EE 5098 – Digital Image Processing

3. Intensity Transformation and Spatial Filtering

Source: Frank Liu

Outline

- Basic Intensity Transformation Functions
- Histogram Processing
- Fundamentals of Spatial Filtering
- Smoothing
- Sharpening
- Highpass, Bandstop, Bandpass Filters from Lowpass Filters
- Combining Spatial Enhancement Methods

Spatial Domain vs. Transform Domain

- Spatial domain
  - Directly manipulate pixels in an image
  - Examples
    - intensity transformation: contrast manipulation, image thresholding
    - spatial filtering: image smoothing, sharpening
- Transform domain
  - Transform an image to the transform domain, process the transform coefficients, and finally bring the results back to the spatial domain by an inverse transform
  - Examples
    - Spectral shaping (e.g. LP-, BP-, HP-filtering)
    - Image compression

Spatial-Domain Processing on a Single Image

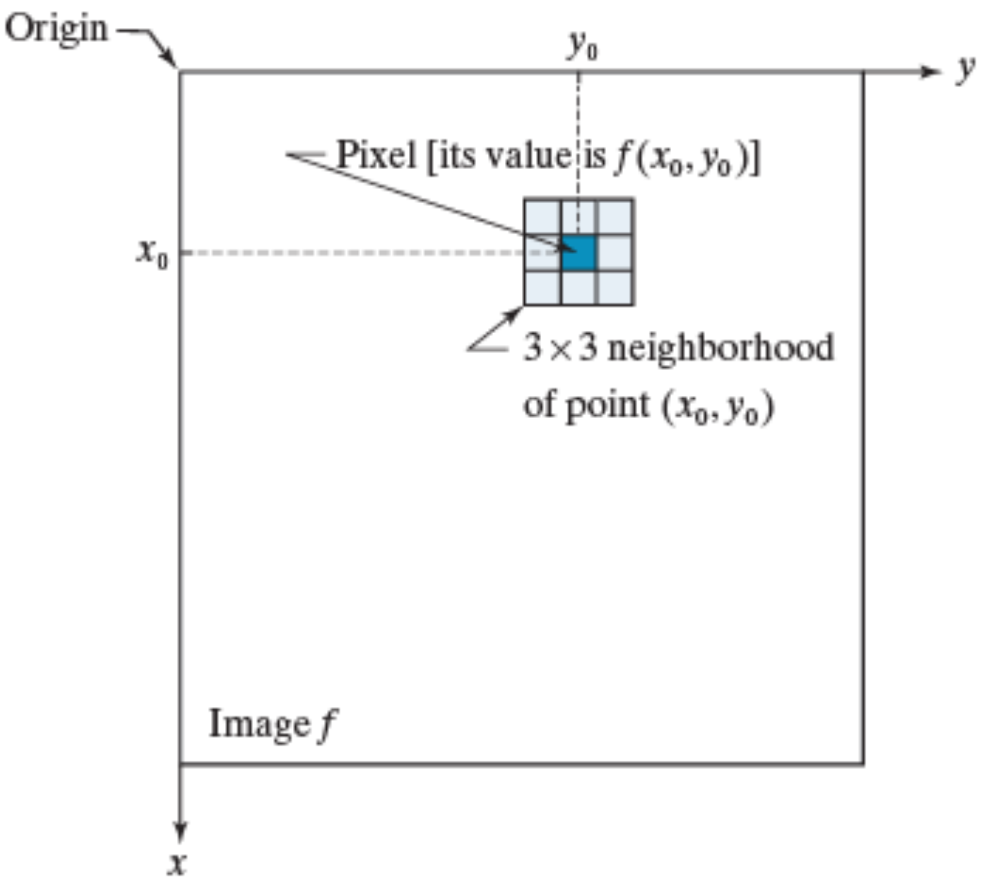
$$g(x,y) = T(f(x,y))$$

$f(x,y)$: input image

$g(x,y)$: output image

T : an operator defined over a neighborhood

FIGURE 3.1
A 3×3 neighborhood about a point (x_0, y_0) in an image. The neighborhood is moved from pixel to pixel in the image to generate an output image. Recall from Chapter 2 that the value of a pixel at location (x_0, y_0) is $f(x_0, y_0)$, the value of the image at that location.



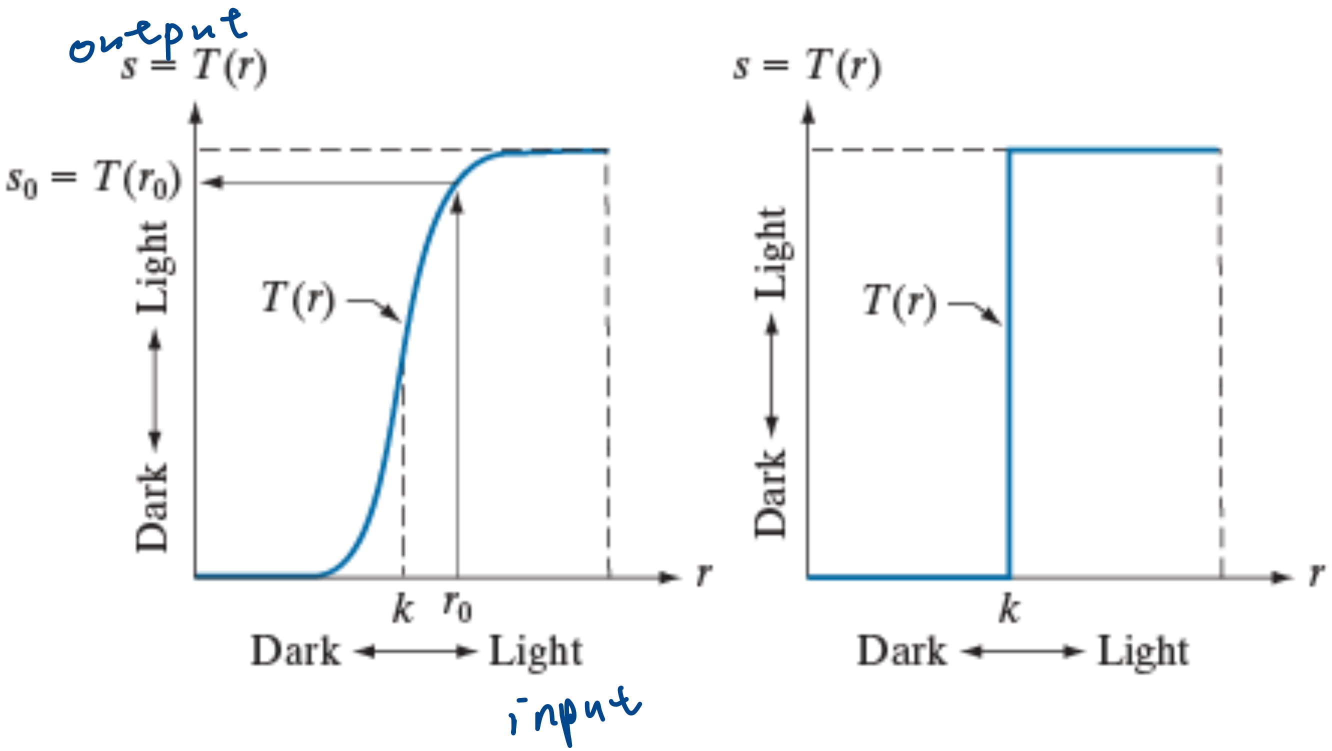
The image processing consists of moving the center of the neighborhood from pixel to pixel and applying the operator to pixels in the neighborhood to yield an output value at that location.

Example: smoothing on 3x3 neighborhood

Intensity transformation: An operator performed on 1x1 neighborhood

Spatial-Domain Processing on a Single Image: Intensity Transformation

$$s = T(r)$$



a b

FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

Some Basic Intensity Transformation Functions

FIGURE 3.3 Some basic intensity transformation functions. Each curve was scaled *independently* so that all curves would fit in the same graph. Our interest here is on the *shapes* of the curves, not on their relative values.

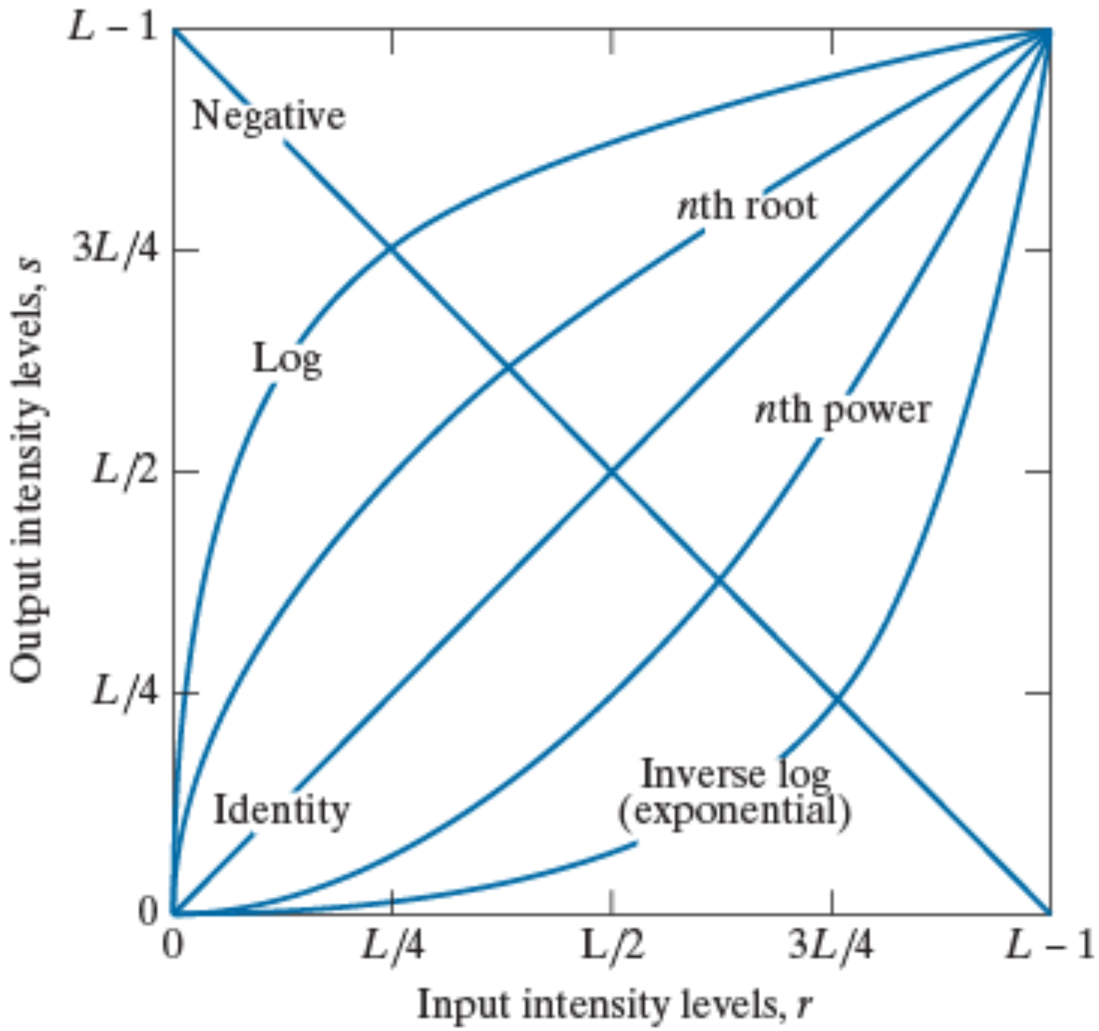
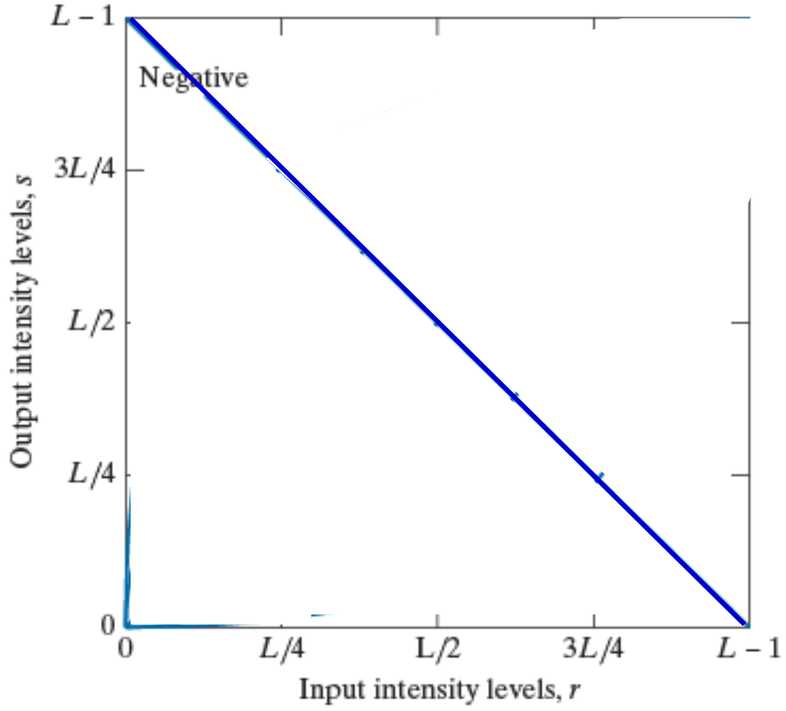


Image Negatives



$$s = L - 1 - r$$

Used to enhance the white or gray detail embedded in dark regions of an image, especially when the black areas are dominant in size.

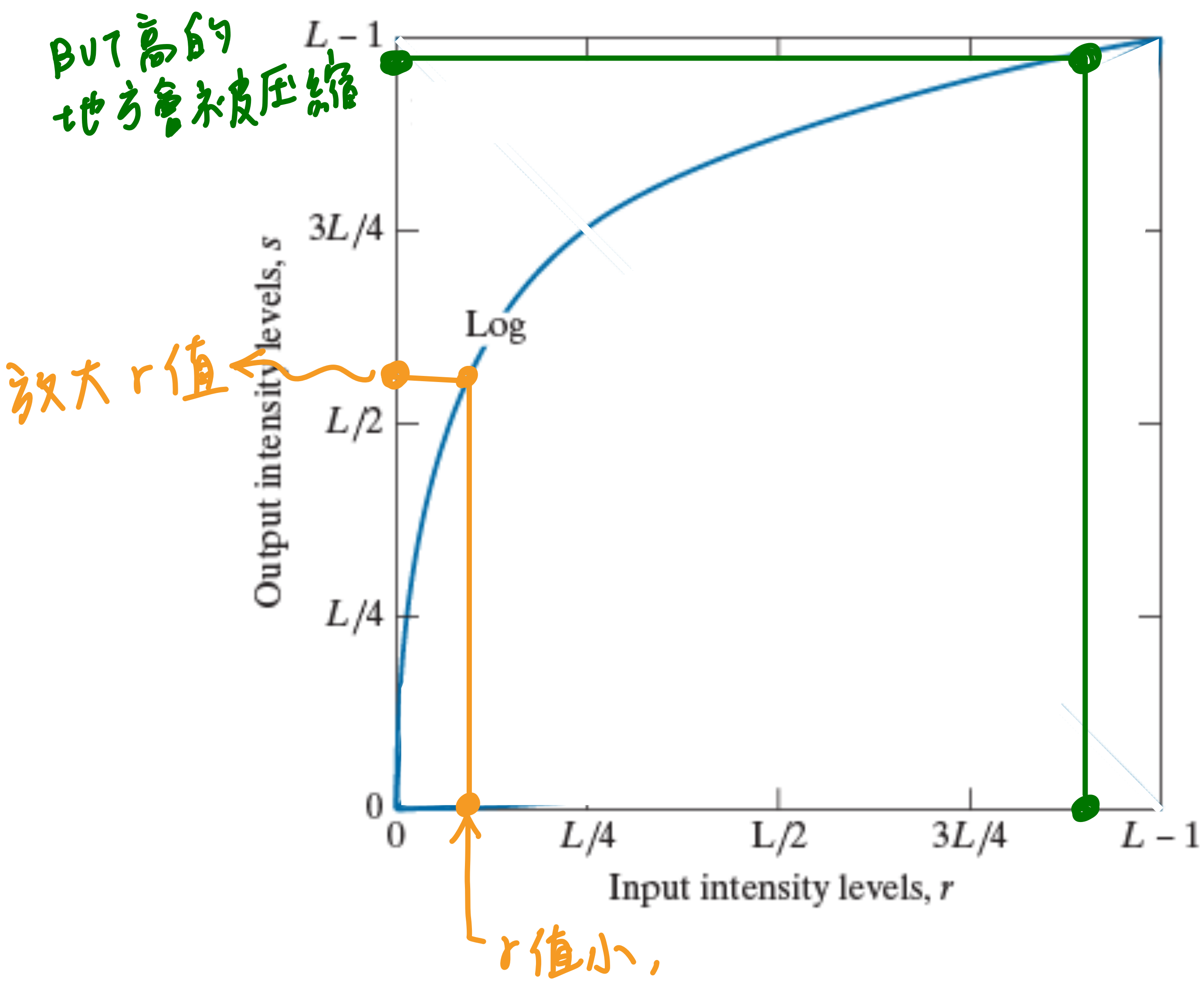
Example: Image Negatives

a b

FIGURE 3.4
(a) A digital mammogram. 乳房X光攝影片
(b) Negative image obtained using Eq. (3-3).
(Image (a) Courtesy of General Electric Medical Systems.)

Small lesion

Log Transformations



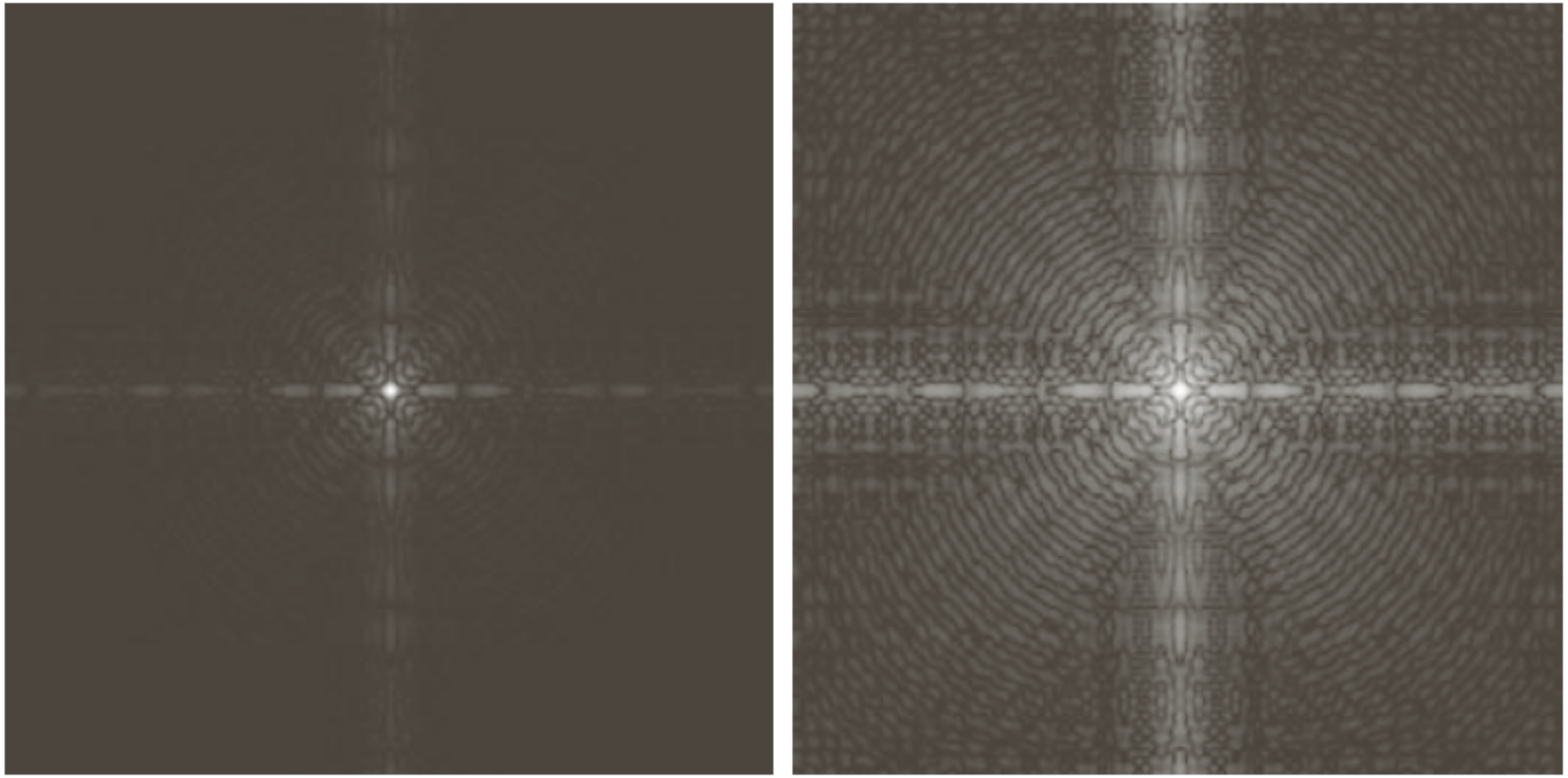
Log Transformations

$$s = c \log(1 + r)$$

$$r \geq 0$$

Used to expand the values of dark pixels in an image while compressing the higher-level values.

Example of Log Transformation

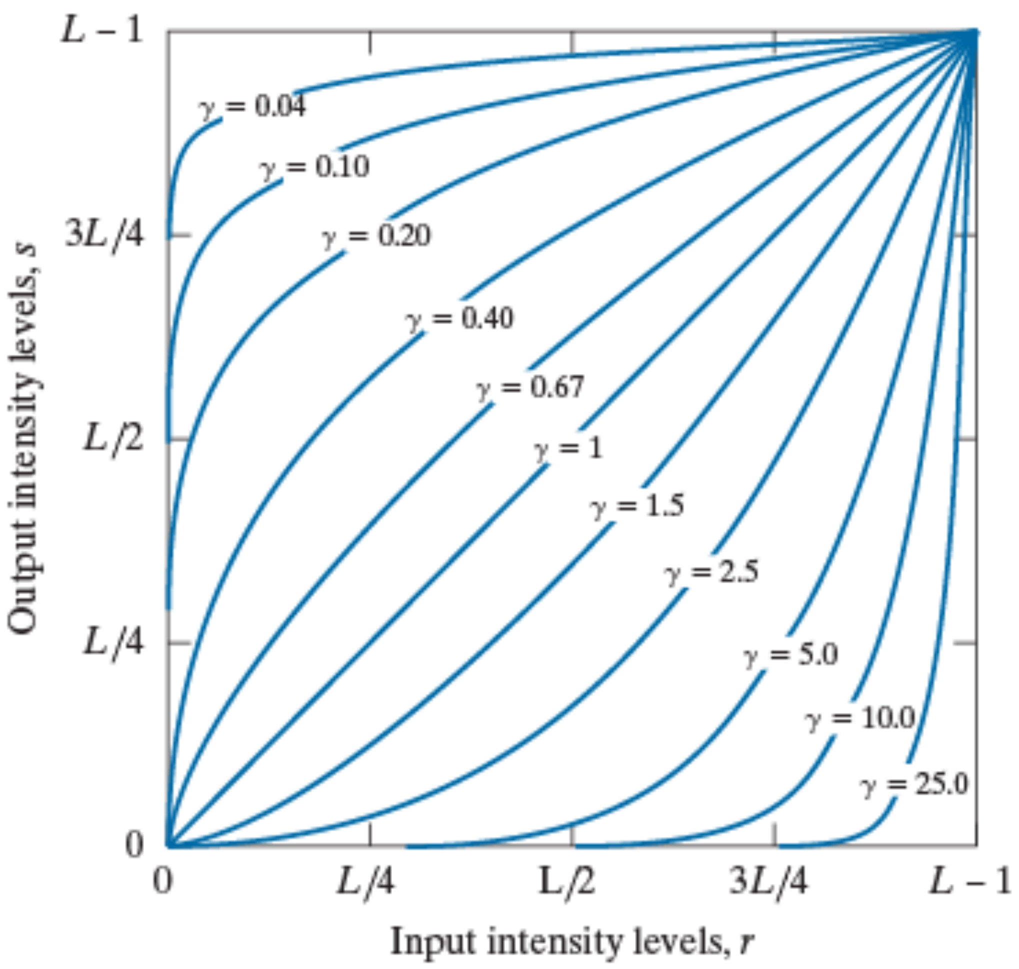


a b

FIGURE 3.5 (a) Fourier spectrum displayed as a grayscale image. (b) Result of applying the log transformation in Eq. (3-4) with $c = 1$. Both images are scaled to the range [0, 255].

Power-Law (Gamma) Transformations

FIGURE 3.6 Plots of the gamma equation $s = cr^\gamma$ for various values of γ ($c = 1$ in all cases). Each curve was scaled *independently* so that all curves would fit in the same graph. Our interest here is on the *shapes* of the curves, not on their relative values.



$$s = cr^\gamma$$

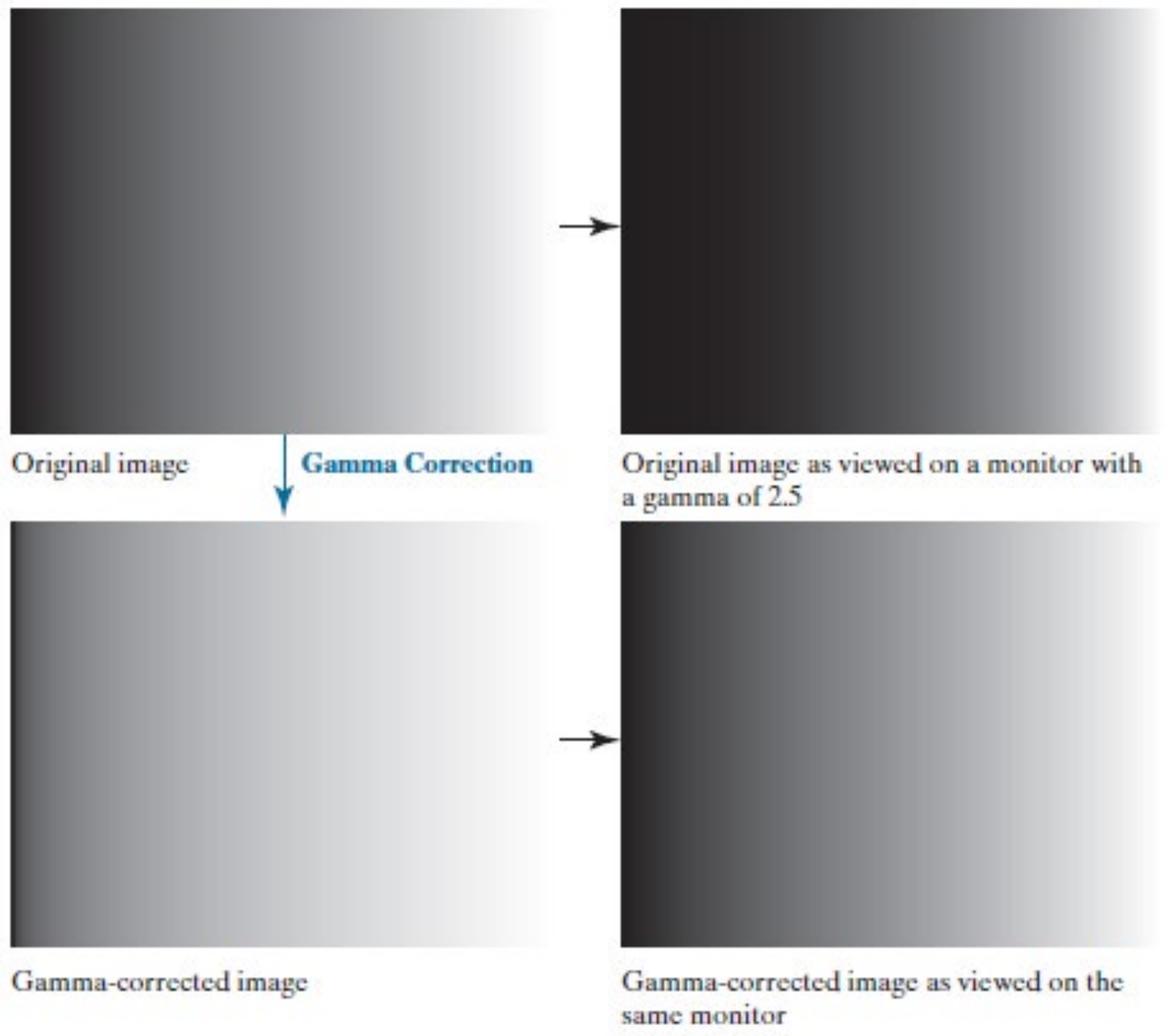
Many devices for image capture and display obey this kind of power law.

The process of correcting such power-law responses is called gamma correction.

Example of Gamma Transformation

a b
c d

FIGURE 3.7
(a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).



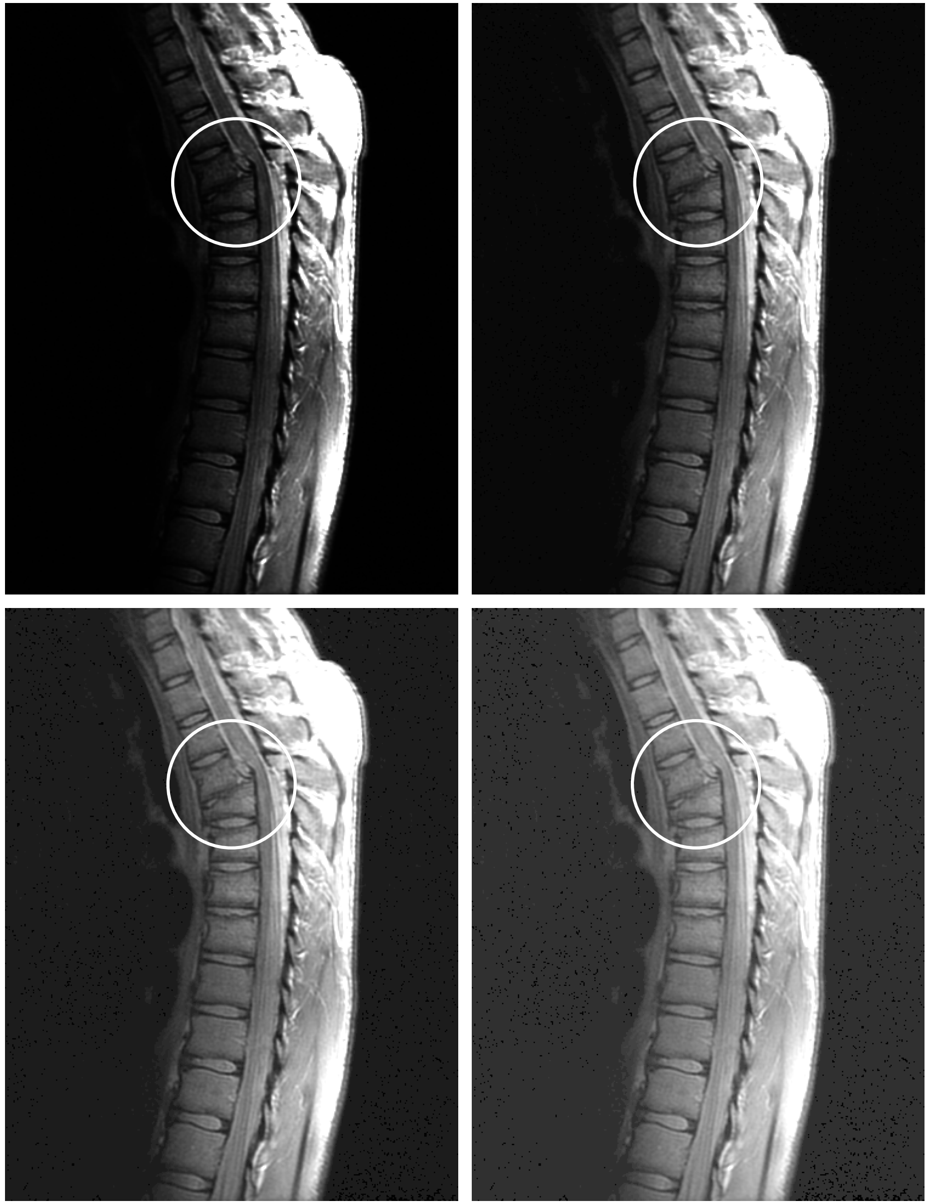
Cathode ray tube (CRT) devices have an intensity-to-voltage response that is a power function, with exponents varying from approximately 1.8 to 2.5

Gamma corrected image is obtained by applying

$$s = r^{1/2.5}$$

to the image before imputing it to the monitor.

Example of Gamma Transformation



a	b
c	d

FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine (the region of the fracture is enclosed by the circle). (b)–(d) Results of applying the transformation in Eq. (3-5) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

Example of Gamma Transformation

a	b
c	d

FIGURE 3.9 (a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3-5) with $\gamma = 3.0, 4.0$, and 5.0, respectively. ($c = 1$ in all cases.) (Original image courtesy of NASA.)

γ=3

γ=4

γ=5

$$s = cr^{\gamma}$$

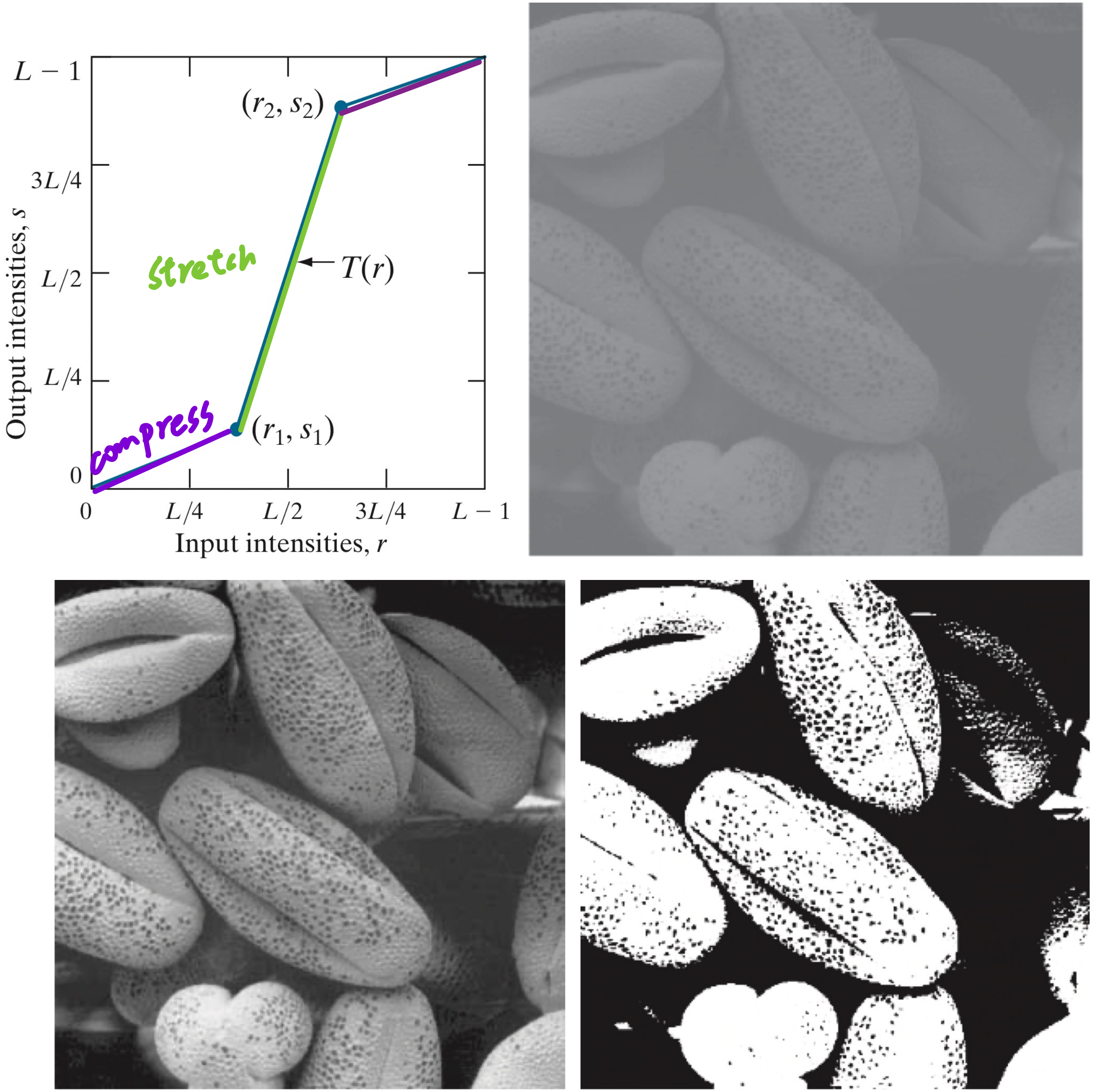
Piecewise-Linear Transformation

- Contrast Stretching
  - Expands the range of intensity levels in an image so that it spans the full intensity range of the recording medium or display device.
  - Applicable to low-contrast images resulted from poor illumination, lack of dynamic range in the image sensor, or wrong setting of a lens aperture during image acquisition.
- Intensity-Level Slicing
  - Highlighting a specific range of intensities in an image

Contrast Stretching

$(r_1,s_1)=(r_{min},0)$
$(r_2,s_2)=(r_{max},L-1)$

$(r_1,s_1)=(m,0)$
$(r_2,s_2)=(m,L-1)$

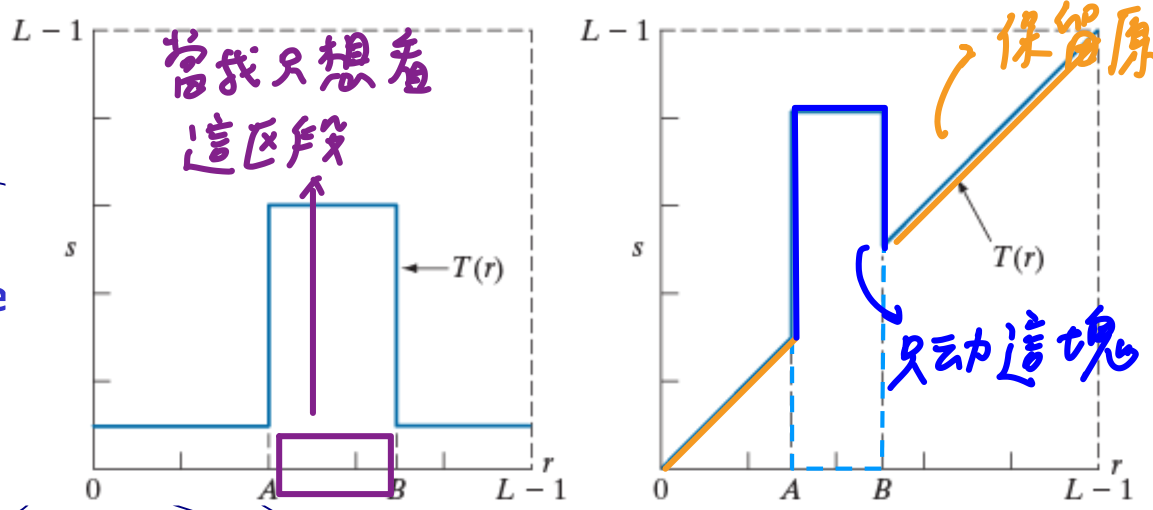


a b
c d

FIGURE 3.10 Contrast stretching. (a) Piecewise linear transformation function. (b) A low-contrast electron microscope image of pollen, magnified 700 times. (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.)

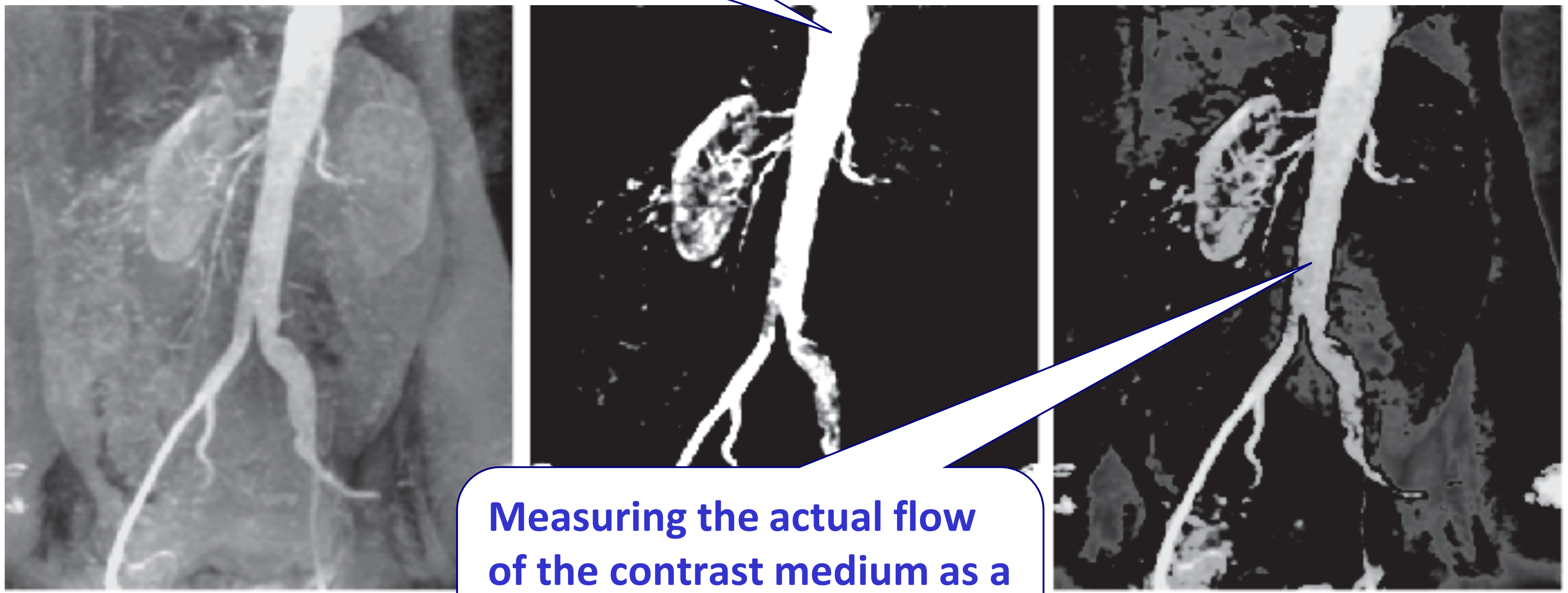
Intensity-Level Slicing

Highlight the major blood vessels and study the shape of the flow of the contrast medium (to detect blockages, etc.)



a b

FIGURE 3.11 (a) This transformation function highlights range $[A, B]$ and reduces all other intensities to a lower level. (b) This function highlights range $[A, B]$ and leaves other intensities unchanged.



Measuring the actual flow of the contrast medium as a function of time in a series of images

a b c

大主動脈的 血管造影

FIGURE 3.12 (a) Aortic angiogram. (... of the type illustrated in Fig. 3.11(a), with the range of intensities of inte... scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected range set near black, so that the grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

Bit-plane Slicing

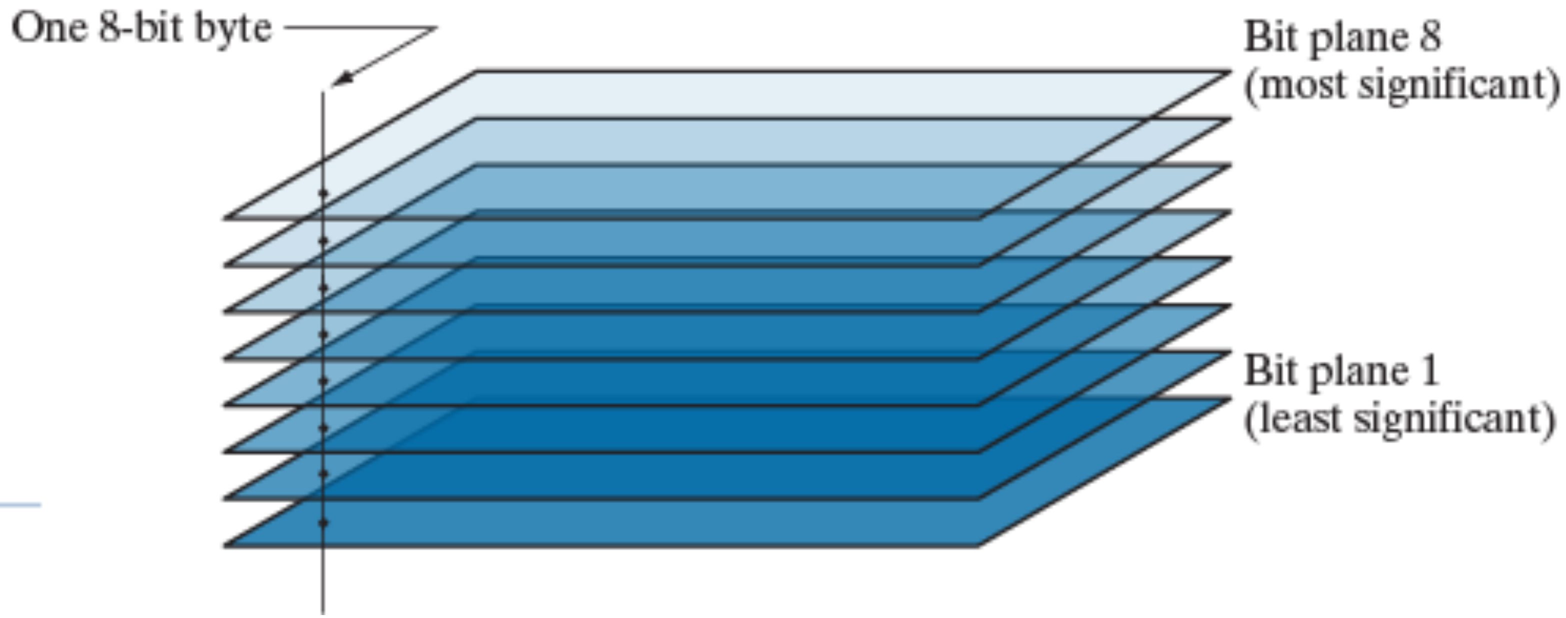
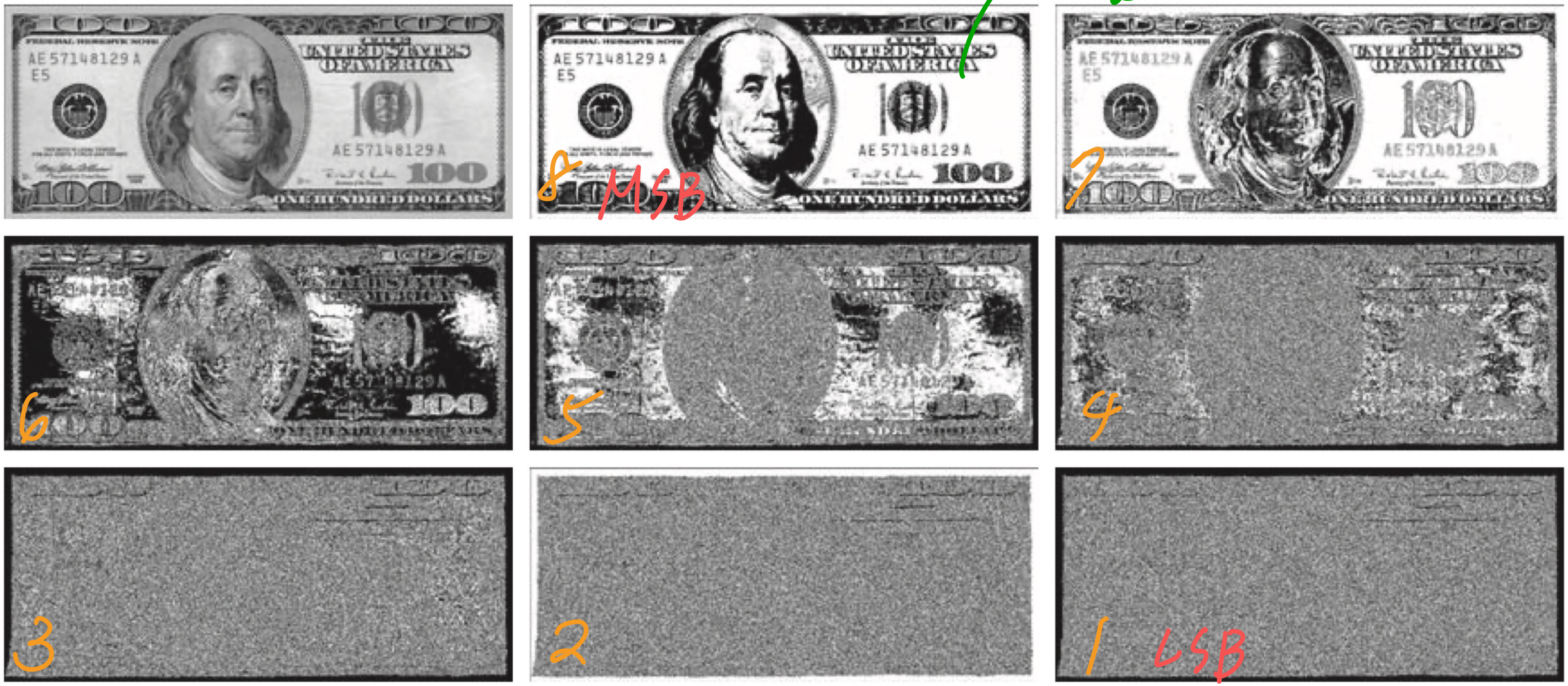


FIGURE 3.13
Bit-planes of an 8-bit image.

Bit-plane Slicing

把最白的部份
表現出來(∵白色值很大)



a	b	c
d	e	f
g	h	i

FIGURE 3.14 (a) An 8-bit gray-scale image of size 550×1192 pixels. (b) through (i) Bit planes 8 through 1, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image..

Bit-plane Slicing

False contour

a b c **FIGURE 3.15** Image reconstructed from bit planes: (a) 8 and 7; (b) 8, 7, and 6; (c) 8, 7, 6, and 5.

Fig. 15(c) is close enough to Fig. 3.14(a).
Therefore, we may store only the four highest-order bits to save 50% storage.

實際上不會這樣做 compression

FIGURE 3.14(a)

Histogram Processing

- Histogram Equalization
- Histogram Matching
- Local Histogram Processing
- Using Histogram Statistics for Image Enhancement

Histogram Processing

- Histogram

$$h(r_k) = n_k$$

r_k is the k^{th} intensity value

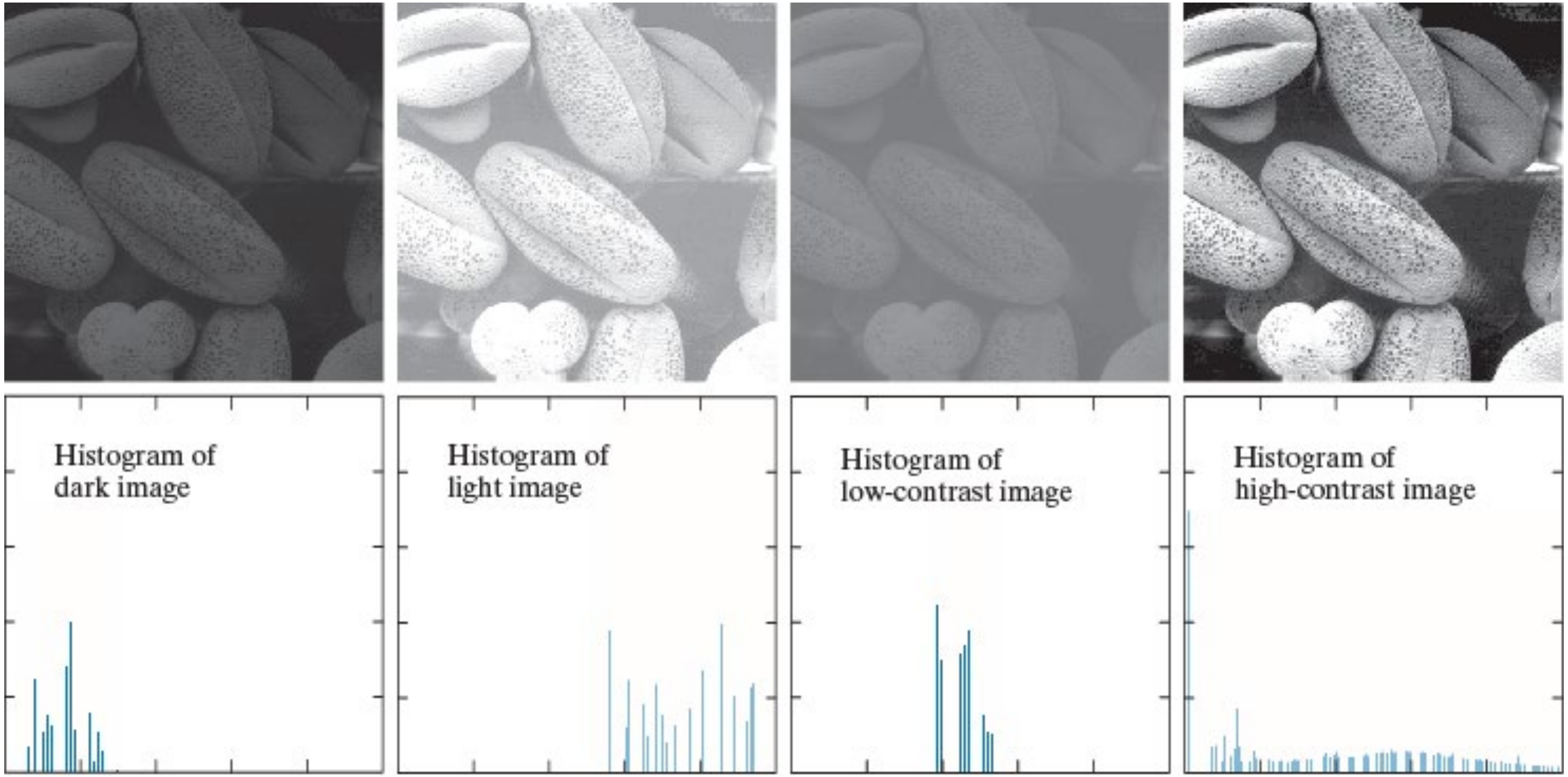
n_k is the number of pixels in the image with intensity r_k

- Normalized histogram

$$p(r_k) = \frac{n_k}{MN}$$

n_k: the number of pixels with intensity r_k in the image of size $\mathrm{M} \times \mathrm{N}$

Histogram



a b c d

FIGURE 3.16 Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of r_k and the vertical axis are values of $p(r_k)$.

Histogram Equalization

$$s = T(r) \qquad 0 \le r \le L-1$$

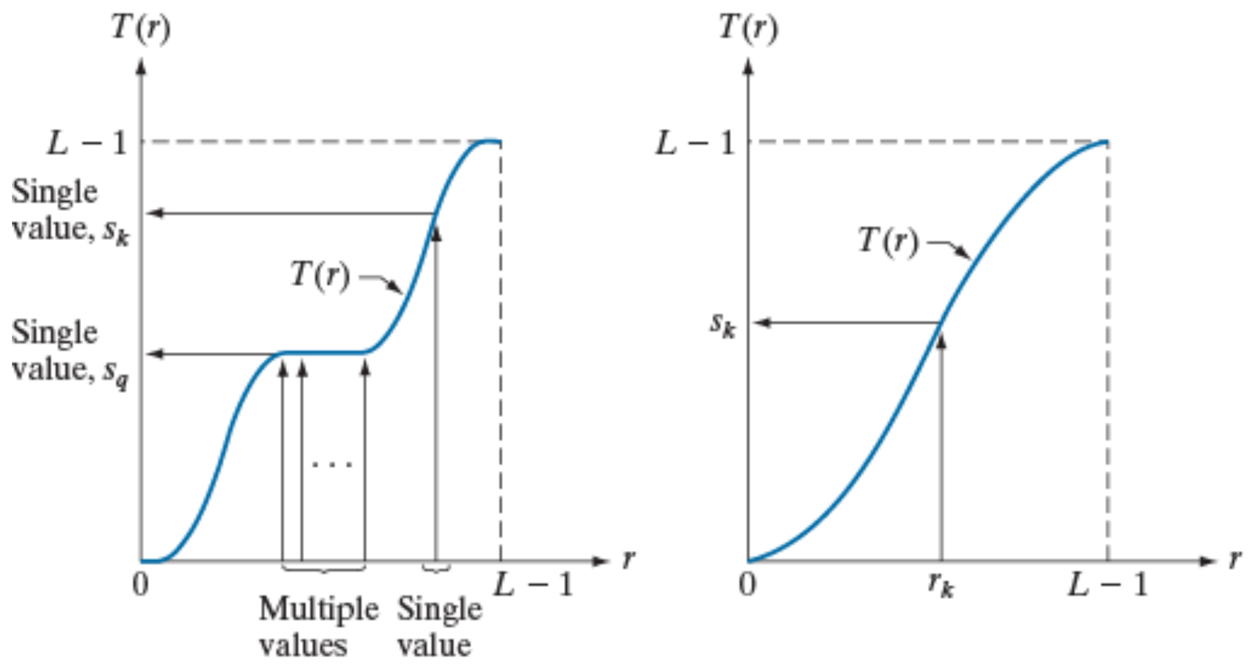
- $T(r)$ is a monotonically increasing function in the interval $0 \le r \le L-1$
- $0 \le T(r) \le L-1 \quad \text{for} \quad 0 \le r \le L-1$

$$r = T^{-1}(s), \qquad 0 \le s \le L-1$$

- $T(r)$ is a strictly monotonically increasing function in the interval $0 \le r \le L-1$
- $0 \le T(r) \le L-1$ for $0 \le r \le L-1$

a b

FIGURE 3.17 (a) Monotonic increasing function, showing how multiple values can map to a single value. (b) Strictly monotonic increasing function. This is a one-to-one mapping, both ways.



Histogram Equalization

- The intensity levels in an image may be viewed as random variables in the interval $[0, L-1]$.
- Let $p_r(r)$ and $p_s(s)$ denote the probability density function (PDF) of random variables r and s.
- If $p_r(r)$ and $T(r)$ are known and $T(r)$ is continuous and differentiable, then

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right|$$

Leibniz rule

- A transformation of particular importance is

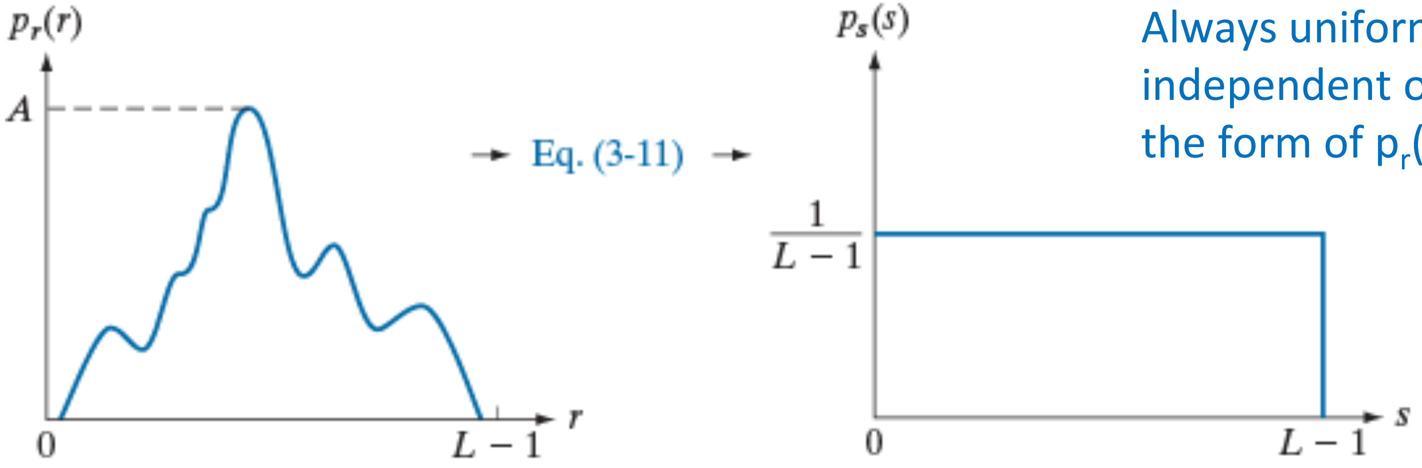
$$s = T(r) = (L-1)\underbrace{\int_0^r p_r(w)\,dw}_{\text{Cumulative distribution function (CDF)}} \qquad (3\text{-}11)$$

保證 s 是 uniform distribution

Histogram Equalization

$$\frac{ds}{dr} = \frac{dT(r)}{dr} = (L-1)\frac{d}{dr}\left[\int_0^r p_r(w)dw\right] = (L-1)p_r(r)$$

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = p_r(r)\left|\frac{1}{(L-1)p_r(r)}\right| = \frac{1}{L-1}, \quad 0 \le s \le L-1$$



Always uniform, independent of the form of $p_r(r)$.

a b

FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying Eq. (3-11) to the input PDF. The resulting PDF is always uniform, independently of the shape of the input.

Example

Suppose that the (continuous) intensity values in an image have the PDF

$$p_r(r) = \begin{cases} \dfrac{2r}{(L-1)^2}, & \text{for } 0 \le \mathrm{r} \le \mathrm{L}-1 \\ 0, & \text{otherwise} \end{cases}.$$

Prove that Eq. (3-11) leads to a uniform PDF.

$$\Rightarrow \quad s = T(r) = (L-1)\int_0^r p_r(w)dw = (L-1)\int_0^r \frac{2w}{(L-1)^2}dw = \frac{r^2}{L-1}$$

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = \frac{2r}{(L-1)^2}\left|\frac{L-1}{2r}\right| = \frac{1}{L-1}$$

Indeed, $p_s(s)$ is a uniform PDF.

Histogram Equalization 改善 constrast

Continuous case:

$$s = T(r) = (L-1)\int_0^r p_r(w)dw$$

Discrete case:

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

$$= (L-1)\sum_{j=0}^{k}\frac{n_j}{MN} = \frac{L-1}{MN}\sum_{j=0}^{k} n_j \qquad k = 0, 1, \ldots, L-1$$

Example 3.5: (Discrete) Histogram Equalization

Suppose that a 3-bit image (L=8) of size 64 × 64 pixels (MN = 4096) has the intensity distribution shown in following table.
Get the histogram equalization transformation function and give the $p_s(s_k)$ for each s_k.

TABLE 3.1
Intensity distribution and histogram values for a 3-bit, 64 × 64 digital image.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

Example 3.5: (Discrete) Histogram Equalization (cont'd)

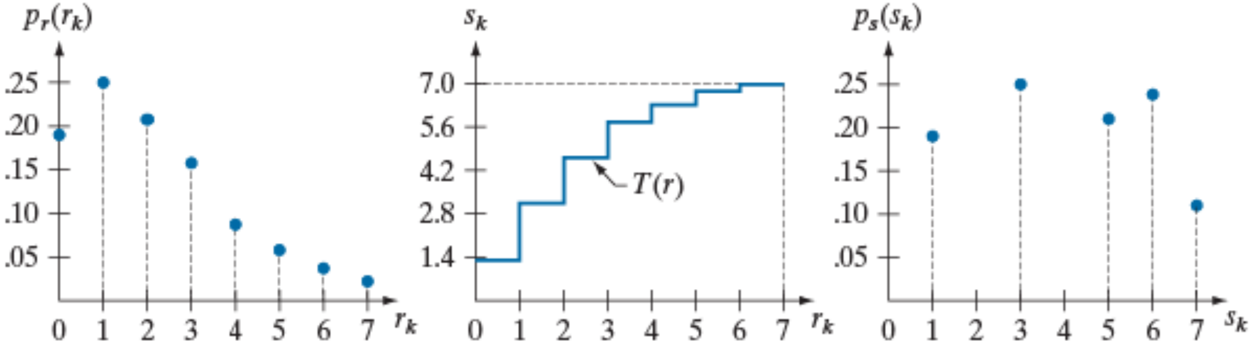
$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7 \times 0.19 = 1.33 \rightarrow 1$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7 \times (0.19 + 0.25) = 3.08 \rightarrow 3$$

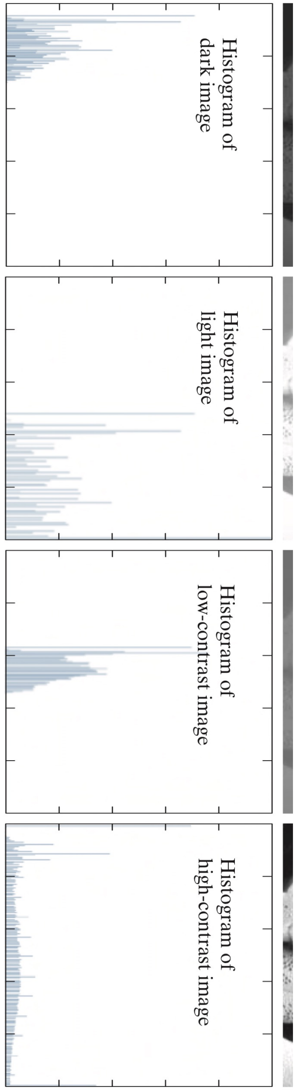
		$p_s(s_k)$
$s_0 = 1.33$	$\rightarrow 1$	0.19
$s_1 = 3.08$	$\rightarrow 3$	0.25
$s_2 = 4.55$	$\rightarrow 5$	0.21
$s_3 = 5.67$	$\rightarrow 6$	0.24
$s_4 = 6.23$	$\rightarrow 6$	
$s_5 = 6.65$	$\rightarrow 7$	0.11
$s_6 = 6.86$	$\rightarrow 7$	
$s_7 = 7.00$	$\rightarrow 7$	

a b c

FIGURE 3.19 Histogram equalization. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.



Perfectly flat histograms are rare.



a b c d

FIGURE 3.16 Four image types and their corresponding histograms. (a) dark; (b) light; (c) low contrast; (d) high contrast. The horizontal axis of the histograms are values of r_k and the vertical axis are values of $p(r_k)$.

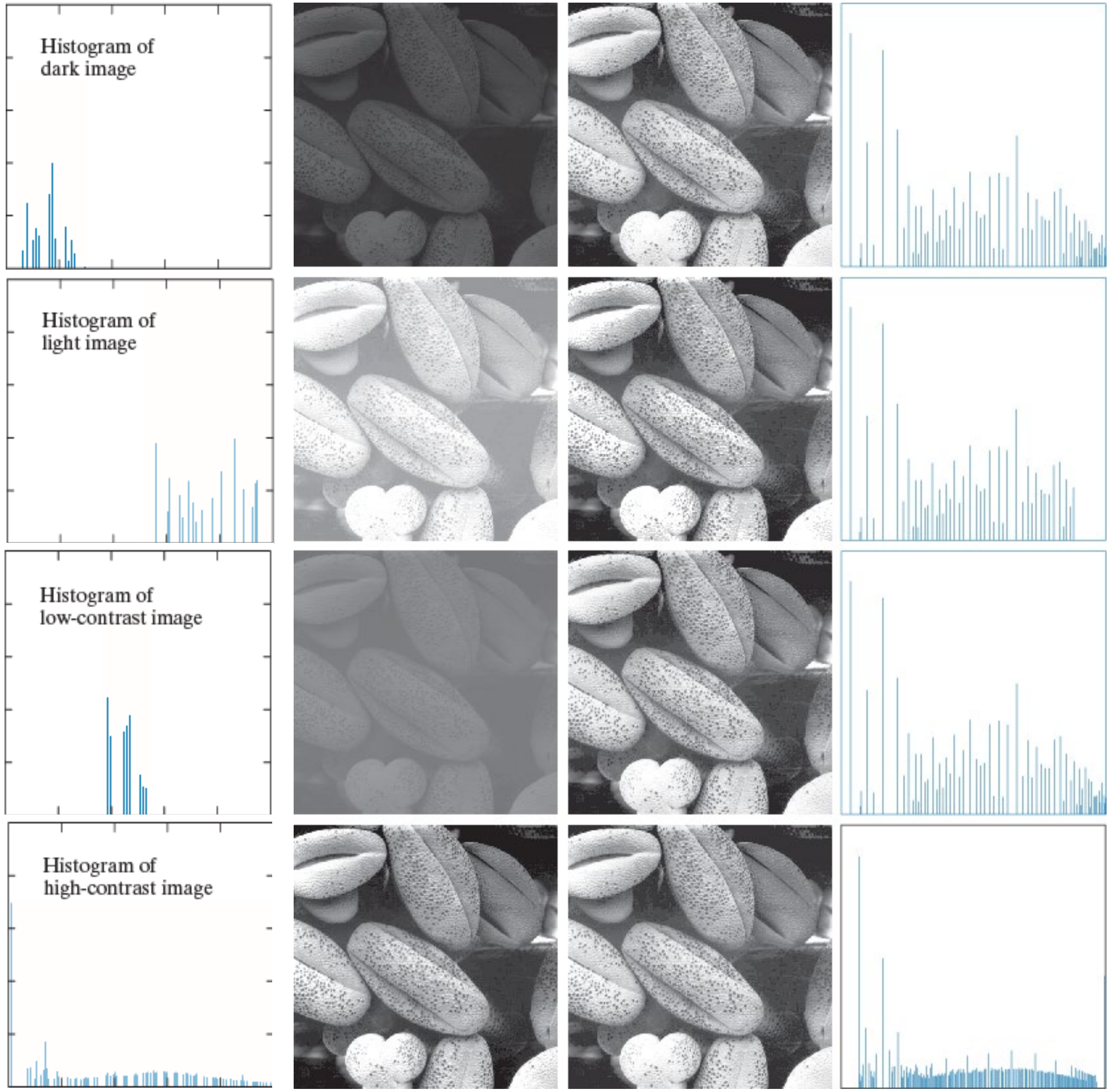
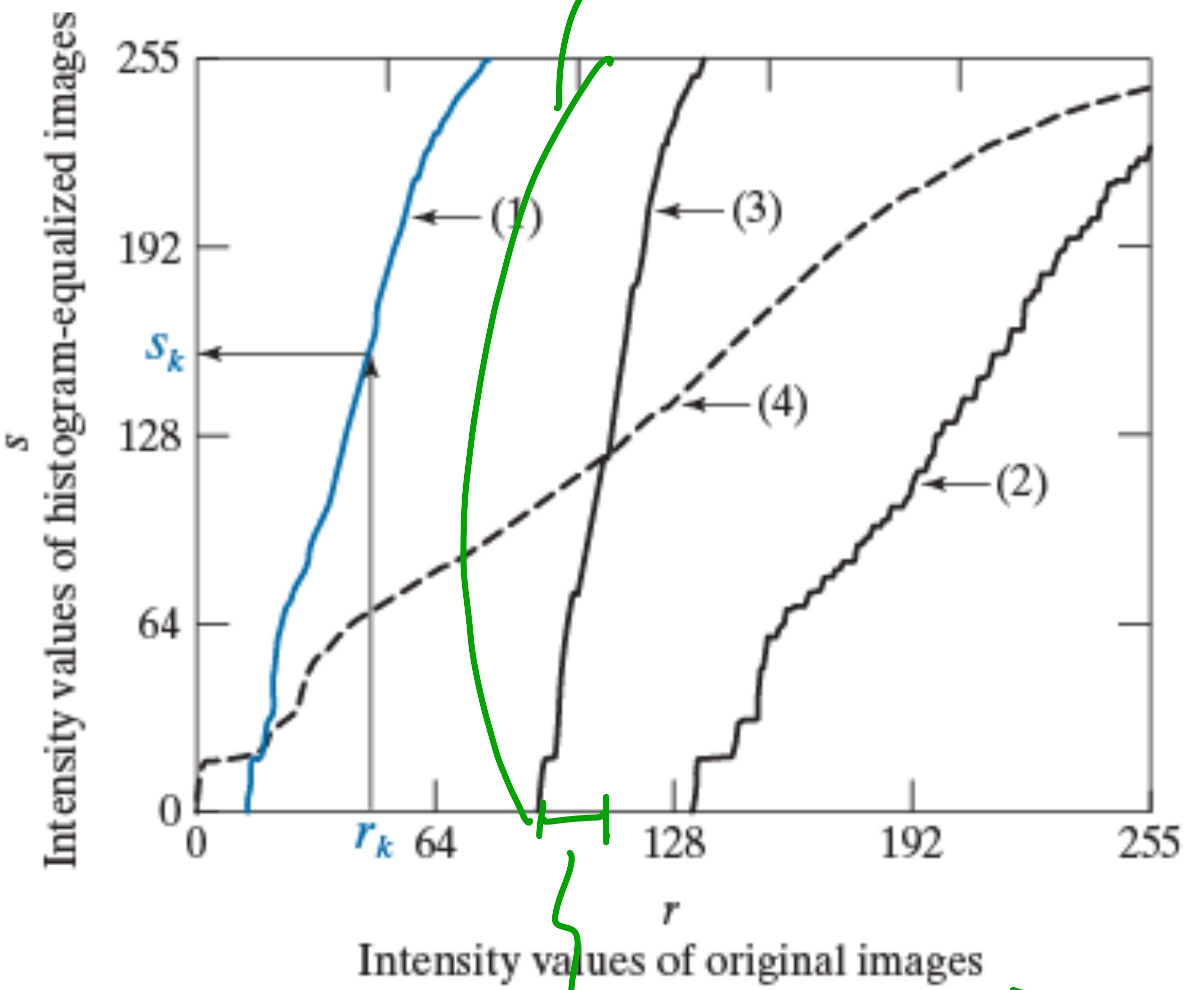


FIGURE 3.20 Left column: Images from Fig. 3.16. Center column: Corresponding histogram-equalized images. Right column: histograms of the images in the center column (compare with the histograms in Fig. 3.16).

Transformation Functions of Fig. 3.20

② 現在把範圍拉成這樣



① 原本範圍只有這樣

FIGURE 3.21 Transformation functions for histogram equalization. Transformations (1) through (4) were obtained using Eq. (3-15) and the histograms of the images on the left column of Fig. 3.20. Mapping of one intensity value r_k in image 1 to its corresponding value s_k is shown.

Histogram Matching (Specification)

Generate a processed image that has a specified histogram

Let $p_r(r)$ and $p_z(z)$ denote the continous probability density functions of the variables r and z.

$p_z(z)$ is the specified PDF that we want the output image to have.

Let s be the random variable with the probability

$$s = T(r) = (L-1)\int_0^r p_r(w)dw.$$

Define a function $G(z)$ with the property

$$G(z) = (L-1)\int_0^z p_z(t)dt = s.$$

Then, z must satisfy the condition

$$z = G^{-1}(s) = G^{-1}\left[T(r)\right].$$ →找到r和z的关系

Histogram Matching: Procedure

- Obtain $p_r(r)$ from the input image and then obtain the values of s

$$s = (L-1)\int_0^r p_r(w)dw$$

- Use the specified PDF and obtain the transformation function $G(z)$

$$G(z) = (L-1)\int_0^z p_z(t)dt = s$$

- Mapping from s to z

$$z = G^{-1}(s)$$

Histogram Matching: Example

For continuous intensity values, suppose that an image has the intensity PDF

$$p_r(r) = \begin{cases} \dfrac{2r}{(L-1)^2}, & \text{for } 0 \le r \le L\text{-}1 \\ 0\ , & \text{otherwise} \end{cases}$$

Find the transformation function that will produce an image whose intensity PDF is

$$p_z(z) = \begin{cases} \dfrac{3z^2}{(L-1)^3}, & \text{for } 0 \le z \le (L\text{-}1) \\ 0, & \text{otherwise} \end{cases}.$$

Histogram Matching: Example

Find the histogram equalization transformation for the input image

$$s = T(r) = (L-1)\int_0^r p_r(w)dw = (L-1)\int_0^r \frac{2w}{(L-1)^2}dw = \frac{r^2}{L-1}$$

Find the histogram equalization transformation for the specified histogram

$$G(z) = (L-1)\int_0^z p_z(t)dt = (L-1)\int_0^z \frac{3t^2}{(L-1)^3}dt = \frac{z^3}{(L-1)^2} = s$$

The transformation function

$$z = \left[(L-1)^2 s\right]^{1/3} = \left[(L-1)^2 \frac{r^2}{L-1}\right]^{1/3} = \left[(L-1)r^2\right]^{1/3}$$

Histogram Matching: Discrete Case

- Obtain $p_r(r_j)$ from the input image, obtain the values of s_k, and then round the value to the integer range [0, $L-1$].

$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j, \quad k = 0,1,\ldots,L-1$$

- Use the specified PDF, obtain the transformation function $G(z_q)$, and then round the value to the integer range [0, $L-1$].

$$G(z_q) = (L-1)\sum_{i=0}^{q} p_z(z_i) = s_k$$

- Mapping from s_k to z_q so that $G(z_q)$ is closes to s_k. When the mapping is not unique, choose the smallest value.

$$z_q = G^{-1}(s_k)$$

Example: Histogram Matching

Suppose that a 3-bit image (L=8) of size 64 × 64 pixels (MN = 4096) has the intensity distribution shown in the left table. Compute the histogram transformation function and make the output image with the histogram specified in the right table.

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

z_q	Specified $p_z(z_q)$
$z_0 = 0$	0.00
$z_1 = 1$	0.00
$z_2 = 2$	0.00
$z_3 = 3$	0.15
$z_4 = 4$	0.20
$z_5 = 5$	0.30
$z_6 = 6$	0.20
$z_7 = 7$	0.15

Example: Histogram Matching

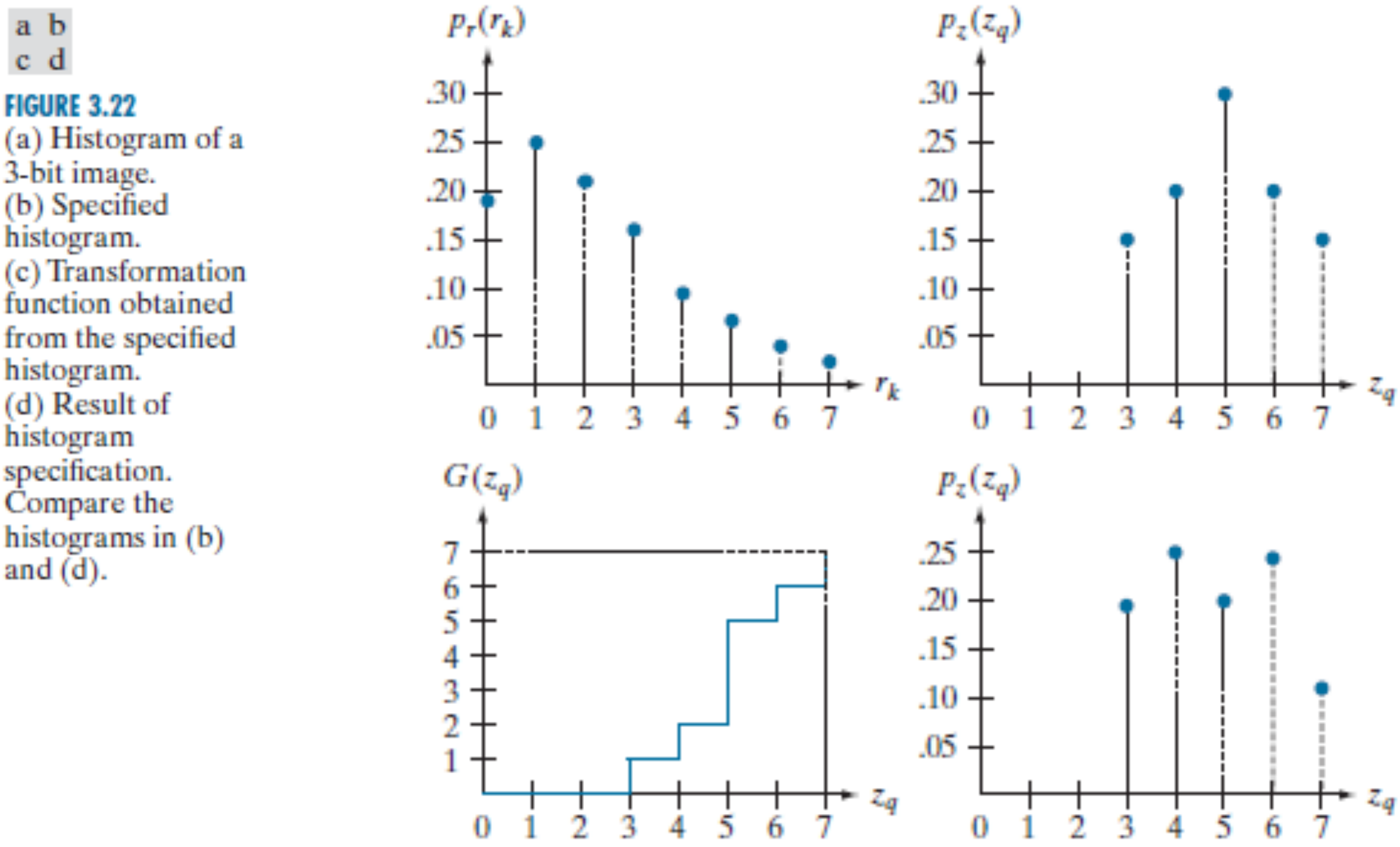


FIGURE 3.22 (a) Histogram of a 3-bit image. (b) Specified histogram. (c) Transformation function obtained from the specified histogram. (d) Result of histogram specification. Compare the histograms in (b) and (d).

Example: Histogram Matching

Obtain the scaled histogram-equalized values (from Example 3.5)

$$s_0 = 1, s_1 = 3, s_2 = 5, s_3 = 6, s_4 = 6, s_5 = 7, s_6 = 7, s_7 = 7.$$

Compute all the values of the transformation function G

$$G(z_0) = 7\sum_{j=0}^{0} p_z(z_j) = 0.00 \rightarrow 0$$

$$G(z_1) = 0.00 \rightarrow 0 \qquad G(z_2) = 0.00 \rightarrow 0$$

$$G(z_3) = 1.05 \rightarrow 1 \qquad G(z_4) = 2.45 \rightarrow 2$$

$$G(z_5) = 4.55 \rightarrow 5 \qquad G(z_6) = 5.95 \rightarrow 6$$

$$G(z_7) = 7.00 \rightarrow 7$$

TABLE 3.2
Specified and actual histograms (the values in the third column are computed in Example 3.7).

z_q	Specified $p_z(z_q)$	Actual $p_z(z_k)$
$z_0 = 0$	0.00	0.00
$z_1 = 1$	0.00	0.00
$z_2 = 2$	0.00	0.00
$z_3 = 3$	0.15	0.19
$z_4 = 4$	0.20	0.25
$z_5 = 5$	0.30	0.21
$z_6 = 6$	0.20	0.24
$z_7 = 7$	0.15	0.11

Example: Histogram Matching

Obtain the scaled histogram-equalized values (from Example 3.5)

$$s_0 = 1, s_1 = 3, s_2 = 5, s_3 = 6, s_4 = 6, s_5 = 7, s_6 = 7, s_7 = 7.$$

Compute all the values of the transformation function G

$$G(z_0) = 7\sum_{j=0}^{0} p_z(z_j) = 0.00 \rightarrow 0$$

$G(z_1) = 0.00 \rightarrow 0$ $\quad$ $G(z_2) = 0.00 \rightarrow 0$

$G(z_3) = 1.05 \rightarrow 1$ $\quad s_0 \rightarrow z_3$ $\quad$ $G(z_4) = 2.45 \rightarrow 2$ $\quad s_1 \rightarrow z_4$

$G(z_5) = 4.55 \rightarrow 5$ $\quad s_2 \rightarrow z_5$ $\quad$ $G(z_6) = 5.95 \rightarrow 6$ $\quad s_3 \rightarrow z_6$

$G(z_7) = 7.00 \rightarrow 7$ $\quad s_4 \quad s_5 \quad s_6 \quad s_7 \rightarrow z_7$

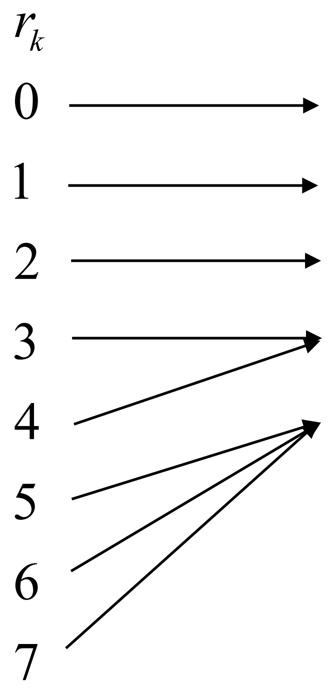
∵要遞增 ∴s_4是map到z_6

> G is not strictly monotonic
> ➔ find the smallest value of z_q so that the value $G(z_q)$ is the closest to s_k, $\forall s_k$.

Example: Histogram Matching

$$s_0 = 1, s_1 = 3, s_2 = 5, s_3 = 6, s_4 = 6, s_5 = 7, s_6 = 7, s_7 = 7.$$

TABLE 3.4 Mapping of values s_k into corresponding values z_q.

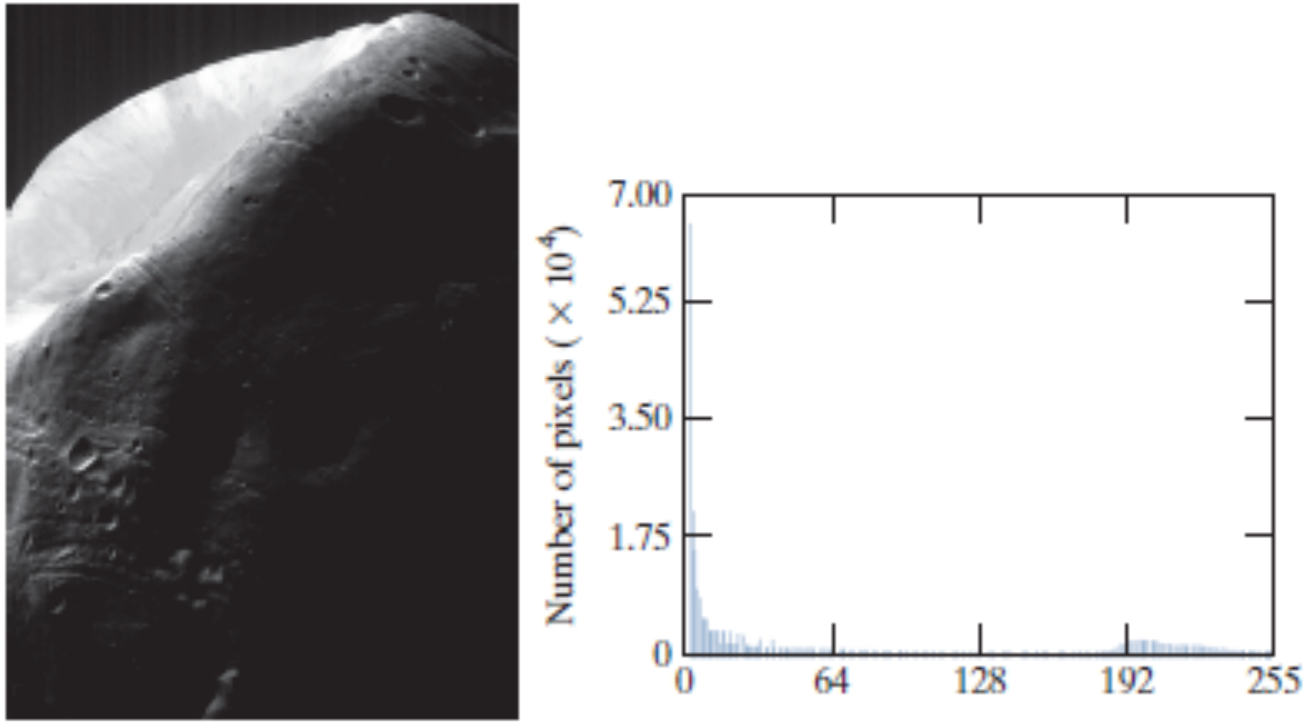


s_k	$\rightarrow$	z_q
1	$\rightarrow$	3
3	$\rightarrow$	4
5	$\rightarrow$	5
6	$\rightarrow$	6
7	$\rightarrow$	7

Histogram Matching vs. Histogram Equalization

a b

FIGURE 3.23
(a) An image, and
(b) its histogram.

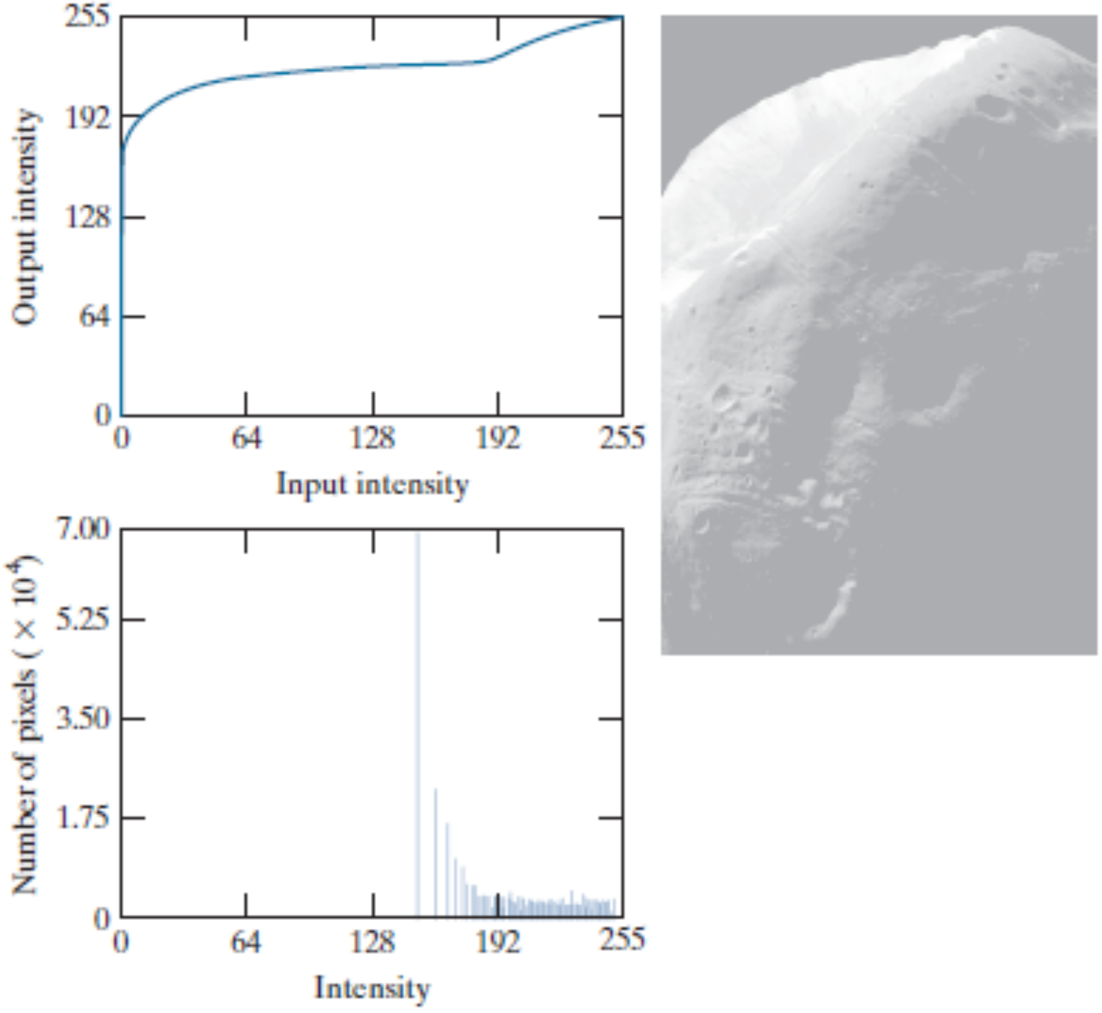


兩極

Histogram Matching vs. Histogram Equalization

a b
c

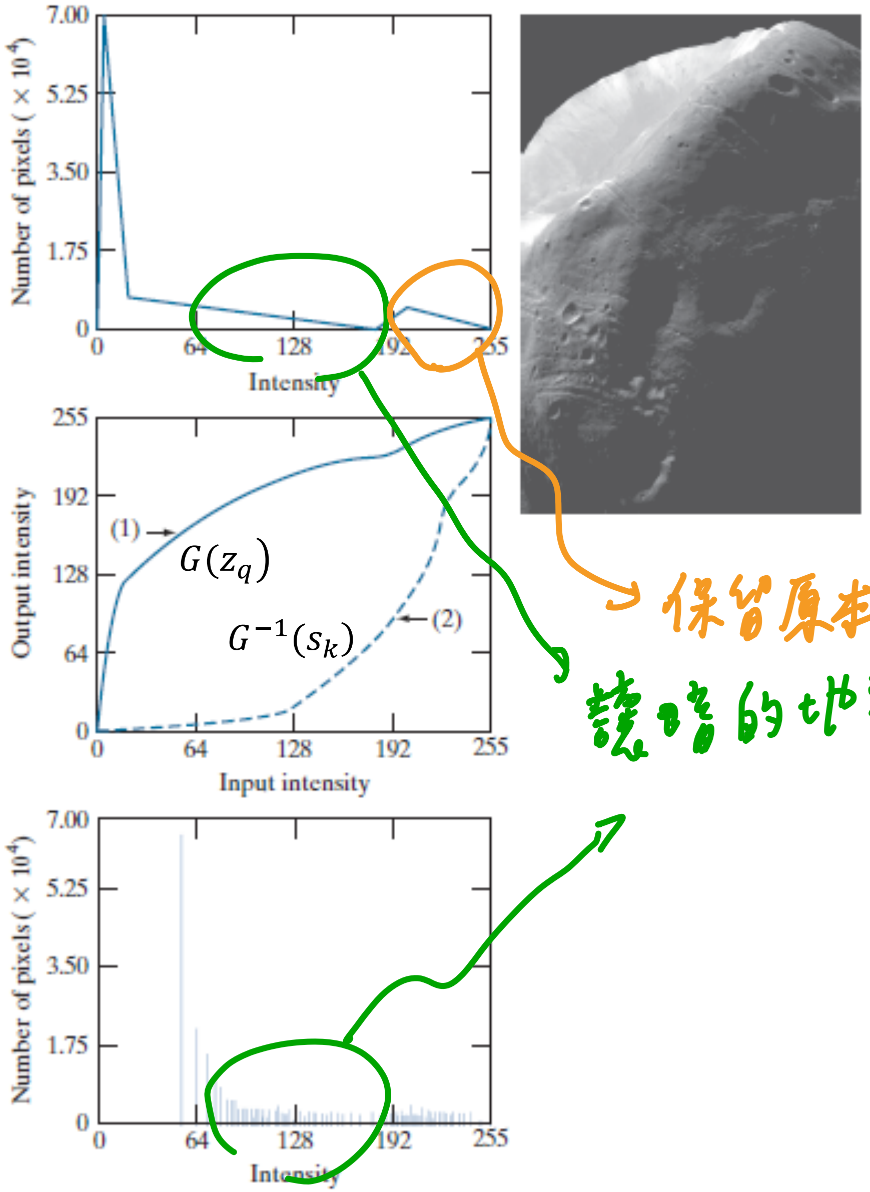
FIGURE 3.24
(a) Histogram equalization transformation obtained using the histogram in Fig. 3.23(b).
(b) Histogram equalized image.
(c) Histogram of equalized image.



Histogram Matching vs. Histogram Equalization

a c
b
d

FIGURE 3.25 Histogram specification. (a) Specified histogram. (b) Transformation $G(z_q)$, labeled (1), and $G^{-1}(s_k)$, labeled (2). (c) Result of histogram specification. (d) Histogram of image (c).



保留原本亮的地方

較暗的地方往中間移

Local Histogram Equalization

Define a neighborhood and move its center from pixel to pixel

At each location, the histogram of the pixels in the neighborhood is computed. Either histogram equalization or histogram specification transformation function is obtained

Map the intensity of the pixel centered in the neighborhood

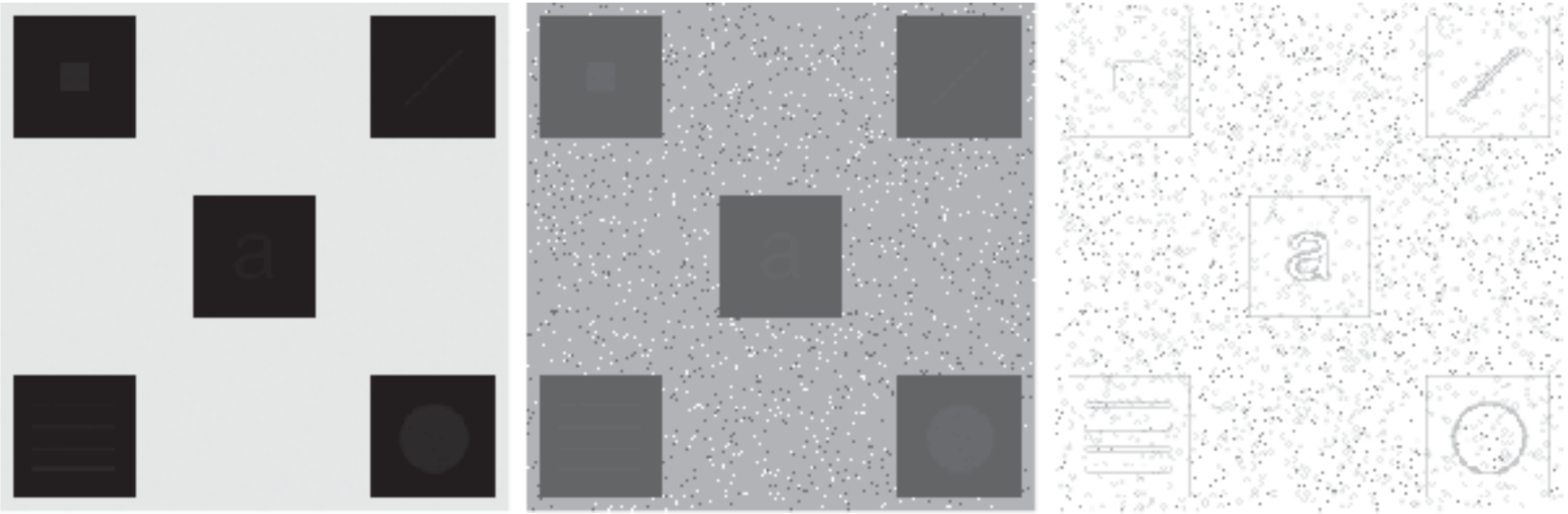
Move to the next location and repeat the procedure

Example of Local Histogram Equalization

Original Image: Imperceptible noise throughout; invisible objects embedded in the dark squares

Global histogram equalization: Noise enhanced but no new details

Local histogram equalization with a neighborhood of size 3x3. Dark details appear.



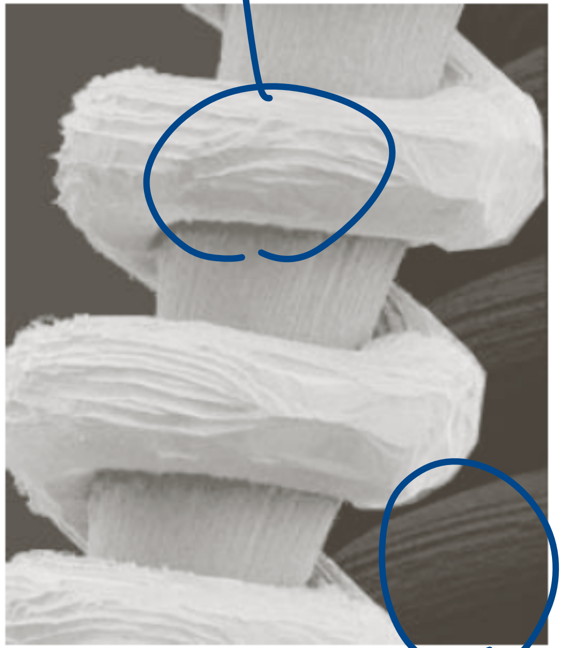
a b c

FIGURE 3.26
(a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization.

Example of Local Histogram Equalization

希望保留亮区不变

亮区被加强了(不好)



Original image

Global histogram equalization

Local histogram equalization

希望暗处增强

Using Histogram Statistics for Image Enhancement

Let r denote a discrete random variable representing intensity values in the range [0, L–1].
Let $p(r_i)$ denote the normalized histogram component corresponding to intensity value r_i .

Moment

$$\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n p(r_i) \quad \leftarrow \ n\text{th moment of } r \text{ about its mean}$$

Mean (average intensity)

$$m = \sum_{i=0}^{L-1} r_i p(r_i) \quad = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y)$$

Variance

$$\sigma^2 = u_2(r) = \sum_{i=0}^{L-1} (r_i - m)^2 p(r_i) \quad = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \left[f(x, y) - m \right]^2$$

The mean is a measure of average intensity, and the variance is a measure of image contrast.

Using Histogram Statistics for Image Enhancement

Let S_{xy} denote a neighborhood of specified size, centered on (x,y).

分区計算 Local average intensity

$$m_{s_{xy}} = \sum_{i=0}^{L-1} r_i p_{s_{xy}}(r_i)$$

告訴

→ 明暗程度

Local variance

告訴

→ detail

$$\sigma^2_{s_{xy}} = \sum_{i=0}^{L-1} (r_i - m_{s_{xy}})^2 p_{s_{xy}}(r_i)$$

Example: Using Histogram Statistics for Image Enhancement

$$g(x,y)=\begin{cases} C\,f(x,y) & \text{if } k_0 m_G \le m_{s_{xy}} \le k_1 m_G \text{ and } k_2\sigma_G \le \sigma_{s_{xy}} \le k_3\sigma_G \\ f(x,y) & \text{otherwise} \end{cases}$$

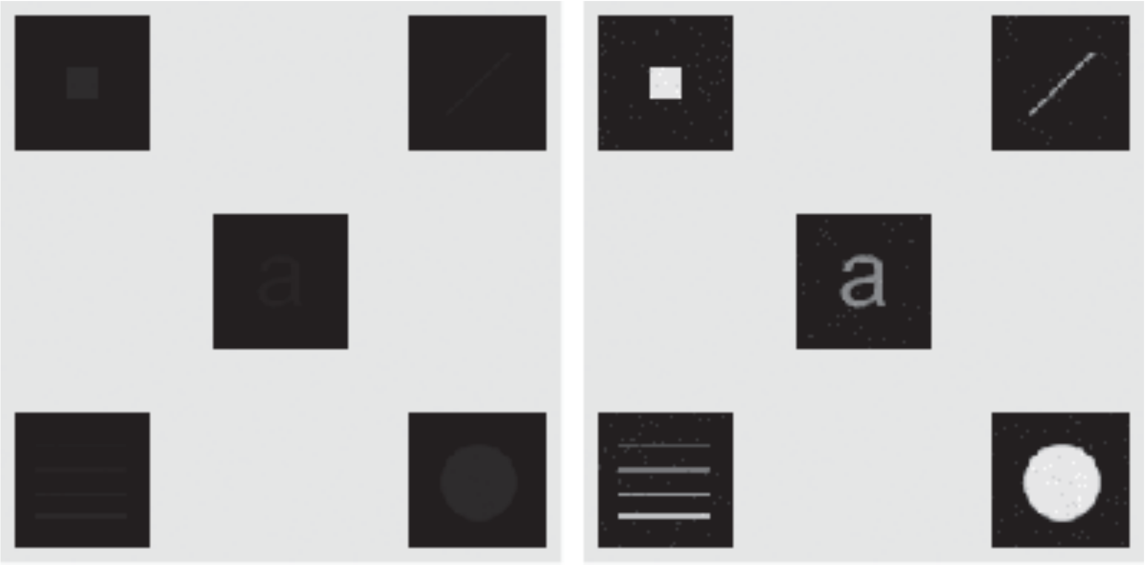
Dark and low contrast

$$m_G \text{ (global mean)} = 161,\quad \sigma_G \text{ (global standard deviation)} = 103;$$

$$r_{\max} = 228 \text{ for image},\quad r_{\max} = 10 \quad \text{for dark squares, } r_{\min} = 0;$$

$$k_0 = 0,\quad k_1 = 0.1,\ k_2 = 0,\ k_3 = 0.1,\ C = 22.8.$$

Set C=22.8 to make maximum intensity = r_{max}.



a b

FIGURE 3.27 (a) Original image. (b) Result of local enhancement based on local histogram statistics. Compare (b) with Fig. 3.26(c).

Spatial Filtering

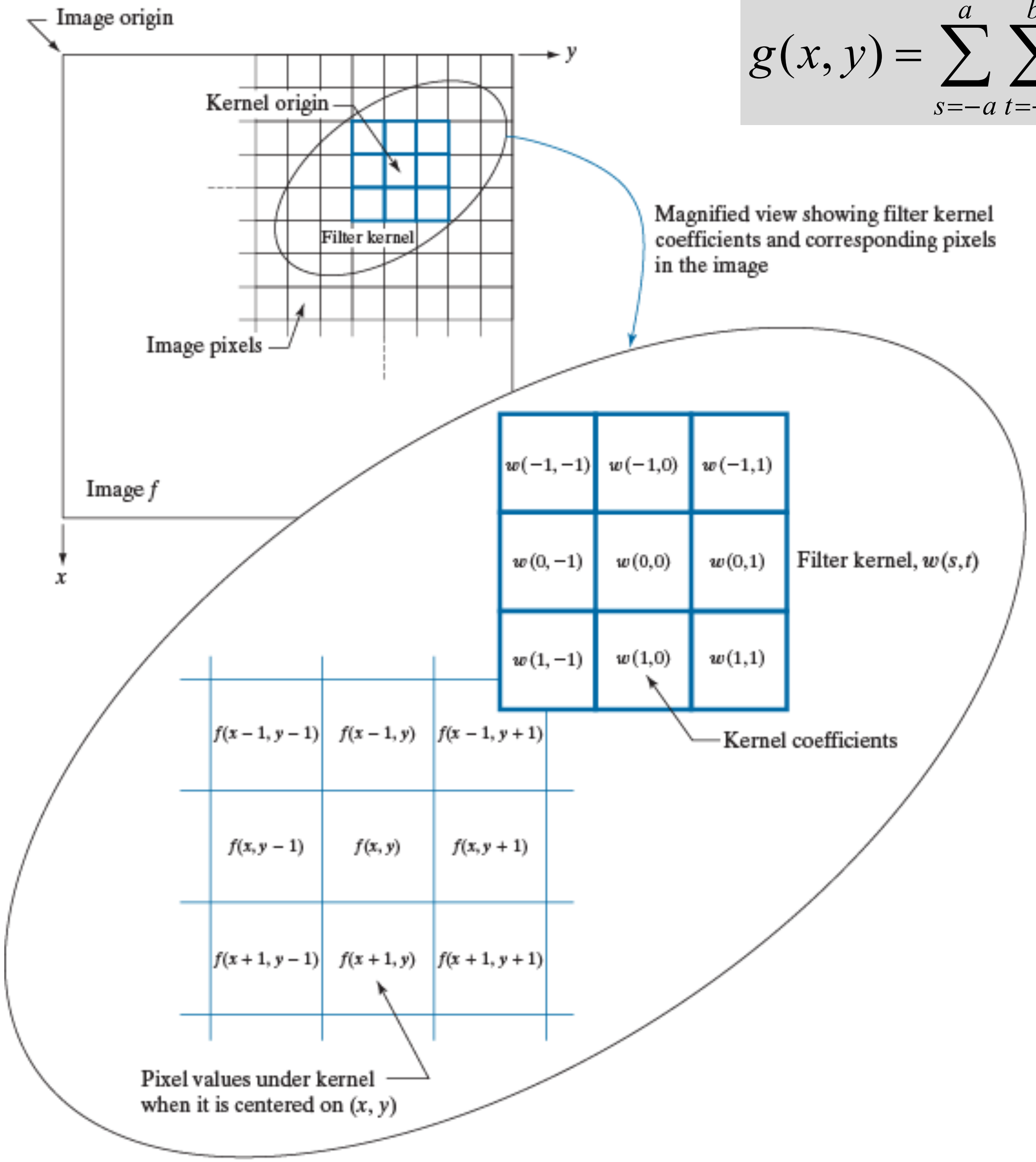
- Filtering refers to passing, modifying, or rejecting specified components of an image. If the operation is linear (nonlinear), it is called a linear (nonlinear) filter.
- Spatial filtering modifies an image by replacing the value of each pixel by a function of the values of the pixel and its neighbors.
- Linear spatial filtering of an image of size M x N with a filter kernel (or kernel) w of size m x n is given by the expression

$$g(x,y) = \underbrace{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t) f(x+s, y+t)}_{\text{A sum-of-products operation}}$$

Spatial Filtering

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

FIGURE 3.28 The mechanics of linear spatial filtering using a 3×3 kernel. The pixels are shown as squares to simplify the graphics. Note that the origin of the image is at the top left, but the origin of the kernel is at its center. Placing the origin at the center of spatially symmetric kernels simplifies writing expressions for linear filtering.



Convolution vs. Correlation

The convolution of a filter $w(x, y)$ of size $m \times n$ with an image $f(x, y)$ is denoted by $w(x, y) \bigstar f(x, y)$

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

The correlation of a filter $w(x, y)$ of size $m \times n$ with an image $f(x, y)$ is denoted by $w(x, y) \Join f(x, y)$

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

If the kernel is symmetric about its center, convolution and correlation yield the same result.

Correlation vs. Convolution

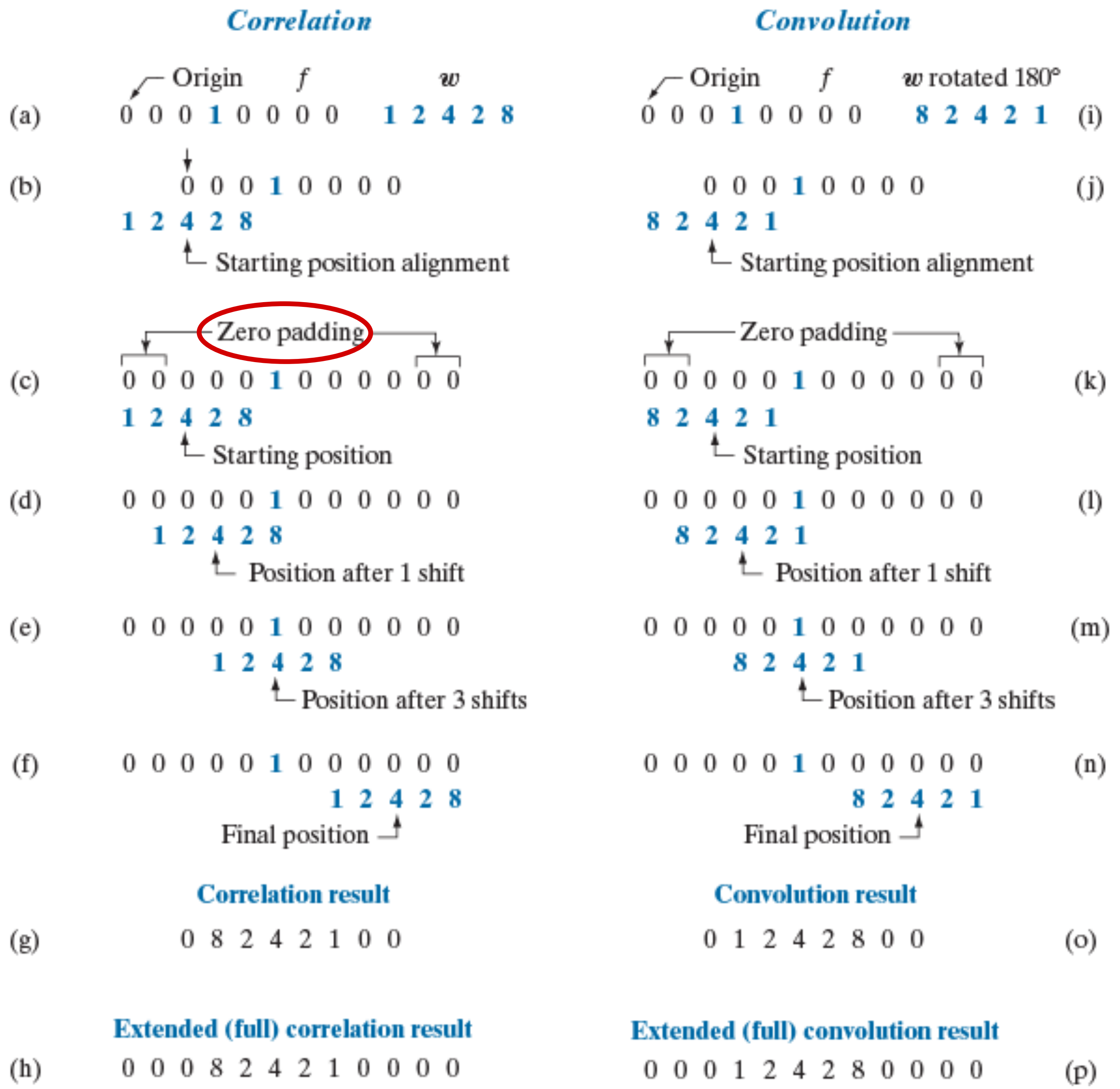


FIGURE 3.29 Illustration of 1-D correlation and convolution of a kernel, *w*, with a function *f* consisting of a discrete unit impulse. Note that correlation and convolution are functions of the variable *x*, which acts to *displace* one function with respect to the other. For the extended correlation and convolution results, the starting configuration places the right-most element of the kernel to be coincident with the origin of *f*. Additional padding must be used.

Origin f

$$\begin{matrix}0&0&0&0&0\\0&0&0&0&0\\0&0&1&0&0\\0&0&0&0&0\\0&0&0&0&0\end{matrix} \qquad w \quad \begin{matrix}1&2&3\\4&5&6\\7&8&9\end{matrix}$$

(a)

Padded f

$$\begin{matrix}0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&1&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\end{matrix}$$

(b)

Initial position for w

$$\begin{matrix}1&2&3&0&0&0&0\\4&5&6&0&0&0&0\\7&8&9&0&0&0&0\\0&0&0&1&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\end{matrix}$$

(c)

Correlation result

$$\begin{matrix}0&0&0&0&0\\0&9&8&7&0\\0&6&5&4&0\\0&3&2&1&0\\0&0&0&0&0\end{matrix}$$

(d)

Full correlation result

$$\begin{matrix}0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&9&8&7&0&0\\0&0&6&5&4&0&0\\0&0&3&2&1&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\end{matrix}$$

(e)

Rotated w

$$\begin{matrix}9&8&7&0&0&0&0\\6&5&4&0&0&0&0\\3&2&1&0&0&0&0\\0&0&0&1&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\end{matrix}$$

(f)

Convolution result

$$\begin{matrix}0&0&0&0&0\\0&1&2&3&0\\0&4&5&6&0\\0&7&8&9&0\\0&0&0&0&0\end{matrix}$$

(g)

Full convolution result

$$\begin{matrix}0&0&0&0&0&0&0\\0&0&0&0&0&0&0\\0&0&1&2&3&0&0\\0&0&4&5&6&0&0\\0&0&7&8&9&0&0\\0&0&0&0&0&0&0\\0&0&0&0&0&0&0\end{matrix}$$

(h)

FIGURE 3.30 Correlation (middle row) and convolution (last row) of a 2-D kernel with an image consisting of a discrete unit impulse. The 0's are shown in gray to simplify visual analysis. Note that correlation and convolution are functions of x and y. As these variable change, they *displace* one function with respect to the other. See the discussion of Eqs. (3-36) and (3-37) regarding full correlation and convolution.

Fundamental Properties

TABLE 3.5
Some fundamental properties of convolution and correlation. A dash means that the property does not hold.

Property	Convolution	Correlation
Commutative	$f \bigstar g = g \bigstar f$	—
Associative	$f \bigstar (g \bigstar h) = (f \bigstar g) \bigstar h$	—
Distributive	$f \bigstar (g + h) = (f \bigstar g) + (f \bigstar h)$	$f \star (g + h) = (f \star g) + (f \star h)$

Three Approaches to Filter Construction

1. Formulating filters based on mathematical properties
   - Computing the average to smooth an image is analogous to integration
   - Computing the derivative of an image sharpens the image
2. Sampling a 2D spatial function whose shape has a desired property
   - For example, Gaussian function in Section 3.5
3. Designing from a specified frequency response, typically using a filter design software
   - Section 3.7

Smoothing (Lowpass) Spatial Filters

- Smoothing filters are used for blurring or noise reduction
- Blurring can be applied to remove small details of images or bridge in small gaps for lines or curves
- Smoothing spatial filters include linear and nonlinear filters
- The general implementation for filtering an M×N image with a weighted averaging filter of size m×n is given

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

where $2a+1=M$ and $2b+1=N$.

Smoothing (Lowpass) Kernels

Box and Gaussian Kernels

a b

FIGURE 3.31 Examples of smoothing kernels: (a) is a *box* kernel; (b) is a *Gaussian* kernel.

Box

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

1	1	1
1	1	1
1	1	1

Gaussian

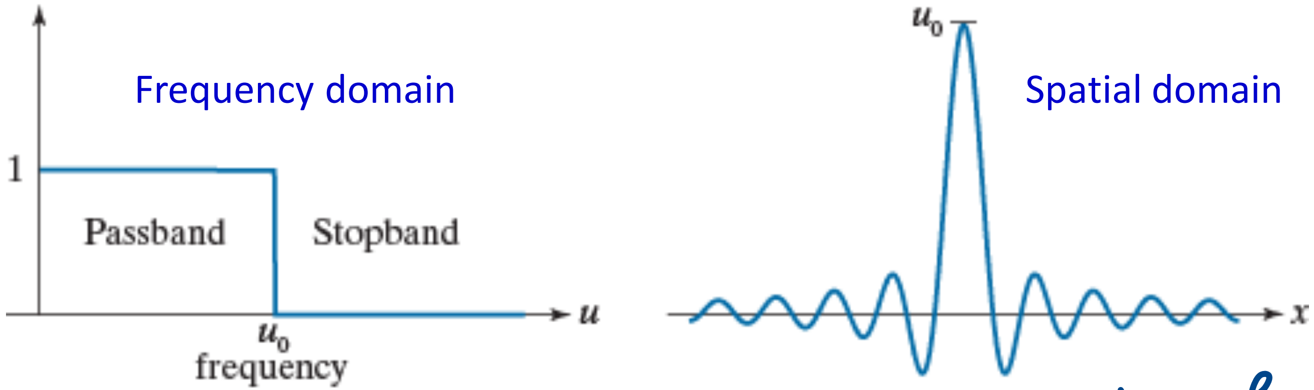
$\frac{1}{4.8976} \times$

0.3679	0.6065	0.3679
0.6065	1.0000	0.6065
0.3679	0.6065	0.3679

1-D Ideal Low-Pass Filter

a b

FIGURE 3.32 (a) Ideal 1-D low-pass filter transfer function in the frequency domain. (b) Corresponding filter kernel in the spatial domain.

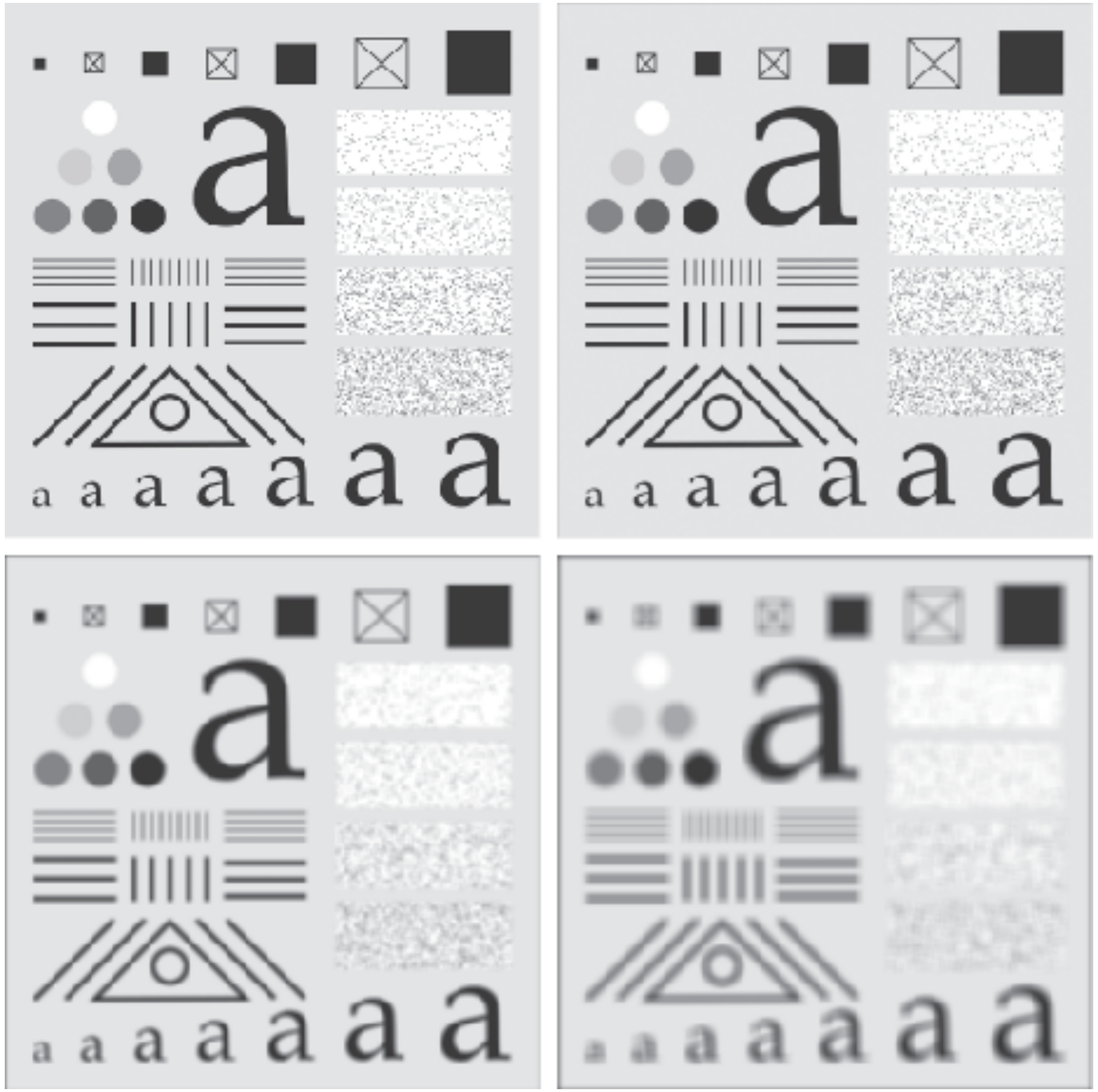


sinc function

Low-Pass Filtering Using a Box Kernel

a b
c d

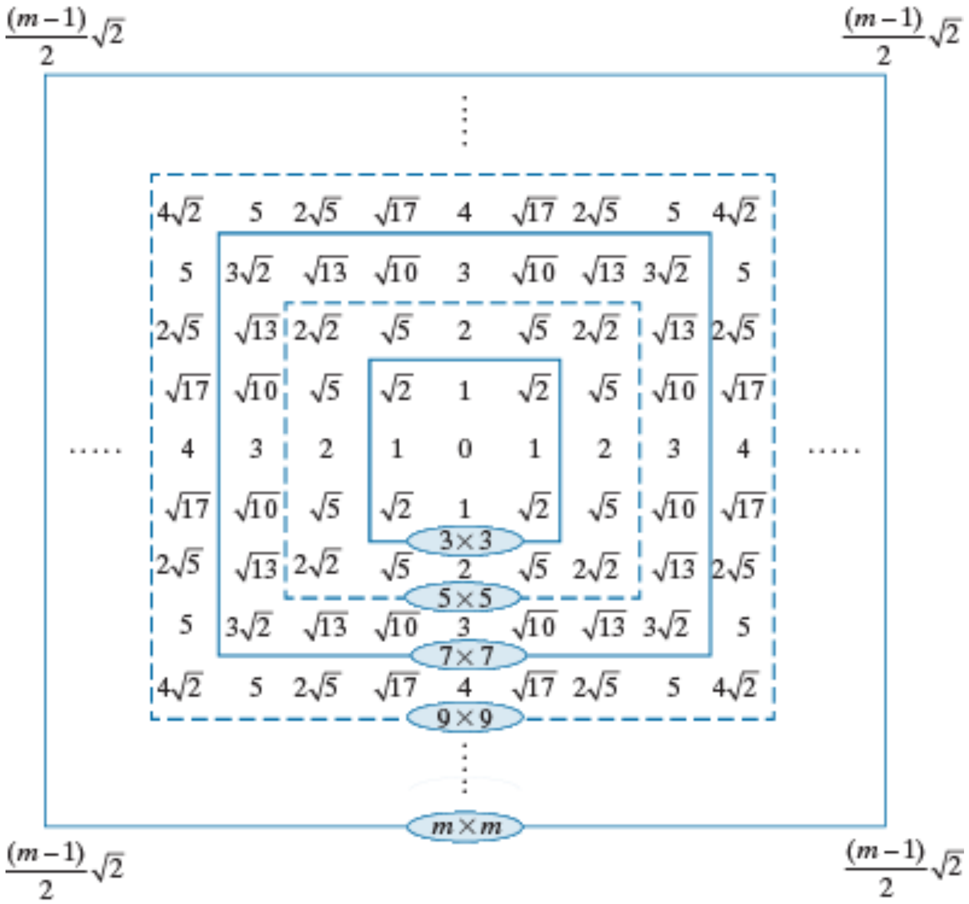
FIGURE 3.33
(a) Test pattern of size 1024×1024 pixels.
(b)-(d) Results of lowpass filtering with box kernels of sizes 3×3, 11×11, and 21×21, respectively.



kernel size ↑
smooth ↑

Distances to Center of Square Gaussian Kernels

FIGURE 3.34 Distances from the center for various sizes of square kernels.



Value is negligible when distance >3σ

$$G(x, y) = Ke^{\frac{x^2+y^2}{2\sigma^2}}$$

$$K = 1, \ \sigma = 1$$

∴ Gaussian filter
大過某程度就肉眼無法辨別
所以 kenel 不用太大

Discretization of Gaussian Filter

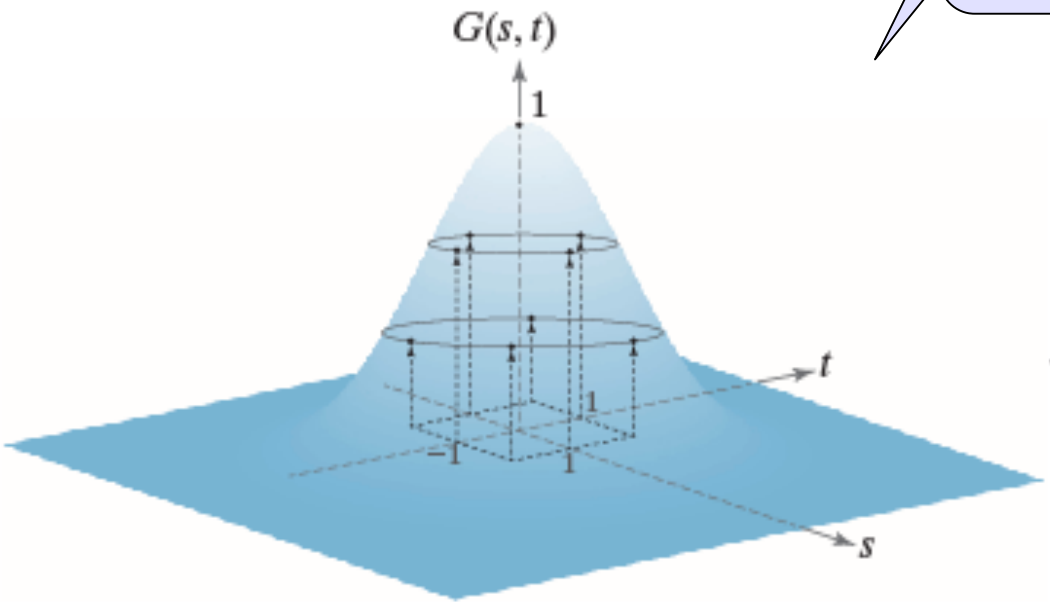
$$G(x,y) = Ke^{\frac{x^2+y^2}{2\sigma^2}}$$

$$K = 1, \ \sigma = 1$$

circularly symmetric (isotropic)

a b

FIGURE 3.35 (a) Sampling a Gaussian function to obtain a discrete Gaussian kernel. The values shown are for $K = 1$ and $\sigma = 1$. (b) Resulting 3×3 kernel [this is the same as Fig. 3.31(b)].



$\frac{1}{4.8976} \times$

0.3679	0.6065	0.3679
0.6065	1.0000	0.6065
0.3679	0.6065	0.3679

Properties of Gaussian Filters

rank=1

1. Separable $\Rightarrow$ $G(x,y) = G_1(x)G_2(y)$
2. Product and convolution of two Gaussians are Gaussian too

TABLE 3.6 Mean and standard deviation of the product ($\times$) and convolution ($\star$) of two 1-D Gaussian functions, f and g. These results generalize directly to the product and convolution of more than two 1-D Gaussian functions (see Problem 3.33).

	f	g	$f \times g$	$f \star g$
Mean	m_f	m_g	$m_{f\times g} = \dfrac{m_f \sigma_g^2 + m_g \sigma_f^2}{\sigma_f^2 + \sigma_g^2}$	$m_{f\star g} = m_f + m_g$
Standard deviation	σ_f	σ_g	$\sigma_{f\times g} = \sqrt{\dfrac{\sigma_f^2 \sigma_g^2}{\sigma_f^2 + \sigma_g^2}}$	$\sigma_{f\star g} = \sqrt{\sigma_f^2 + \sigma_g^2}$

Low-Pass Filtering with a Gaussian Kernel

21x21

43x43

Result comparable to Fig. 3.33(d) with a 21x21 averaging filter

a b c

FIGURE 3.36 (a)A test pattern of size 1024×1024. (b) Result of lowpass filtering the pattern with a Gaussian kernel of size 21×21, with standard deviations $\sigma = 3.5$. (c) Result of using a kernel of size 43×43, with $\sigma = 7$. This result is comparable to Fig. 3.33(d). We used $K = 1$ in all cases.

Gaussian Filtering with Different Kernel Sizes

43x43 和 85x85 差不多.

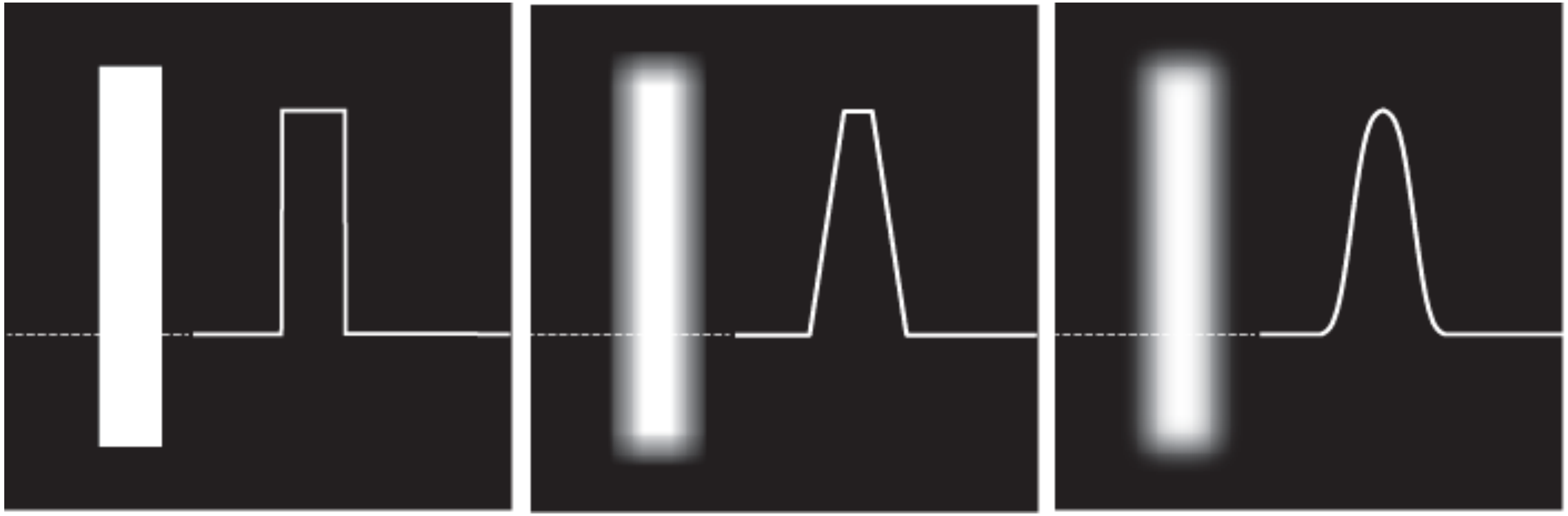
43x43 85x85 Difference

a b c

FIGURE 3.37 (a) Result of filtering Fig. 3.36(a) using a Gaussian kernels of size 43×43, with $\sigma = 7$. (b) Result of using a kernel of 85×85, with the same value of σ. (c) Difference image.

There is little to be gained by using a Gaussian kernel larger than $\lceil 6\sigma \rceil \times \lceil 6\sigma \rceil$.

Comparison of Gaussian and Box Filter Smoothing



71x71 Box

151x151 Gaussian

smoother around edges

a b c

FIGURE 3.38 (a) Image of a white rectangle on a black background, and a horizontal intensity profile along the scan line shown dotted. (b) Result of smoothing this image with a box kernel of size 71×71, and corresponding intensity profile. (c) Result of smoothing the image using a Gaussian kernel of size 151×151, with $K = 1$ and $\sigma = 25$. Note the smoothness of the profile in (c) compared to (b). The image and rectangle are of sizes 1024×1024 and 768×128 pixels, respectively.

Effect of Padding on Gaussian Filtering

要会的!

1024x1024 image

Dark border

Zero padding | Mirror padding | Replicate padding

187x187

a b c

FIGURE 3.39 Result of filtering the test pattern in Fig. 3.36(a) using (a) zero padding, (b) mirror padding, and (c) replicate padding. A Gaussian kernel of size 187 × 187, with $K = 1$ and $\sigma = 31$ was used in all three cases.

Smoothing Performance as a Function of Image and Kernel Size

4096x4096 image

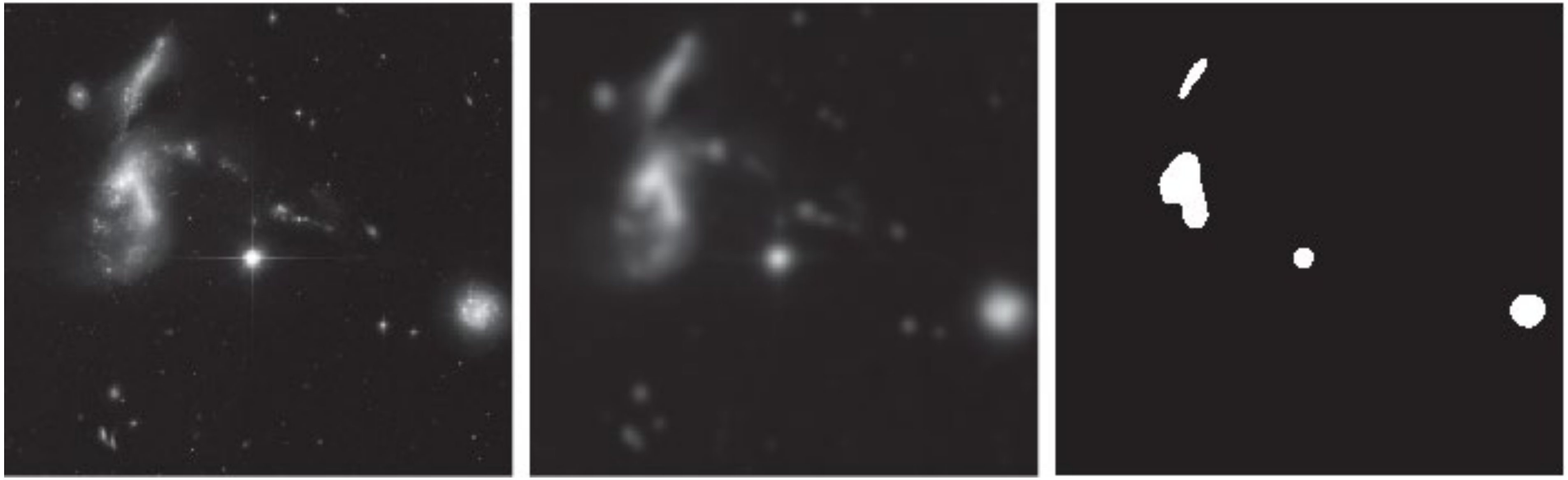
187x187

745x745

a b c

FIGURE 3.40 (a) Test pattern of size 4096 × 4096 pixels. (b) Result of filtering the test pattern with the same Gaussian kernel used in Fig. 3.39. (c) Result of filtering the pattern using a Gaussian kernel of size 745 × 745 elements, with $K = 1$ and $\sigma = 124$. Mirror padding was used throughout.

Using Lowpass Filtering and Thresholding for Region Extraction

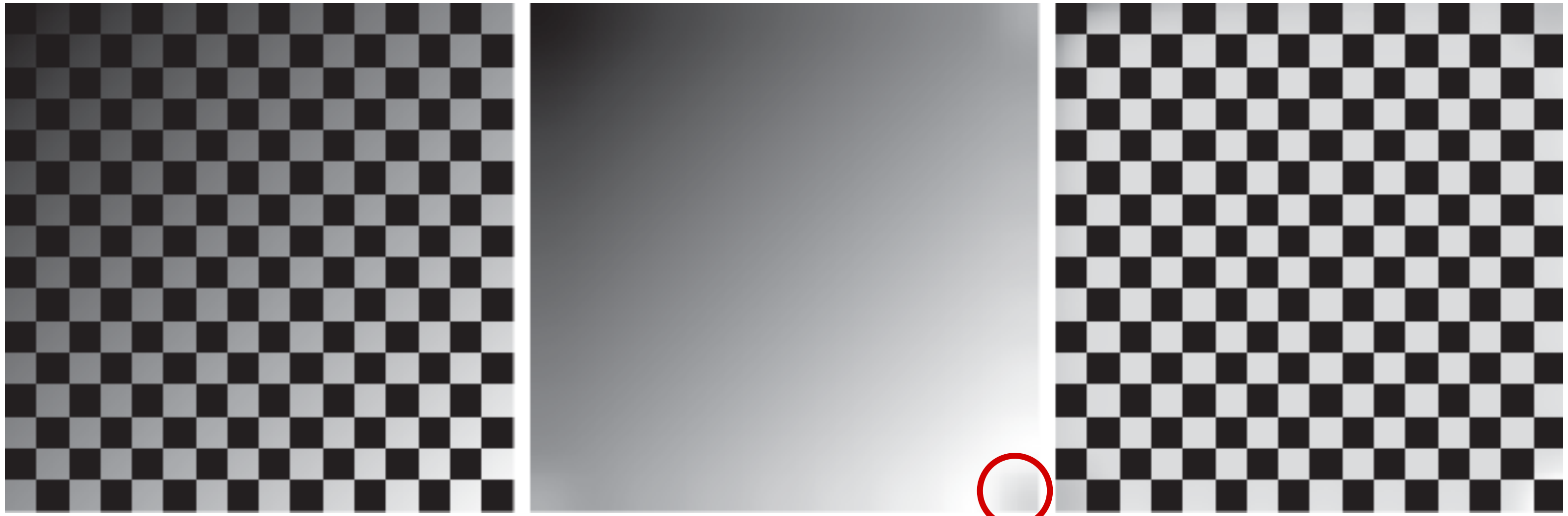


a b c

FIGURE 3.41 (a) A 2566×2758 Hubble Telescope image of the *Hickson Compact Group*. (b) Result of lowpass filtering with a Gaussian kernel. (c) Result of thresholding the filtered image (intensities were scaled to the range [0, 1]). The Hickson Compact Group contains dwarf galaxies that have come together, setting off thousands of new star clusters. (Original image courtesy of NASA.)

Shading Correction Using Lowpass Filtering

- Image: 2048x2048
- Checkerboard square: 128x128
- Gaussian kernel: 512x512, K=1, σ=128
- The kernel is just large enough to blur out the squares.
- A kernel three times the size of the squares is too small to blur them out sufficiently.



mirroring by padding

a b c

FIGURE 3.42 (a) Image shaded by a shading pattern oriented in the −45° direction. (b) Estimate of the shading patterns obtained using lowpass filtering. (c) Result of dividing (a) by (b). (See Section 9.8 for a morphological approach to shading correction).

Order-Statistic Filters

- Nonlinear
- Based on ordering (ranking) the pixels contained in the filter mask
- Replacing the value of the center pixel with the value determined by the ranking result
- For example, median filter, max filter, min filter, etc.

Provides excellent noise reduction with less blurring than linear smoothing filters.

{1, 2, 3, 5, 6, 7, 8}

3	15	6
15	~~20~~	14
5	10	3

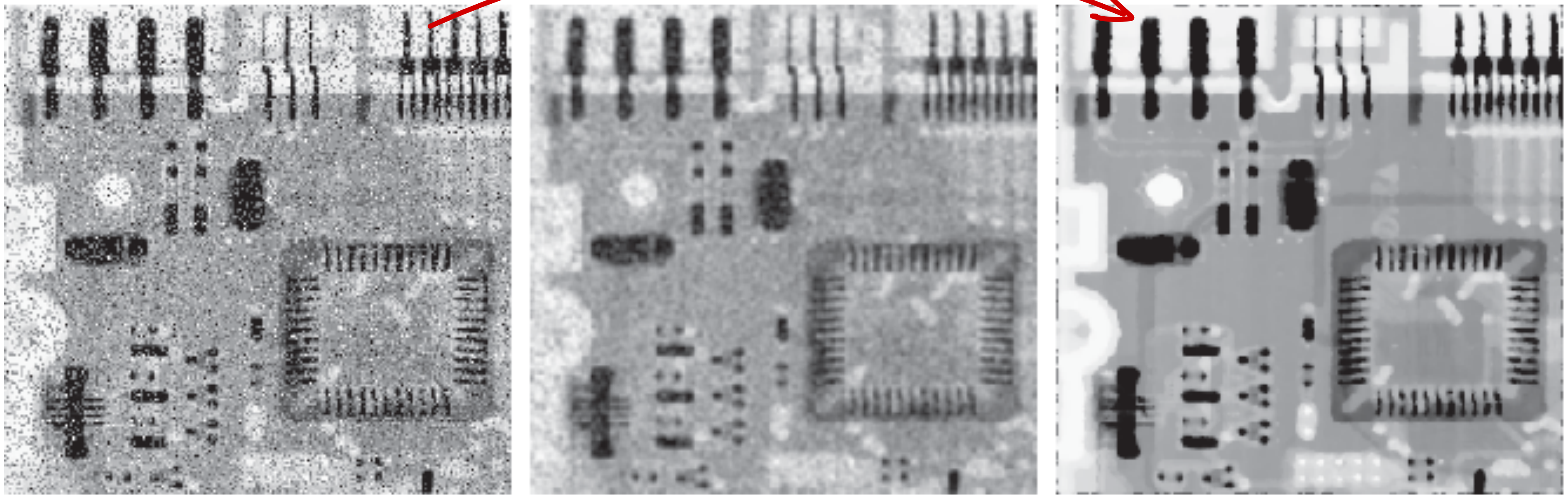
改成 10 (median)

3, 3, 6, 10, 14, 15, 15, 20

median!

Median Filtering for Noise Reduction

WHY



Gaussian

Median

a b c

FIGURE 3.43 (a) X-ray image of a circuit board, corrupted by salt-and-pepper noise. (b) Noise reduction using a 19×19 Gaussian lowpass filter kernel with $\sigma = 3$. (c) Noise reduction using a 7×7 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Sharpening Spatial Filters

- Sharpening highlights transitions in intensity
- Can be accomplished by spatial differentiation, which enhances edges and other discontinuities (such as noise) and de-emphasizes areas with slowly varying intensities.
- Referred to as highpass filtering

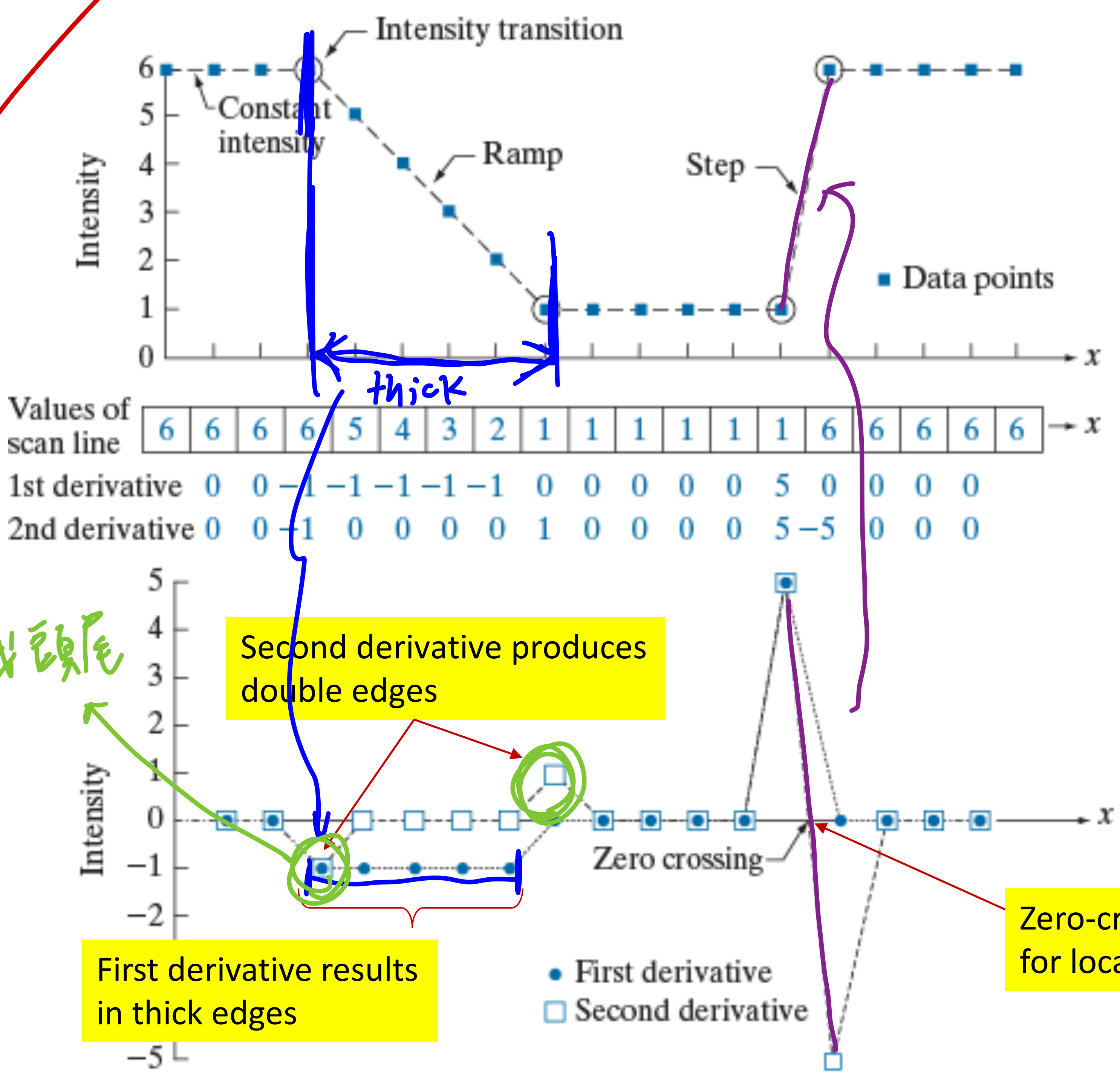
Sharpening Spatial Filters: Foundation

- The first-order derivative of a one-dimensional function f(x) is the difference

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

- The second-order derivative of f(x) as the difference

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



Second derivative produces double edges

First derivative results in thick edges

Zero-crossing is useful for locating edges

a
b
c

FIGURE 3.44
(a) A section of a horizontal scan line from an image, showing ramp and step edges, as well as constant segments.
(b) Values of the scan line and its derivatives.
(c) Plot of the derivatives, showing a zero crossing. In (a) and (c) points were joined by dashed lines as a visual aid.

Sharpening Spatial Filters: Laplace Operator

Laplacian for an image f(x,y):

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

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

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

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

Sharpening Spatial Filters: Laplace Operator

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

a b c d

FIGURE 3.45 (a) Laplacian kernel used to implement Eq. (3-53). (b) Kernel used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other Laplacian kernels.

Sharpening Spatial Filters: Laplace Operator

Image sharpening by adding the Laplacian image to the original:

$$g(x, y) = f(x, y) + c\left[\nabla^2 f(x, y)\right]$$

difference between p84

都是sharpen

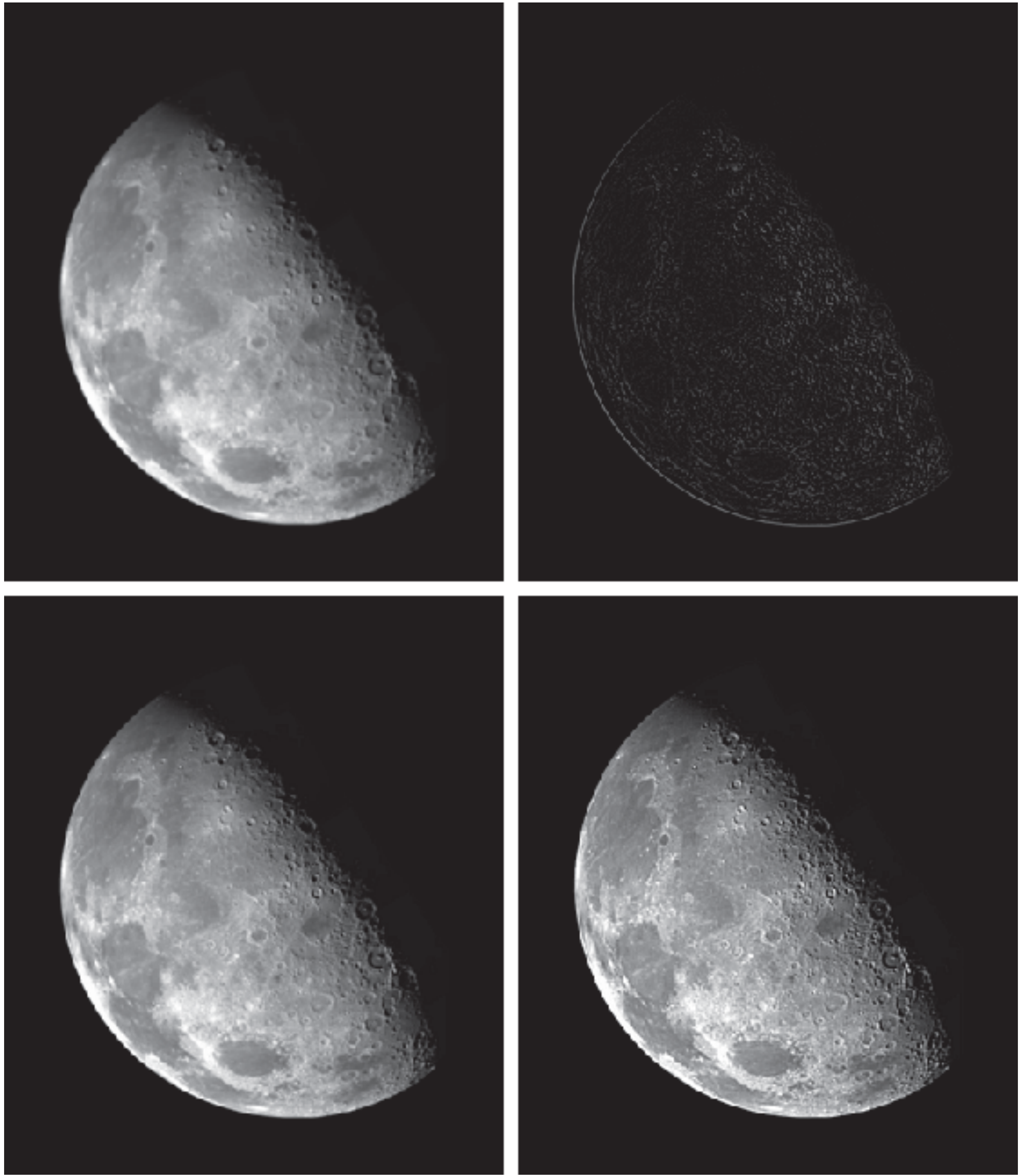
$f(x, y)$: input image

$g(x, y)$: sharpenend images

$$c = \begin{cases} -1 & \text{if } \nabla^2 f(x, y) \text{ corresponding to Fig. 3.45(a) or (b)} \\ 1 & \text{if } \nabla^2 f(x, y) \text{ corresponding to Fig. 3.45(c) or (d)} \end{cases}$$

→ 其2次微分

Image Sharpening Using the Laplacian



a	b
c	d

FIGURE 3.46
(a) Blurred image of the North Pole of the moon.
(b) Laplacian image obtained using the kernel in Fig. 3.45(a).
(c) Image sharpened using Eq. (3-54) with $c = -1$.
(d) Image sharpened using the same procedure, but with the kernel in Fig. 3.45(b).
(Original image courtesy of NASA.)

Image Sharpening Using the Laplacian

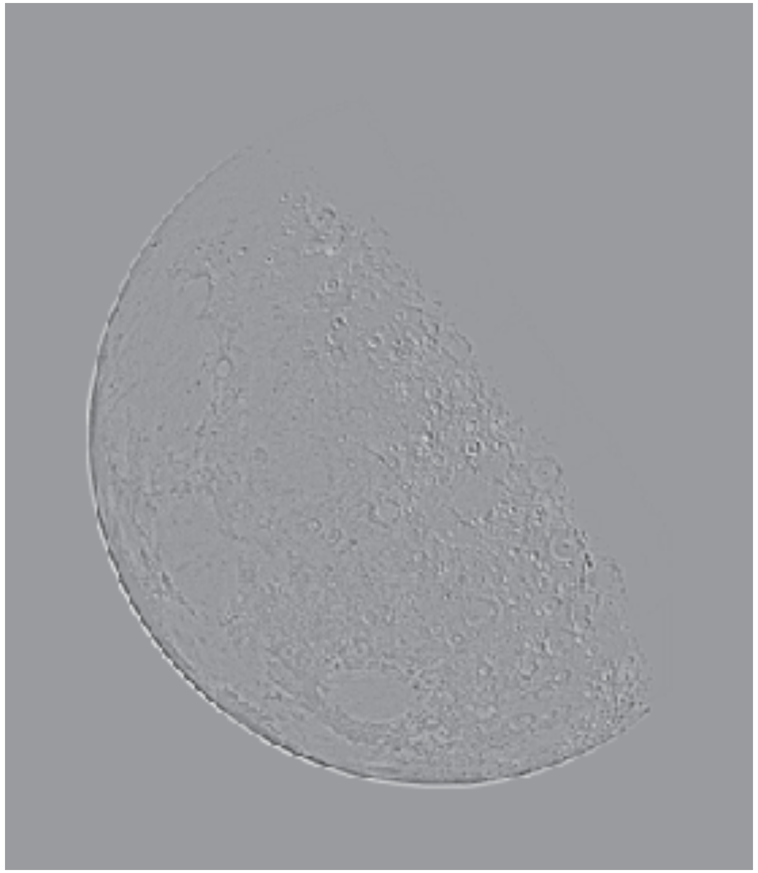


FIGURE 3.47 The Laplacian image from Fig. 3.46(b), scaled to the full [0, 255] range of intensity values. Black pixels correspond to the most negative value in the unscaled Laplacian image, grays are intermediate values, and white pixels corresponds to the highest positive value.

Unsharp Masking and Highboost Filtering

- **Unsharp masking**
  - Subtracting an unsharp (smoothed) version of an image from the original image
  - Often used in printing and publishing industry

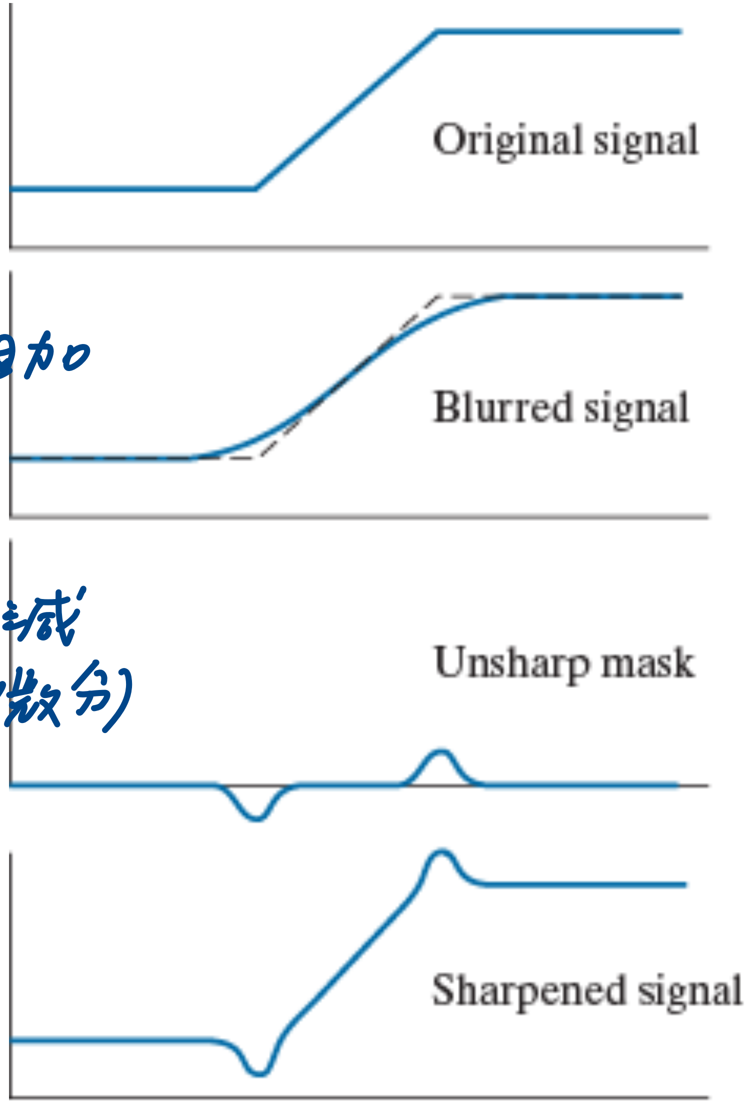
- **Steps**

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

⇒ 共一次微分

相加

相減
(有微分)



Unsharp Masking and Highboost Filtering

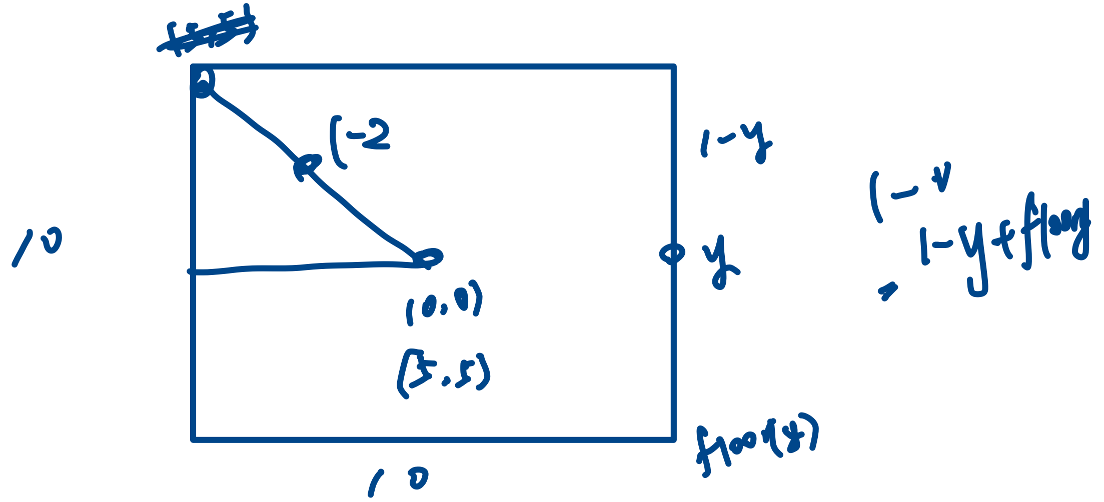
First, obtain an unsharp mask by substracting the blurred image $\bar{f}(x, y)$ from the original

$$g_{mask}(x, y) = f(x, y) - \bar{f}(x, y)$$

Then add a weighted portion of the mask back to the original

$$g(x, y) = f(x, y) + k * g_{mask}(x, y) \qquad k \geq 0$$

when $k > 1$, the process is referred to as "highboost filtering."



Unsharp Masking and Highboost Filtering

a b c
d e

FIGURE 3.49 (a) Original image of size 600×259 pixels. (b) Image blurred using a 31×31 Gaussian lowpass filter with $\sigma = 5$. (c) Mask. (d) Result of unsharp masking using Eq. (3-56) with $k = 1$. (e) Result of highboost filtering with $k = 4.5$.

Image Sharpening by First-Order Derivatives

For function $f(x, y)$, the gradient of f at coordinates (x, y) is defined as

$$\nabla f \equiv \text{grad}(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

The *magnitude* of vector ∇f, denoted by $M(x, y)$, is computed by

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

Gradient Image

Image Sharpening by First-Order Derivatives

In some implementations, it is more suitable computationally to approximate the squares and square root operations by absolute values

简化

$$M(x,y) \approx |g_x| + |g_y|$$

$$M(x,y) = |z_8 - z_5| + |z_6 - z_5|$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Roberts Cross-gradient Operators

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

Sobel Operators

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

Image Sharpening Based on First-Order Derivatives

a
b c
d e

FIGURE 3.50 (a) A 3×3 region of an image, where the zs are intensity values. (b)–(c) Roberts cross-gradient operators. (d)–(e) Sobel operators. All the kernel 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

Roberts cross-gradient operators

−1	−2	−1
0	0	0
1	2	1

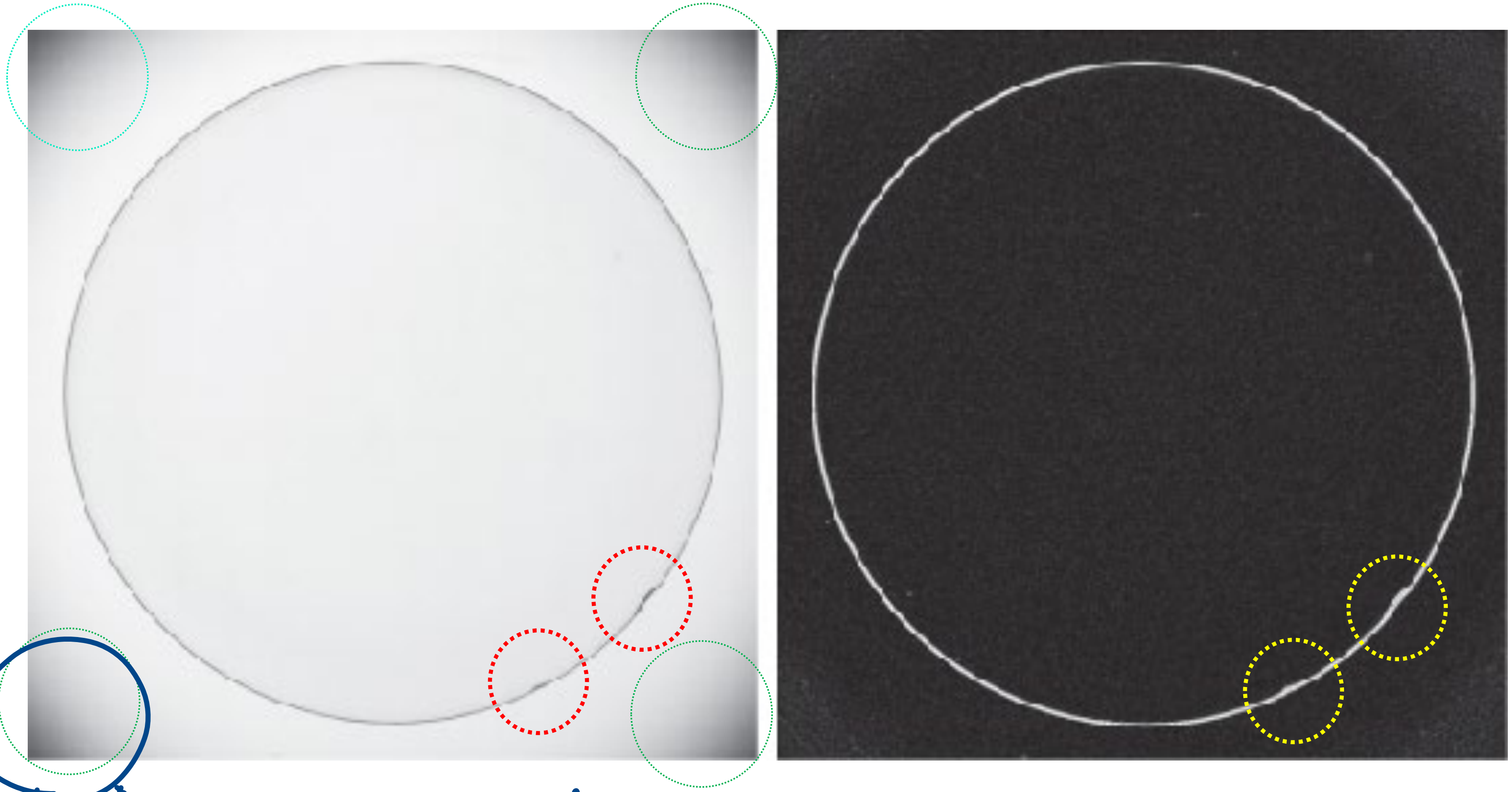
−1	0	1
−2	0	2
−1	0	1

Sobel operators

Using the Gradient for Edge Enhancement

a b

FIGURE 3.51 (a) Image of a contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Perceptics Corporation.)

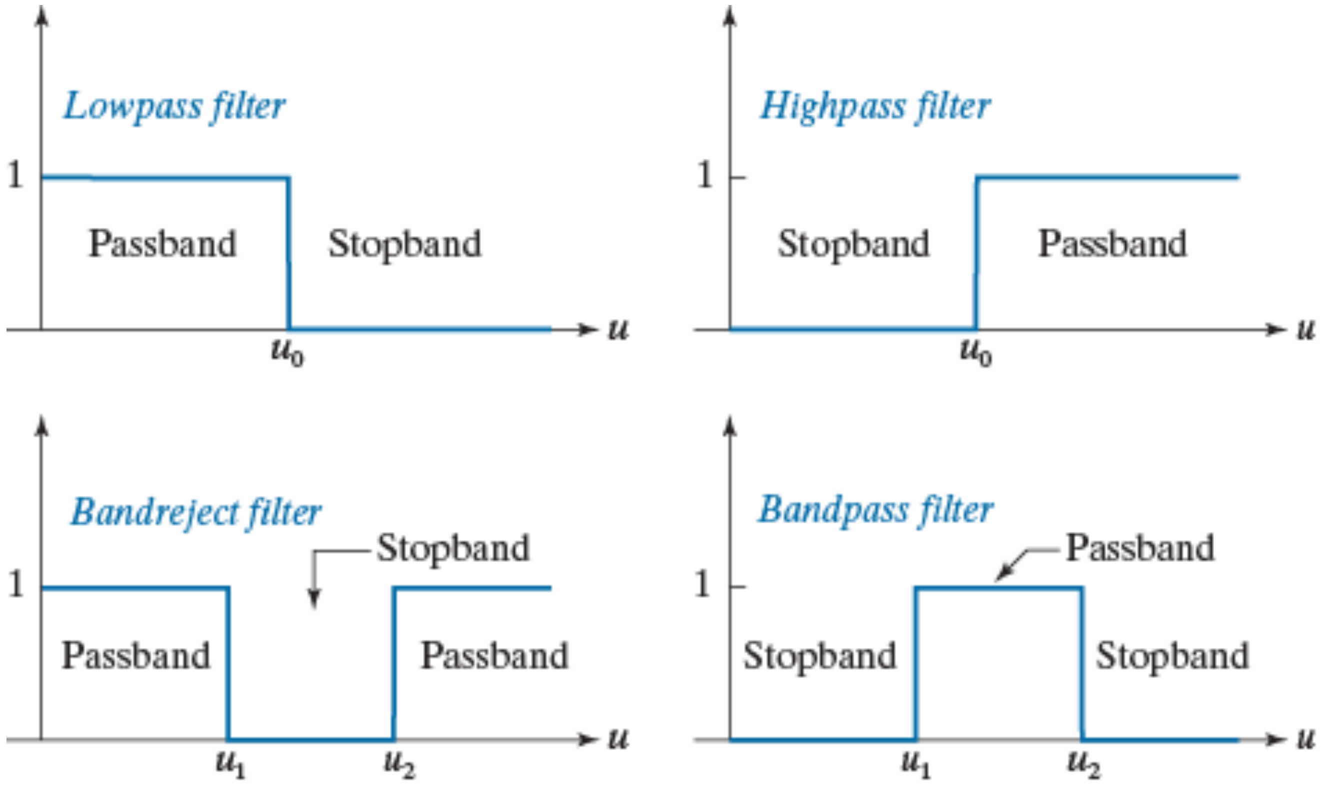


Optical vignetting

But 實際上 enhancement ≠ sharpening

enhancement 主要處理 edge, sharpen 是讓整個圖銳利

Highpass, Bandreject, and Bandpass Filters from Lowpass Filter



a b
c d

FIGURE 3.52 Transfer functions of ideal 1-D filters in the frequency domain (u denotes frequency).
(a) Lowpass filter.
(b) Highpass filter.
(c) Bandreject filter.
(d) Bandpass filter.
(As before, we show only positive frequencies for simplicity.)

Spatial Kernels Derived from Lowpass Kernel

Lowpass $\quad f_{lp}(x,y)$

Highpass $\quad f_{hp}(x,y)=\delta(x,y)-f_{lp}(x,y)$

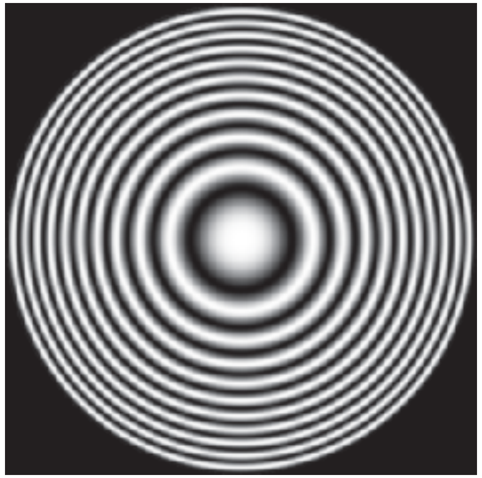
Bandreject $\quad f_{br}(x,y)=f_{lp}(x,y)+f_{hp}(x,y)=f_{lp}(x,y)+\left[\delta(x,y)-f_{lp}(x,y)\right]$

Bandpass $\quad f_{bp}(x,y)=\delta(x,y)-f_{br}(x,y)=\delta(x,y)-\left[f_{lp}(x,y)+\left[\delta(x,y)-f_{lp}(x,y)\right]\right]$

找細節，給img，求做了甚麼？

Zone Plate Filtered with Lowpass Kernels

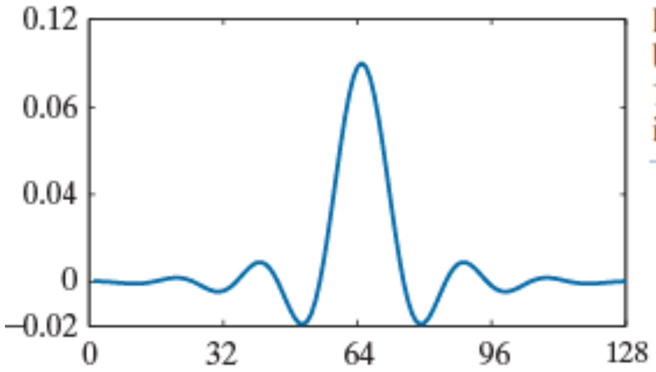
FIGURE 3.53 A zone plate image of size 597 × 597 pixels.



$$z(x,y)=\frac{1}{2}[1+\cos(x^2+y^2)]$$

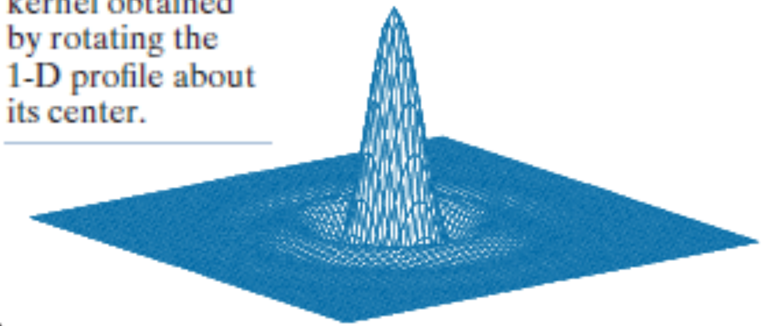
$$x,y:[-8.2,\ 8.2]$$

$$\Delta:0.0275$$



a b

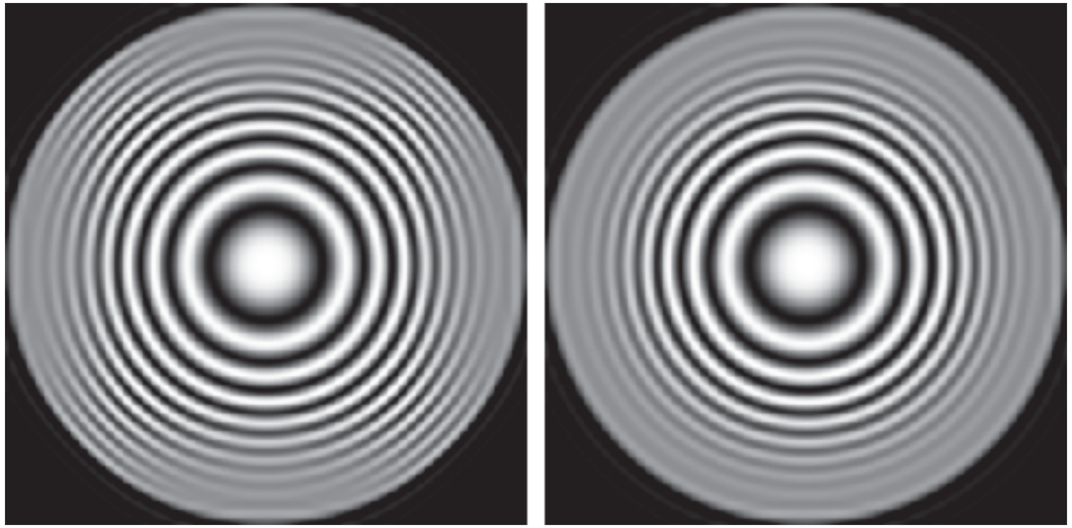
FIGURE 3.54 (a) A 1-D spatial lowpass filter function. (b) 2-D kernel obtained by rotating the 1-D profile about its center.



a b

FIGURE 3.55 (a) Zone plate image filtered with a separable lowpass kernel. (b) Image filtered with the isotropic lowpass kernel in Fig. 3.54(b).

WHY different?



Highpassed, Bandrejected, and Bandpassed Zone Plate

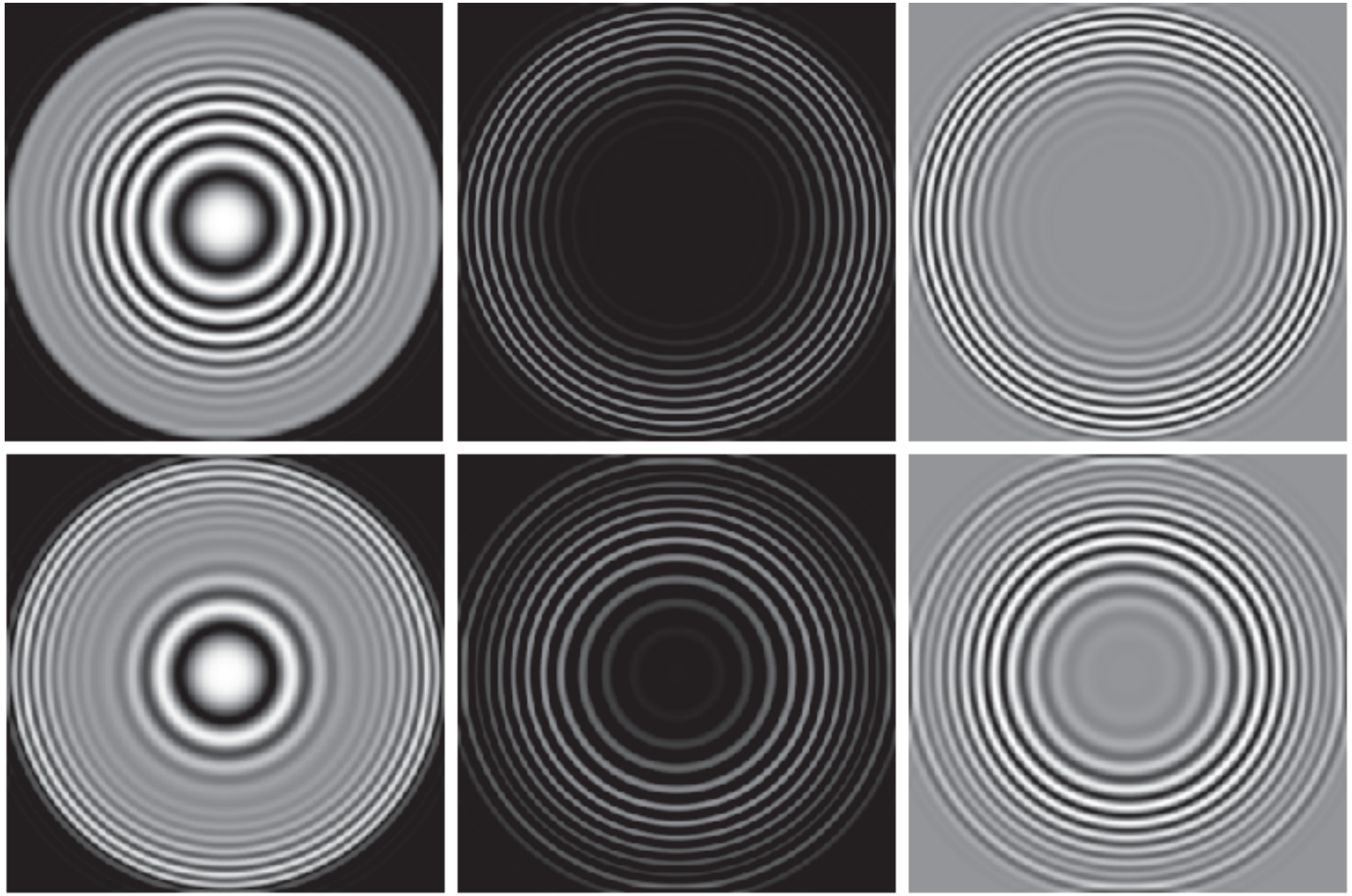
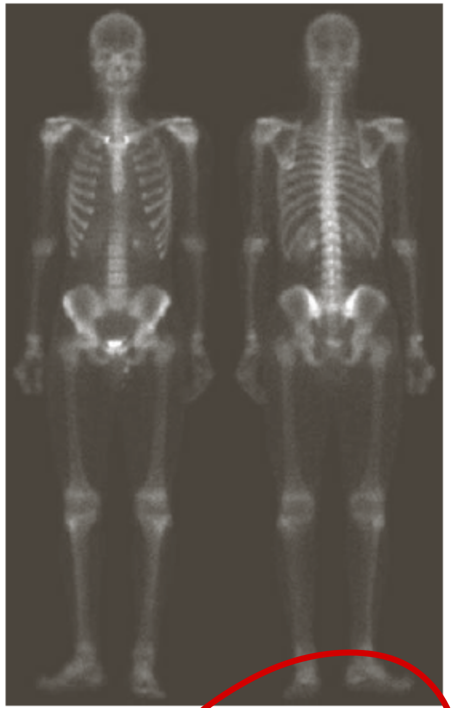


FIGURE 3.56
Spatial filtering of the zone plate image. (a) Lowpass result; this is the same as Fig. 3.55(b). (b) Highpass result. (c) Image (b) with intensities scaled. (d) Bandreject result. (e) Bandpass result. (f) Image (e) with intensities scaled.

Combination of Techniques

Goal:
Bring out more skeletal details by image enhancement

Challenges:
Small dynamic range
Noisy



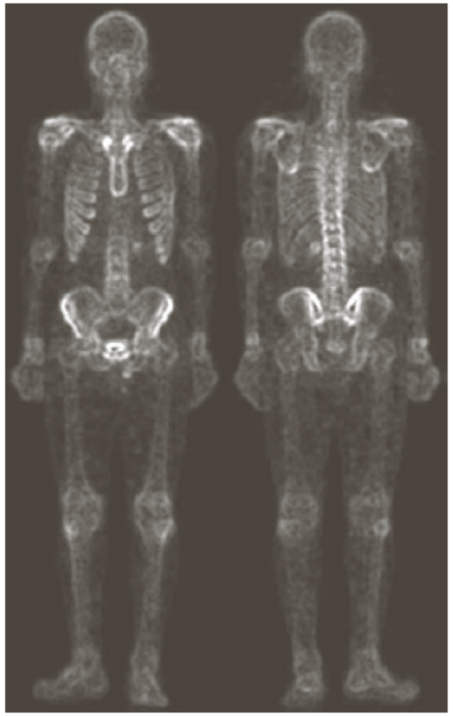
p 193
p195

WHY 相乘?

a	b
c	d

FIGURE 3.57
(a) Image of whole body bone scan.
(b) Laplacian of (a).
(c) Sharpened image obtained by adding (a) and (b).
(d) Sobel gradient of image (a). (Original image courtesy of G.E. Medical Systems.)

Combination of Techniques



e f
g h

FIGURE 3.57 *(Continued)* (e) Sobel image smoothed with a 5×5 box filter. (f) Mask image formed by the product of (b) and (e). (g) Sharpened image obtained by the adding images (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare images (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

Histogram processing is not a good choice for images with histograms characterized by dark and bright components.

Study p. 194 of the textbook.

目標: noise, smooth dynamic range.