Chapter 3 Intensity Transformations and Spatial Filtering

Spatial domain refers to the image plane itself, and image processing methods in this category are based on direct manipulation of pixels in an image.

Two principal categories of spatial processing are intensity transformations and spatial filtering.

Intensity transformations operate on single pixels of an image for the purpose of contrast manipulation and image thresholding.

Spatial filtering deals with performing operations, such as image sharpening, by working in a neighbourhood of every pixel in an image.

3.1 Background

The Basics of Intensity Transformations and Spatial Filtering

Generally, spatial domain techniques are more efficient computationally and require less processing resources to implement.

The spatial domain processes can be denoted by the expression

$$g(x, y) = T[f(x, y)] \qquad (3.1\text{-}1)$$

where $f(x, y)$ is the input image, $g(x, y)$ is the output image, and T is an operator on f defined over a neighbourhood of point (x, y). The operator can apply to a single image or to a set of images.

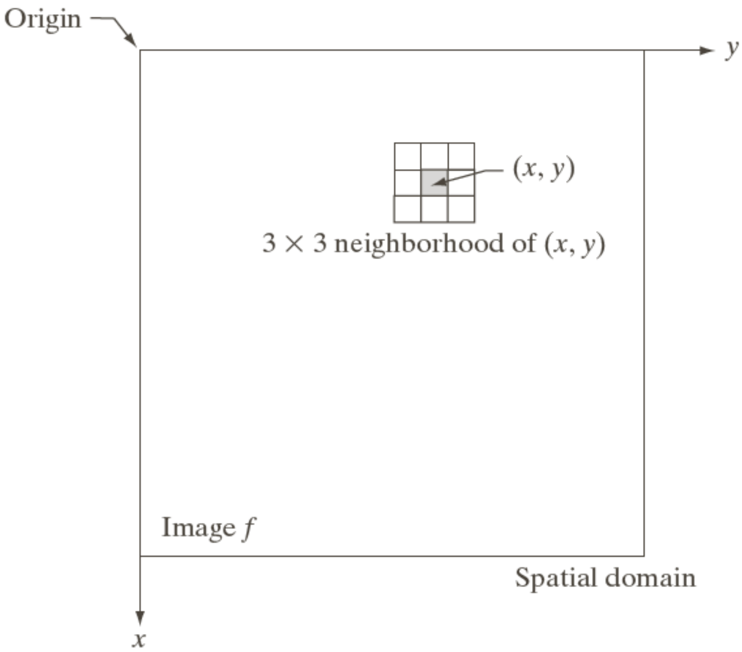


FIGURE 3.1 A 3 × 3 neighborhood about a point (x, y) in an image in the spatial domain. The neighborhood is moved from pixel to pixel in the image to generate an output image.

Typically, the neighbourhood is rectangular, centered on *(x, y)*, and much smaller than the image.

Example: Suppose that the neighbourhood is a square of size 3×3 and the operator *T* is defined as "compute the average intensity of the neighbourhood."

At an arbitrary location in an image, say *(10, 15)*, the output *g(10, 15)* is computed as the sum of *f(10, 15)* and its 8-neighbourhood is divided by *9*.

The origin of the neighbourhood is then moved to the next location and the procedure is repeated to generate the next value of the output image *g*.

The smallest possible neighbourhood is of size 1×1.

Example:

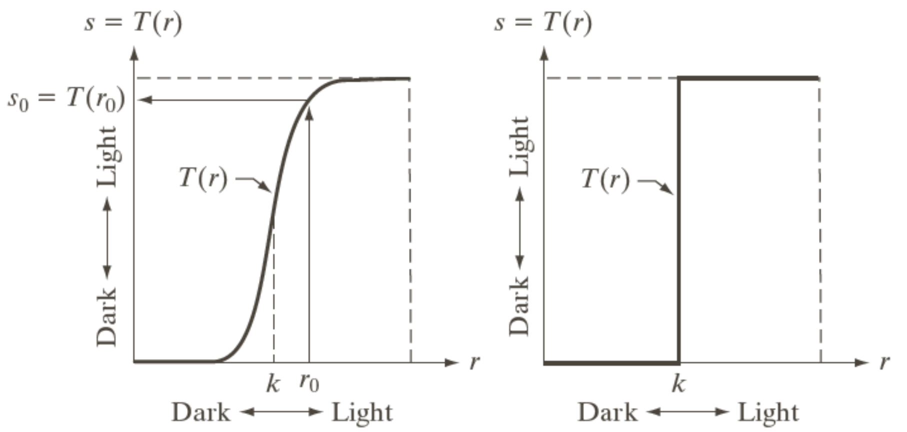


FIGURE 3.2 Intensity transformation functions. (a) Contrast-stretching function. (b) Thresholding function.

3.2 Some Basic Intensity Transformation Functions

Intensity transformations are among the simplest of all image processing techniques. We use the following expression to indicate a transformation

$$s = T(r)$$

where T is a transformation that maps a pixel value r into a pixel value s.

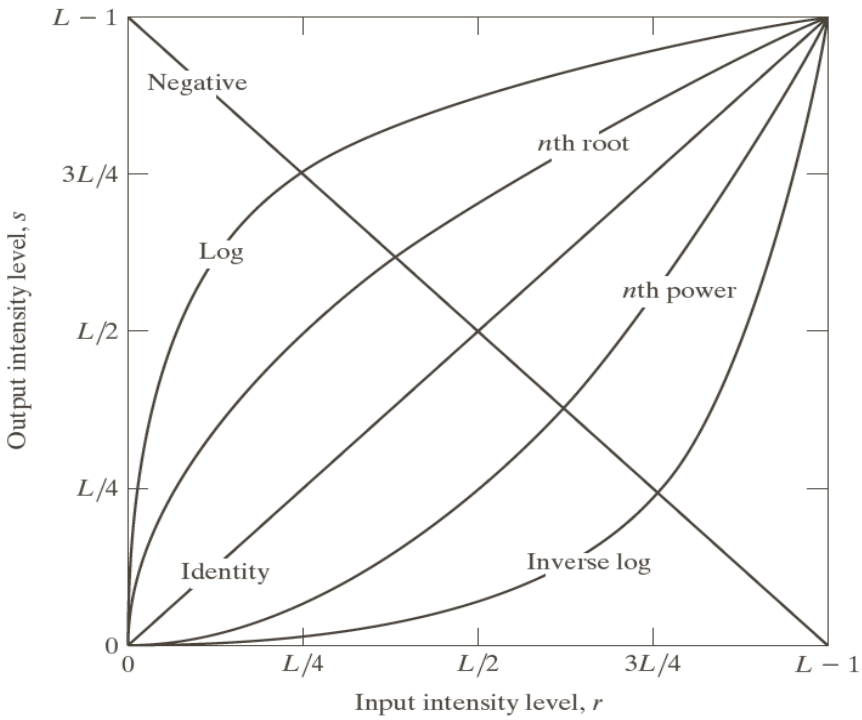


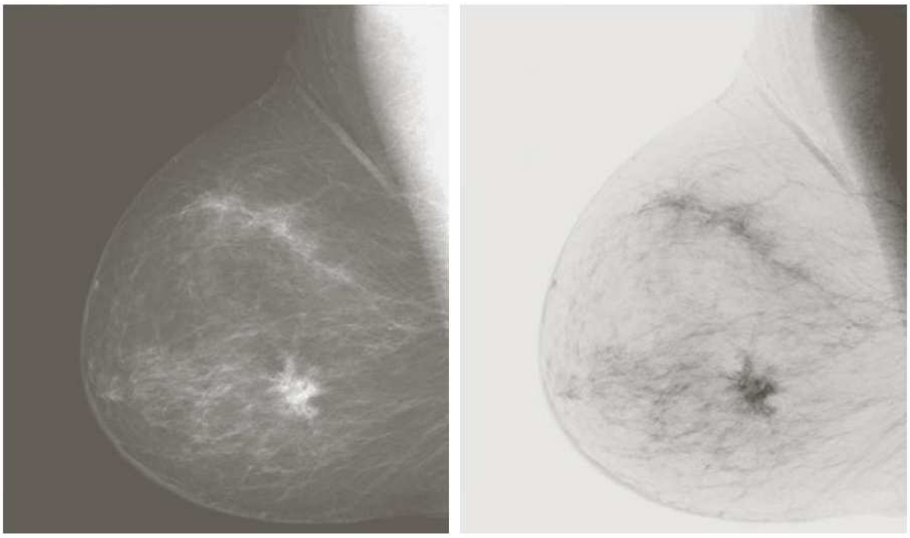
FIGURE 3.3 Some basic intensity transformation functions. All curves were scaled to fit in the range shown.

Image Negatives

The negative of an image with intensity levels in the range $[0, L-1]$ is obtained by using the negative transformation shown in Figure 3.3, which is given by

$$s = L - 1 - r \tag{3.2-1}$$

Example:



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.)

The negative transformation can be used to enhance white or gray detail embedded in dark regions of an image.

Log Transformations

The general form of the log transformations is

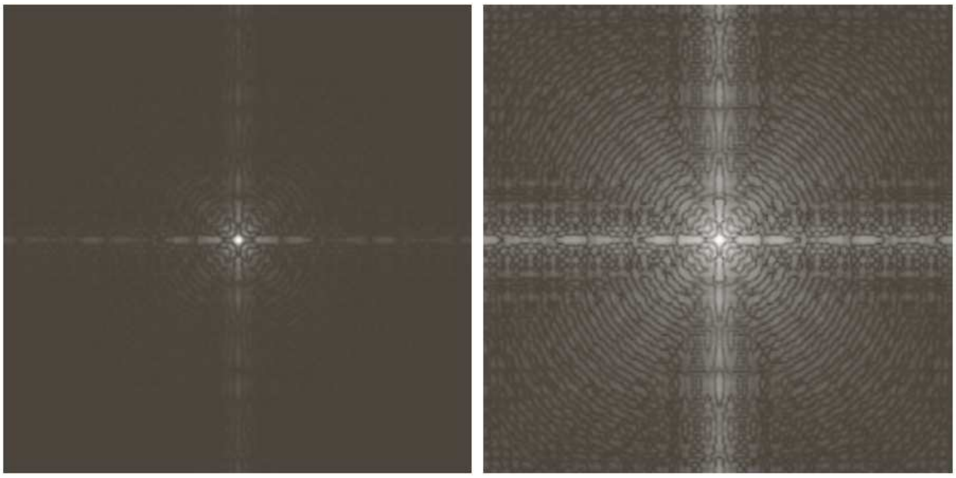
$$s = c\log(1+r) \qquad (3.2\text{-}2)$$

where c is a constant, and $r \geq 0$.

The log transformation maps a narrow range of low intensity values in the input into a wider range of output levels. We use the transformation of this type to expend the values of dark pixels in an image while compress the higher-level values.

The opposite is true of the inverse log transformation.

Example:



a b

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

Figure 3.5(a) shows a Fourier spectrum with values in the range 0 to 1.5×10^6.

Figure 3.5(b) shows the result of applying (3.2-2) to the spectrum values, which will rescale the values to a range of 0 to 6.2, and displaying the results with an 8-bit system.

Power-Law (Gamma) Transformations

Power-law transformations have the basic form

$$s = cr^{\gamma} \qquad (3.2\text{-}3)$$

where c and γ are positive constants.

A variety of devices used for image capture, printing, and display according to a power-law. By convention, the exponent in the power-law equation is referred to as gamma.

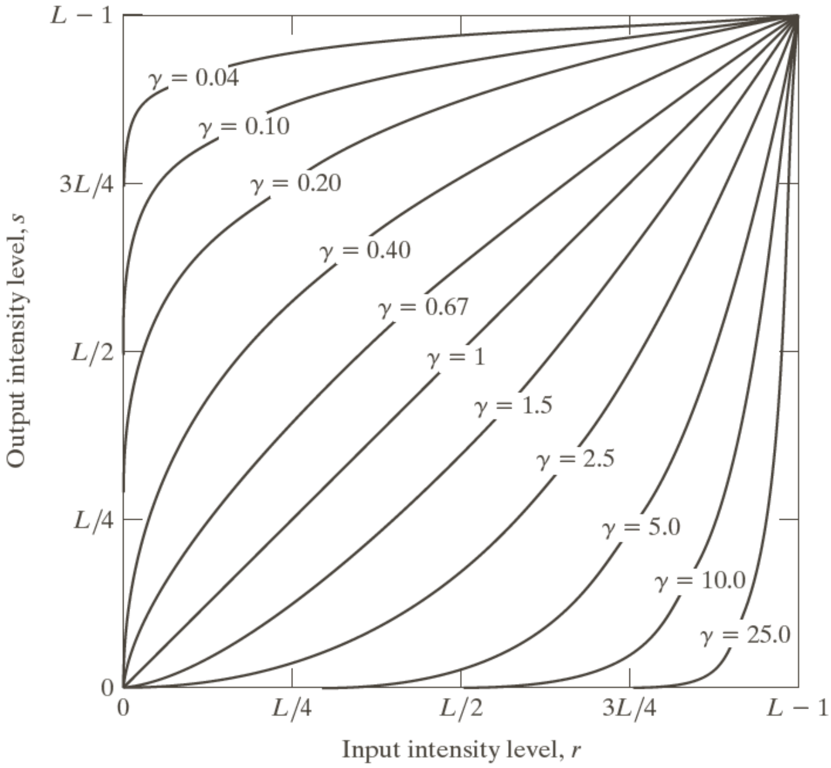
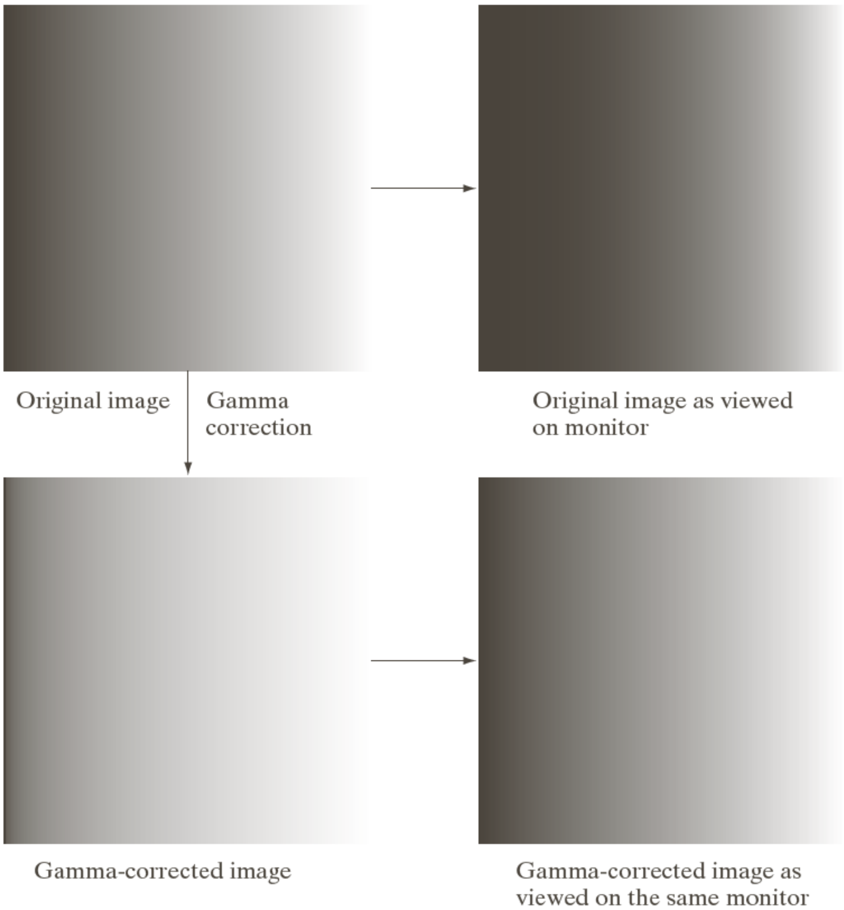


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

Unlike the *log* function, changing the value of γ will obtain a family of possible transformations. As shown in Figure 3.6, the curves generated with values of $\gamma > 1$ have exactly the opposite effect as those generated with values of $\gamma < 1$.

The process used to correct these power-law response phenomena is called gamma correction.

Example:



a b
c d

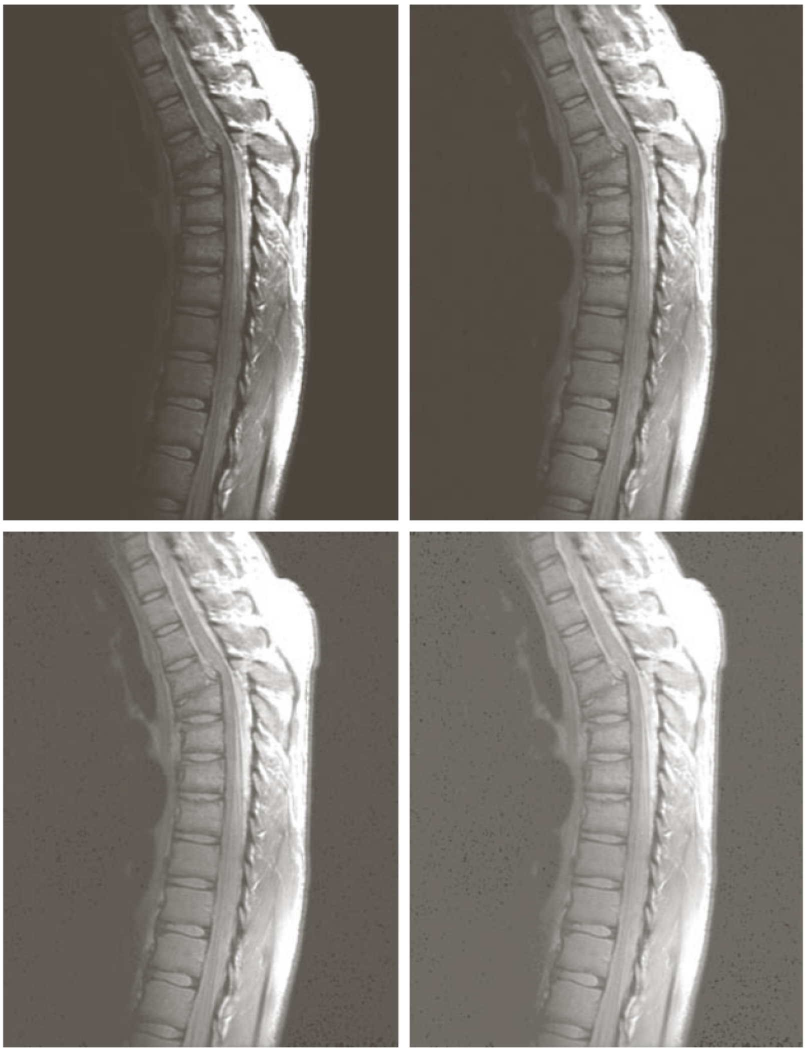
FIGURE 3.7 (a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).

Gamma correction is important if displaying an image accurately on a computer screen is of concern.

Gamma correction has become increasingly important as the use of digital images over the Internet has increased.

In addition to gamma correction, power-law transformations are very useful for general-purpose contrast manipulation.

Example 3.1: Contrast enhancement using power-law transformations.



a b
c d

FIGURE 3.8 (a) Magnetic resonance image (MRI) 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 courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

Example 3.2: Another illustration of power-law transformations.

a b
c d

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

Figure 3.9(a) shows the opposite problem of Figure 3.8(a).

Piecewise-Linear Transformation Functions

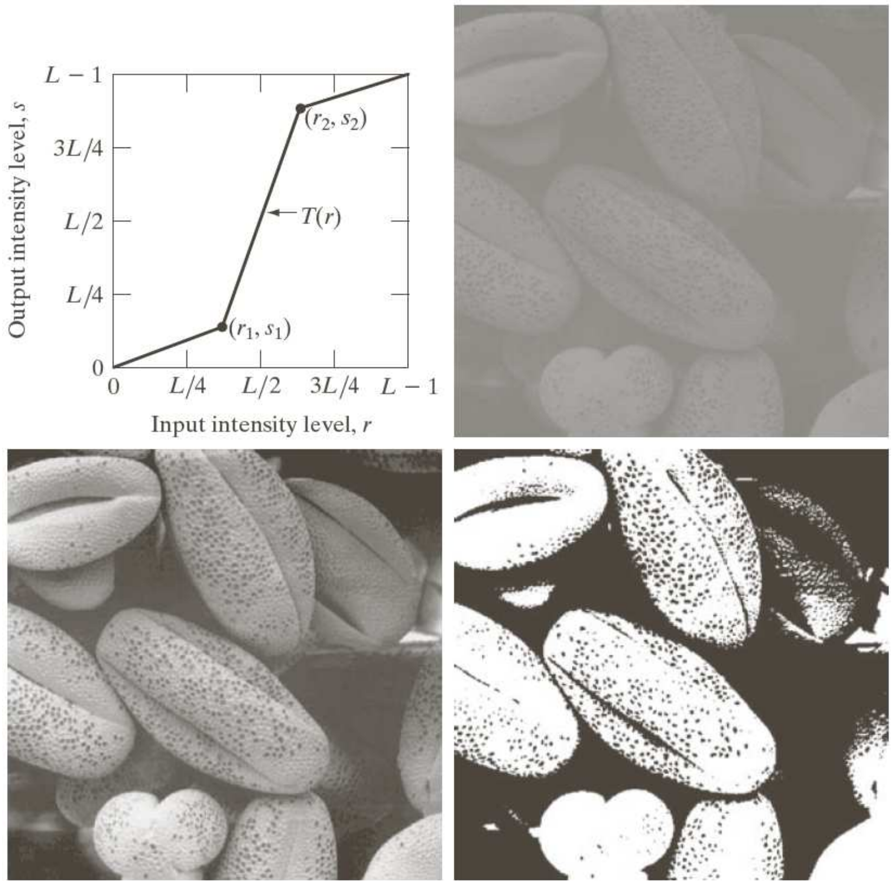
A complementary approach to the abovementioned methods is to use piecewise linear functions.

Contrast stretching

One of the simplest piecewise linear functions is a contrast-stretching transformation.

Contrast-stretching transformation is a process that expands the range of intensity levels in an image so that it spans the full intensity range of the recording medium or display device.

Example:



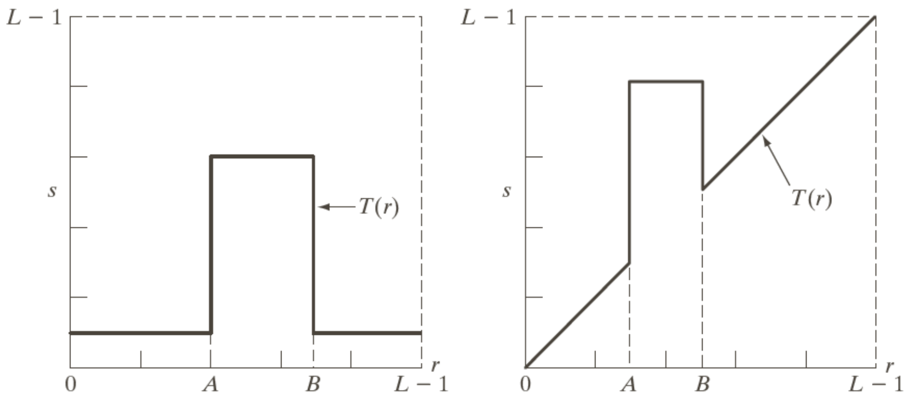
a b
c d

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.)

Intensity-level slicing

Highlighting a specific range of intensities in an image often is of interest. The process, often called intensity-level slicing, can be implemented in several ways, though basic themes are mostly used.

One approach is to display in one value all the values in the range of interest and in another all other intensities, as shown in Figure 3.11 (a).

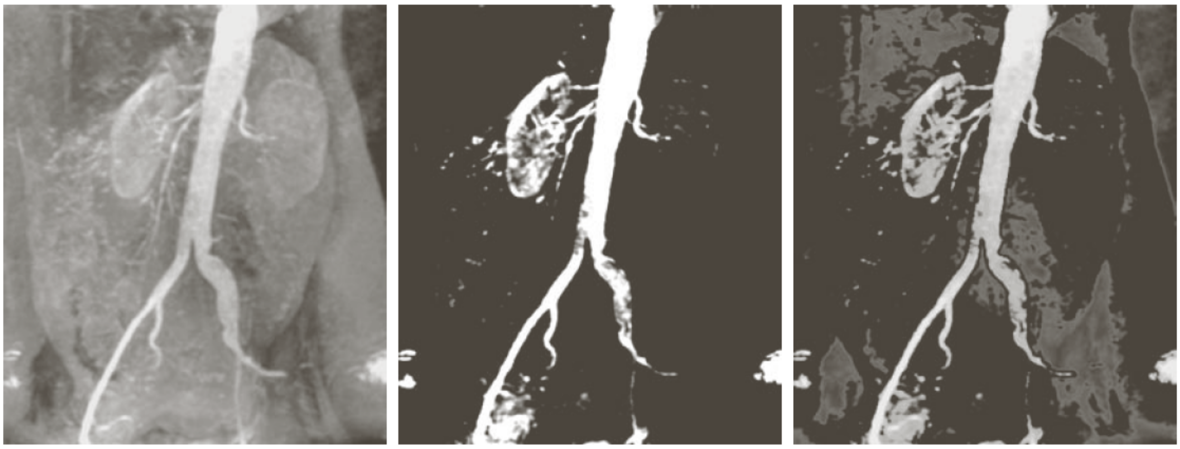


a b

FIGURE 3.11 (a) This transformation highlights intensity range [A, B] and reduces all other intensities to a lower level. (b) This transformation highlights range [A, B] and preserves all other intensity levels.

Another approach is based on the transformation in Figure 3.11(b), which brightens (or darkens) the desired range of intensities but leaves all other intensities levels in the image unchanged.

Example 3.3: Intensity-level slicing



a b c

FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that 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.)

Figure 3.12 (b) shows the result of using a transformation of the form in Figure 3.11 (a), with the selected band near the top of the scale, because the range of interest is brighter than the background.

Figure 3.12 (c) shows the result of using the transformation in Figure 3.11 (b) in which a band of intensities in the mid-gray region around the mean intensity was set to black, while all other intensities were unchanged.

Bit-plane slicing

Instead of highlighting intensity-level ranges, we could highlight the contribution made to total image appearance by specific bits.

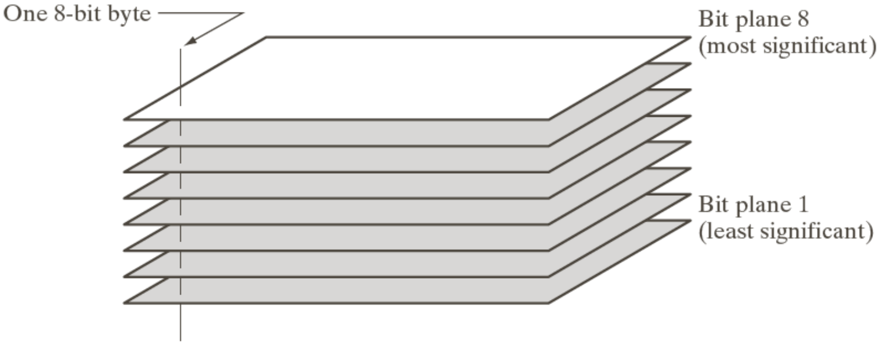


FIGURE 3.13 Bit-plane representation of an 8-bit image.

Figure 3.13 shows an 8-bit image, which can be considered as being composed of eight 1-bit planes, with plane 1 containing the lowest-order bit of all pixels in the image and plane 8 all the highest-order bits.

Example:

a b c
d e f
g h i

FIGURE 3.14 (a) An 8-bit gray-scale image of size 500 × 1192 pixels. (b) through (i) Bit planes 1 through 8, with bit plane 1 corresponding to the least significant bit. Each bit plane is a binary image.

Note that each bit plane is a binary image.

For example, all pixels in the border have values 1 1 0 0 0 0 1 0, which is the binary representation of decimal 194. Those values can be viewed in Figure 3.14 (b) through (i).

Decomposing an image into its bit planes is useful for analyzing the relative importance of each bit in the image.

Example:

a b c

FIGURE 3.15 Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

3.3 Histogram Processing

The histogram of a digital image with intensity levels in the range $[0, L-1]$ is a discrete function $h(r_k) = n_k$, where r_k is the kth intensity value and n_k is the number of pixels in the image with intensity r_k.

It is common practice to normalize a histogram by diving each of its components by the total number of pixels in the image, denoted by MN, where M and N are the row and column dimensions of the image.

A normalized histogram is given by

$$p(r_k) = \frac{n_k}{MN}, \text{ for } k = 0, 1, 2, \ldots, L-1.$$

$p(r_k)$ can be seen as an estimate of the probability of occurrence of intensity level r_k in an image. The sum of all components of a normalized histogram is equal to 1.

Histograms are the basic for numerous spatial domain processing techniques.

Example:

Figure 3.16, which is the pollen image of Figure 3.10 shown in four basic intensity characteristics: dark, light, low contrast, and high contrast, shows the histograms corresponding to these image.

The vertical axis corresponds to value of $h(r_k) = n_k$ or $p(r_k) = n_k / MN$ if the values are normalized.

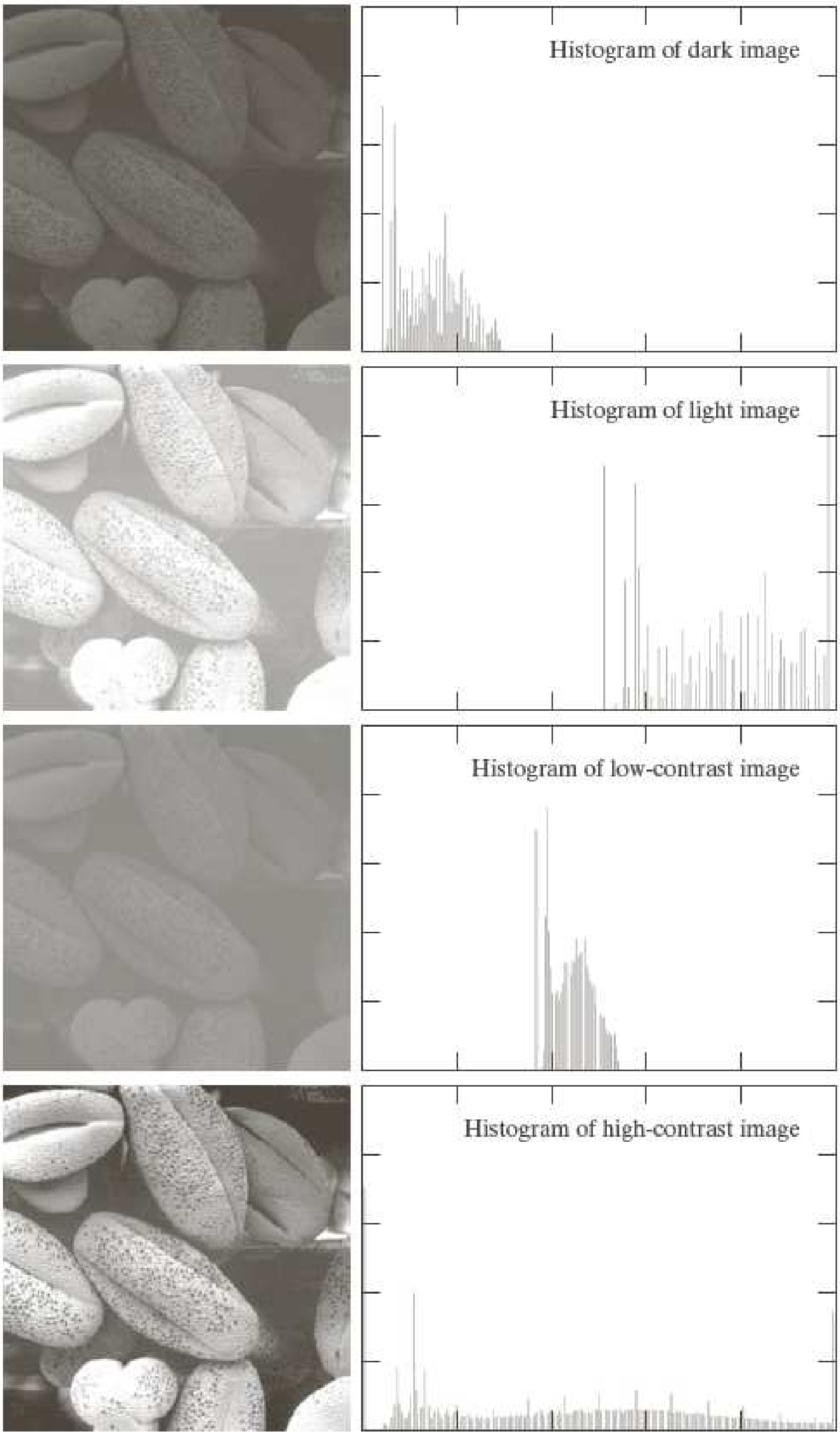


FIGURE 3.16 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms.

Histogram Equalization

We consider the continuous intensity values and let the variable r denote the intensities of an image. We assume that r is in the range $[0, L-1]$.

We focus on transformations (intensity mappings) of the form

$$s = T(r) \quad 0 \le r \le L-1 \tag{3.3-1}$$

that produce an output intensity level s for every pixel in the input image having intensity r. Assume that

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

(b) $0 \le T(r) \le L-1$ for $0 \le r \le L-1$.

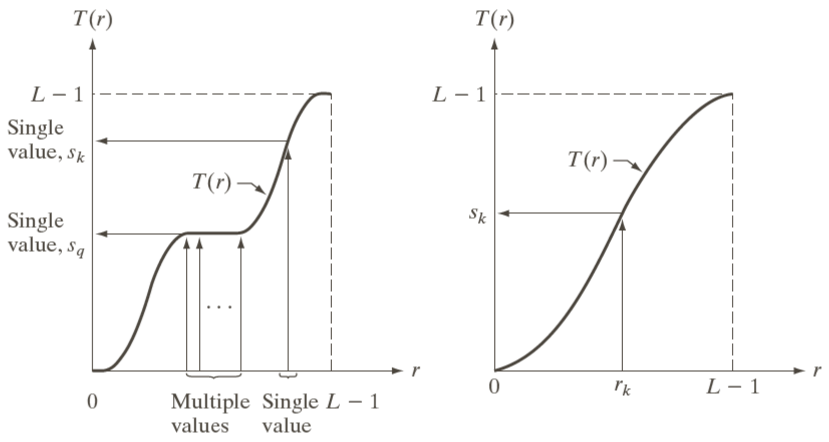
In some formations to be discussed later, we use the inverse

$$r = T^{-1}(s) \quad 0 \le s \le L-1 \tag{3.3-2}$$

in which case we change condition (a) to

(a') $T(r)$ is a strictly monotonically increasing function in the interval $0 \le r \le L-1$.

Figure 3.17 (a) shows a function that satisfies conditions (a) and (b).



a b

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

From Figure 3.17 (a), we can see that it is possible for multiple values to map to a single value and still satisfy these two conditions, (a) and (b). That is, a monotonic transformation function can perform a one-to-one or many-to-one mapping, which is perfectly fine when mapping from r to s.

However, there will be a problem if we want to recover the values of r uniquely from the mapped values.

As Figure 3.17 (b) shows, requiring that $T(r)$ be strictly monotonic guarantees that the inverse mappings will be single valued. This is a theoretical requirement that allows us to derive some important histogram processing techniques.

The intensity levels in an image may be viewed as random variables in the interval $[0, L-1]$. A fundamental descriptor of a random variable is its probability density function (PDF).

Let $p_r(r)$ and $p_s(s)$ denote the probability density functions of r and s. A fundamental result from basic probability theory is that if $p_r(r)$ and $T(r)$ are known, and $T(r)$ is continuous and differentiable over the range of values of interest, then the PDF of the transformed variable s can be obtained using the formula

$$p_s(s) = p_r(\mathrm{r})\left|\frac{dr}{ds}\right| \qquad (3.3\text{-}3)$$

A transformation function of particular importance in image processing has the form

$$s = T(\mathrm{r}) = (L-1)\int_0^r p_r(\omega)d\omega \qquad (3.3\text{-}4)$$

where ω is a dummy variable of integration.

The right side of (3.3-4) is recognized as the cumulative distribution function of random variable r. Since PDFs always are positive, the transformation function of (3.3-4) satisfies condition (a) because the area under the function cannot decreases as r increases.

When the upper limit in (3.3-4) is $r = (L-1)$, the integral evaluates to 1 (the area under a PDF curve always is 1), so the maximum value of s is $(L-1)$ and condition (b) satisfies as well.

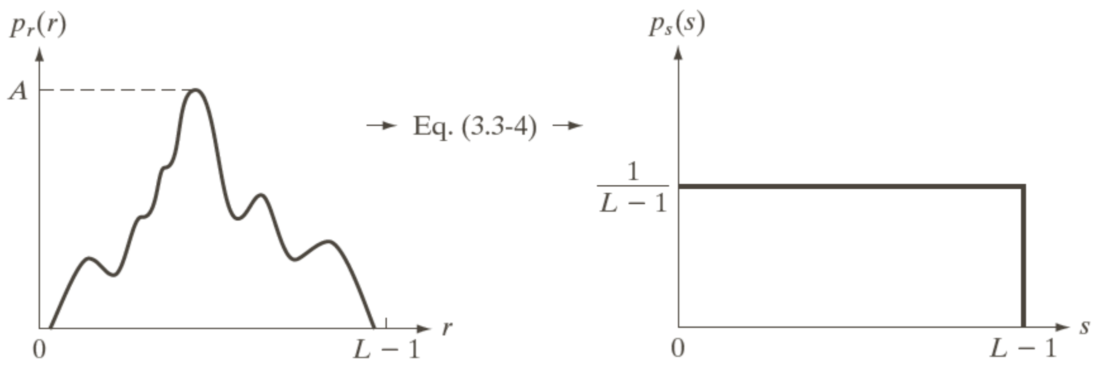
Using (3.3-3) and recalling the Leibniz's rule that saying the derivative of a definite integral with respect to its upper limit is the integrand evaluated at the limit, we have

$$\begin{aligned}\frac{ds}{dr} &= \frac{dT(r)}{dr} \\ &= (L-1)\frac{d}{dr}\left[\int_0^r p_r(\omega)d\omega\right] \\ &= (L-1)p_r(r)\end{aligned} \qquad (3.3\text{-}5)$$

Substituting this result for dr/ds in (3.3-3), yields

$$\begin{aligned} p_s(s) &= p_r(r)\left|\frac{dr}{ds}\right| \\ &= p_r(r)\left|\frac{1}{(L-1)p_r(r)}\right| \qquad\qquad (3.3\text{-}6) \\ &= \frac{1}{L-1} \qquad 0 \le s \le L-1 \end{aligned}$$

which shows the that $p_s(s)$ always is uniform, independently of the form of $p_r(r)$.



a b

FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying the transformation in Eq. (3.3-4) to all intensity levels, r. The resulting intensities, s, have a uniform PDF, independently of the form of the PDF of the r's.

Example 3.4: Illustration of (3.3-4) and (3.3.6)

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

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

From (3.3-4),

$$s = T(r) = (L-1)\int_0^r p_r(\omega)d\omega \qquad (3.3\text{-}4)$$
$$= \frac{2}{(L-1)}\int_0^r \omega d\omega = \frac{r^2}{L-1}$$

Consider an image in which $L = 10$, and suppose that a pixel at (x, y) in the input image has intensity $r = 3$. Then, the pixel at (x, y) in the new image is $s = T(r) = r^2/9 = 1$.

We can versify that the PDF of the intensities in the new image is uniform by substituting $p_r(r)$ into (3.3-6) and using the facts that $s = r^2/(L-1)$, r is nonnegative, and $L > 1$:

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| \qquad (3.3\text{-}6)$$
$$= \frac{2r}{(L-1)^2}\left|\left[\frac{ds}{dr}\right]^{-1}\right|$$
$$= \frac{2r}{(L-1)^2}\left|\left[\frac{d}{dr}\frac{r^2}{L-1}\right]^{-1}\right|$$
$$= \frac{2r}{(L-1)^2}\left|\frac{(L-1)}{2r}\right| = \frac{1}{L-1}$$

For discrete values, we deal with probabilities (histogram values) and summations instead of probability density functions and integrals.

The probability of occurrence of intensity level r_k in a digital image is approximated by

$$p_r(r_k) = \frac{n_k}{MN} \qquad k = 0, 1, 2, \ldots, L-1 \tag{3.3-7}$$

where MN is the total number of pixels in the image, n_k is the number of pixels having intensity r_k, and L is the number of possible intensity levels in the image.

The discrete form of the transformation in

$$s = T(r) = (L-1)\int_0^r p_r(\omega)\,d\omega \tag{3.3-4}$$

is

$$\begin{aligned} s_k &= T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j) \\ &= \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j \qquad k = 0, 1, 2, \ldots, L-1 \end{aligned} \tag{3.3-8}$$

The transformation (mapping) $T(r_k)$ in (3.3-8) is called a histogram equalization transformation.

Example 3.5: A simple illustration of history equalization.

Suppose that a 3-bit image ($L = 8$) of size 64×64 pixels ($MN = 4096$) has the intensity distribution shown in Table 3.1.

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

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

The histogram of our hypothetical image is sketched in Figure 3.19 (a).

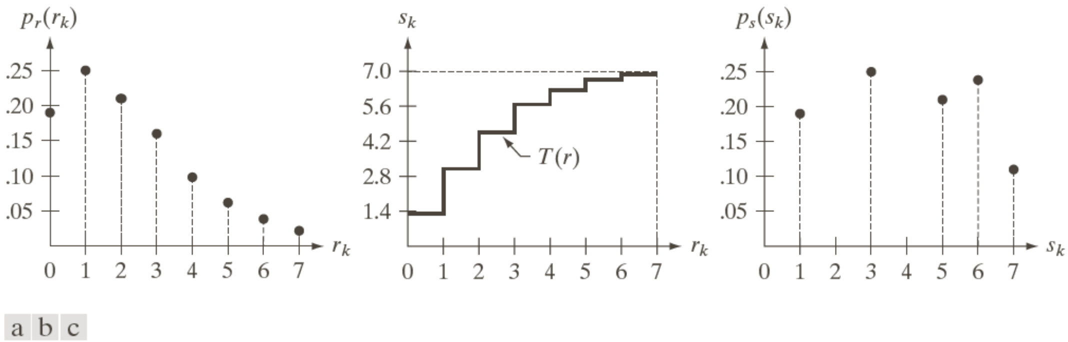


FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

By using (3.3-8), we can obtain values of the histogram equalization function:

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7p_r(r_0) = 1.33,$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7p_r(r_0) + 7p_r(r_1) = 3.08,$$

$s_2 = 4.55$, $s_3 = 5.67$, $s_4 = 6.23$, $s_5 = 6.65$, $s_6 = 6.86$, and $s_7 = 7.00$. This function is shown in Figure 3.19 (b).

Then, we round them to the nearest integers:

$$s_0 = 1.33 \to 1 \quad s_1 = 3.08 \to 3 \quad s_2 = 4.55 \to 5 \quad s_3 = 5.67 \to 6$$
$$s_4 = 6.23 \to 6 \quad s_5 = 6.65 \to 7 \quad s_6 = 6.86 \to 7 \quad s_7 = 7.00 \to 7$$

which are the values of the equalized histogram.

Observe that there are only five distinct levels:

$s_0 \to 1$:	790 pixels
$s_1 \to 3$:	1023 pixels
$s_2 \to 5$:	850 pixels
$s_3 \to 6$:	985 (656+329) pixels
$s_5 \to 7$:	448 (245+122+81) pixels
Total:	4096

Dividing these numbers by $MN = 4096$ would yield the equalized histogram shown in Figure 3.19 (c).

Since a histogram is an approximation to probability density function, and no new allowed intensity levels are created in the process, perfectly flat histograms are rare in practical applications of histogram equalization.

Therefore, in general, it cannot be proved that discrete histogram equalization results in a uniform histogram.

Given an image, the process of histogram equalization consists simply of implementing

$$s_k = \frac{(L-1)}{MN}\sum_{j=0}^{k} n_j \quad , \qquad (3.3\text{-}8)$$

which is based on information that can be extracted directly from the given image, without the need for further parameter specifications.

The inverse transformation from s back to r is denoted by

$$r_k = T^{-1}(s_k) \qquad k = 0, 1, 2, \ldots, L-1 \qquad (3.3\text{-}9)$$

Although the inverse transformation is not used in the histogram equalization, it plays a central role in the histogram-matching scheme.

Example 3.6: Histogram equalization

The left column in Figure 3.20 shows the four images from Figure 3.16.

The center column in Figure 3.20 shows the result of performing histogram equalization on each of the images in left.

The histogram equalization did not have much effect on the fourth image because the intensities of this image already span the full intensity scale.

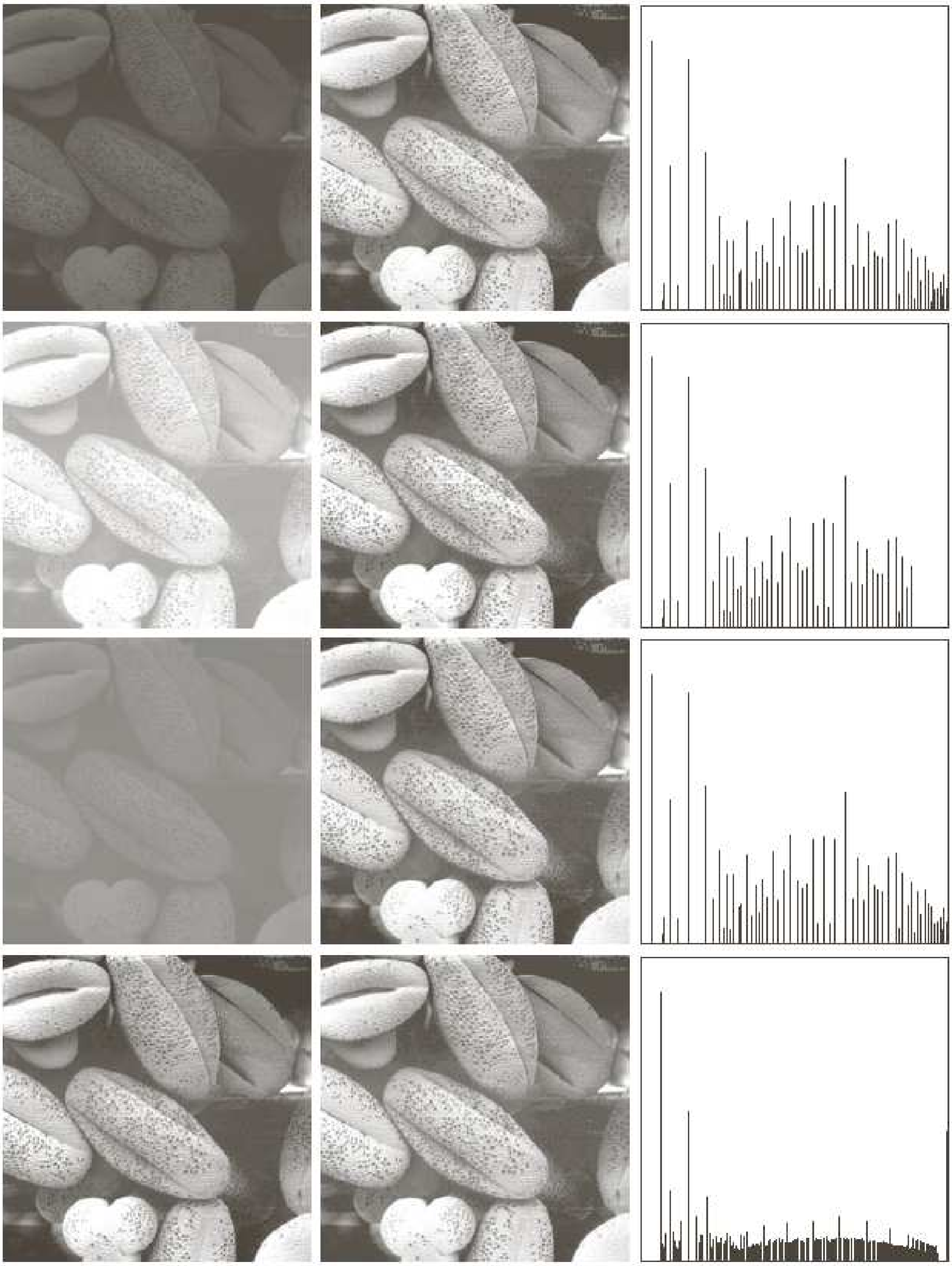


FIGURE 3.20 Left column: images from Fig. 3.16. Center column: corresponding histogram-equalized images. Right column: histograms of the images in the center column.

Figure 3.21 shows the transformation functions used to generate the equalized images in Figure 3.20.

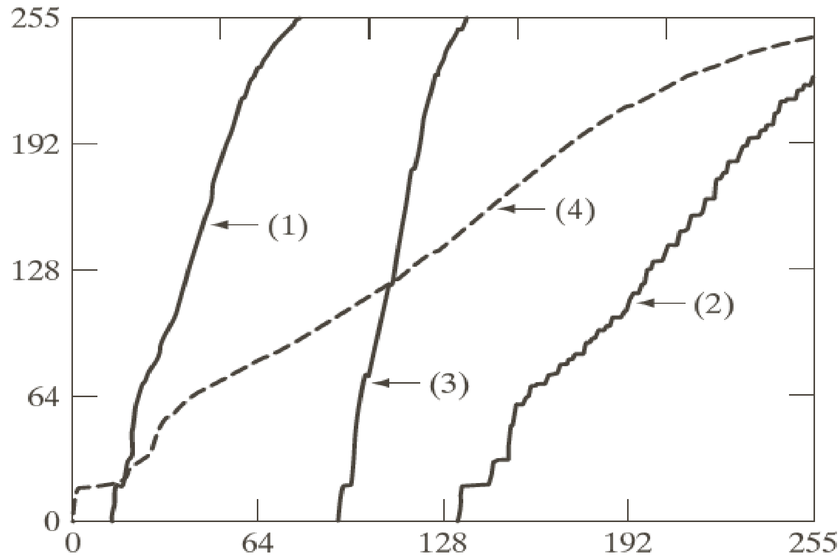


FIGURE 3.21 Transformation functions for histogram equalization. Transformations (1) through (4) were obtained from the histograms of the images (from top to bottom) in the left column of Fig. 3.20 using Eq. (3.3-8).