Chapter 3: Image Enhancement in Spatial Domain

Chapter Overview:

- ✓ To briefly introduce the concept of image enhancement
- ✓ To introduce the fundamental techniques in image enhancement in spatial domain
- ✓ To discuss these techniques in real world applications

Principle Objective of Enhancement

- Process an image so that the result will be more suitable than the original image for a specific application.
- The suitableness is up to each application.
- A method which is quite useful for enhancing an image may not necessarily be the best approach for enhancing another images.
- Based on visual evaluation of image quality, thus it is a highly subjective process. – to compare one enhancement method from the other.

2 Domains

- Spatial domain : (image plane)
  - Techniques are based on direct manipulation of pixels in an image
- Frequency Domain :
  - Techniques are based on modifying the Fourier transform of an image.
- There are some enhancement techniques based on various combinations of methods from these two categories.

“Good” Quality of Images

- For human visual
  - The visual evaluation of image quality is a highly subjective process.
  - It is hard to standardize the defination of a good image.
- For machine perception
  - The evaluation task is easier.
  - A good image is one which gives the best machine recognition results.
- A certain amount of trial and error usually is required before a particular image enhancement approach is selected.

Image Enhancement in the Spatial Domain

•Procedures that operate directly on pixels.

$$g(x,y) = \mathrm{T}\,[f(x,y)]$$

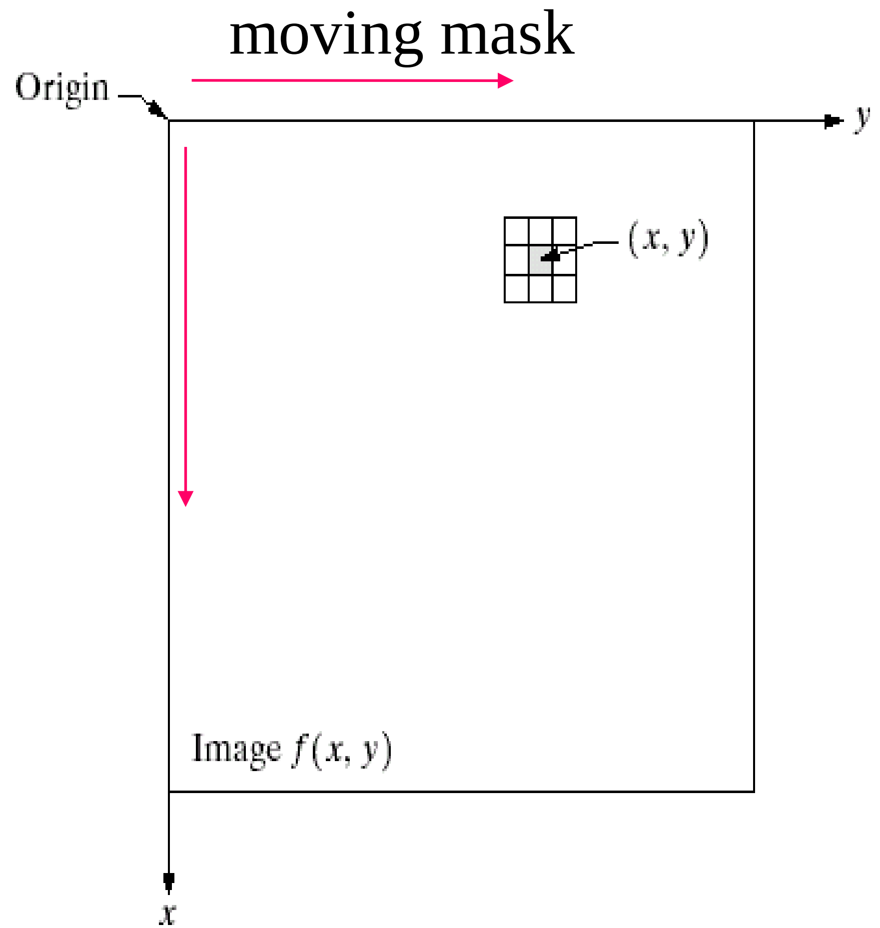
where

- $f(x,y)$ is the input image
- $g(x,y)$ is the processed image
- T is an operator on f defined over some neighborhood of (x,y).

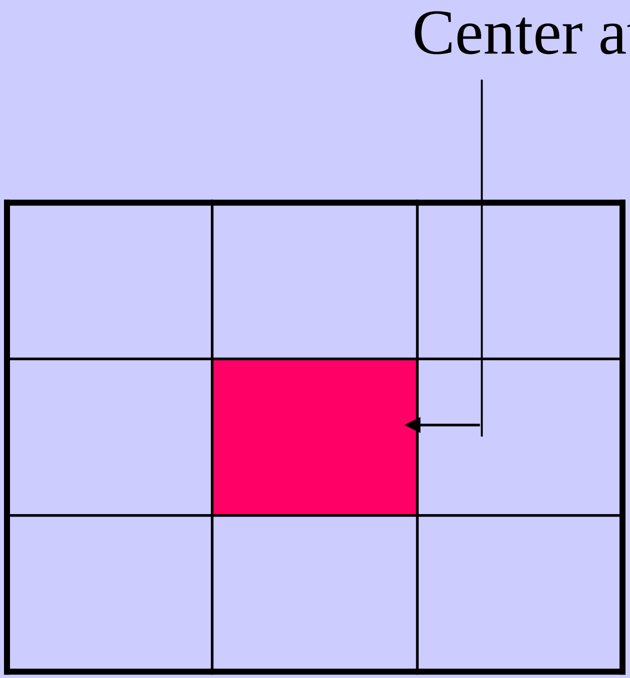
T can operate on a set of images, such as performing the pixel-by-pixel sum of *K* images for noise reduction (will be discussed in detailed in later section)

Image Enhancement in the Spatial Domain

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



Mask/Filter



- Neighborhood of a point (x,y)can be defined by using a square/rectangular (commonly used) or circular subimage area centered at (x,y)

- The center of the subimage is moved from pixel to pixel starting at the top left corner.

Point Processing

- Simplest form of T, when mask = 1 x 1 pixel (single pixel)
- *g* depends on only the value of *f* at (*x*,*y*)
- Thus, *T* = gray level (or *intensity* or *mapping*) transformation function

$$s = T(r)$$

where

r = gray level of ***f(x,y)***

s = gray level of ***g(x,y)***

Gray Level Transformation Function for Contrast Enhancement

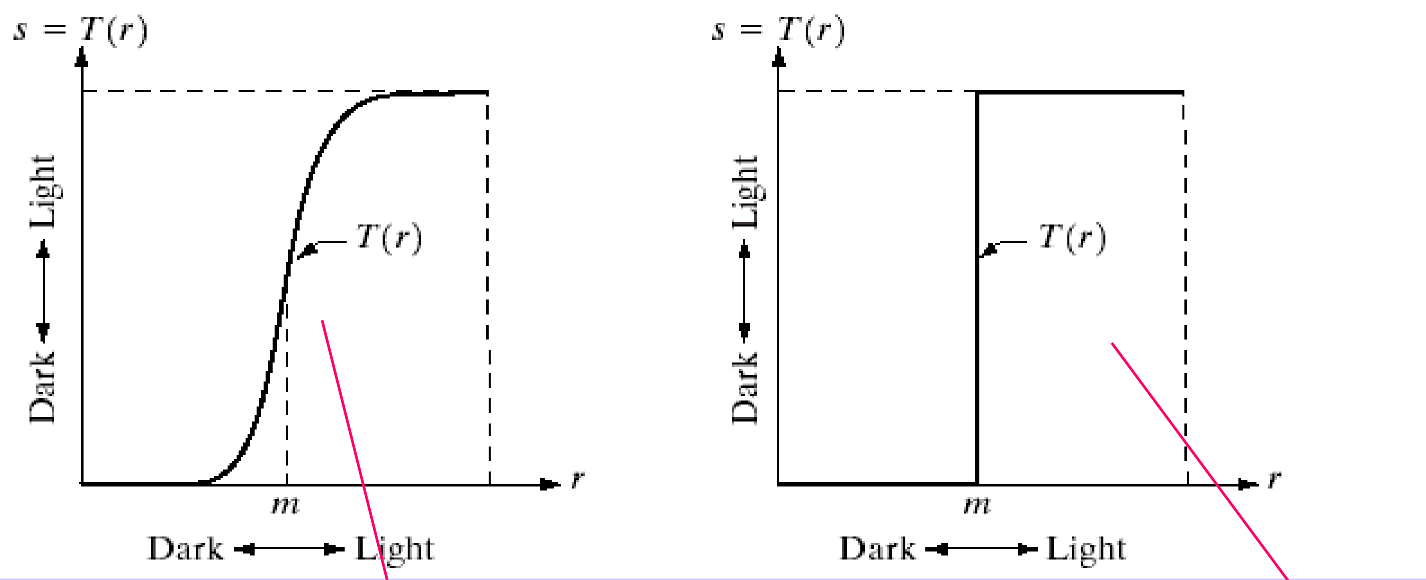


FIGURE 3.2 Gray-level transformation functions for contrast enhancement.

Contrast Stretching

- Producer higher contrast than the original by
  - Darkening the levels below *m* in the original image.
  - Brightening the levels above *m* in the original image.

Thresholding

- Produce a two-level (binary) image

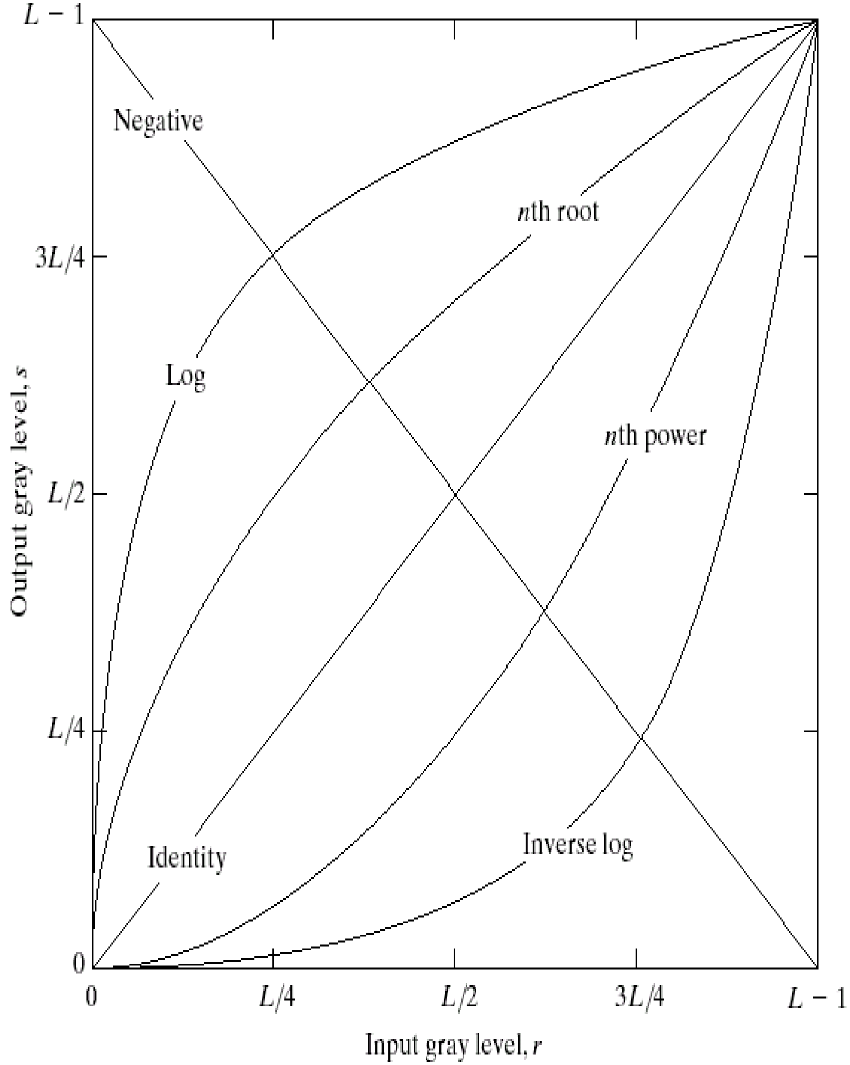
Mask Processing or Filter

- Neighborhood is bigger than 1 x 1 pixel – to provide more flexibility, say 3 x 3 mask
- Use a function of the values of *f* in a predefined neighborhood of (*x*,*y*) to determine the value of *g* at (*x*,*y*)
- This technique is known as mask (filter, kernel, template, or window)
- The value of the mask coefficients determine the nature of the process
- Used for:
  - Image Sharpening
  - Image Smoothing

Basic Gray Level Transformation Functions

- **Linear function**
- **Negative and identity**
- **Logarithm function**
    - Log and inverse-log transformation
- **Power-law function**
    - *n*th power and *n*th root transformations

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



Identity Function

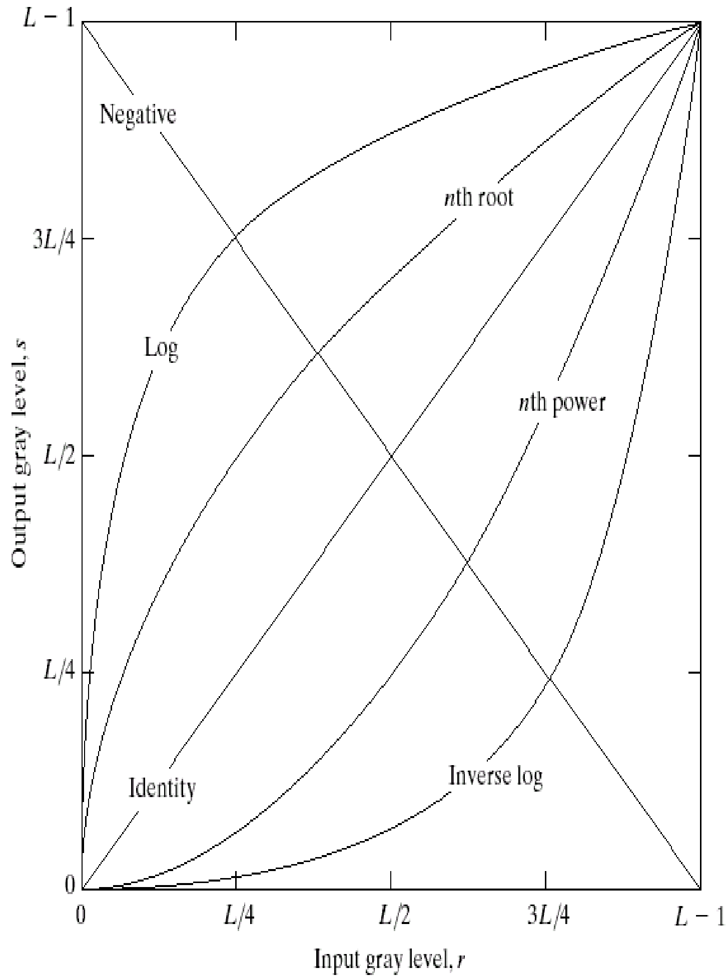


FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.

- Output intensities are identical to input intensities.
- Included in the graph only for completeness

Negative Function

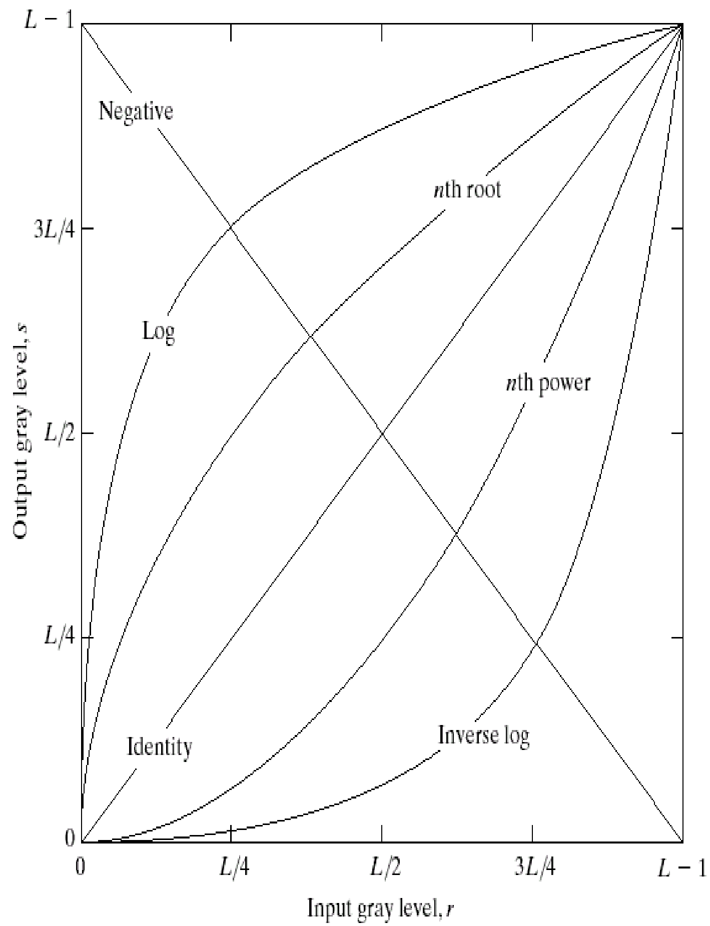


FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.

- An image with gray level in the range *[0, L-1]* where $L=2^n$ and $n = 1, 2, 3, \ldots, m$.

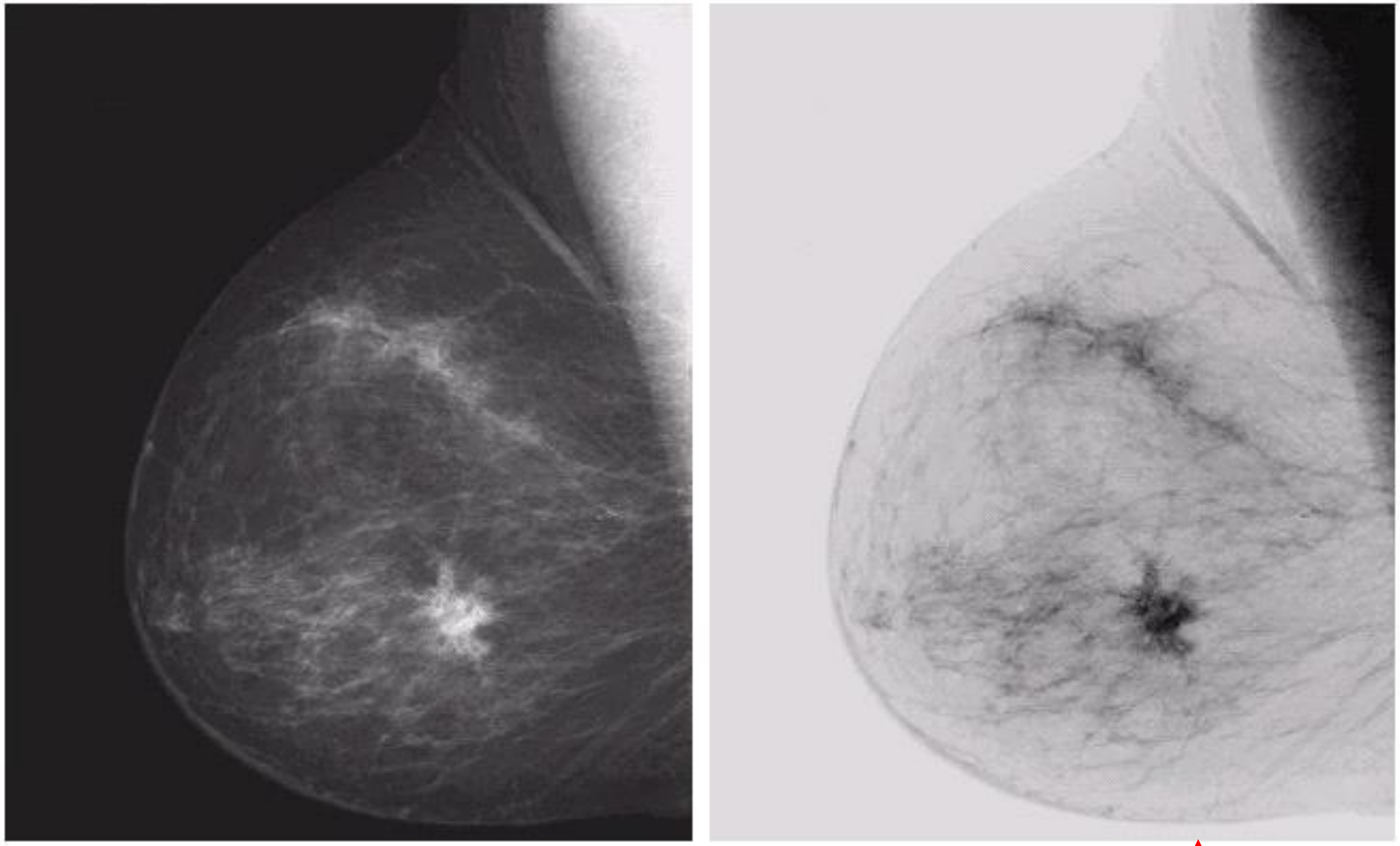
- Negative transformation :

$$s = L - 1 - r$$

- Reversing the intensity levels of an image.

- Suitable for enhancing white or gray detail embedded in dark regions of an image, especially when the black area dominant in size.

Example of Negative Transformation



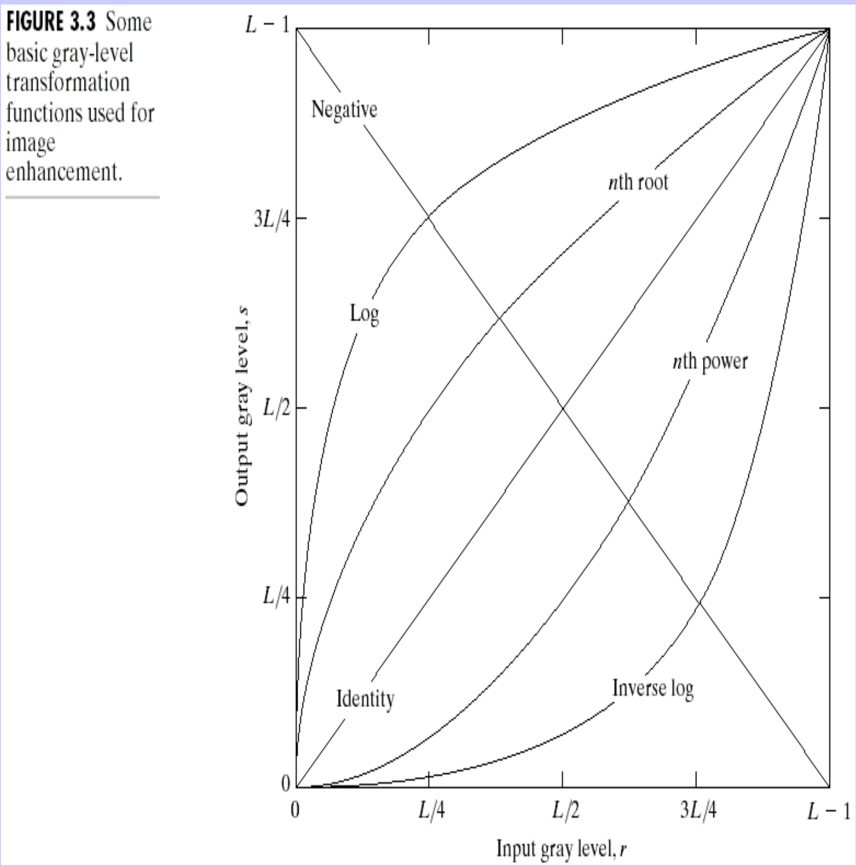
a b

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.)

Original mammogram showing a small lesion of breast

Negative image: gives a better vision to analyze the image

Log Transformations



- $s = c \log(1+r)$
- c is a constant and r is assumed to be >= 0
- Log curve maps a narrow range of low gray-level values in the input image into a wider range of output levels.
- Used to expand the values of dark pixels in an image while compressing the higher levels values.

Log Transformations

- It compresses the dynamic range of images with large variations in pixel values
- Example of image with dynamic range: Fourier spectrum image.
- It can have intensity range from 0 to 10^6 or higher.
- We can’t see the significant degree of detail as it will be lost in the display.

Example of Log Transformation

a b

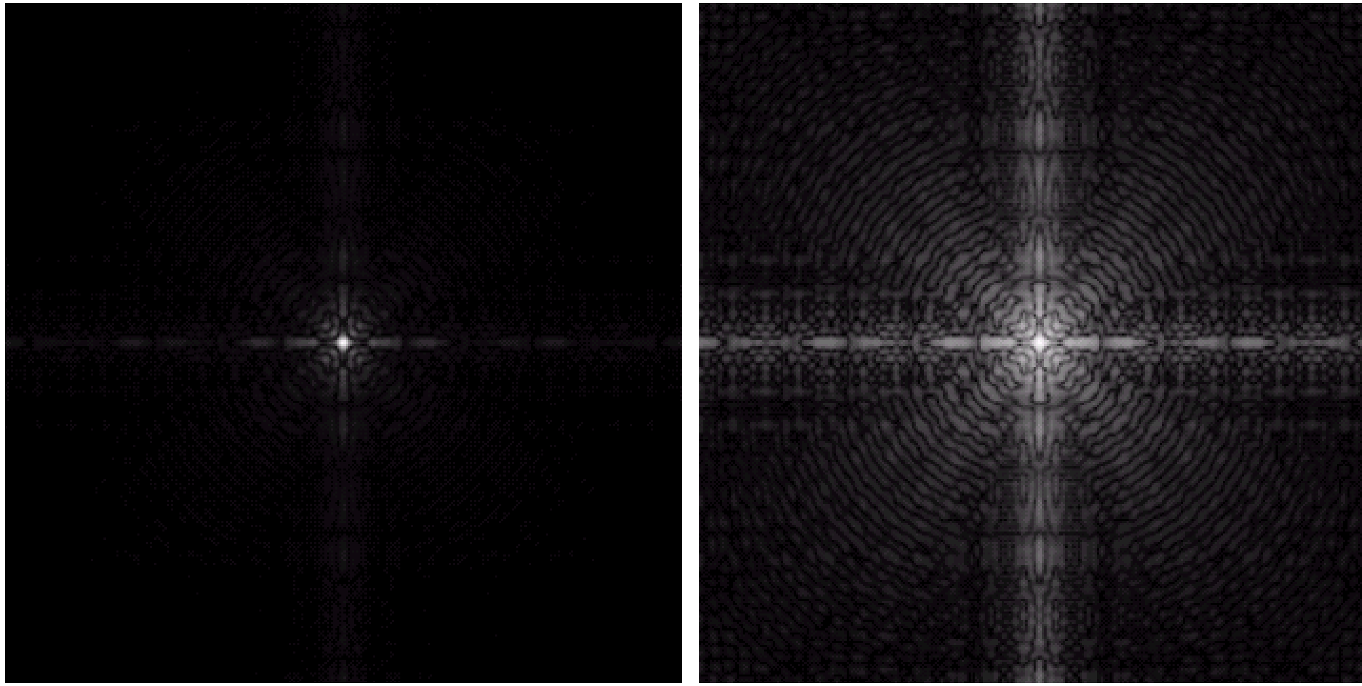


FIGURE 3.5 (a) Fourier spectrum. (b) Result of applying the log transformation given in Eq. (3.2-2) with $c = 1$.

Fourier Spectrum with range = 0 to 1.5×10^6

Result after apply the log transformation with c=1, Range = 0 to 6.2

Inverse Logarithm Transformations

- Do opposite to the Log Transformations
- Used to expand the values of high pixels in an image while compressing the darker-level values.

Power-Law Transformations

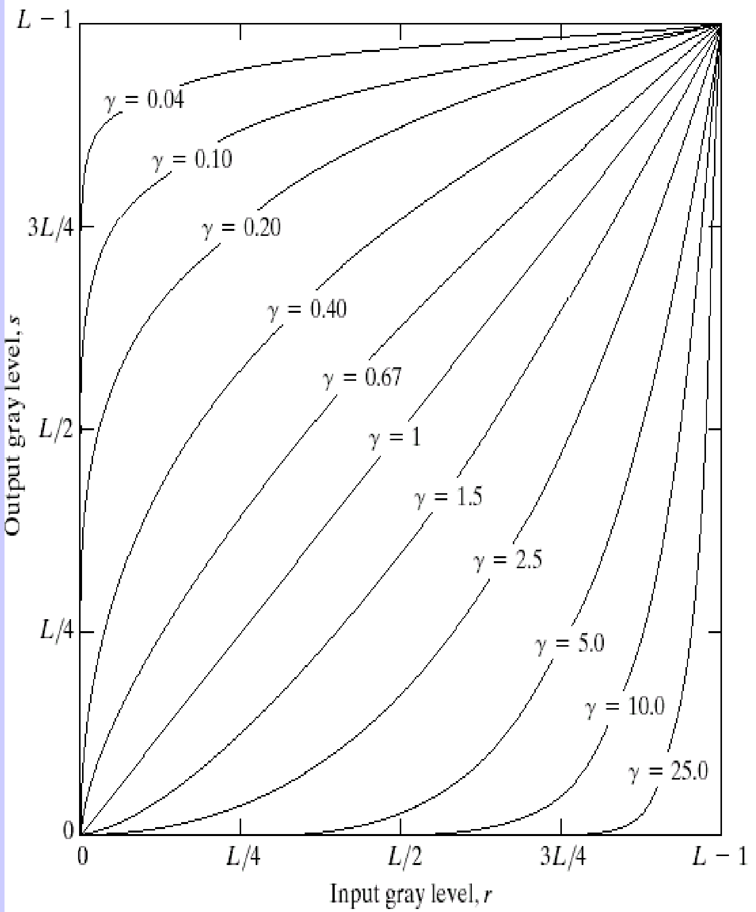
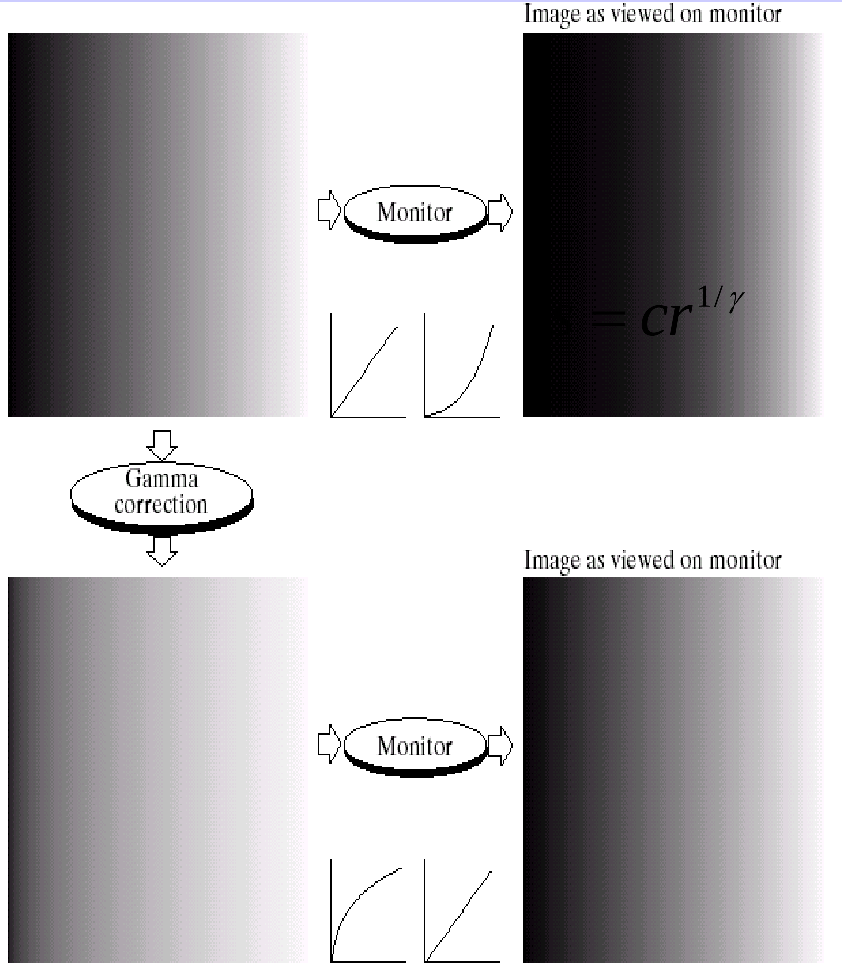


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

- $s = c\, r^{\gamma}$
- c and γ are positive constants
- Power-law curves with fractional values of y map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher values of input levels.
- $c = y = 1$ → Identity function

Gamma Correction

The process to correct the image using power law transformation is called Gamma correction



a b
c d

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

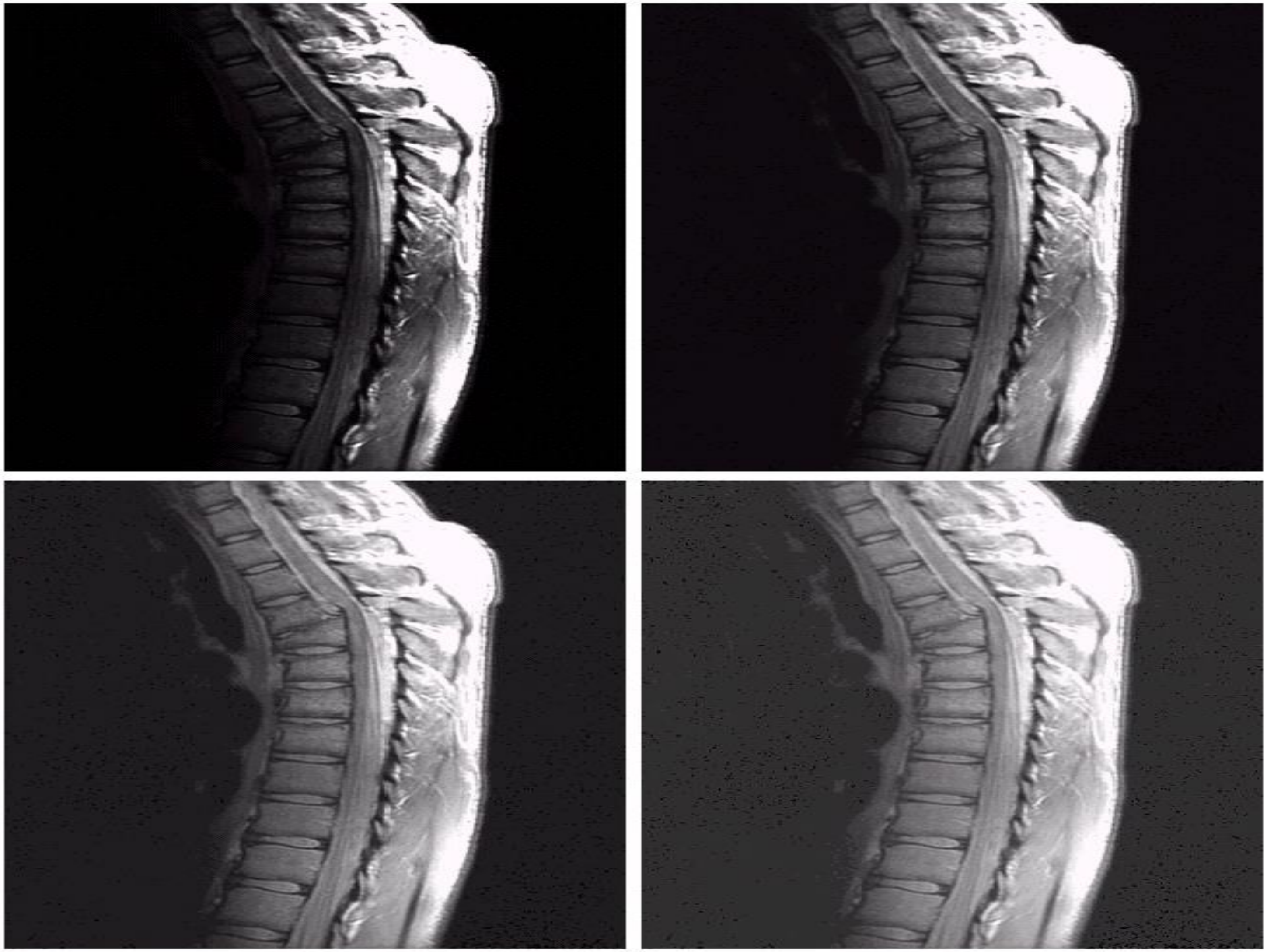
- Cathode ray tube (CRT) devices have an intensity-to-voltage response that is a power function, with γ varying from 1.8 to 2.5

- The picture will become darker

- Gamma correction is done by preprocessing the image before inputting it to the monitor with

$$s = cr^{1/\gamma}$$

Another example of contrast enhancement using power transformation: MRI Image



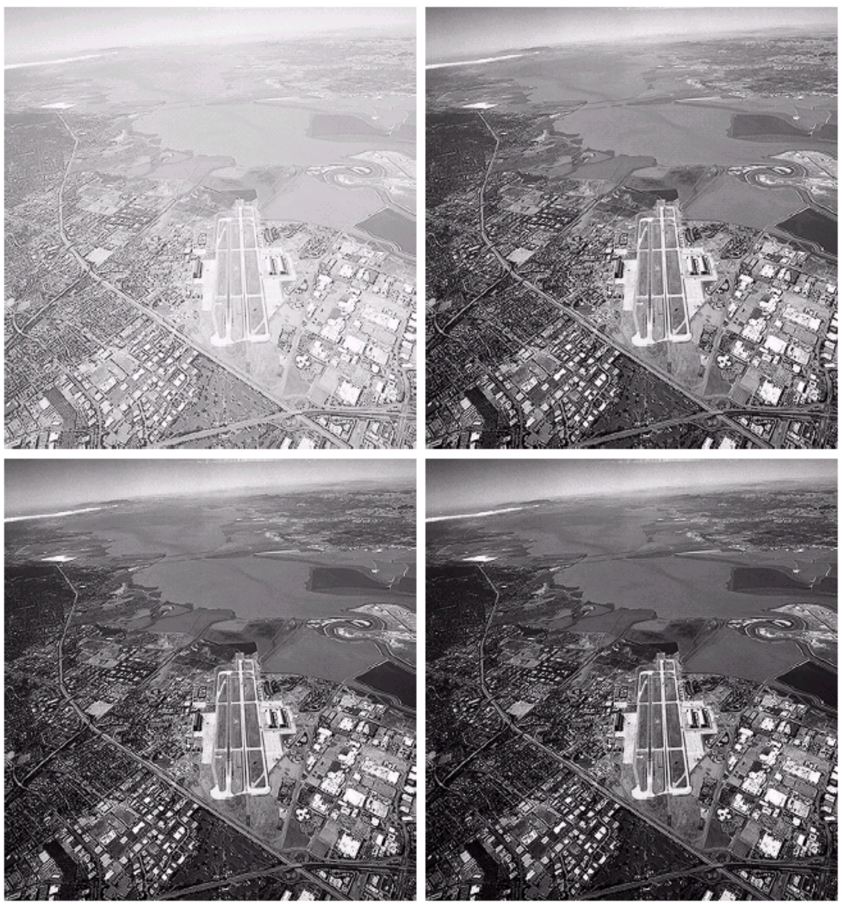
a b
c d

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.)

The Effect of Decreasing Gamma Value

- When the γ is reduced too much, the image begins to reduce contrast to the point where the images started to have very slight "wash-out" look, especially in the background.

Remote Sensing Example under Power Law Transformation



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.)

(a) Image has a washed-out appearance, it needs a compression of gray levels
$\longrightarrow$ needs $\gamma > 1$

(b) result after power-law transformation with $\gamma = 3.0$ (suitable)

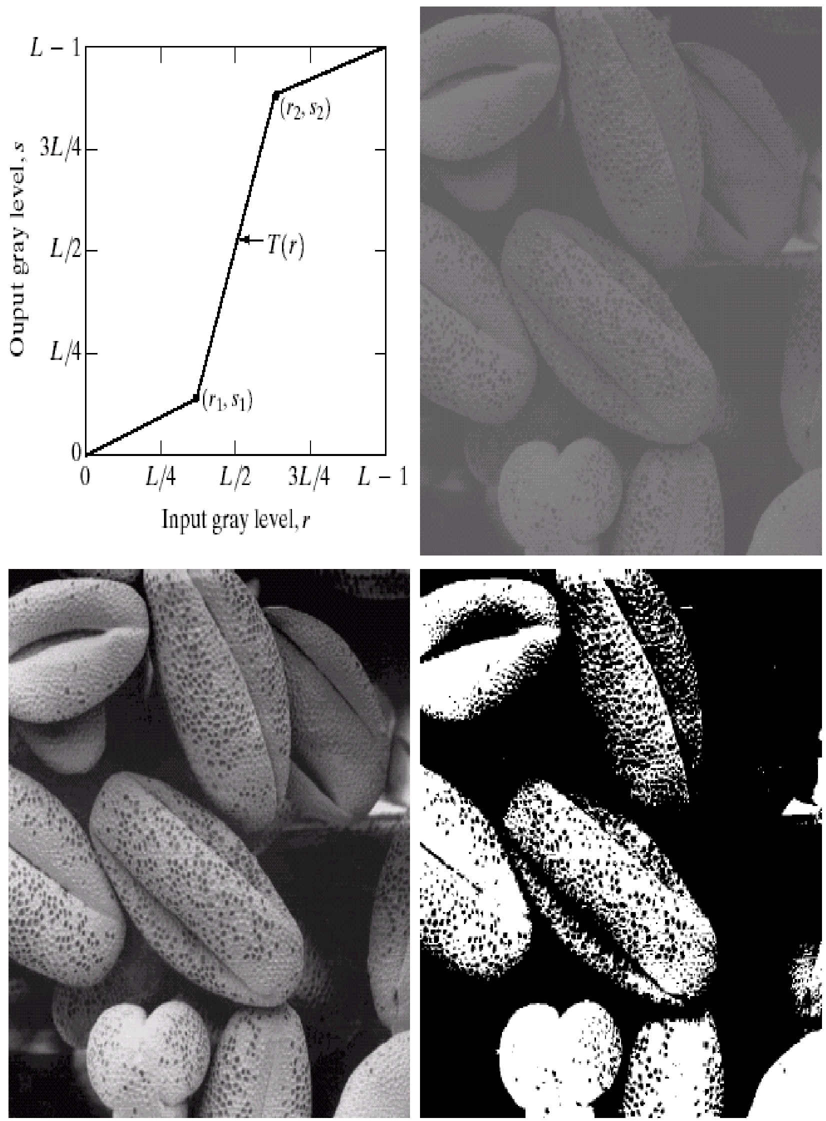
(c) transformation with $\gamma = 4.0$ (suitable)

(d) transformation with $\gamma = 5.0$ (high contrast, the image has areas that are too dark, some details is lost).

Piecewise-Linear Transformation Functions

- A complementary method to the previous methods
- Advantage:
  - The form of piecewise functions can be arbitrarily complex – can be formulated using a piecewise function
- Disadvantage:
  - Their specification requires considerably more user input

Contrast Stretching

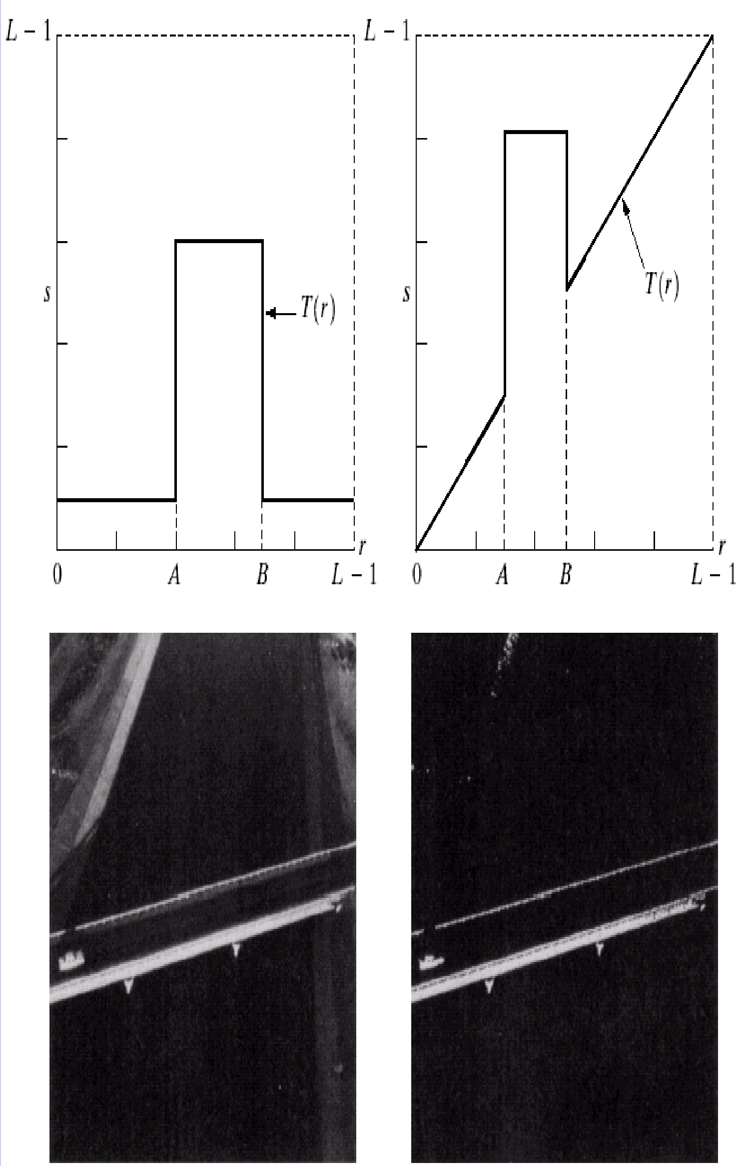


a b
c d

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.)

- one of the simplest piecewise linear function is contrast stretching method
- increase the dynamic range of the gray levels in the image
- (b) a low-contrast image : result from poor illumination, lack of dynamic range in the imaging sensor, or even wrong setting of a lens aperture of image acquisition
- (c) result of contrast stretching: $(r_1,s_1)=(r_{min},0)$ and $(r_2,s_2) = (r_{max},L-1)$
- (d) result of thresholding

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).

- Highlighting a specific range of gray levels in an image.
  - Display a high value of all gray levels in the range of interest and a low value for all other gray levels

- (a) transformation highlights range [A,B] of gray level and reduces all others to a constant level.

- (b) transformation highlights range [A,B] but preserves all other levels.

Bit-plane Slicing

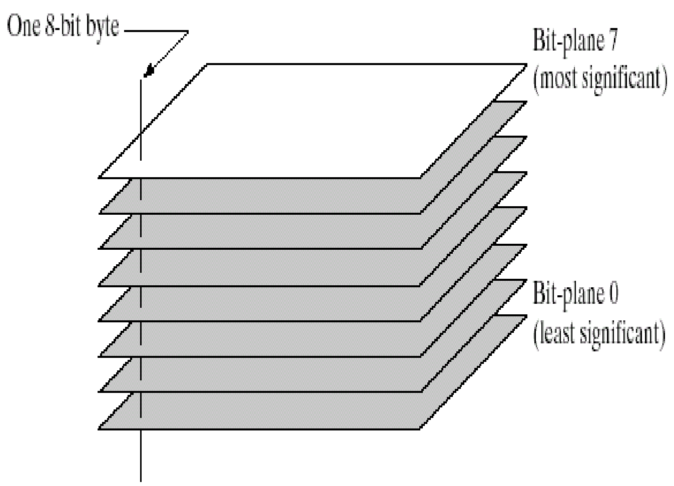


FIGURE 3.12 Bit-plane representation of an 8-bit image.

- Highlighting the contribution made to total image appearance by specific bits.
- Suppose each pixel is represented by 8 bits.
- Higher-order bits contain the majority of the visually significant data.
- Useful for analyzing the relative importance played by each bit of the image.

Example of Bit Plane Transformation

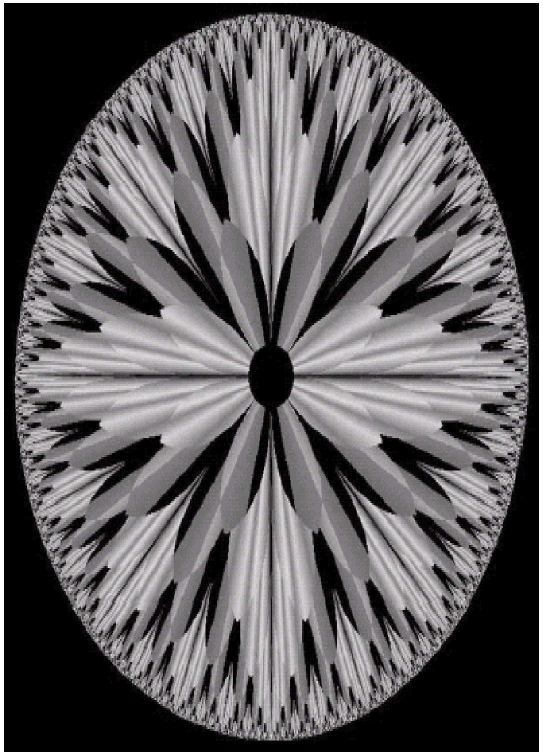


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)

Original Image

- The (binary) image for bit-plane 7 can be obtained by processing the input image with a thresholding gray-level transformation.
  - Map all levels between 0 and 127 to 0
  - Map all levels between 129 and 255 to 255.

8 Bit Planes Example

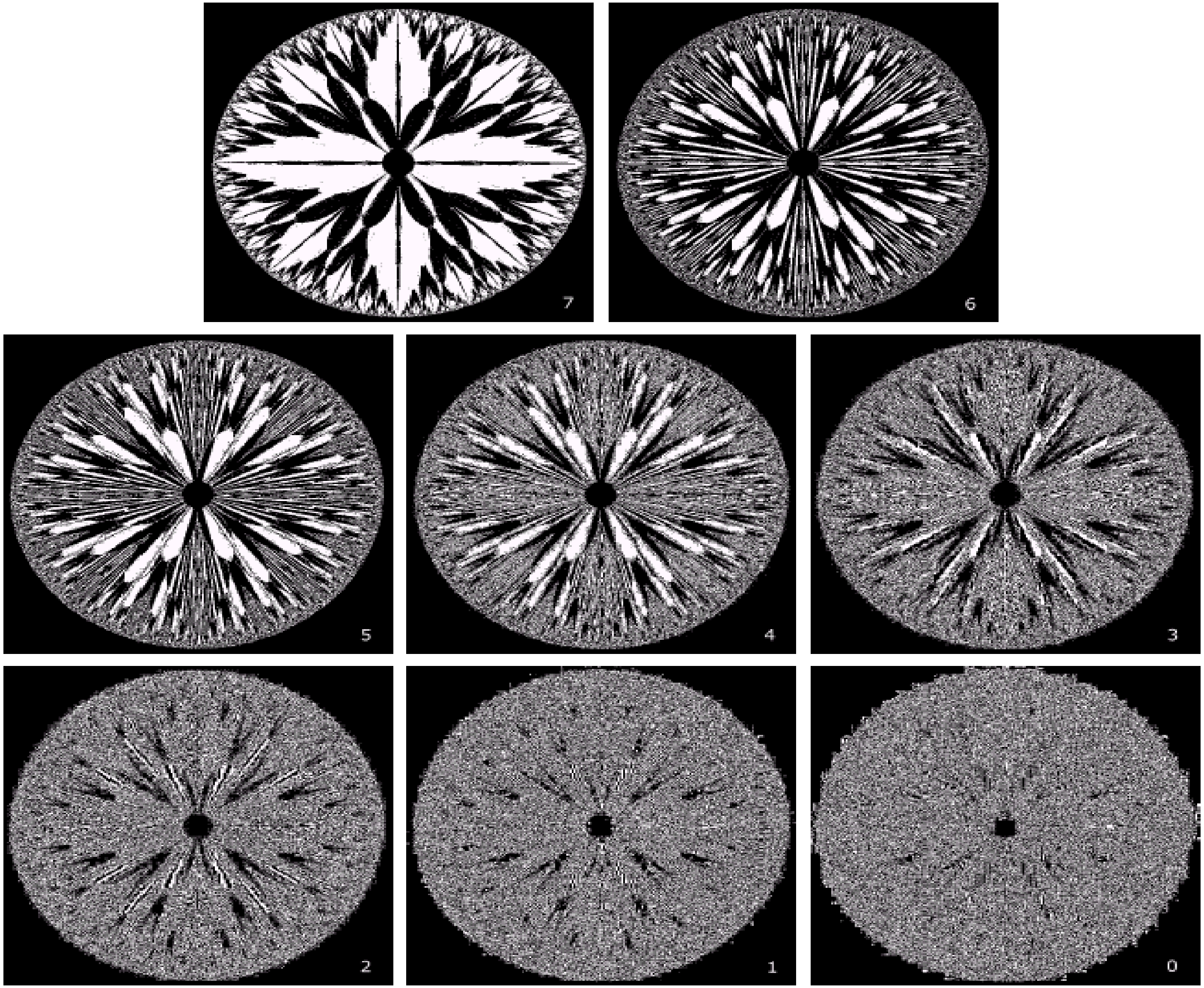


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

Histogram Processing

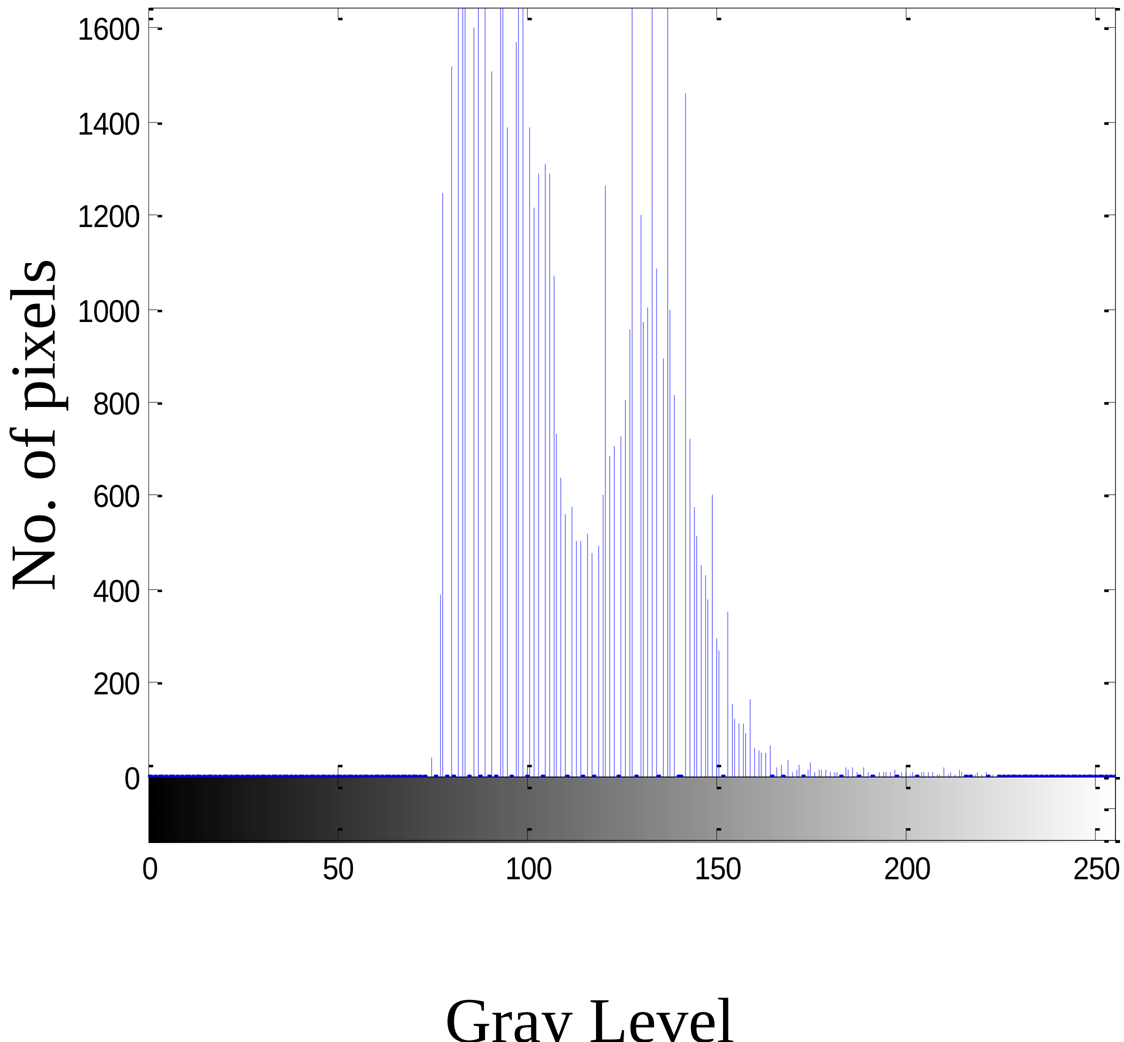
- Histogram of a digital image with gray levels in the range [0, L-1] is a discrete function

$$h(r_k) = n_k$$

where

- r_k : the kth gray level
- n_k : the number of pixels in the image having gray level r_k
- $h(r_k)$: histogram of a digital image with gray levels r_k.

Example of Histogram of an Image



Normalized Histogram

- Dividing each of histogram at gray level r_k by the total number of pixels in the image, *n*

$$p(r_k) = n_k / n$$

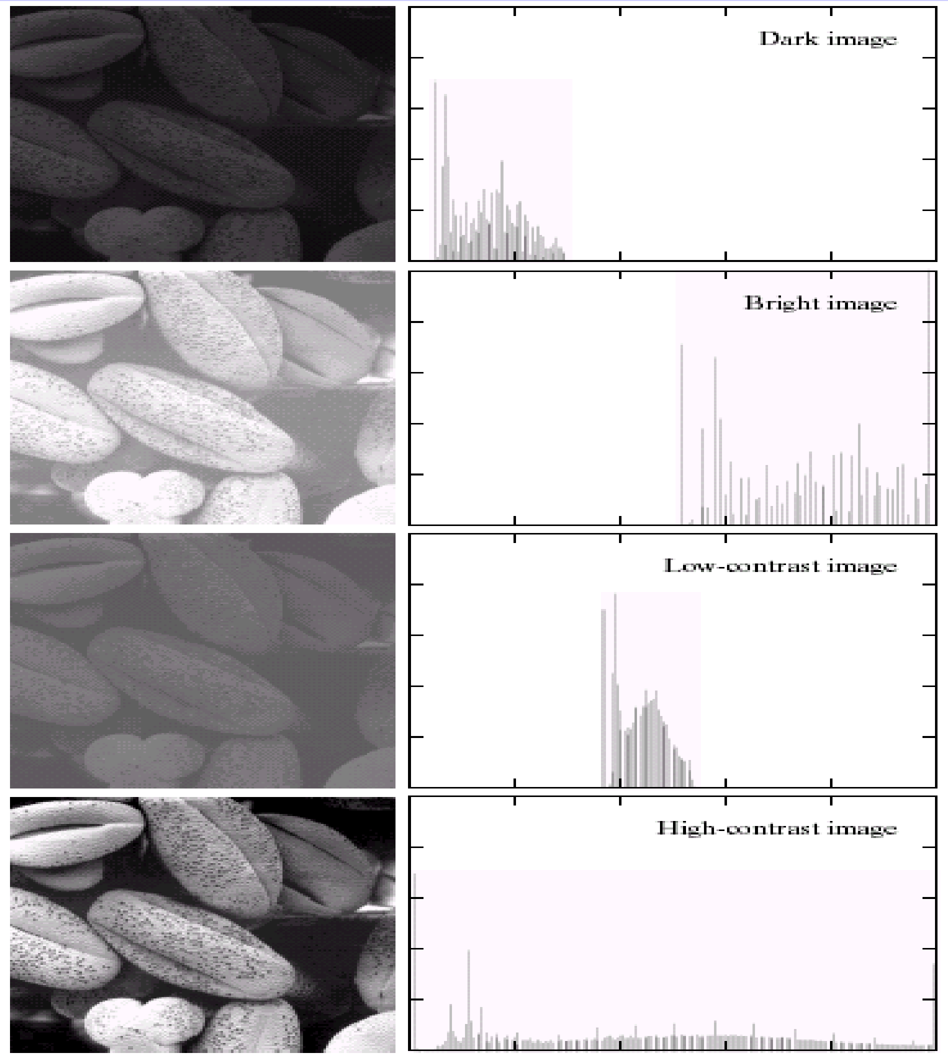
- For $k = 0,1,\ldots., L-1$
- $p(r_k)$ gives an estimate of the probability of occurrences of gray level r_k
- The sum of all components of a normalized histogram is equal to 1.

$$\rightarrow \quad \sum_{i=0}^{k} p(r_k) = 1$$

Histogram Processing

- Basic for numerous spatial domain processing techniques.
- Used effectively for image enhancement
- Information inherent in histograms also is useful in image compression and segmentation.

Example of Histogram Distribution of an Image

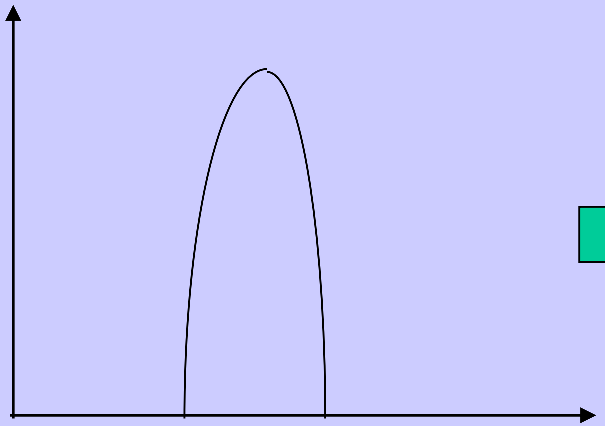


- Dark image
  - Components of histogram are concentrated on the low side of the gray scale.
- Bright image
  - Components of histogram are concentrated on the high side of the gray scale.
- Low-contrast image
  - histogram is narrow and centered toward the middle of the gray scale.
- High-contrast image
  - Histogram covers broad range of the gray scale and the distribution of pixels is not too far from uniform, with very few vertical lines being much than the others

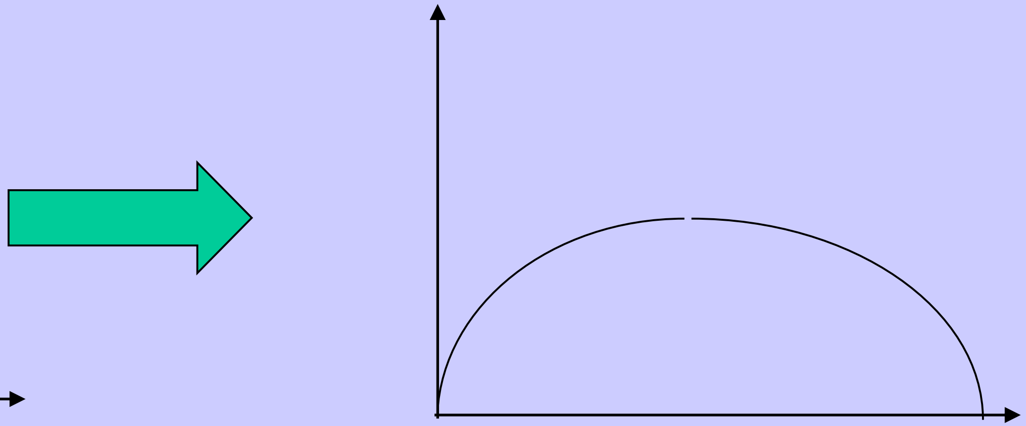
Histogram Equalization

- As the low-contrast image's histogram is narrow
  and centered toward the middle of the gray scale, if we distribute the histogram to a wider range the quality of the image will be improved.

- We can do it by adjusting the probability density function of the original histogram of the image so that the probability spread equally.



Original Image Histogram



After Histogram Equalization Process

Histogram Transformation

$s = T(r) \qquad 0 <= r <= 1$

where $T(r)$ satisfy the following 2 conditions:

(a) $T(r)$ is a single value (one to one relationship) and monotonically increasing in the range $0 <= r <= 1$

(b) $0 <= T(r) <= 1$ for $0 <= r <= 1$

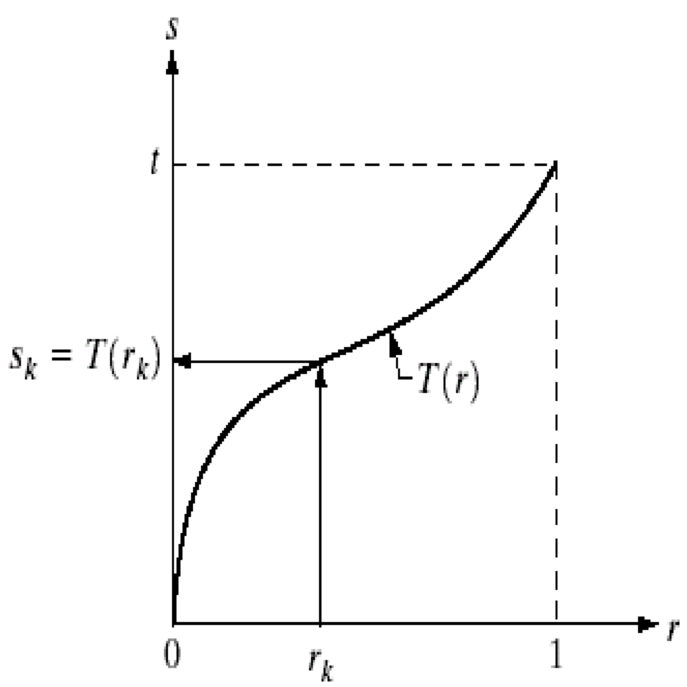


FIGURE 3.16 A gray-level transformation function that is both single valued and monotonically increasing.

2 Conditions of T(r)

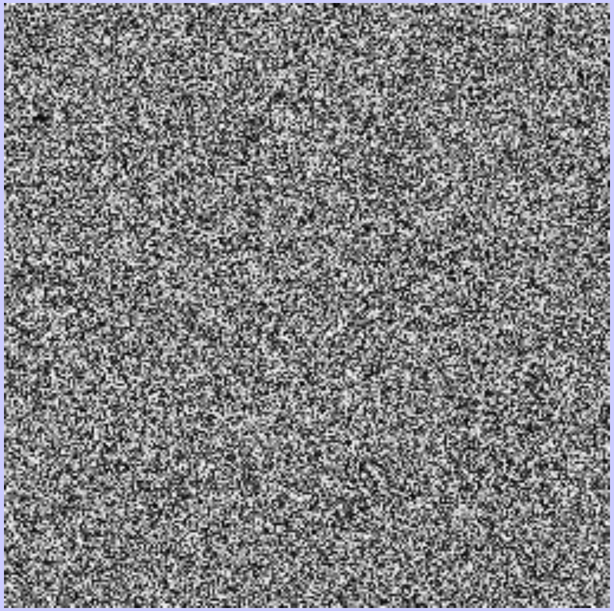
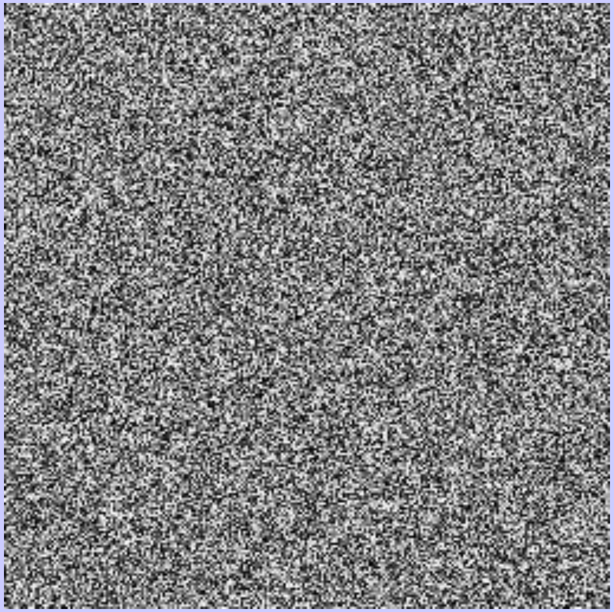
- Condition (a) : Single –valued (one-to-one relationship) guarantees that the inverse transformation will exist.
  - Monotonicity condition preserves the increasing order from black to white in the output image thus it won’t cause a negative image.
- Condition (b) : Guarantees that the output gray levels will be in the same range as the input levels.

Thus, the inverse transformation from *s* back to *r* :

$$r = T^{-1}(s) \qquad 0 \leq s \leq 1$$

Random Variables & Digital Images

- The gray levels in an image may be viewed as random variables in the interval [0,1]

Example of images produced from random numbers

Probability Density Function

In general,

- If a random variable ***x*** is transformed by a monotonic transformation function ***T(x)*** to produce a new random variable *y*,
- The probability density function of ***y*** can be obtained from knowledge of ***T(x)*** and the probability density function of *x*, as follows:

$$P_y(y) = P_x(x)\left|\frac{dx}{dy}\right|$$

where the vertical bars signify the absolute value

PDF Applied to an Image

One of the most fundamental descriptors of a random variable is its probability density function (PDF).

- Let
  - $P_r(r)$ denote the PDF of random variable r
  - $P_s(s)$ denote the PDF of random variable s
- If $p_r(r)$ and $T(r)$ are know and $T^{-1}(s)$ satisfies condition (a) then $P_s(s)$ can be obtained using a formula :

$$P_s(s) = P_r(r)\left|\frac{dr}{ds}\right|$$

Thus, the PDF of a transformed variable, s, is determined by the gray level PDF of the input image and the chosen transformation function.

Transformation Function

- A transformation function is a cumulative distribution function (CDF) of random variable *r*:

$$s = T(r) = \int_0^r p_r(w)\,dw$$

where *w* is a dummy variable of integration.

Cumulative Distribution Function (CDF)

- CDF is an integral of a probability function (always positive) is the area under the function.
- Thus, CDF is always single valued and monotonically increasing
- Thus,CDF satisfies the condition (a)
- We can use CDF as a transformation function in image processing.

Finding $P_s(s)$ from given $T(r)$

Given transformation function, $T(r)$, $P_s(s)$ could be calculated using:

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

In other words,

$$\frac{ds}{dr} = \frac{dT(r)}{dr}$$

$$= \frac{d}{dr}\left[\int_0^r p_r(w)\,dw\right]$$

$$= p_r(r)$$

$\longrightarrow$

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

$$= \mathrm{p_r(r)}\left|\frac{1}{\mathrm{p_r(r)}}\right|$$

$$= 1 \;\; where\; 0 \le \mathrm{s} \le 1$$

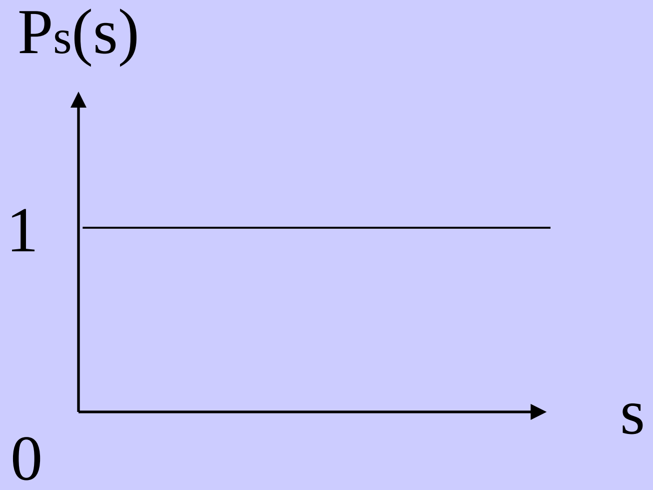
Substitute and yield

$P_s(s)$

- As $p_s(s)$ is a probability function, it must be zero outside the interval [0,1] in this case because its integral over all values of s must equal 1.
- **Called $p_s(s)$ as a uniform probability density function**
- $P_s(s)$ is always a uniform, independent of the form of $p_r(r)$.

$$s = T(r) = \int_0^r p_r(w)\,dw$$

- A random variable s characterized by a uniform probability function



Discrete transformation function

- The probability of occurrence of gray level in an image is approximated by

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

- The discrete version of transformation

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

$$= \sum_{j=0}^{k} \frac{n_j}{n} \quad \text{where } k = 0,1,\ldots\ldots,L-1$$

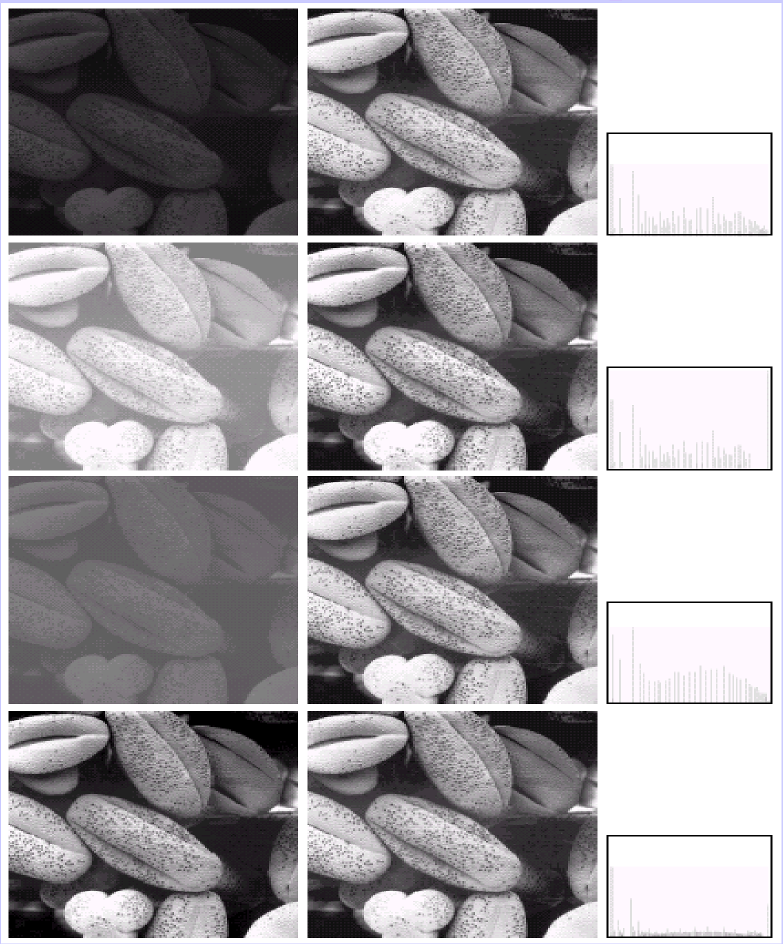
Histogram Equalization

- Thus, an output image is obtained by mapping each pixel with level r_k in the input image into a corresponding pixel with level s_k in the output image.

- In discrete space, it cannot be proved in general that this discrete transformation will produce the discrete equivalent of a uniform probability density function, which would be a uniform histogram.

Effects of Histogram Equalization

Before After Histogram equalization



The quality is not improved much because the original image already has a broaden gray-level scale.

a b c

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

Image Enhancement in the Spatial Domain

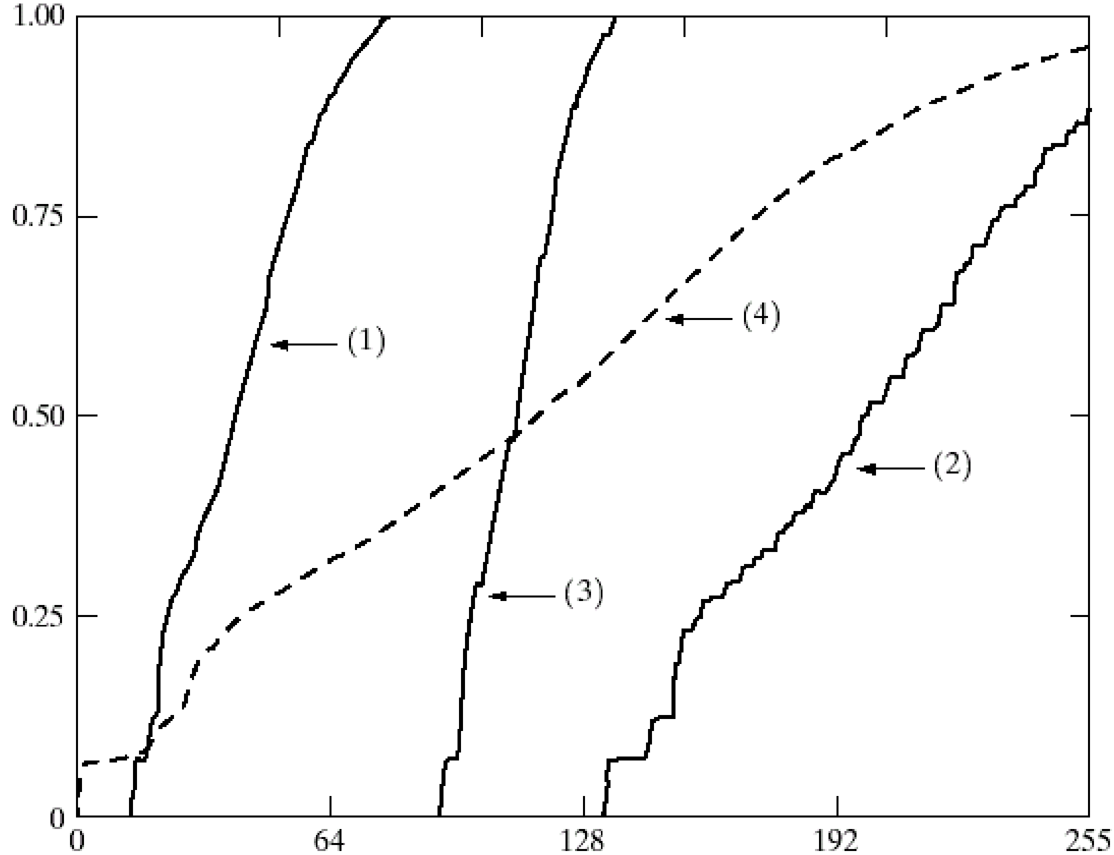
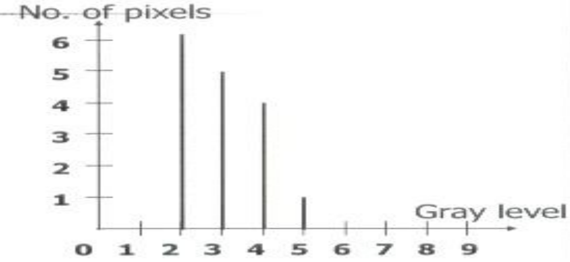


FIGURE 3.18 Transformation functions (1) through (4) were obtained from the histograms of the images in Fig.3.17(a), using Eq. (3.3-8).

Example

2	3	3	2
4	2	4	3
3	2	3	5
2	4	2	4

4x4 image
Gray scale = [0,9]



histogram

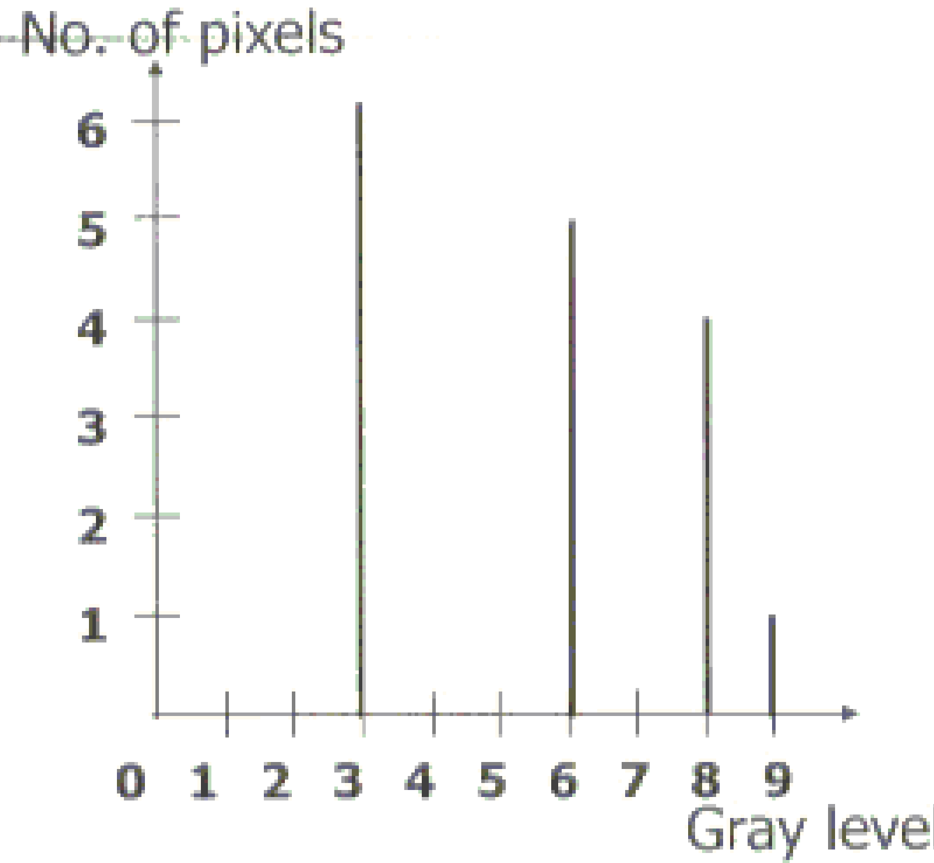
Gray Level(j)	0	1	2	3	4	5	6	7	8	9
No. of pixels	0	0	6	5	4	1	0	0	0	0
$\sum_{j=0}^{k} n_j$	0	0	6	11	15	16	16	16	16	16
$s = \sum_{j=0}^{k} \frac{n_j}{n}$	0	0	6 / 16	11 / 16	15 / 16	16 / 16	16 / 16	16 / 16	16 / 16	16 / 16
s x 9	0	0	3.3 ≈3	6.1 ≈6	8.4 ≈8	9	9	9	9	9

Example

3	6	6	3
8	3	8	6
6	3	6	9
3	8	3	8

Output image

Gray scale = [0,9]



Histogram equalization

Note

- It is clearly seen that
  - Histogram equalization distributes the gray level to reach the maximum gray level (white) because the cumulative distribution function equals 1 when $0 \leq r \leq L-1$
  - If the cumulative numbers of gray levels are slightly different, they will be mapped to little different or same gray levels as we may have to approximate the processed gray level of the output image to integer number
  - Thus the discrete transformation function can't guarantee the one to one mapping relationship

Histogram Matching/Specification

- Histogram equalization has the disadvantage which is that it can generate **only one type of output image**.
- On the other hand, Histogram Specification or sometimes referred to as Histogram Matching, the shape of the histogram is specified manually so that desired the image output is produced.
- In Histogram Matching, histogram doesn't have to be uniform.

Consider the continuous domain

Let $p_r(r)$ denote continuous probability density function of gray-level of input image, r

Let $p_z(z)$ denote desired (specified) continuous probability density function of gray-level of output image, z

Let s be a random variable with the property

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

Histogram equalization

Where w is a dummy variable of integration

Next, we define a random variable z with the property

$$g(z) = \int_0^z p_z(t)\,dt = s$$

Histogram equalization

Where t is a dummy variable of integration

thus

$$s = T(r) = G(z)$$

Therefore, z must satisfy the condition

$$z = G^{-1}(s) = G^{-1}[T(r)]$$

Assume G^{-1} exists and satisfies the condition (a) and (b)

We can map an input gray level r to output gray level z

Procedure Conclusion

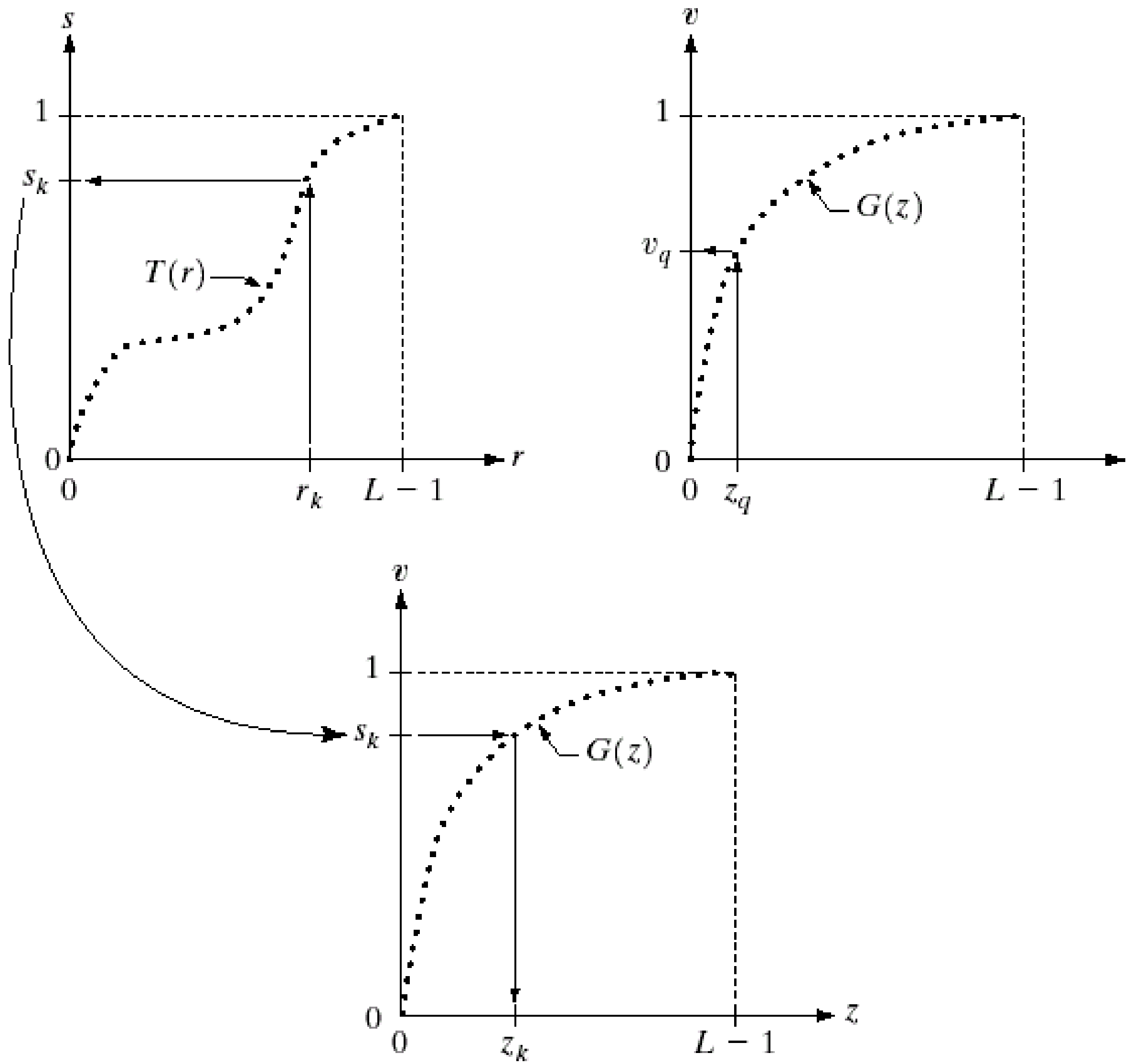
1. Obtain the transformation function T(r) by calculating the histogram equalization of the input image

$$s = T(r) = \int_0^r p_r(w)\,dw$$

2. Obtain the transformation function G(z) by calculating histogram equalization of the desired density function

$$G(z) = \int_0^z p_z(t)\,dt = s$$

Graphical Representation of Histogram Specification/Matching



a b
c

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.

Procedure Conclusion

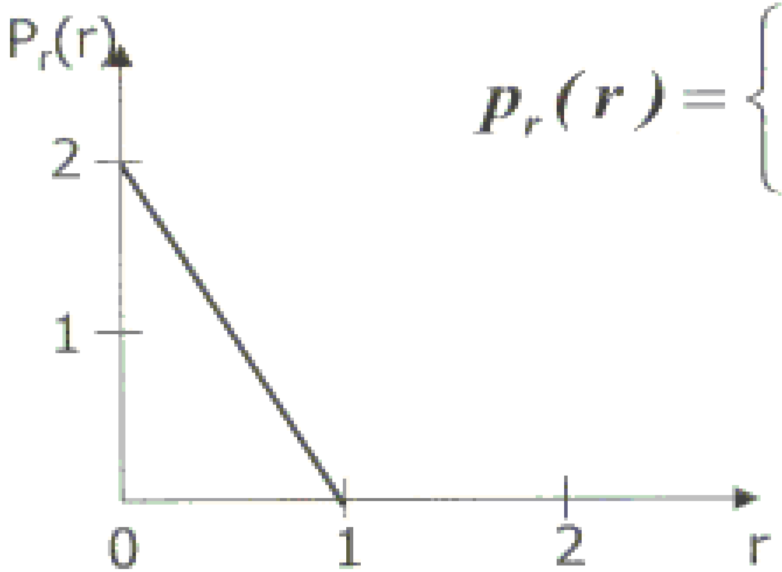
3. Obtain the inversed transformation function G^{-1}

$$z = G^{-1}(s) = G^{-1}[T(r)]$$

4. Obtain the output image by applying the processed gray-level from the inversed transformation function to all the pixels in the input image

Example

Assume an image has a gray level probability density function $p_r(r)$ as shown.

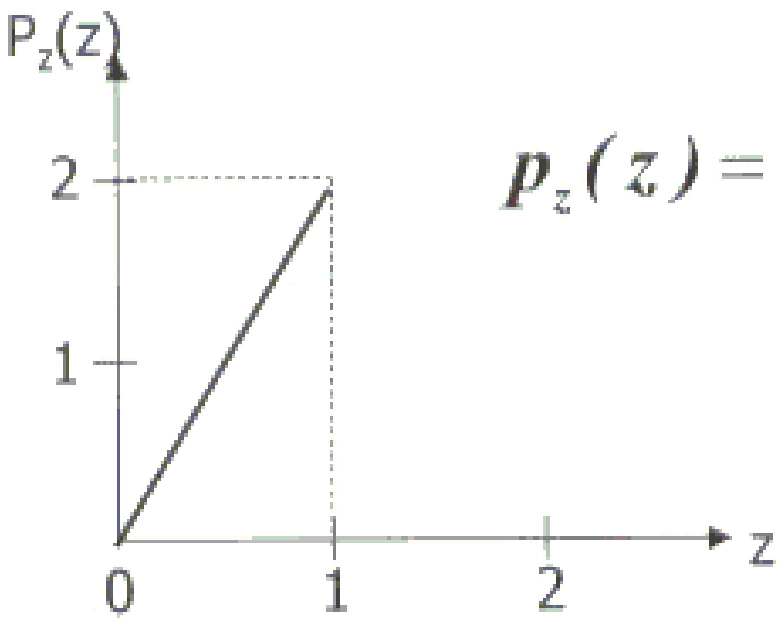


$$p_r(r) = \begin{cases} -2r + 2 & ;0 \le r \le 1 \\ 0 & ;\text{elsewhere} \end{cases}$$

$$\int_0^r p_r(w)\,dw = 1$$

Example

We would like to apply the histogram specification with the desired probability density function $p_z(z)$ as shown.

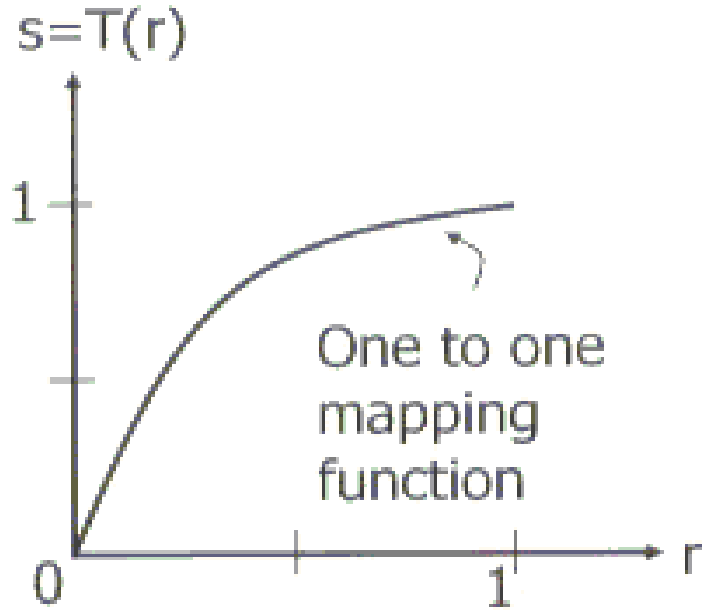


$$p_z(z) = \begin{cases} 2z & ;0 \le z \le 1 \\ 0 & ;\text{elsewhere} \end{cases}$$

$$\int_0^z p_z(w)\,dw = 1$$

Step 1:

Obtain the transformation function T(r)



$$s = T(r) = \int_0^r p_r(w)\,dw$$

$$= \int_0^r (-2w + 2)\,dw$$

$$= -w^2 + 2w \Big|_0^r$$

$$= -r^2 + 2r$$

Step 2:

Obtain the transformation function G(z)

$$G(z) = \int_0^z (2w)\,dw \quad = z^2 \Big|_0^z \quad = z^2$$

Step 3:

Obtain the inversed transformation function G^{-1}

$$G(z) = T(r)$$

$$z^2 = -r^2 + 2r$$

$$z = \sqrt{2r - r^2}$$

We can guarantee that $0 \le z \le 1$ when $0 \le r \le 1$

Discrete formulation

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

$$= \sum_{j=0}^{k} \frac{n_j}{n} \qquad k = 0,1,2,\ldots,L-1$$

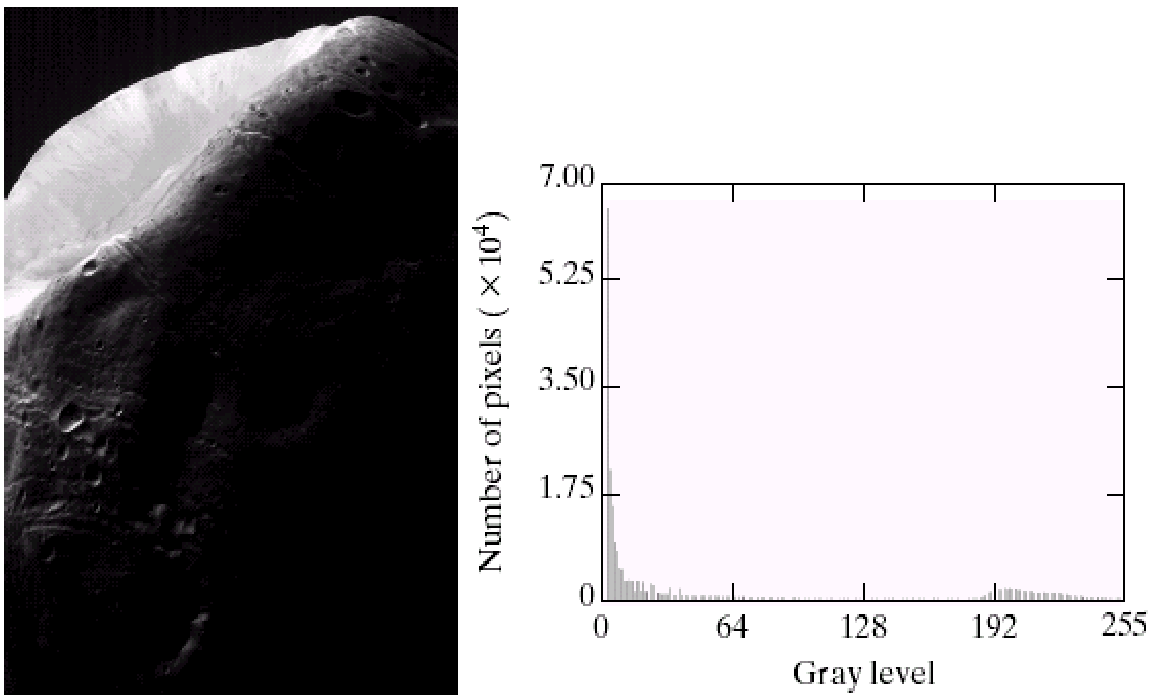
$$G(z_k) = \sum_{i=0}^{k} p_z(z_i) = s_k \qquad k = 0,1,2,\ldots,L-1$$

$$z_k = G^{-1}\left[T(r_k)\right]$$

$$= G^{-1}\left[s_k\right] \qquad k = 0,1,2,\ldots,L-1$$

Comparing Histogram Equalisation vs Histogram Matching

Consider the following image:

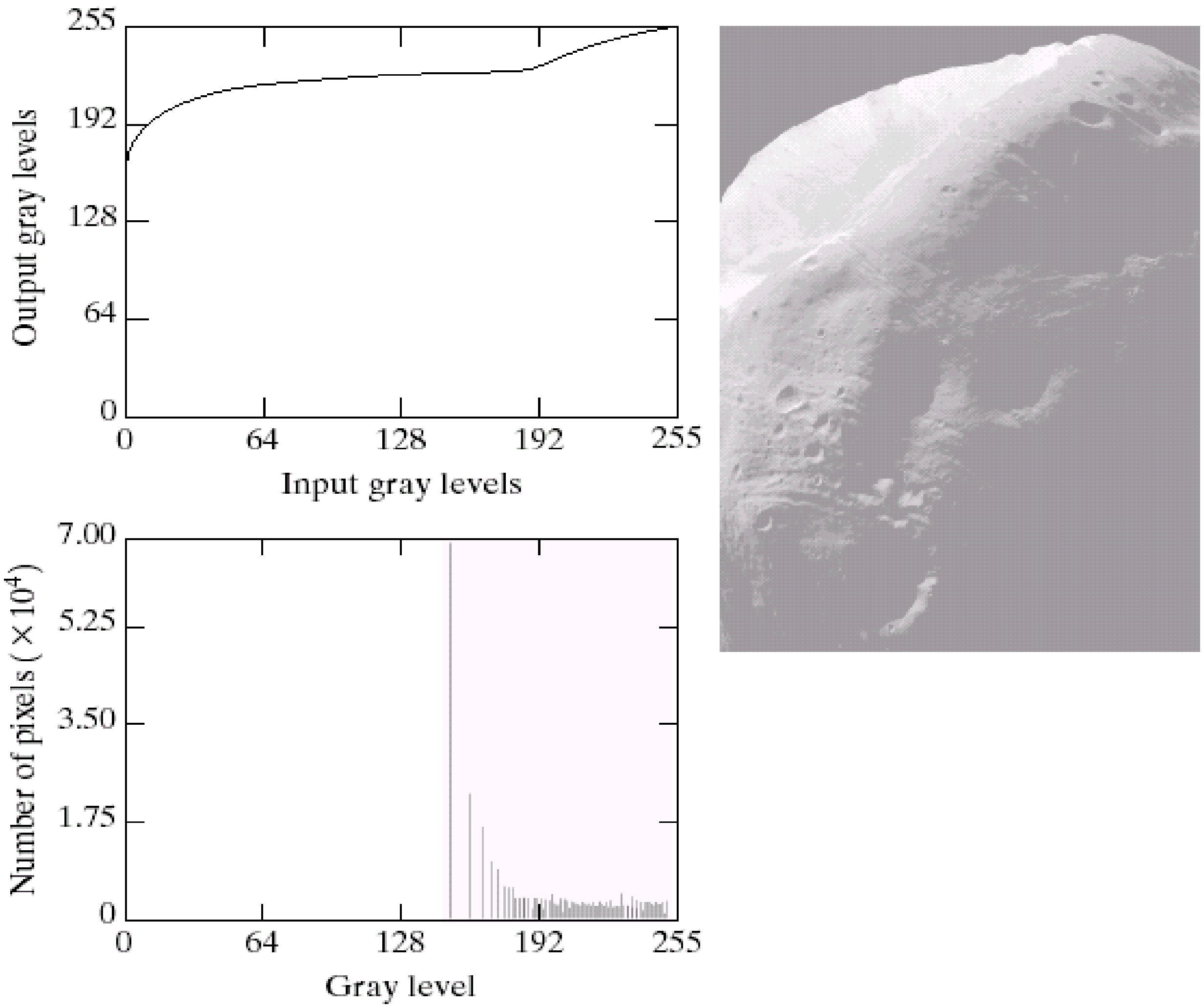


a b

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

Image is dominated by large dark areas resulting in a histogram characterized by a large concentration of pixels in the left of the section of histogram

Results of Histogram Equalisation



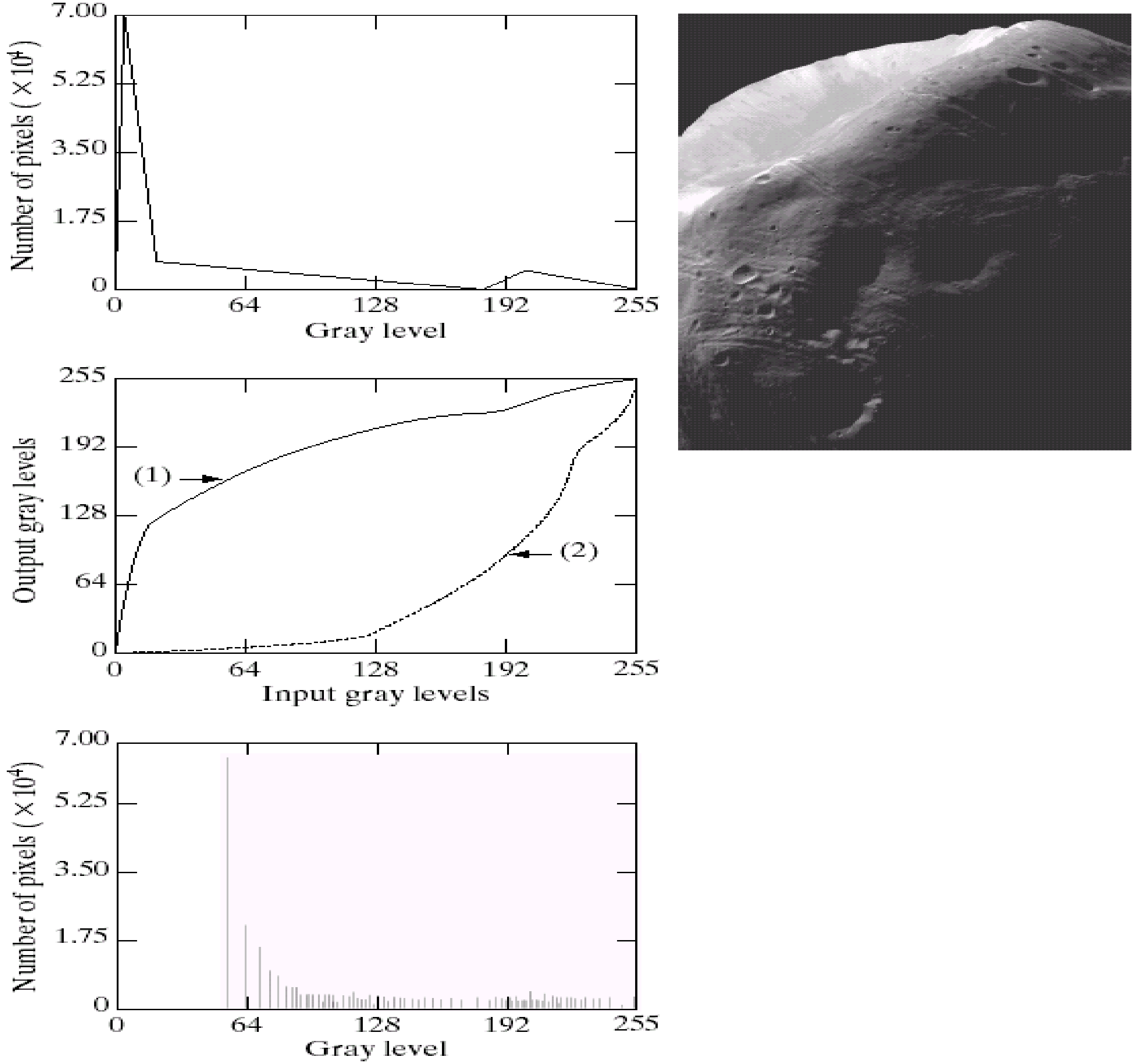
a b
c

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

Results from Histogram Matching/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).



Note on Histogram Matching

- Histogram matching is a manual trial and error process
- There are no rules for specifying histogram, and one must resort to analysis on a case by case basis for any given enhancement task.
- Histogram processing methods are global processing → pixels are modified by a transformation function based on the gray level content of an entire image.
- Sometimes, we may need to enhance image details over small area in an image → local enhancement

Use of Histogram Statistic for Image Enhancement

Instead of using the histogram directly for enhancement, we can use some statistical parameters obtainable from the histogram.

Let, r denote a discrete random variable representing discrete gray levels in the range [0, L-1] and $p(r_i)$ denote the normalised component corresponding to the ith value of r.

Thus, the nth moment of r about its mean is defined:

$$\mu_n(r) = \sum_{i=0}^{L-1} (r_i - m)^n \, p(r_i)$$

where m is the mean value of r (its average gray level).

$$m = \sum_{i=0}^{L-1} r_i \, p(r_i)$$

Therefore, $\mu_0 = 1$ *and* $\mu_1 = 0$.

Use of Histogram Statistic for Image Enhancement

The second moment (variance of *r*) is given by:

$$\mu_2(r) = \sigma^2(r) = \sum_{i=0}^{L-1} (r_i - m)^2 \, p(r_i)$$

Thus, standard deviation,

$$\sigma(r) = \sqrt{\sigma^2(r)}$$

Use of Histogram Statistic for Image Enhancement

The global mean and variance are measured over an entire image are useful primirily for gross adjustments of overall intensity and contrast.:

A much more powerful of these two measures is in local enhancement, where the local mean and variance are used as a basis for making changes that depends on image characteristics in a predefined region about each pixel in an image.

Use of Histogram Statistic for Image Enhancement

Let, (x,y) be the coordinates of a pixel in an image, and let S_{xy} denote the neighborhood (sub image) of specified size, centered at (x, y).

The mean value of this sub image could be expressed as

$$m_{S_{xy}} = \sum_{(s,t)\in S_{xy}}^{L-1} r_{s,t}\; p(s,t)$$

where $r_{s,t}$ is the gray level at coordinates (s,t) in the neighborhood, and $p(r_{s,t})$ is the neighborhood normalized histogram component corresponding to that value of gray level.

Similarly, the gray level variance of the pixels in the region S_{xy} :

$$\sigma^2_{S_{xy}} = \sum_{(s,t)\in S_{xy}}^{L-1} [r_{s,t} - m_{S_{xy}}]^2\, p(r_{s,t})$$

Example of the Results of Local Histogram Equalization Operations

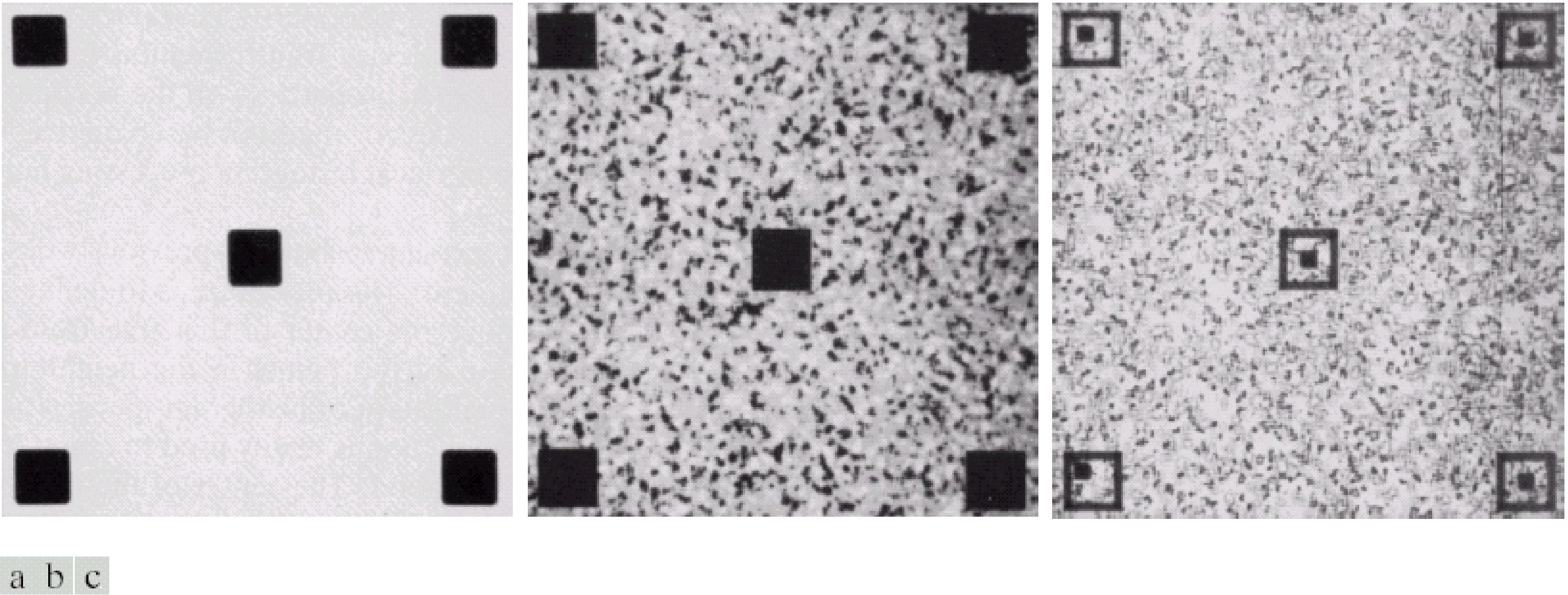


FIGURE 3.23 (a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization using a 7×7 neighborhood about each pixel.

Explain the result in c)

- Basically, the original image consists of many small squares inside the larger dark ones.
- However, the small squares were too close in gray level to the larger ones, and their sizes were too small to influence global histogram equalization significantly.
- So, when we use the local enhancement technique, it reveals the small areas.
- Note also the finer noise texture is resulted by the local processing using relatively small neighborhoods.

Examples of Enhancement Based on Local Statistic

The problem is to enhance dark areas while leaving the light area as unchanged as possible since it does not require 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).

Examples of Enhancement Based on Local Statistic

Solutions:

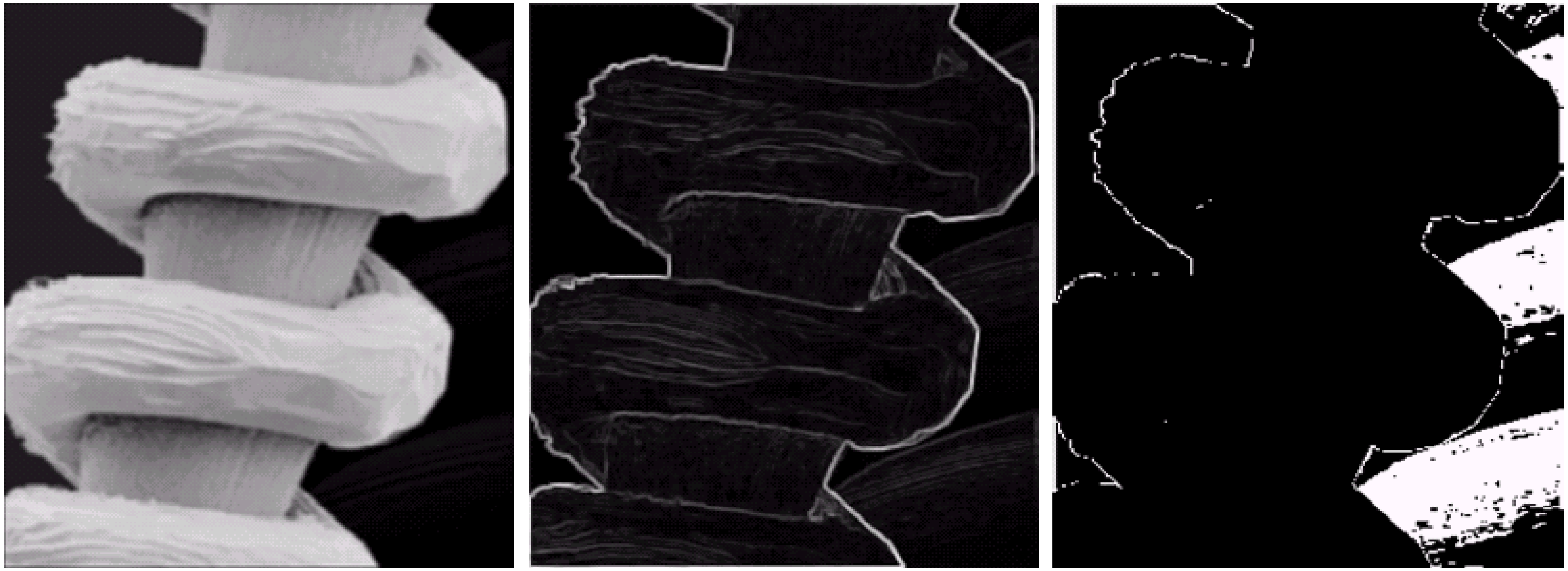
A measure of whether an area is relatively light/dark at a point (x,y) is to compare the local average gray level, $m_{S_{xy}}$, to the average image Gray level, M_G.

Let, $f(x,y)$ be the image coordinates (x,y) and $g(x,y)$ be the corresponding enhanced pixel at those coordinates. Then :

$$g(x,y) = \begin{cases} E \bullet f(x,y) & \text{if } m_{S_{xy}} \le k_0 M_G \text{ AND } k_1 D_G \le \sigma_{S_{xy}} \le k_2 D_G \\ f(x,y) & \text{otherwise} \end{cases}$$

where E, k_0, k_1, k_2 are specified parameters; M_G is the global mean of the input image and D_G is the global standard deviation

Examples of Enhancement Based on Local Statistic



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.

Value by experimentation: $E = 4.0$, k_0=0.4, k_1 =0.02, and k_2 = 0.4.

Examples of Enhancement Based on Local Statistic

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

Enhancement using Arithmetic/Logic Operations

- Arithmetic/Logic operations perform on pixel by pixel basis between two or more images
- except NOT operation which perform only on a single image

Logic Operations

- Logic operation performs on gray-level images, the pixel values are processed as binary numbers
- light represents a binary 1, and dark represents a binary 0
- NOT operation = negative transformation

Examples of AND and OR Operations

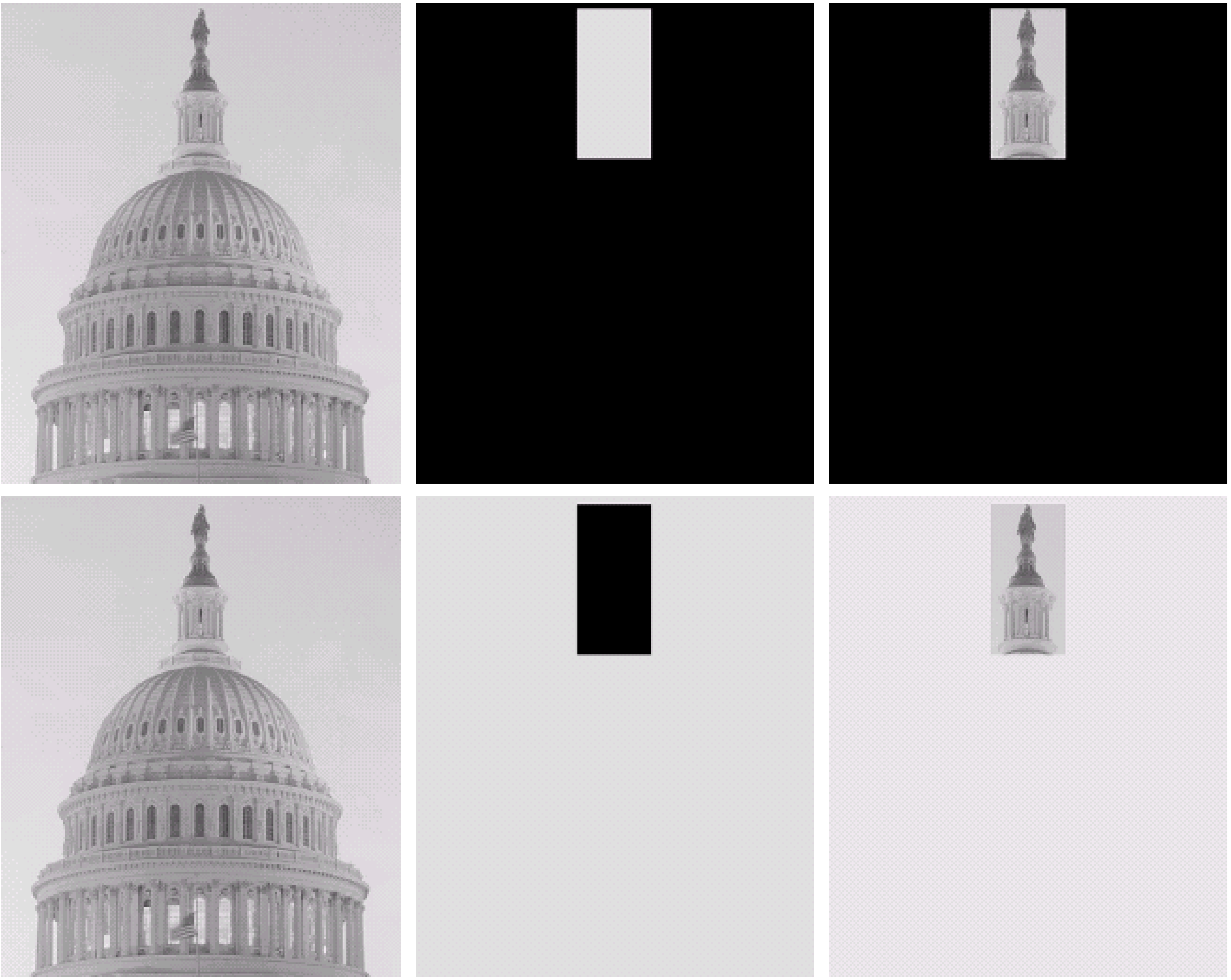


FIGURE 3.27 (a) Original image. (b) AND image mask. (c) Result of the AND operation on images (a) and (b). (d) Original image. (e) OR image mask. (f) Result of operation OR on images (d) and (e).

Original Image Mask Image Results

Image Subtraction

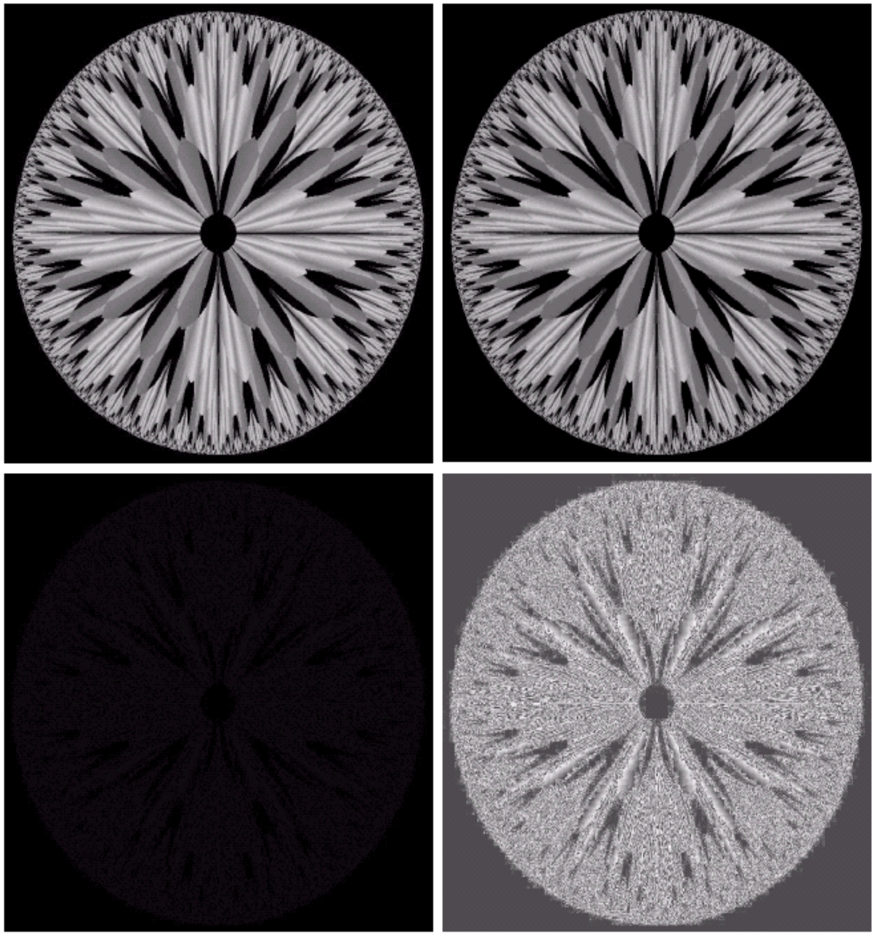
Consider the differences between two images, $f(x,y)$ and $h(x,y)$:

Image subtraction, can be expressed as:

$$g(x,y) = f(x,y) - h(x,y)$$

- The key features is the enhancement of the differences between images

Example of Image Subtraction



a b
c d

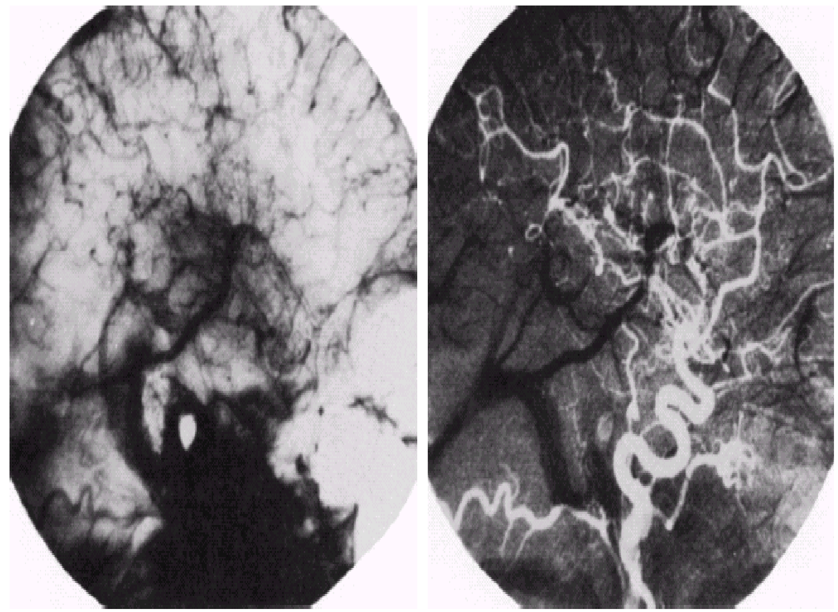
FIGURE 3.28 (a) Original fractal image. (b) Result of setting the four lower-order bit planes to zero. (c) Difference between (a) and (b). (d) Histogram-equalized difference image. (Original image courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA).

- a). original fractal image
- b). result of setting the four lower-order bit planes to zero
  - refer to the bit-plane slicing
  - the higher planes contribute significant detail
  - the lower planes contribute more to fine detail
  - image b). is nearly identical visually to image a), with a very slightly drop in overall contrast due to less variability of the gray-level values in the image.
- c). difference between a). and b). (nearly black)
- d). histogram equalization of c). (perform contrast stretching transformation)

Example in Medical Imaging

Use of image in *mask mode radiography*:

- h(x,y) is the mask, an X-ray image of a region of a patient's body captured by an intensified TV camera (instead of traditional X-ray film) located opposite an X-ray source
- f(x,y) is an X-ray image taken after injection a contrast medium into the patient's bloodstream
- images are captured at TV rates, so the doctor can see how the medium propagates through the various arteries in the area being observed (the effect of subtraction) in a movie showing mode.



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.

X-ray image of the top patient's head prior to the injection of an iodine in a blood stream (mask image)

Note

- We may have to adjust the gray-scale of the subtracted image to be [0, 255] (if 8-bit is used)
  - first, find the minimum gray value of the subtracted image
  - second, find the maximum gray value of the subtracted image
  - set the minimum value to be zero and the maximum to be 255
  - while the rest are adjusted according to the interval [0, 255], by timing each value with 255/max
- Subtraction is also used in segmentation of moving pictures to track the changes
  - after subtract the sequenced images, what is left should be the moving elements in the image, plus noise

Image Averaging

Consider a noisy image $g(x,y)$ formed by the addition of noise $\eta(x,y)$ to an original image $f(x,y)$:

$$g(x,y) = f(x,y) + \eta(x,y)$$

where the assumption is that at every pair of coordinates (x,y) the noise is uncorrelated.

The objective is to reduce the noise content by a set of noisy images, $\{ g_i(x,y) \}$.

Image Averaging

- If noise has zero mean and uncorrelated then it can be shown that the result of averaging K noisy images as follows:

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

Image Averaging

Then,

$$E\{\overline{g}(x,y)\} = f(x,y)$$

and

$$\sigma^2_{\overline{g}(x,y)} = \frac{1}{K}\sigma^2_{\eta(x,y)}$$

where $E\{\overline{g}(x,y)\}$ is the expected value of $\overline{g}$ and $\sigma^2_{\overline{g}(x,y)}$ and $\sigma^2_{\bar{\eta}}(x,y)$ are the variances of $\overline{g}$ and η, at all the coordinates.

Image Averaging

The standard deviation at any point in the average image is:

$$\sigma^2_{\bar{g}(x,y)} = \frac{1}{\sqrt{K}} \sigma_{\eta(x,y)}$$

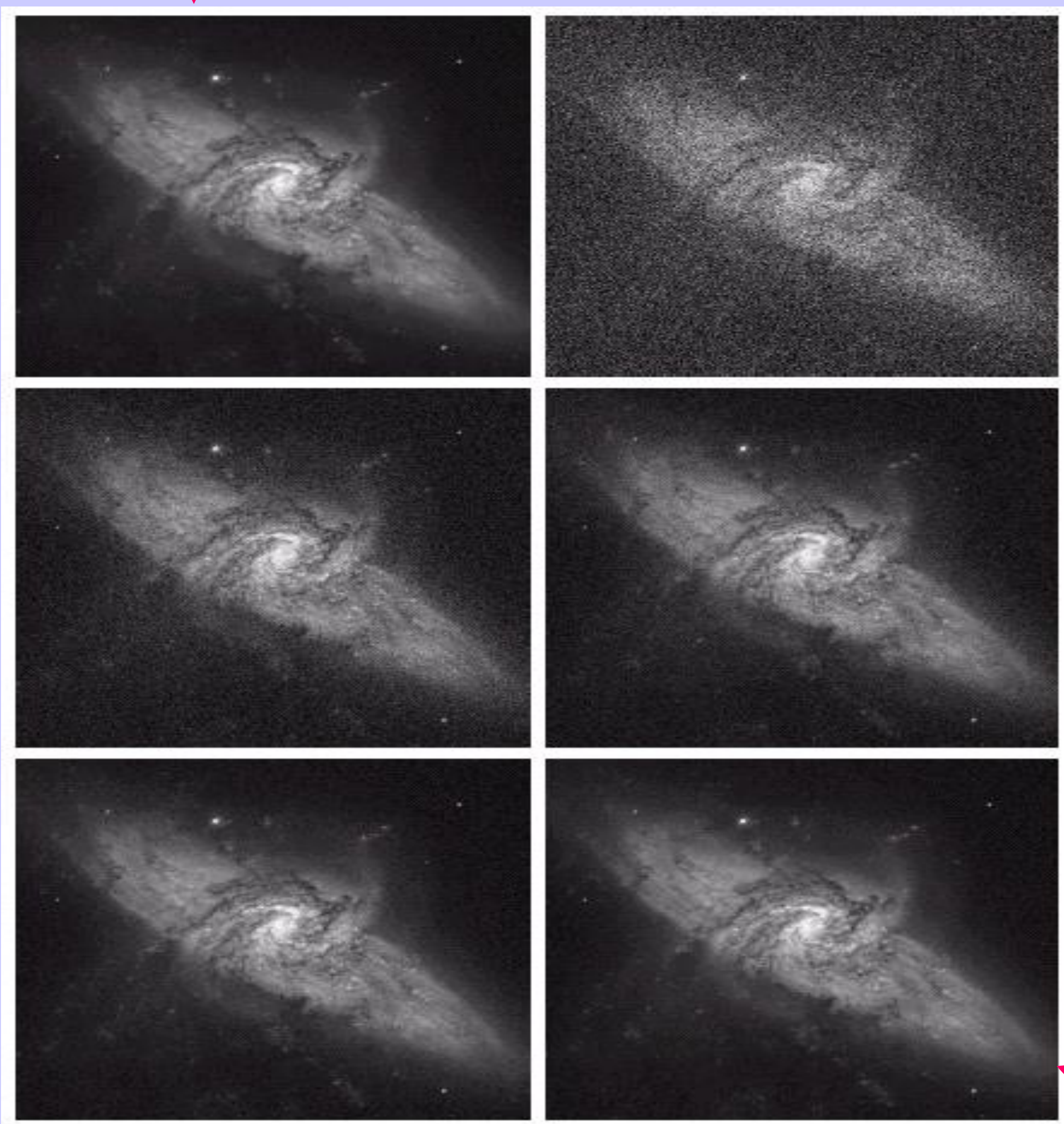
As *K* increase, it indicates that the variability (noise) of the

pixel values at each location (*x*,*y*) decreases.

In practice, the image gi(x,y) must be registered (aligned) to avoid the introduction of blurring and other artifacts in the output image.

Examples of Noise Reduction by Image Averaging

Original image



Problem in astronomy where imaging with very low light level causing sensor noise frequently to render single images virtually useless for analysis.

e.g. Image captured by Hubble Space Telescope

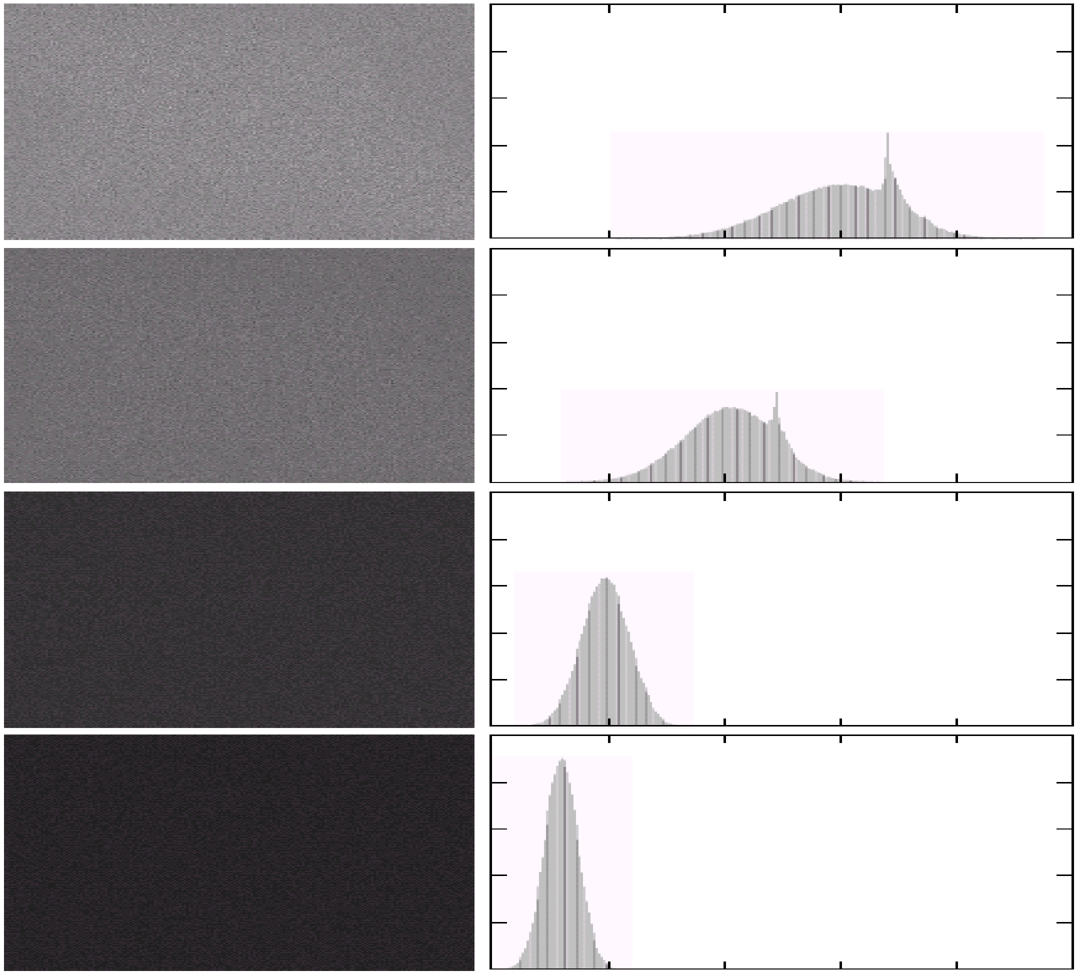
a	b
c	d
e	f

FIGURE 3.30 (a) Image of Galaxy Pair NGC3314. (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.)

Resulted image averaging

Examples of Noise Reduction by Image Averaging (cont')

Reduction in visual appearance of noise take place as a function of Increasing *K*.



a b

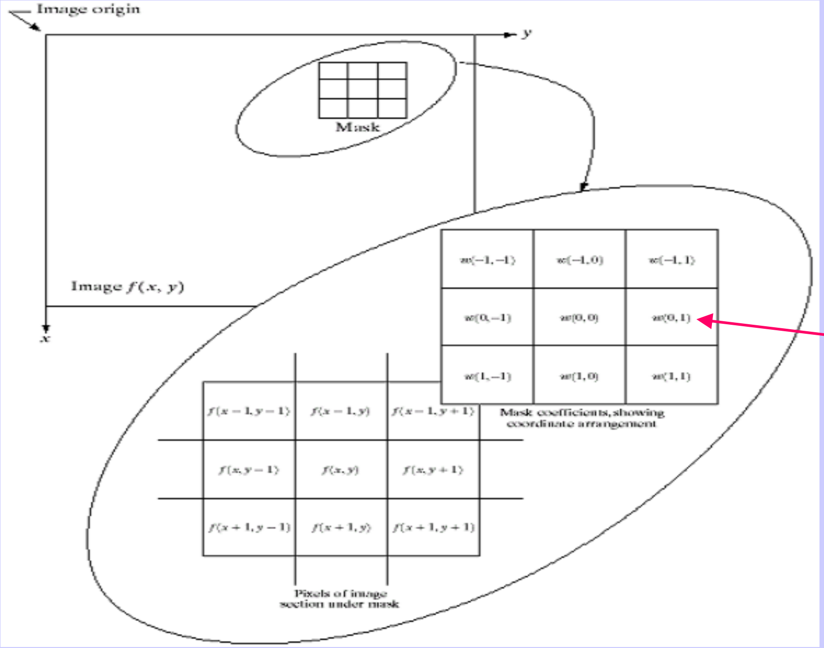
FIGURE 3.31 (a) From top to bottom: Difference images between Fig. 3.30(a) and the four images in Figs. 3.30(c) through (f), respectively. (b) Corresponding histograms.

Spatial Filtering

- Use filter (also called as mask/kernel/template or window)

- The values in a filter sub image are referred to as *coefficients* rather than pixel.

- Our focus will be on mask of odd sizes, e.g. 3x3, 5x5, 7x7, etc.



coefficient

Linear Filtering

In general, linear filtering of an image f of size $M \times N$ with a filter mask of size $m \times n$ is given by the expression:

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

$$\text{where } a = \frac{m\text{-}1}{2} \text{ and } b = \frac{n\text{-}1}{2}$$

To generate a complete filtered image, this equation must be applied For $x = 0, 1, 2, \ldots, M\text{-}1$ and $y = 0, 1, 2, \ldots, N\text{-}1$.

This process sometimes referred to as "*convolving a mask with an image*" – or sometime as *convolution mask* or *convolution kernel*

Graphical Concept of Spatial Filtering

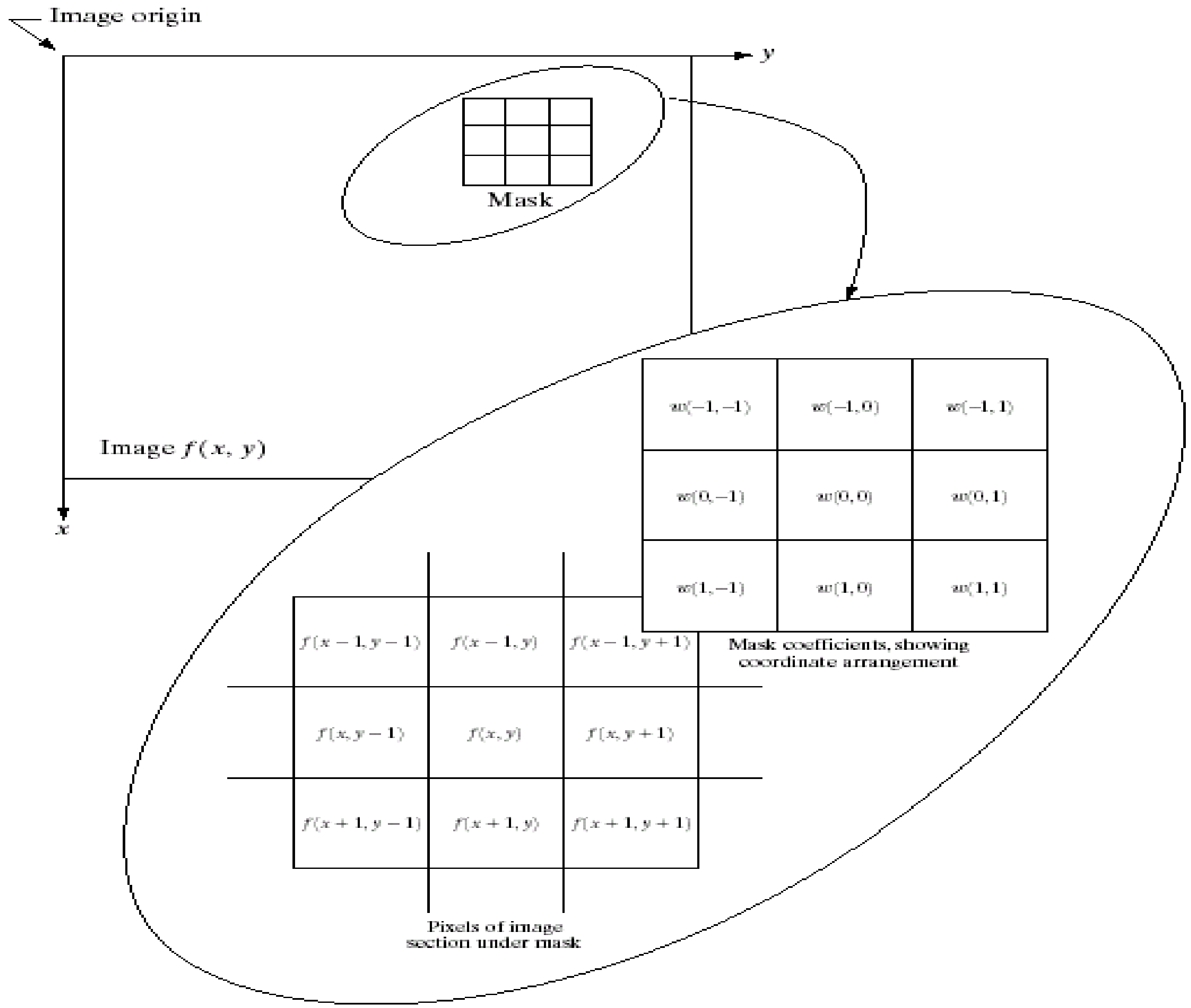


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

Spatial Filtering Process

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

- Simply move mask from point to point from the origin to the end of an image.
- At each point (x,y), the response of a filter at that point is calculated using a predefined relationship.
- The response, *R* could be expressed as:

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

Smoothing Spatial Filters

- used for blurring and for noise reduction
- blurring is used in preprocessing steps, such as
  - removal of small details from an image prior to object extraction
  - bridging of small gaps in lines or curves
- noise reduction can be accomplished by blurring with a linear filter and also by a nonlinear filter

Smoothing Linear Filters

- output is simply the average of the pixels contained in the neighborhood of the filter mask.
- called averaging filters or lowpass filters.

Smoothing Linear Filters

- replacing the value of every pixel in an image by the average of the gray levels in the neighborhood will reduce the "sharp" transitions in gray levels.
- sharp transitions
  - random noise in the image
  - edges of objects in the image
- thus, smoothing can reduce noises (desirable) and blur edges (undesirable)

Smoothing Linear Filters

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

1	1	1
1	1	1
1	1	1

Box Filter

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

1	2	1
2	4	2
1	2	1

Weighted Average

a b

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

The center is the most important and other pixels are inversely weighted as a function of their distance from the center of the mask

Weighted Average Filter

The basic strategy behind weighting the center point the highest and then reducing the value of the coefficients as a function of increasing distance from the origin is simply an attempt to reduce blurring in the smoothing process

General form : smoothing mask

- filter of size mxn (m and n odd)

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

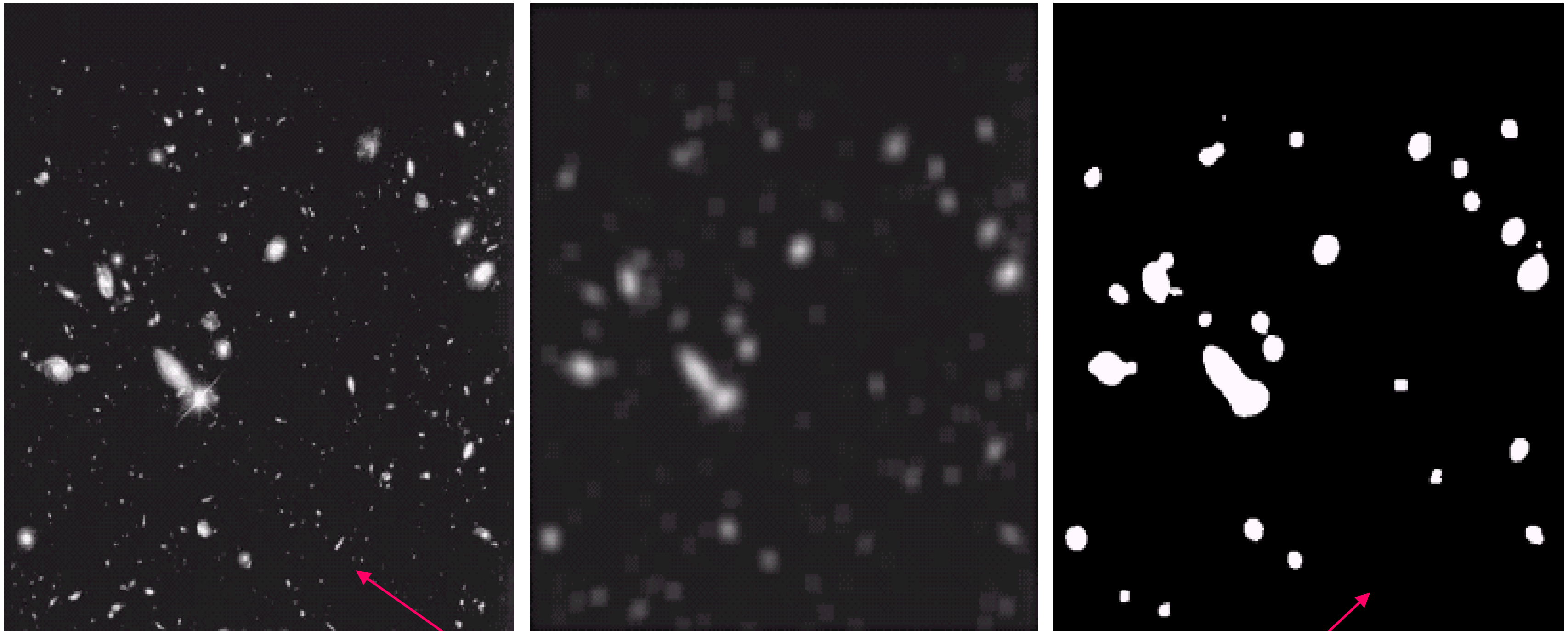
summation of all coefficient of the mask

Example: Results of applying averaging filter mask

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.

Example: Applying 15x15 averaging mask and 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.)

Small bright lights is removed after thresholding

Order-Statistics Filters (Nonlinear Filter)

- The response is based on ordering (ranking) of 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.

  Example:

  - Medium filter : R = median$\{z_k \mid k = 1,2, \ldots, n \text{ x } n\}$
  - Max filter : R = max$\{z_k \mid k = 1,2, \ldots, n \text{ x } n\}$
  - Min filter : R = min$\{z_k \mid k = 1,2, \ldots, n \text{ x } n\}$

  where n x n is the mask size

Median Filter

- Replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel (the original value of the pixel is included in the computation of the medium).

- Particularly effective in the presence of *impulse noise* (*salt and paper*)

Example:

10	20	20
20	15	20
20	25	100

sorted

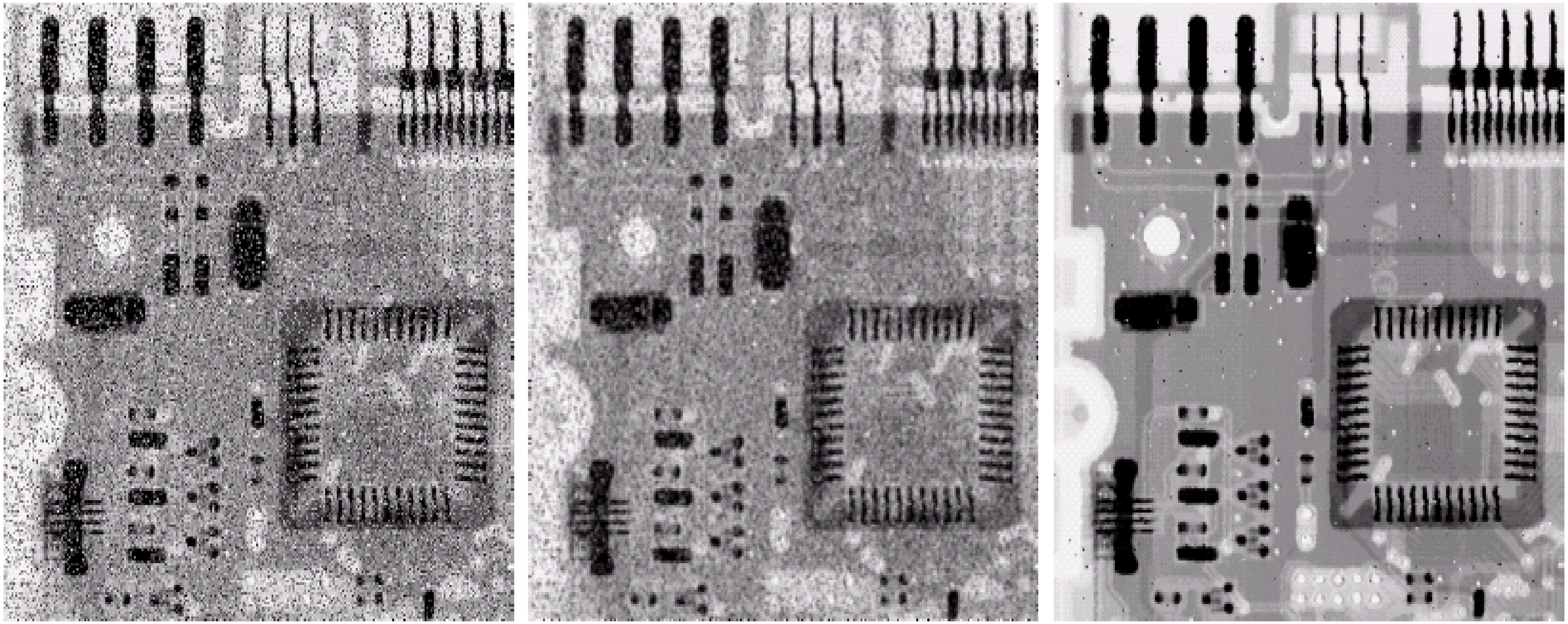
Median

(10, 15, 20, 20, 20, 20, 20, 25, 100)

Median Filters

- forces the points with distinct gray levels to be more like their neighbors.
- isolated clusters of pixels that are light or dark with respect to their neighbors, and whose area is less than $n^2/2$ (one-half the filter area), are eliminated by an $n \times n$ median filter.
- eliminated = forced to have the value equal the median intensity of the neighbors.
- larger clusters are affected considerably less

Example of Median Filter in Noise Reduction Application



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.)

Sharpening Spatial Filters

- to highlight fine detail in an image
- or to enhance detail that has been blurred, either in error or as a natural effect of a particular method of image acquisition.

Max and Min Filters

- Max filter (100th percentile) is useful in finding the brightest point in an image.
- Min filter (0th percentile) could be used for the opposite purpose.
- By far, the median filter is by far the most useful order-statistics filter in image processing.

Blurring vs. Sharpening

- as we know that blurring can be done in spatial domain by pixel averaging in a neighbors
- since averaging is analogous to integration
- thus, we can guess that the sharpening must be accomplished by spatial differentiation.

Derivative operator

- the strength of the response of a derivative operator is proportional to the degree of discontinuity of the image at the point at which the operator is applied.
- thus, image differentiation
  - enhances edges and other discontinuities (noise)
  - deemphasizes area with slowly varying gray-level values.

First-order derivative

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

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

Second-order derivative

- similarly, we define the second-order derivative of a one-dimensional function f(x) is the difference

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

First and Second-order derivative of f(x,y)

- when we consider an image function of two variables, f(x,y), at which time we will dealing with partial derivatives along the two spatial axes.

Gradient operator

$$\nabla \mathrm{f} = \frac{\partial f(x,y)}{\partial x \partial y} = \frac{\partial f(x,y)}{\partial x} + \frac{\partial f(x,y)}{\partial y}$$

Laplacian operator

(linear operator)

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

Discrete Form of Laplacian

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

yield,

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

Result Laplacian mask

0	1	0
1	−4	1
0	1	0

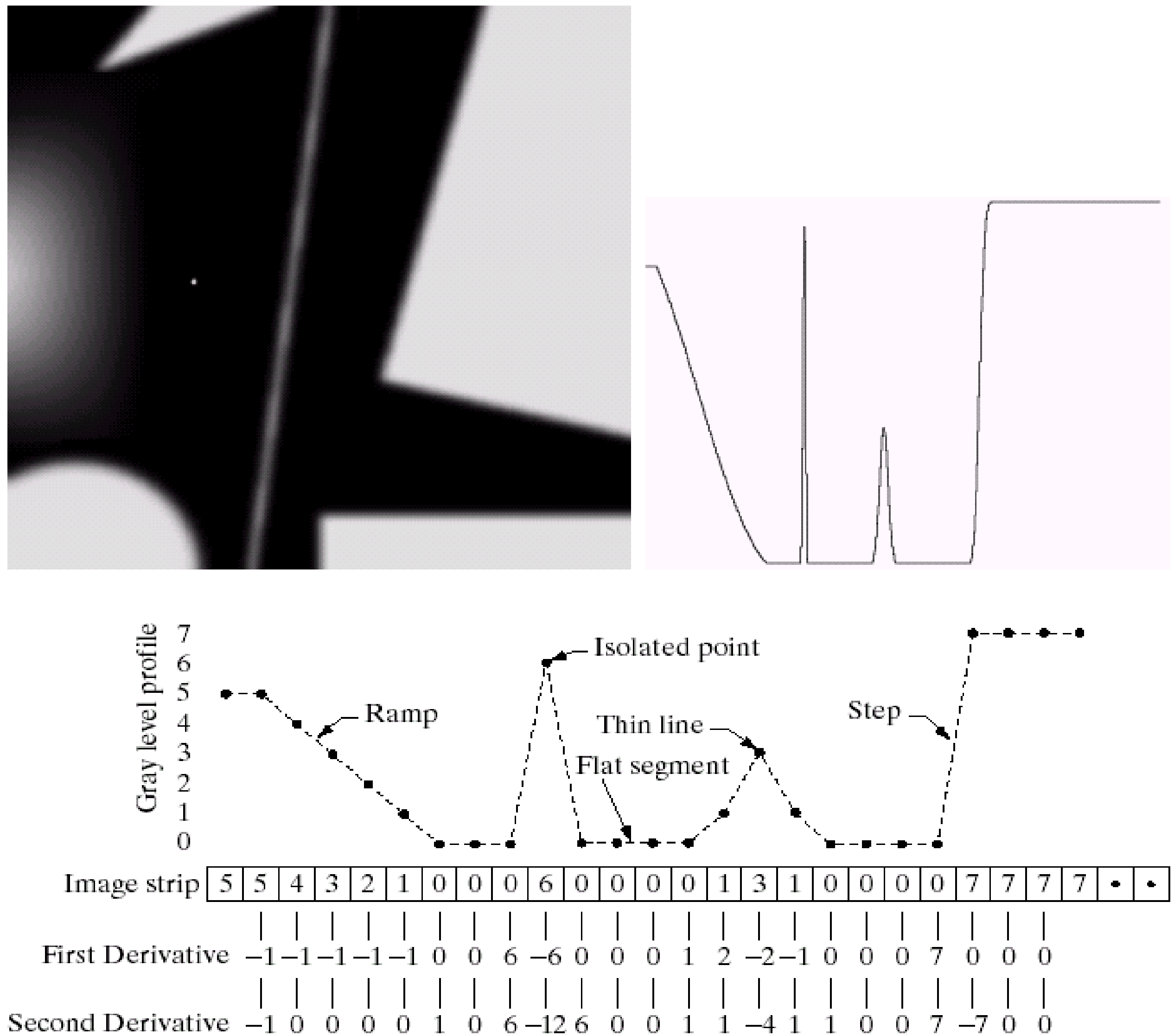
Laplacian mask implemented an extension of diagonal neighbors

1	1	1
1	−8	1
1	1	1

Example of First Derivative and Second Derivative

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).



Filter Mask Implementation of the Laplacian

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 3.39 (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.

Effect of Laplacian Operator

- as it is a derivative operator,
  - it highlights gray-level discontinuities in an image
  - it deemphasizes regions with slowly varying gray levels
- tends to produce images that have
  - grayish edge lines and other discontinuities, all superimposed on a dark,
  - featureless background.

Correct the effect of featureless background

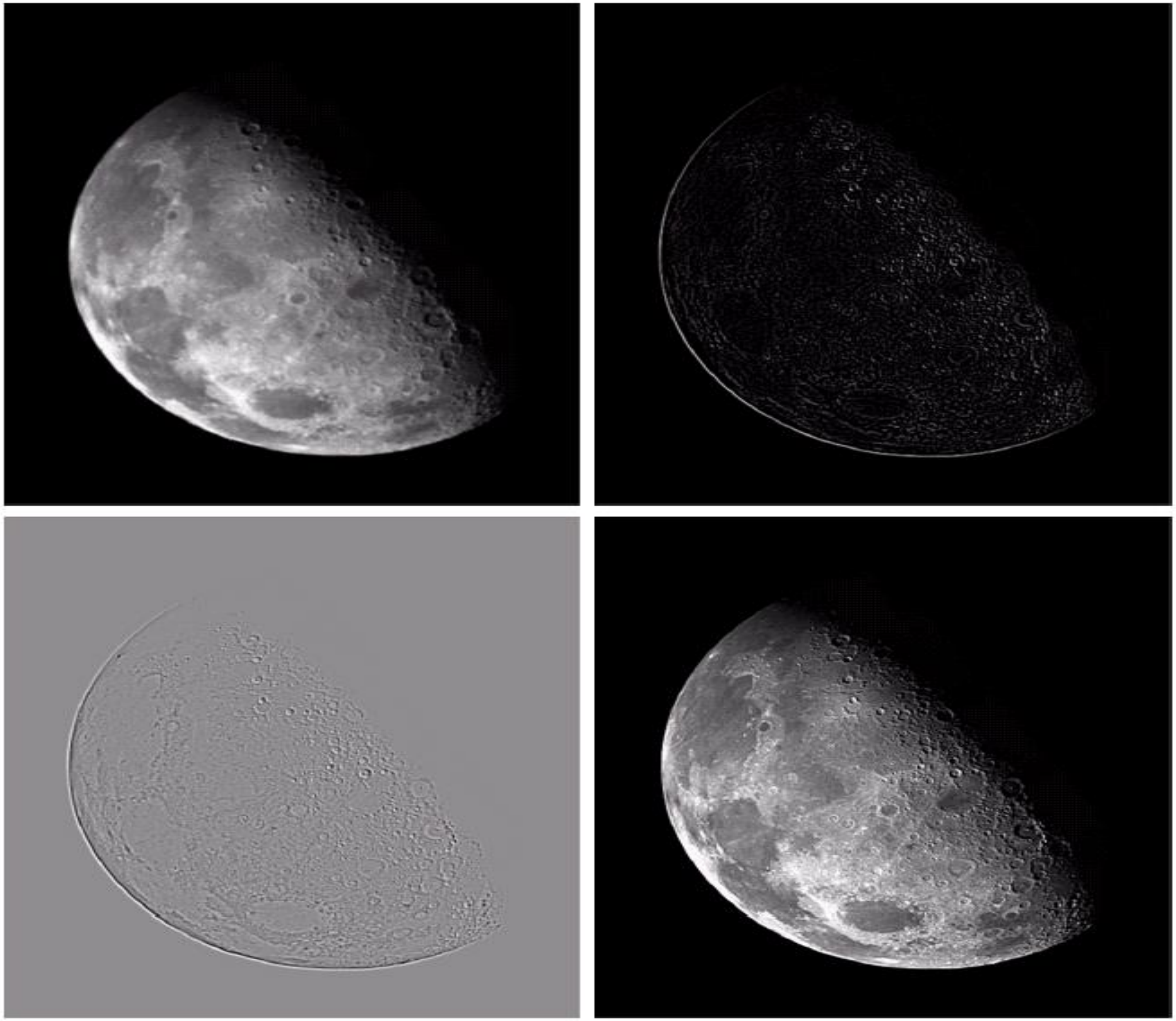
- easily by adding the original and Laplacian image.
- be careful with the Laplacian filter used

$$g(x,y) = \begin{cases} f(x,y) - \nabla^2 f(x,y) \\ f(x,y) + \nabla^2 f(x,y) \end{cases}$$

if the center coefficient of the Laplacian mask is negative

if the center coefficient of the Laplacian mask is positive

Imaging Sharpening with the Laplacian



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.)

Mask of Laplacian + addition

- to simply the computation, we can create a mask which do both operations, Laplacian Filter and Addition the original image.

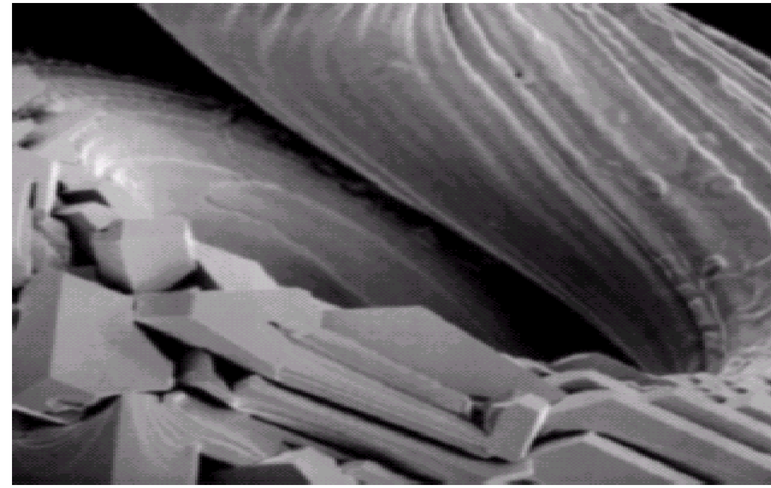
$$
\begin{aligned}
g(x,y) &= f(x,y) - [f(x+1,y) + f(x-1,y) \\
&\quad + f(x,y+1) + f(x,y-1) + 4f(x,y)] \\
&= 5f(x,y) - [f(x+1,y) + f(x-1,y) \\
&\quad + f(x,y+1) + f(x,y-1)]
\end{aligned}
$$

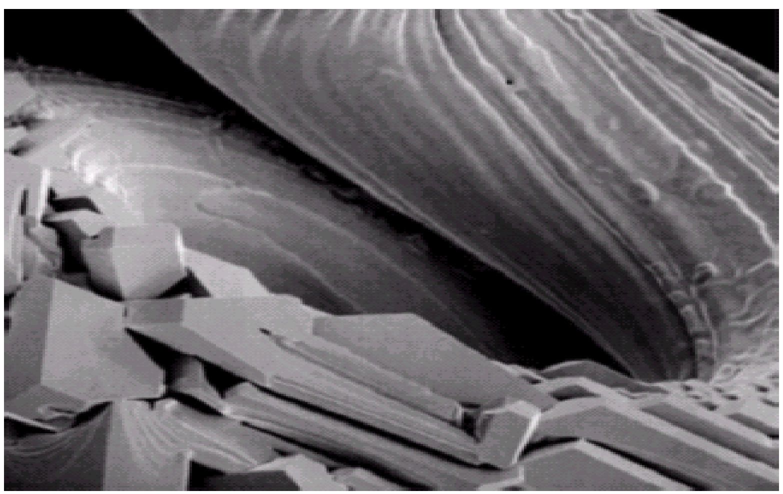
0	-1	0
-1	5	-1
0	-1	0

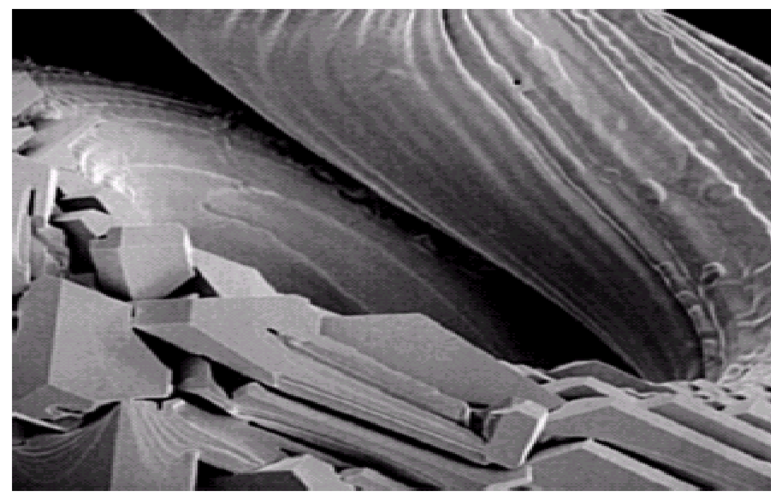
Example of Image Enhancement using a Composite Laplacian Mask

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.)

Note

$$g(x,y)=\begin{cases} f(x,y)-\nabla^2 f(x,y) \\ f(x,y)+\nabla^2 f(x,y) \end{cases}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 9 & -1 \\ 0 & -1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ -1 & 8 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Unsharp masking

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

sharpened image = original image - blurred image

- to subtract a blurred version of an image produces sharpening output image.

High-boost filtering

$$f_{hb}(x,y) = Af(x,y) - \bar{f}(x,y)$$

$$\begin{aligned} f_{hb}(x,y) &= (A-1)f(x,y) - f(x,y)\bar{f}(x,y) \\ &= (A-1)f(x,y) - f_s(x,y) \end{aligned}$$

- generalized form of Unsharp masking
- $A \geq 1$

High-boost filtering

$$f_{hb}(x,y) = (A-1)f(x,y) - f_s(x,y)$$

- if we use Laplacian filter to create sharpen image $f_s(x,y)$ with addition of original image

$$f_s(x,y) = \begin{cases} f(x,y) - \nabla^2 f(x,y) \\ f(x,y) + \nabla^2 f(x,y) \end{cases}$$

High-boost filtering

- yields

$$f_{hb}(x,y) = \begin{cases} Af(x,y) - \nabla^2 f(x,y) \\ Af(x,y) + \nabla^2 f(x,y) \end{cases}$$

if the center coefficient of the Laplacian mask is negative

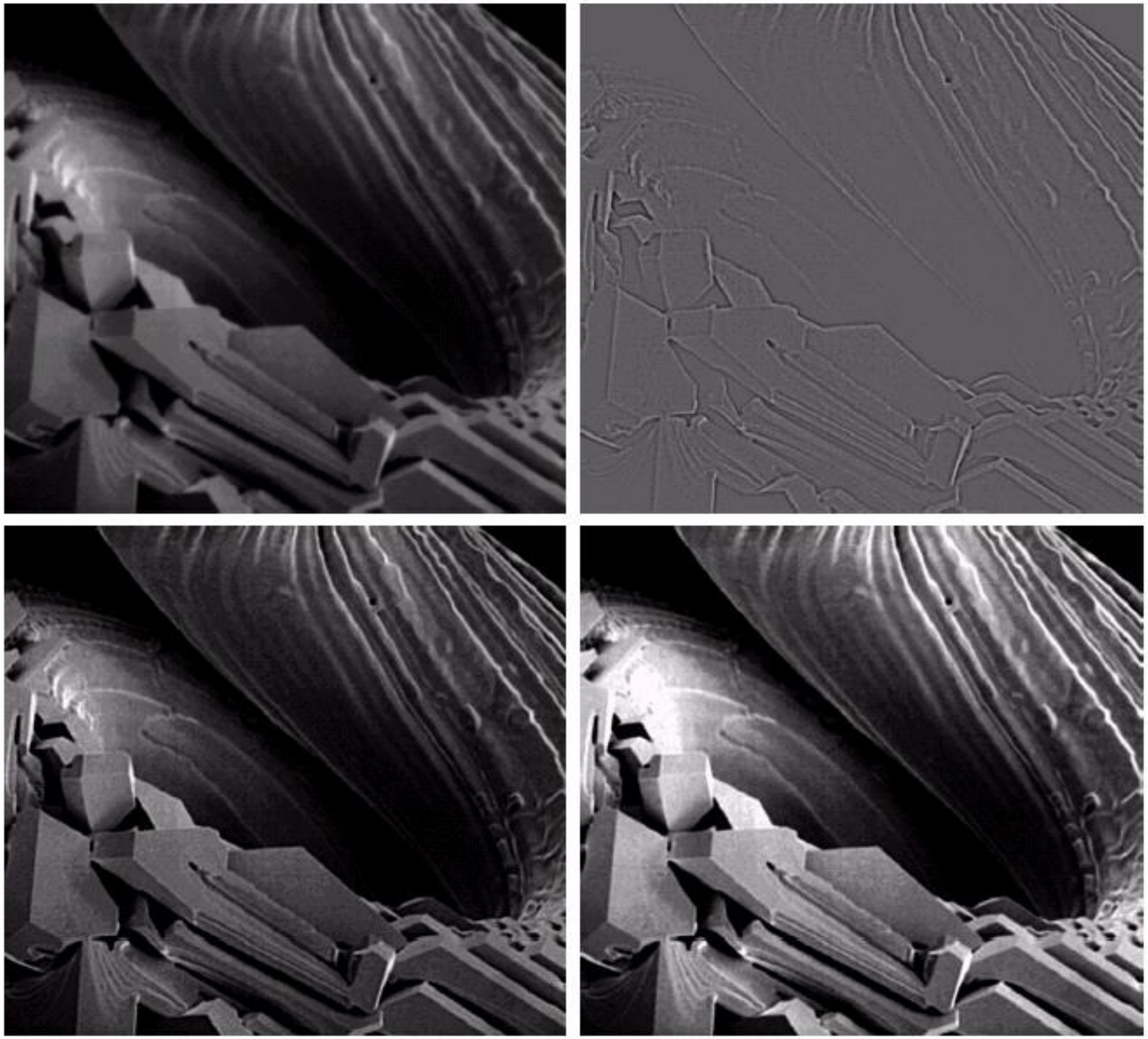
if the center coefficient of the Laplacian mask is positive

High-boost Masks

0	−1	0		−1	−1	−1
−1	$A+4$	−1		−1	$A+8$	−1
0	−1	0		−1	−1	−1

- $A \geq 1$
- if $A = 1$, it becomes "standard" Laplacian sharpening

Example: Image Enhancement with High-Boost Filter



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 $A = 1.7$.

Gradient Operator

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

- first derivatives are implemented using the magnitude of the gradient.

$$\nabla f = mag(\nabla \mathrm{f}) = [G_x^2 + G_y^2]^{1/2}$$

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

commonly approx.

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

the magnitude becomes nonlinear

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

Gradient Mask

- simplest approximation, 2x2

$$G_x = (z_8 - z_5) \quad \text{and} \quad G_y = (z_6 - z_5)$$

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

$$\nabla f \approx |z_8 - z_5| + |z_6 - z_5|$$

Gradient Mask

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

- Roberts cross-gradient operators, 2x2

$$G_x = (z_9 - z_5) \quad \text{and} \quad G_y = (z_8 - z_6)$$

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

$$\nabla f \approx |z_9 - z_5| + |z_8 - z_6|$$

-1	0
0	1

0	-1
1	0

Gradient Mask

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

- Sobel operators, 3x3

$$G_x = (z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)$$

$$G_y = (z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)$$

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

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

the weight value 2 is to achieve smoothing by giving more important to the center point

The First Derivative for Enhancement

$$
\text{(a)}\quad
\begin{bmatrix} z_1 & z_2 & z_3 \\ z_4 & z_5 & z_6 \\ z_7 & z_8 & z_9 \end{bmatrix}
$$

$$
\text{(b)}\quad
\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}
\qquad
\text{(c)}\quad
\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}
$$

$$
\text{(d)}\quad
\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}
\qquad
\text{(e)}\quad
\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}
$$

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.

Note: The summation of coefficients in all masks equal zero (0), indicating that they would give a response of 0 in the area of constant gray level.

Example: Use of Gradient for Edge Detection

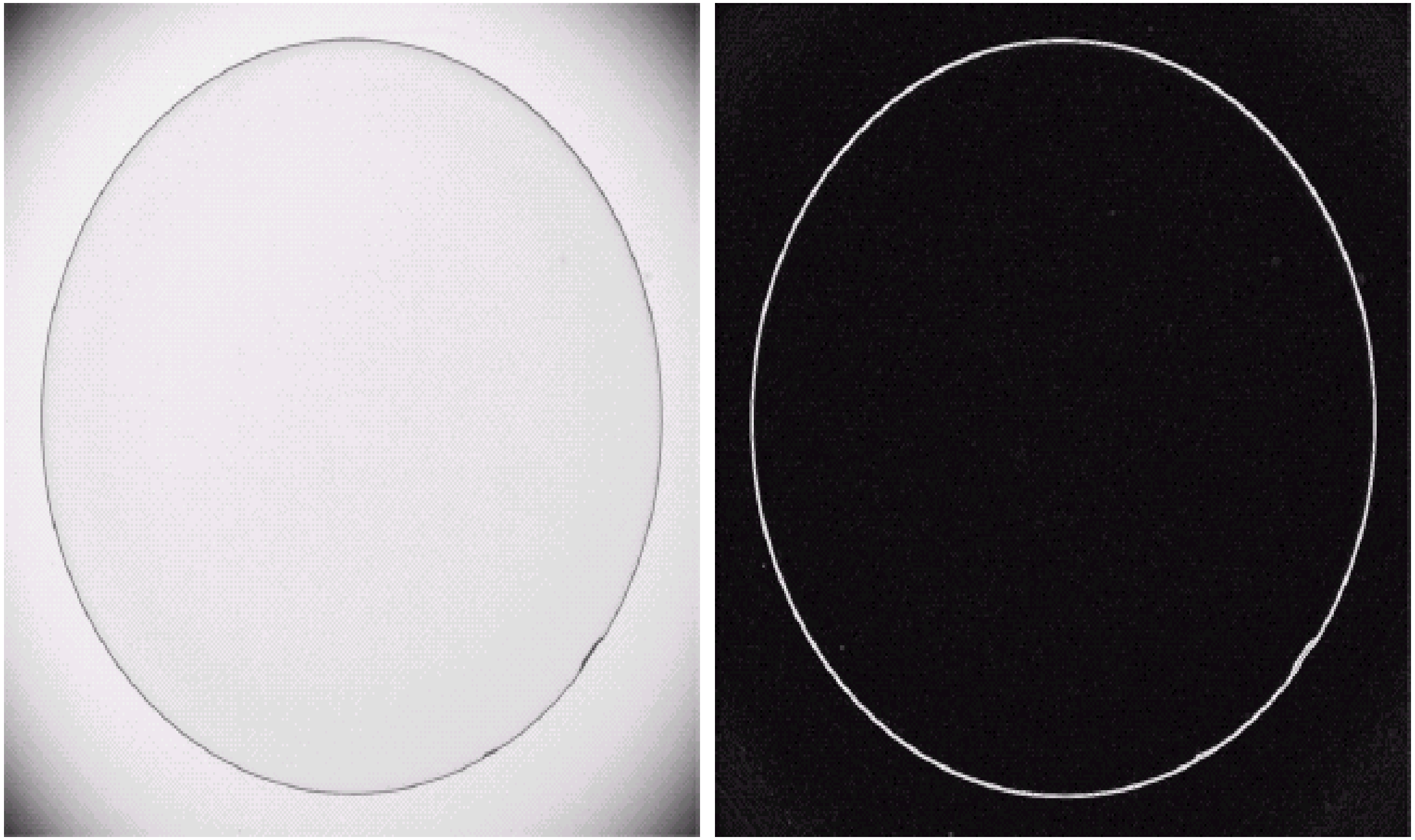


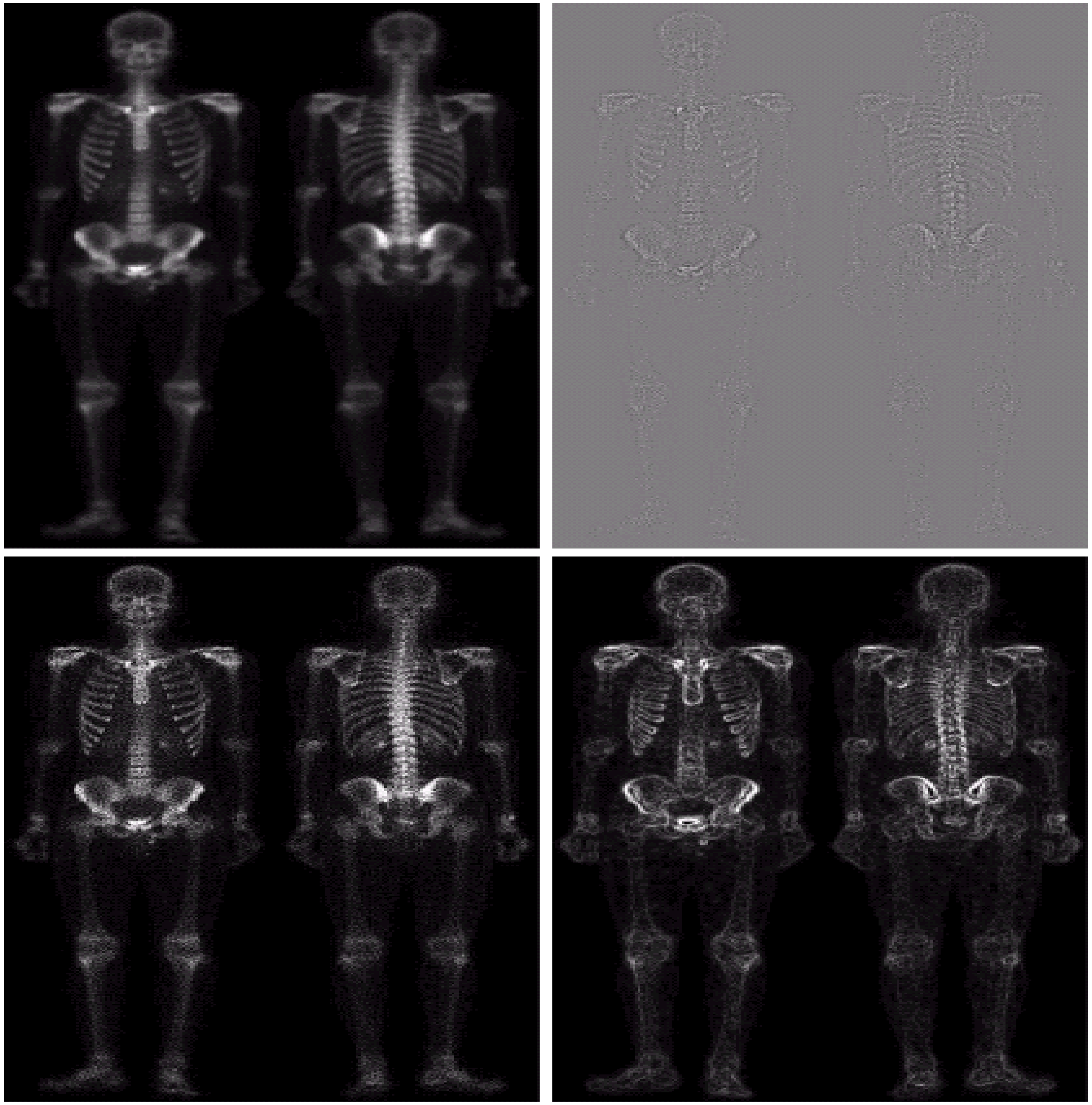
FIGURE 3.45 Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Mr. Pete Sites, Perceptics Corporation.)

Example of Combining Spatial Enhancement Methods

- solve :

1. Laplacian to highlight fine detail
2. gradient to enhance prominent edges
3. gray-level transformation to increase the dynamic range of gray levels

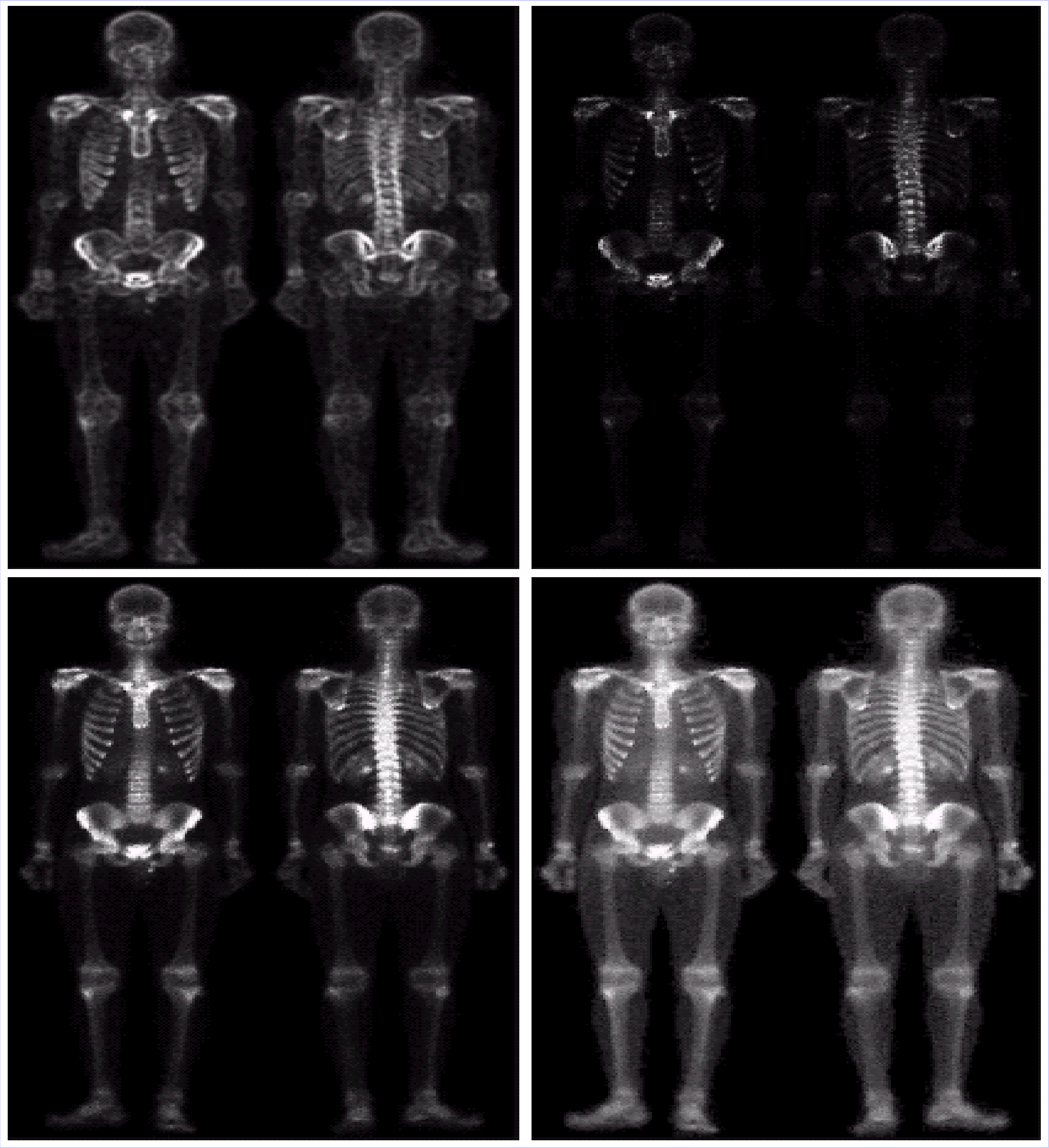
Example: Combining Spatial Enhancement Method



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).

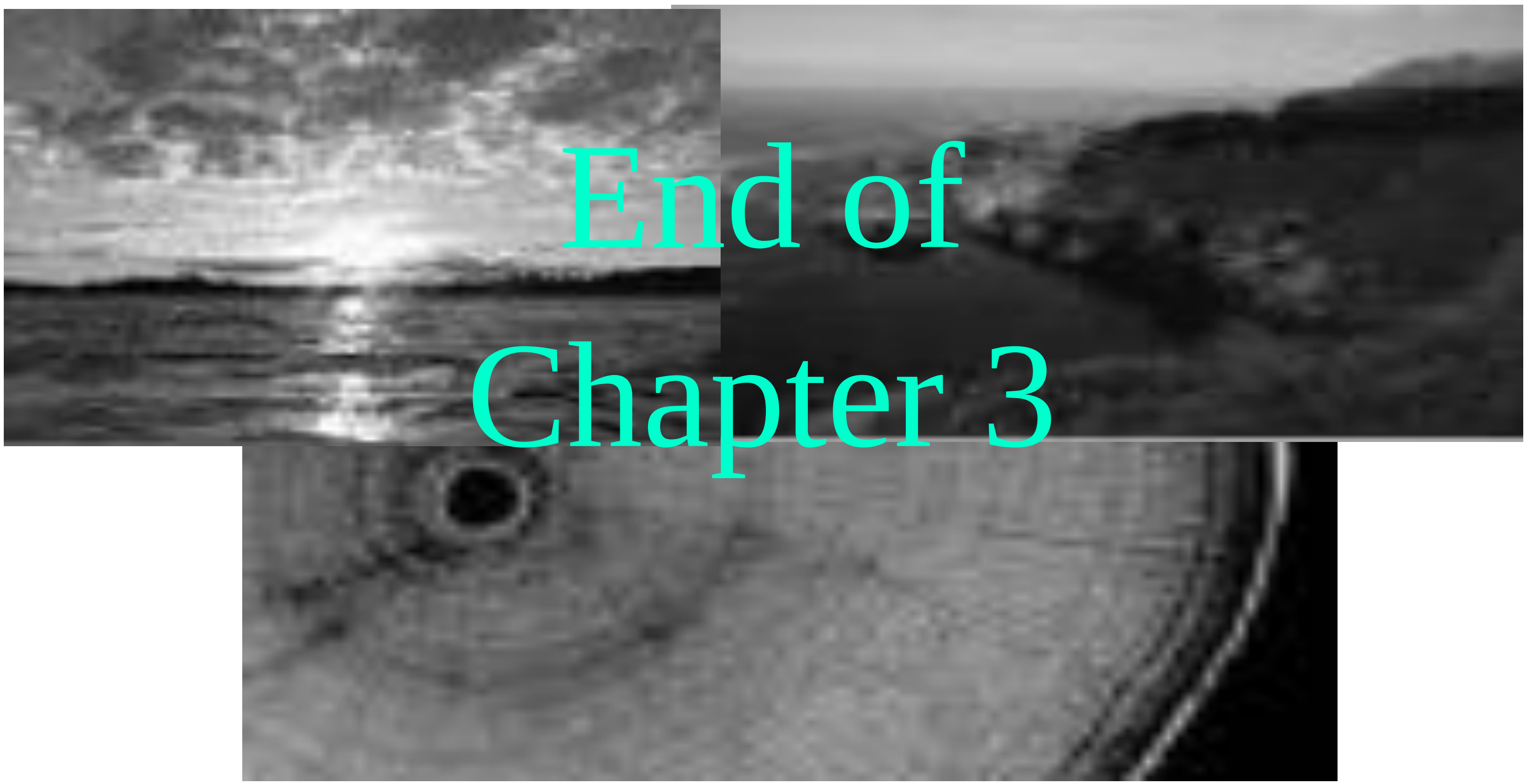
Combining Spatial Enhancement Method (cont')



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.)

Image Enhancement in Spatial Domain



End of

Chapter 3