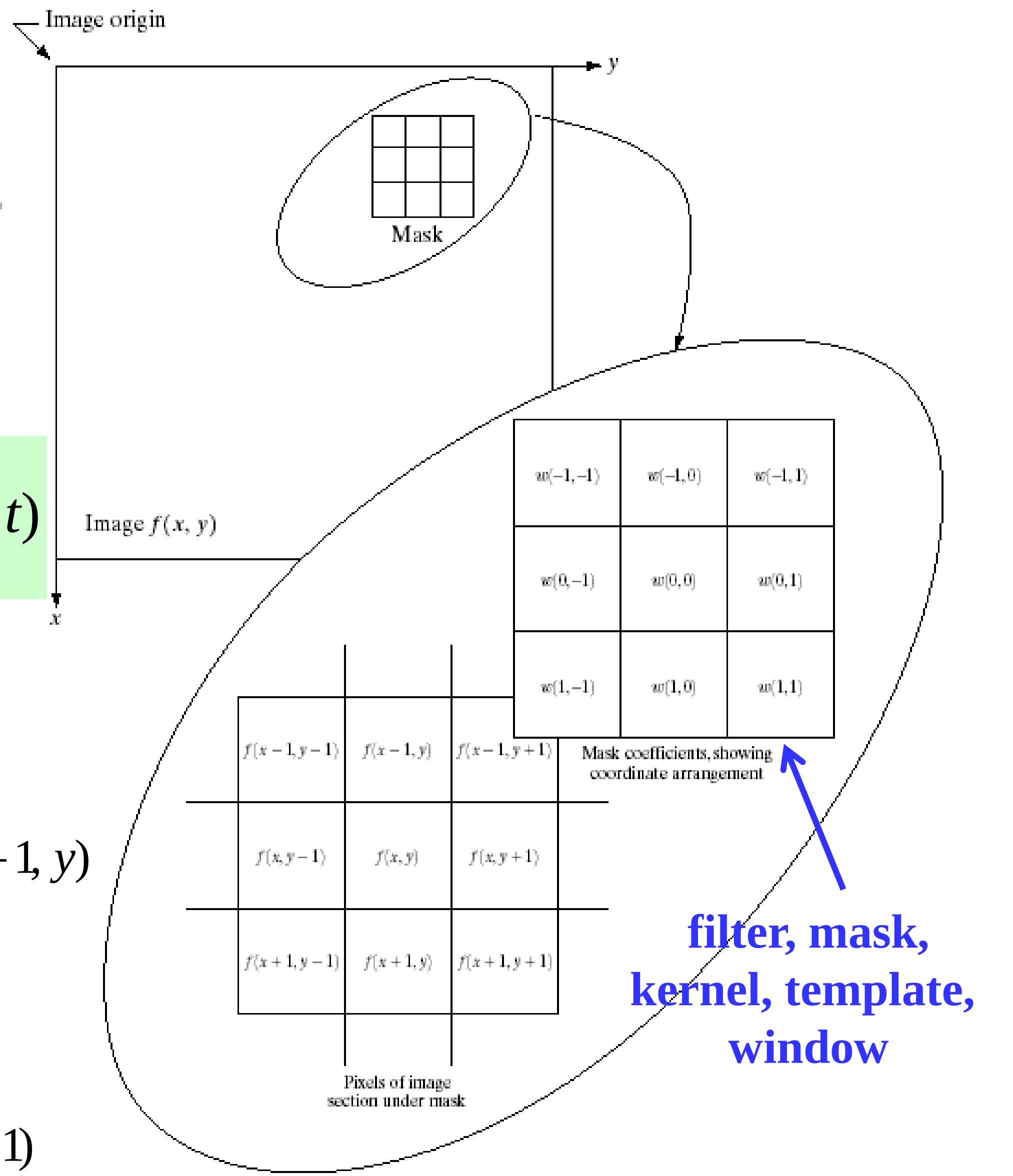
a b

FIGURE 3.31 (a) From top to bottom: Difference images between Fig. 3.30(a) and the four images in Figs. 3.30(c) through (f), respectively. (b) Corresponding histograms.

3.5 Basics of Spatial Filtering

- Filtering operation (mask operation)

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



$$
\begin{aligned}
R &= w(-1,-1)\,f(x-1,y-1)+w(-1,0)\,f(x-1,y)\\
&\quad+\cdots\\
&\quad+w(0,0)\,f(x,y)\\
&\quad+\cdots\\
&\quad+w(1,0)\,f(x+1,y)+w(1,1)\,f(x+1,y+1)
\end{aligned}
$$

3.6 Smoothing Spatial Filters
3.6.1 Smoothing Linear Filters

- Lowpass filter shown in Fig. 3.34 is a weighted average (averaging filter)

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

a b

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

1	1	1
1	1	1
1	1	1

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

1	2	1
2	4	2
1	2	1

FIGURE 3.34 Two 3×3 smoothing (averaging) filter masks. The constant multipli er in front of each mask is equal to the sum of the values of its coefficients, as is required to compute an average.

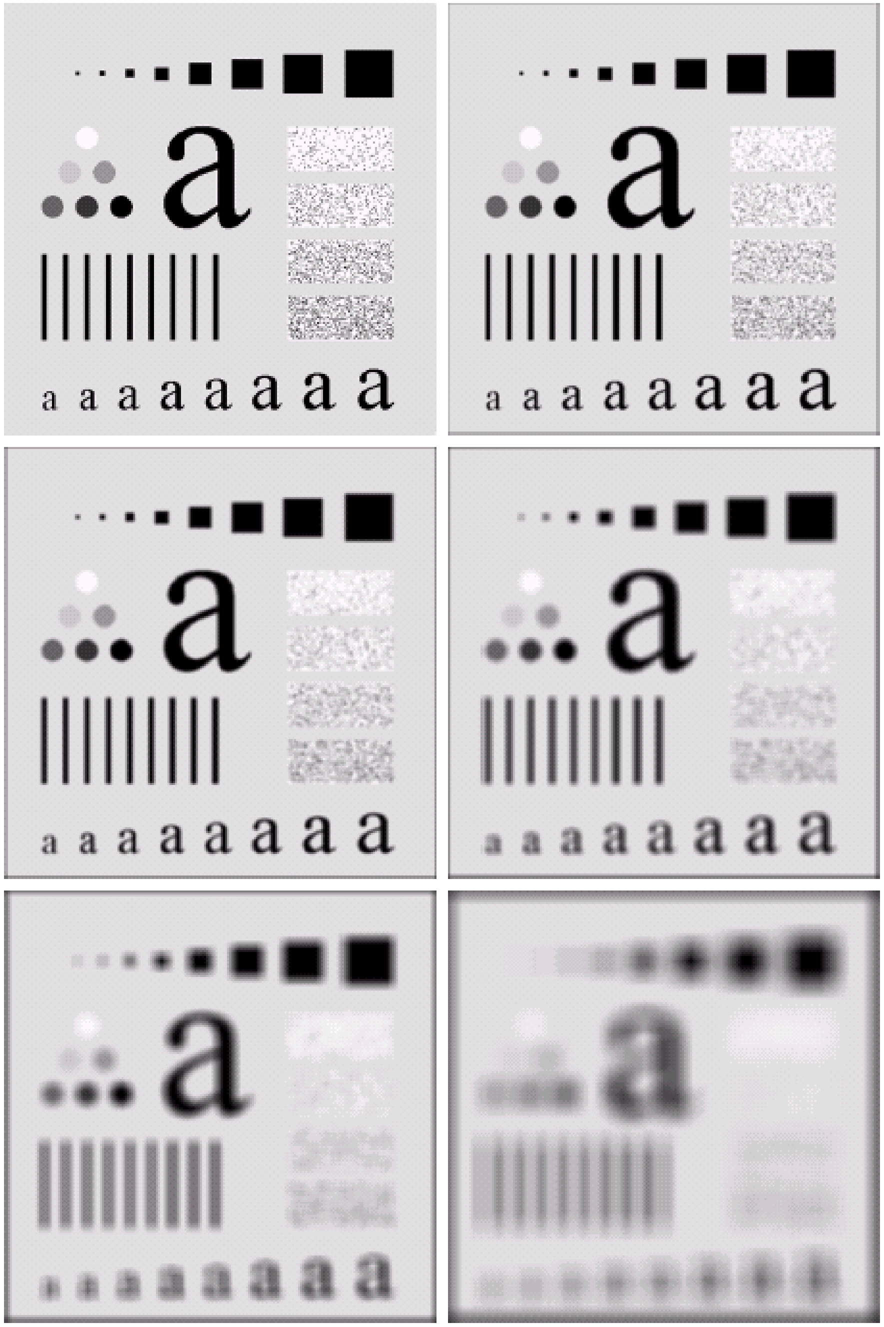
3.6.1 Smoothing Linear Filters

- General form:

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

3.6.1 Smoothing Li

- Ex. 3.9 Image smoothing with masks of various sizes

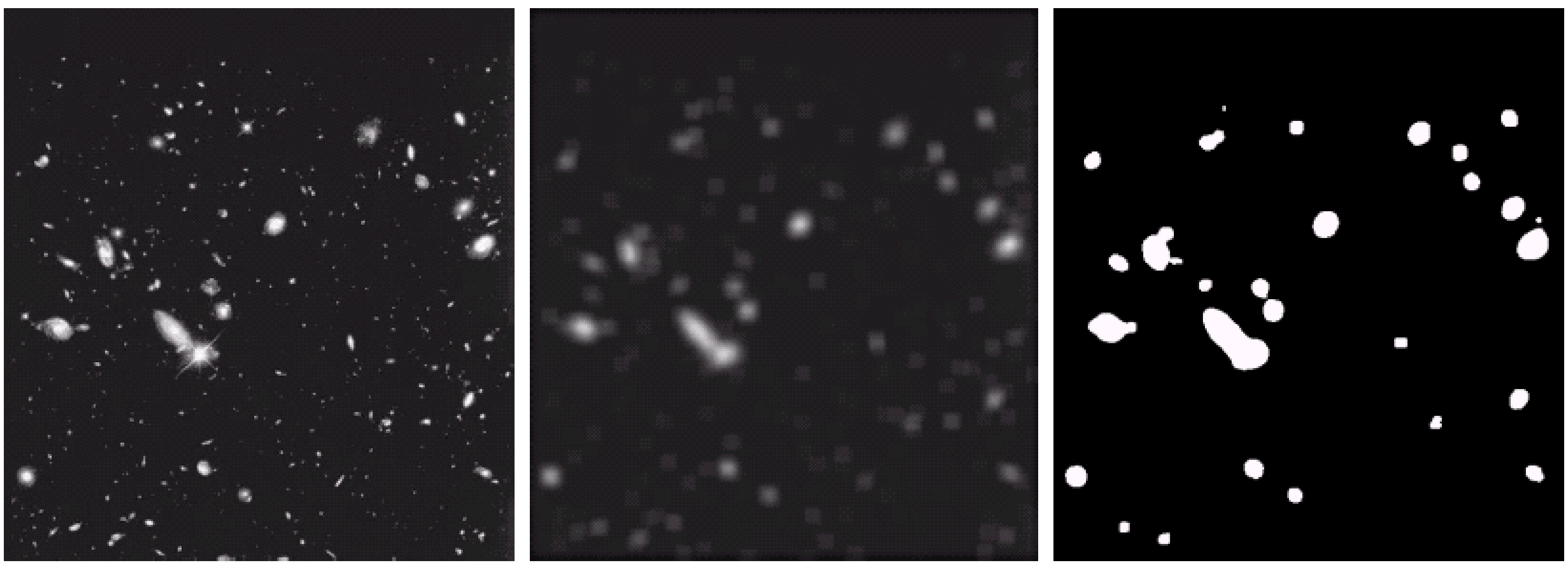


a b
c d
e f

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.

3.6.1 Smoothing Linear Filters

- Ex. Image from Hubble space telescope



a b c

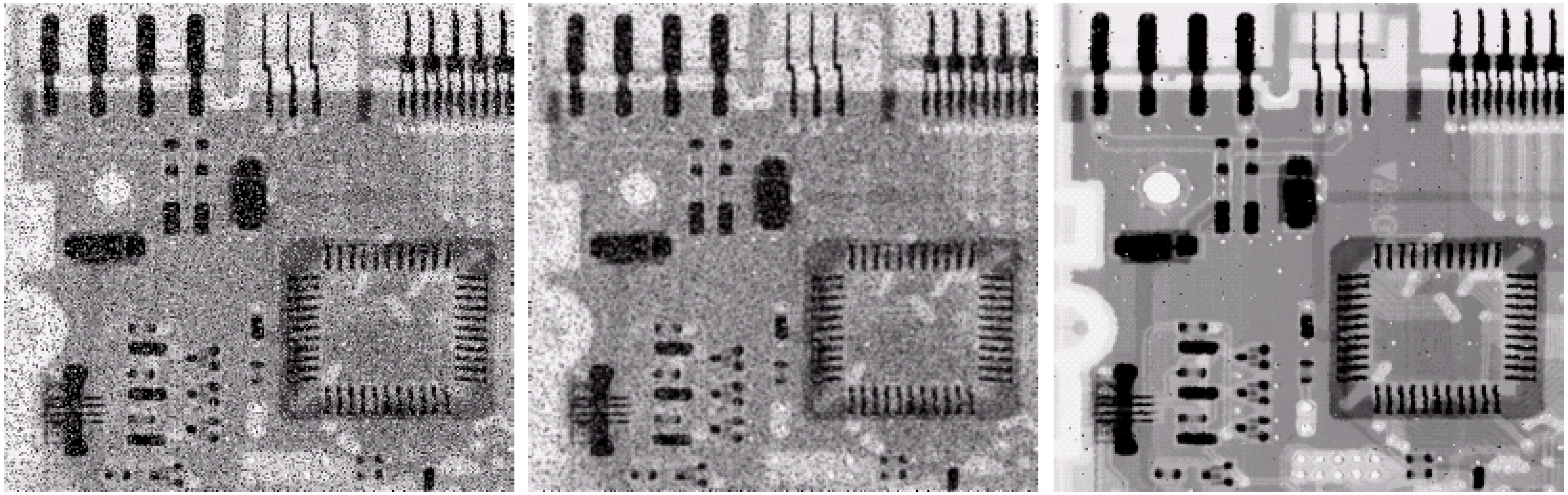
FIGURE 3.36 (a) Image from the Hubble Space Telescope. (b) Image processed by a 15 × 15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

3.6.2 Order-Statistics Filters

- Order-statistics filter are nonlinear filters whose response is based on ordering the pixels contained in the image area encompassed by the filter, and then replacing the value of the center pixel with the value determined by the ranking result.
- Median filters: provide excellent noise-reduction capabilities (particularly for impulse noise or salt-and-pepper noise), with considerably less blurring than linear filters.
- Max filter
- Min filter

3.6.2 Order-Statistics Filters

- Ex. 3.10 Removing of salt-and-pepper noises.



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.)

3.7 Sharpening Spatial Filters
3.7.1 Foundation

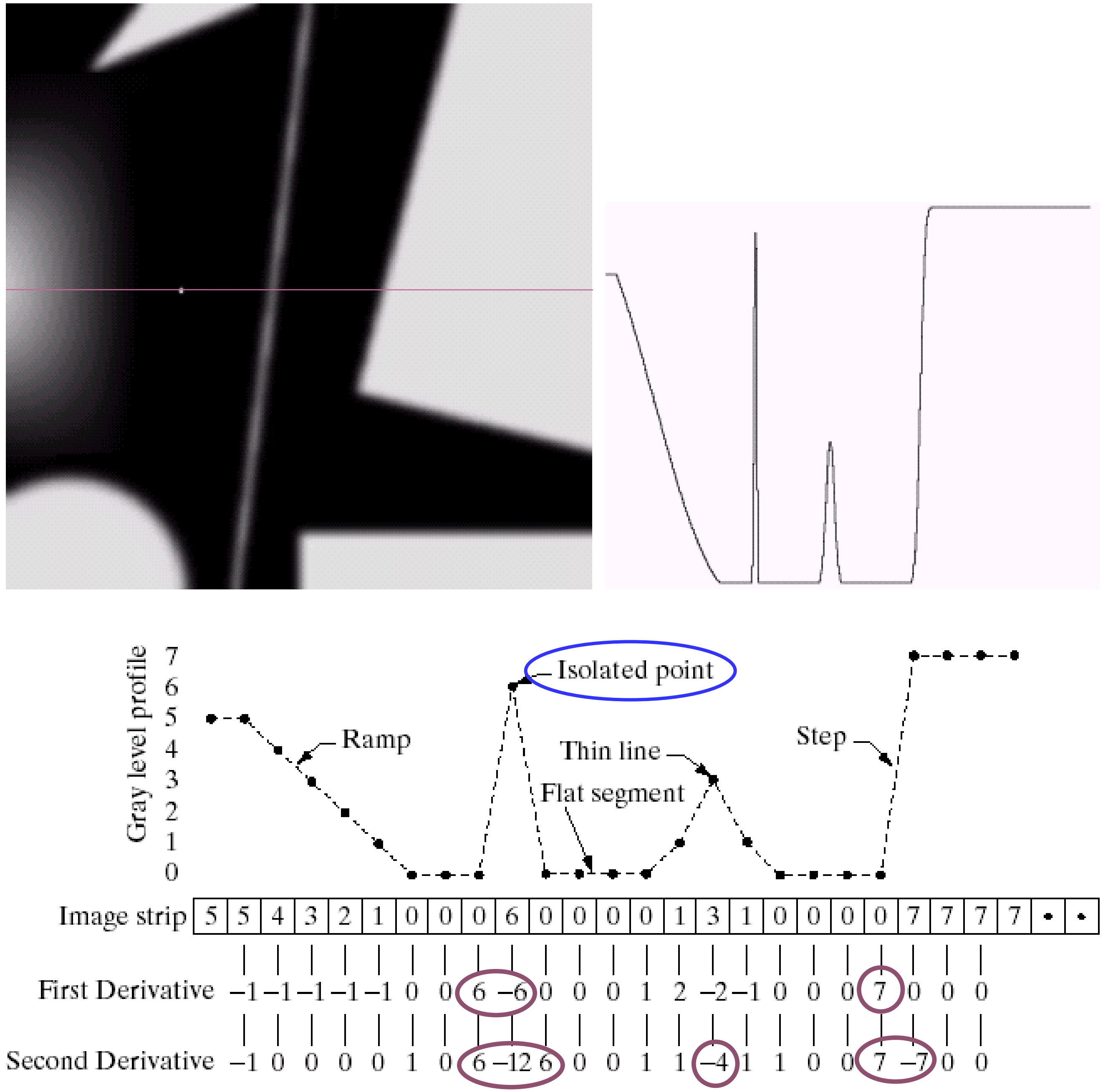
- Sharpening – highlight fine details in an image or to enhance detail that has been blurred
- Apply the 1st and 2nd derivatives to sharpen the images

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

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

a b
c

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).



3.7.1 Foundation

- Conclusion:
  - $f^{(1)}(x,y)$ is nonzero along the entire ramp, while $f^{(2)}(x,y)$ is nonzero only at the onset and end of the ramp
  - $f^{(1)}(x,y)$ generates thicker edge and has stronger response to a step edge.
  - $f^{(2)}(x,y)$ generates thin edge or isolated points and has double response to a step edge.
  - In most application, the second derivative is better suited than the first derivative for image enhancement.

3.7.2 Use of second derivative for enhancement-The Laplacian

- Laplacian operator

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

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

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

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

3.7.2 Use of second derivative for enhancement-The Laplacian

(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

adding diagonal directions

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.

3.7.2 Use of second derivative for enhancement-The Laplacian

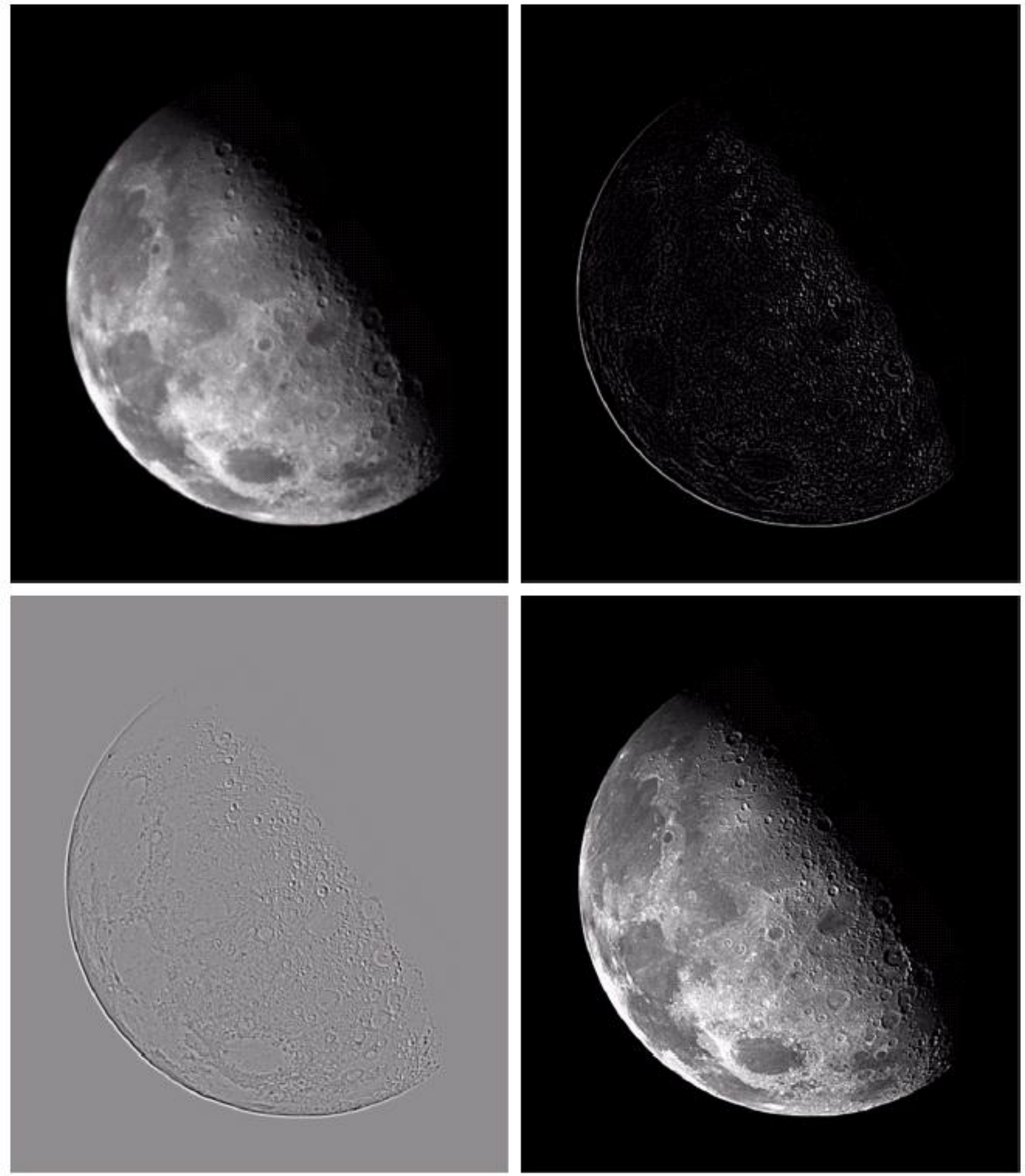
- Basic Laplacian image enhancement method- adding/subtracting the original and Laplacian images

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

- Simplification

  $g(x,y)=5f(x,y) - [f(x+1,y)+f(x-1,y)+f(x,y+1)+f(x,y-1)]$

- Ex. 3.11



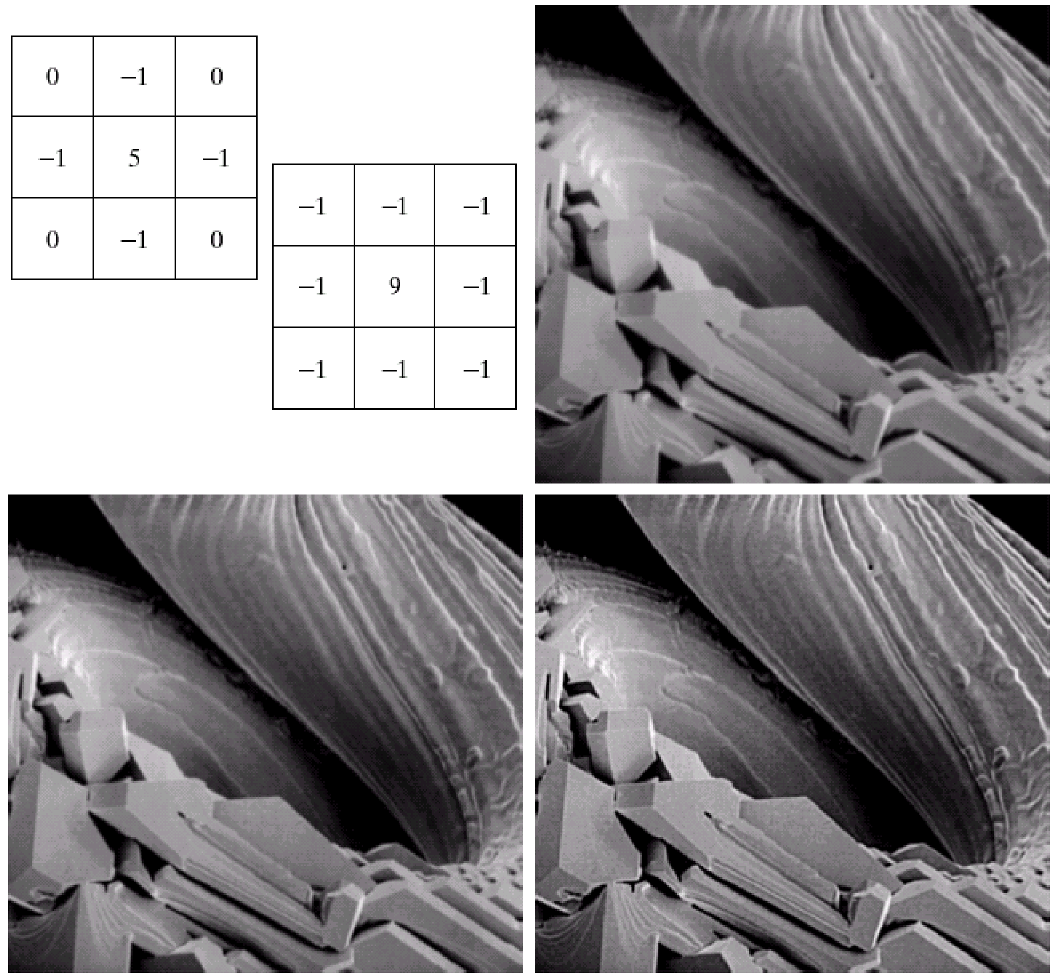
a b
c d

FIGURE 3.40
(a) Image of the North Pole of the moon.
(b) Laplacian-filtered image.
(c) Laplacian image scaled for display purposes.
(d) Image enhanced by using Eq. (3.7-5).
(Original image courtesy of NASA.)

- Ex. 3.12

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 and High-boost filtering

- Unsharp masking - subtracting a blurred version of an image from the image itself

$$f_s(x, y) = f(x, y) - \overline{f}(x, y)$$

  - $\overline{f}(x, y)$: a blurred version of f(x,y)
  - $f_s(x, y)$: sharpen image obtained by unsharp masking
- High-boost filtering

$$\begin{aligned} f_{hb}(x, y) &= Af(x, y) - \overline{f}(x, y) \qquad A \geq 1 \\ &= (A-1)f(x, y) + f(x, y) - \overline{f}(x, y) \\ &= (A-1)f(x, y) + f_s(x, y) \end{aligned}$$

Unsharp masking and High-boost filtering

- If the sharpened image $f_s(x,y)$ is replaced with the Laplacian:

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

Unsharp masking and High-boost filtering

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

Laplacian filtering

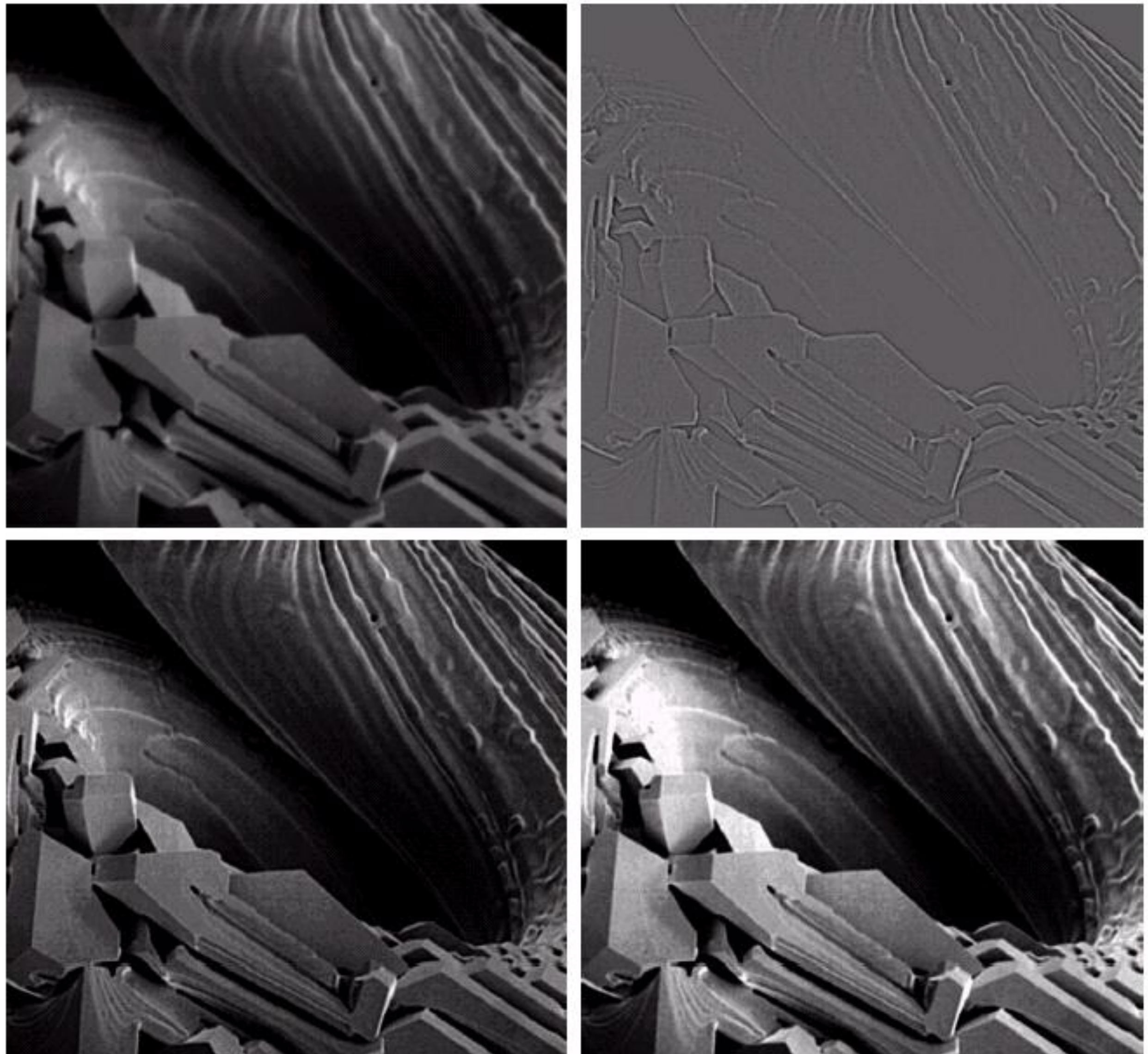
0	−1	0
−1	$A + 4$	−1
0	−1	0

−1	−1	−1
−1	$A + 8$	−1
−1	−1	−1

a b

FIGURE 3.42 The high-boost filtering technique can be implemented with either one of these masks, with $A \geq 1$.

Unsharp masking and High-boost filtering



a	b
c	d

FIGURE 3.43 (a) Same as Fig. 3.41(c), but darker. (a) Laplacian of (a) computed with the mask in Fig. 3.42(b) using $A = 0$. (c) Laplacian enhanced image using the mask in Fig. 3.42(b) with $A = 1$. (d) Same as (c), but using $A = 1.7$.

3.7.3 Use of First Derivatives for Enhancement – The Gradient

- First derivatives in image processing are implemented using the <u>magnitude</u> of the <u>gradient vector</u>:

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

  - The magnitude of the gradient vector is given by

$$\nabla f = mag(\nabla \mathbf{f}) = [G_x^2 + G_y^2]^{1/2} = \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{1/2}$$

  - Common practice to approximate the magnitude of the gradient is using <u>absolute values</u>:

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

3.7.3 Use of First Derivatives for Enhancement – The Gradient

- Computing the gradients:
  - Roberts cross-gradient operators

$$|G_x| = z_9 - z_5$$

$$|G_y| = z_8 - z_6$$

$$\nabla f = \left[(z_9 - z_5)^2 + (z_8 - z_6)^2\right]^{1/2}$$

$$\text{or } \nabla f \approx |z_9 - z_5| + |z_8 - z_6|$$

  - Sobel operators:

$$|G_x| = (z_7 + 2z_8 + z_9) - (z + 2z_1 + z_3)$$

$$|G_y| = (z_3 + 2z_6 + z_9) - (z + 2z_4 + z_7)$$

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 to zero, as expected of a derivative operator.

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

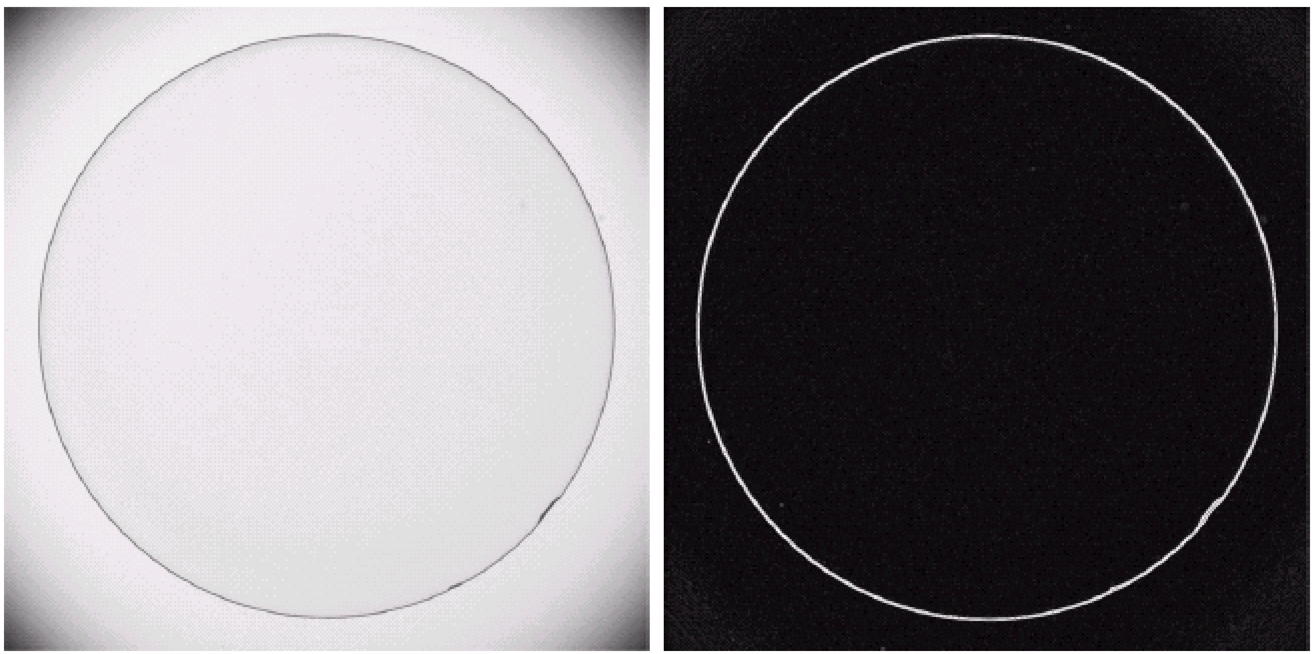
0	−1
1	0

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

3.7.3 Use of First Derivatives for Enhancement – The Gradient

- Ex. 3.14 Edge enhancement



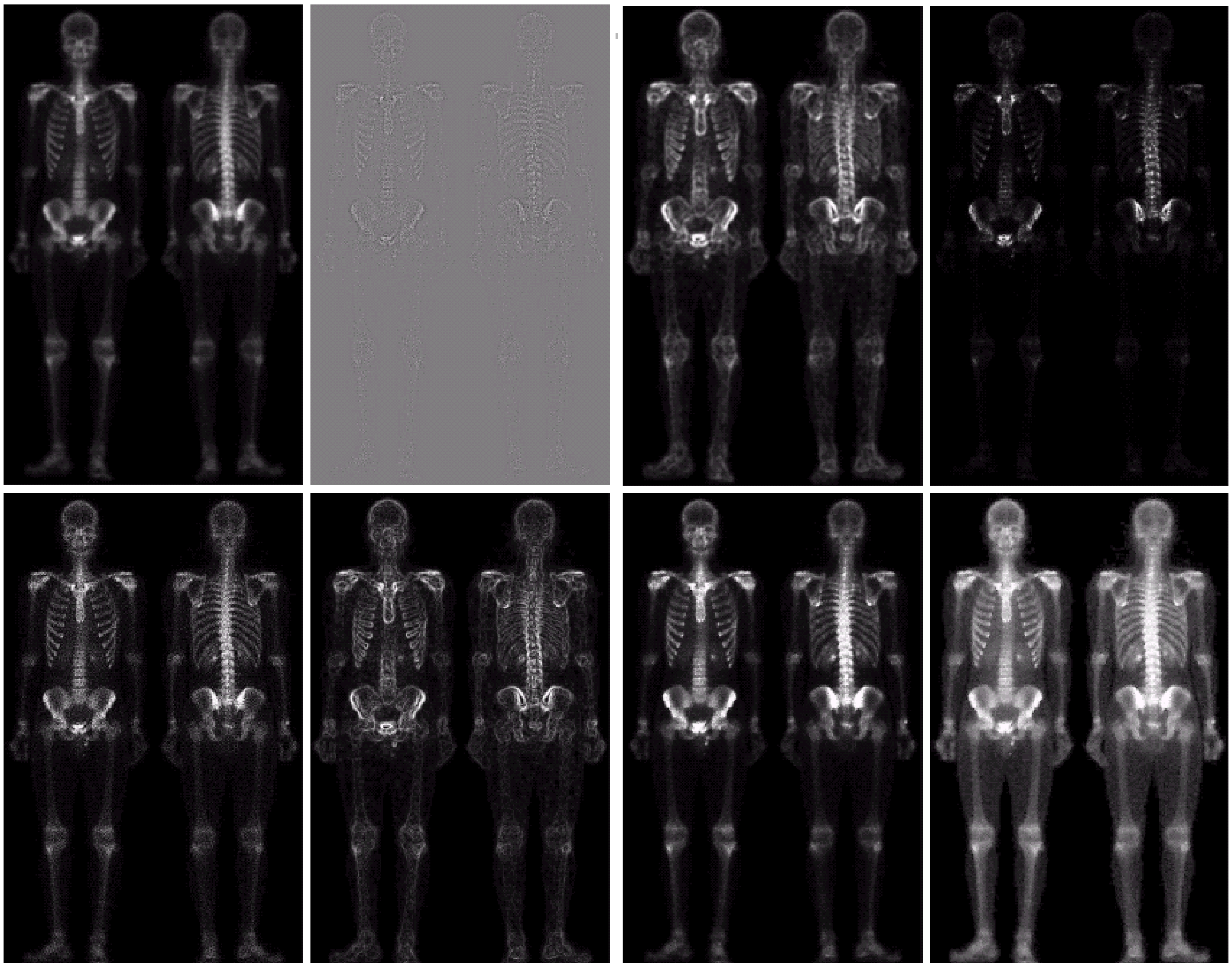
a b

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.)

3.8 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.)