Image Manipulation in Spatial Domain

Image Enhancement I
(Intensity Transformations)

January 16, 22, 23, 2019

Image Enhancement

- Enhancement: to process an image so that the result is more suitable than the original image for a specific application.
- Enhancement approaches:

  1. Spatial domain
  2. Frequency domain

- Spatial domain (image plane) techniques are techniques that operate directly on pixels.
- 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 Images

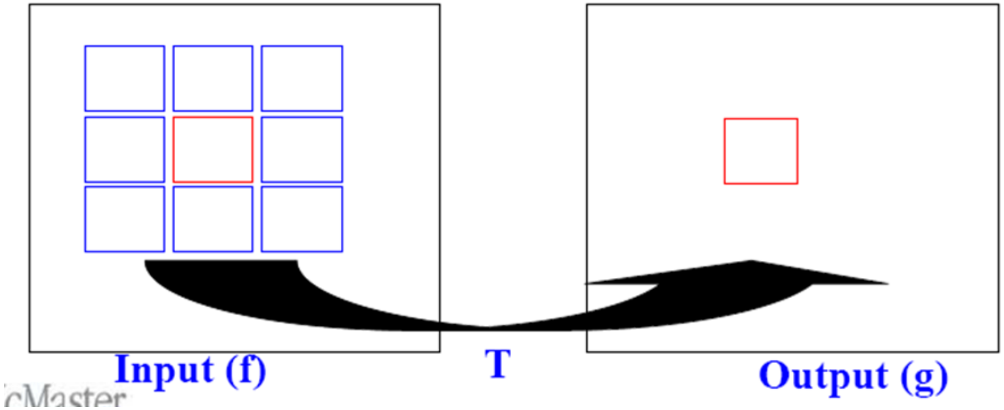
- For human visual
  - The visual evaluation of image quality is a highly subjective process.
  - It is hard to standardize the definition of a good image.
- For machine perception
  - The evaluation task is easier.
  - A good image is one which gives the best machine recognition results.
- A certain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

Spatial Domain: Point Processing

- $g(x,y)=T[f(x,y)]$

$f(x,y)$: input image, $g(x,y)$: processed image, T: an operator

- Simplest form of T
  - Neighborhood is of size 1 x 1 (single pixel)

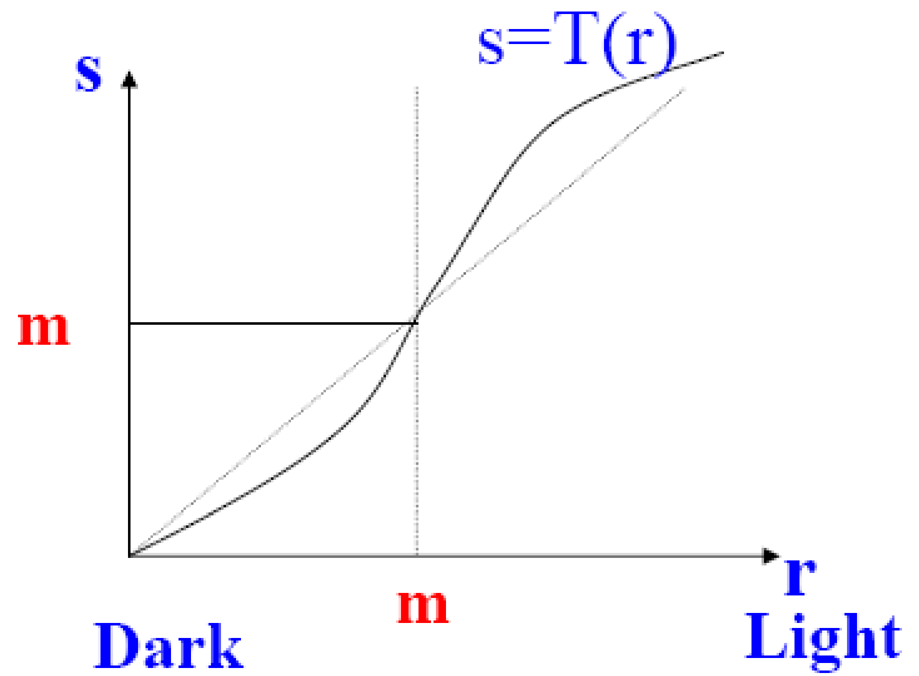


- $s=T(r)$
- r: gray-level at (x,y) in original image $f(x,y)$
- s: gray-level at (x,y) in processed image $g(x,y)$
- T is called gray-level transformation or mapping

Spatial Domain: Point Processing

- Point Processing Techniques: enhancement at any point in an image depends only on the gray level at that point.
- Contrast Stretching: to get an image with higher contrast than the original image.
- The gray levels below m are darkened and the levels above m are brightened.

Contrast Stretching



Spatial Domain: Point Processing

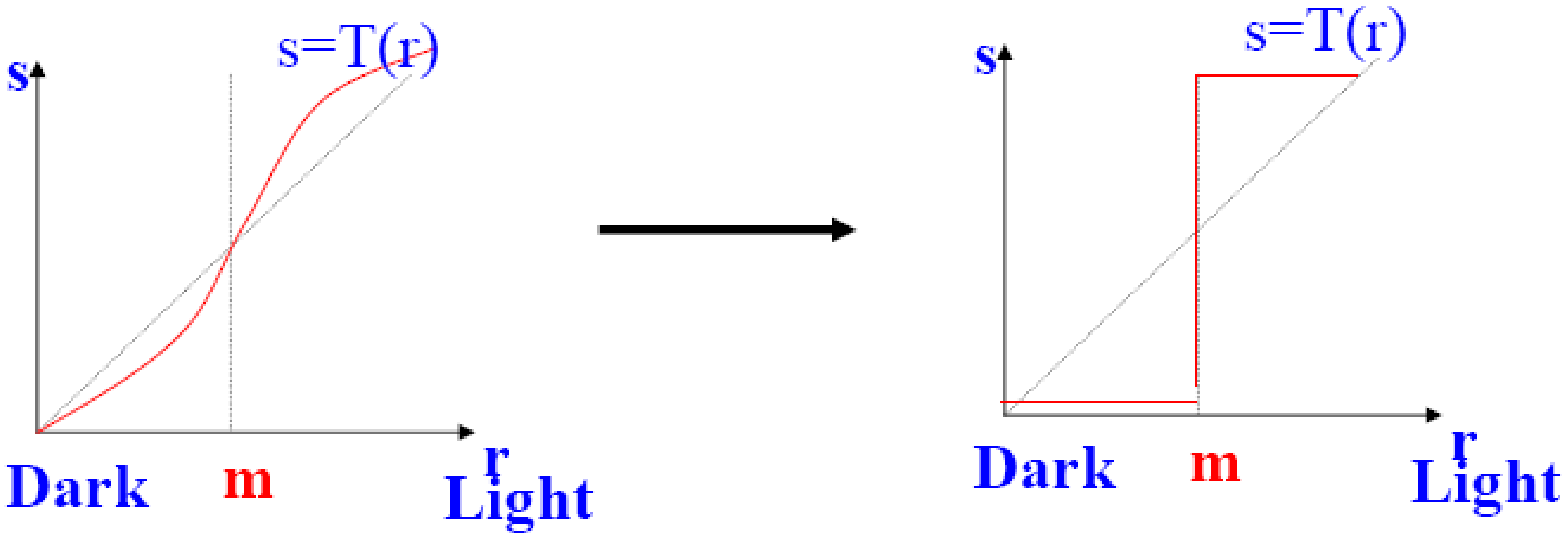
Contrast Stretching

Original

Enhanced

Spatial Domain: Point Processing

- Limiting case: produces a binary image (two level) from the input image



Thresholding Function

Spatial Domain: Point Processing

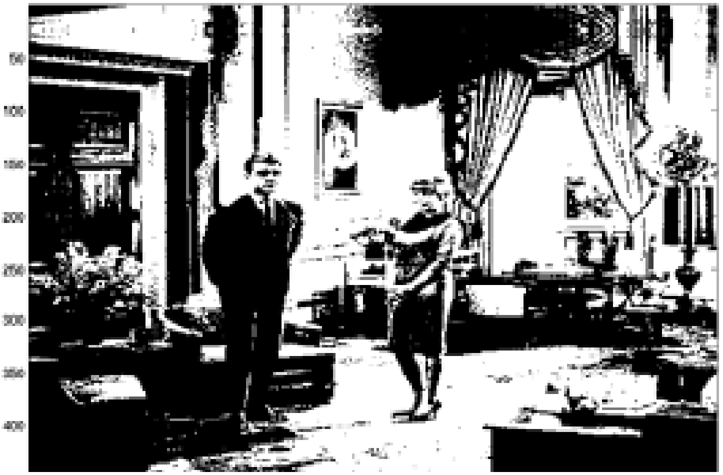
Contrast Stretching: Thresholding

Original

Enhanced

Spatial Domain: Point Processing

A Comparison

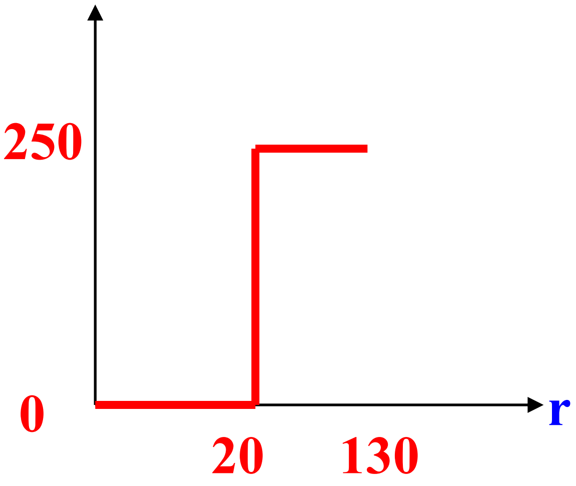


Original **Enhanced**

Example

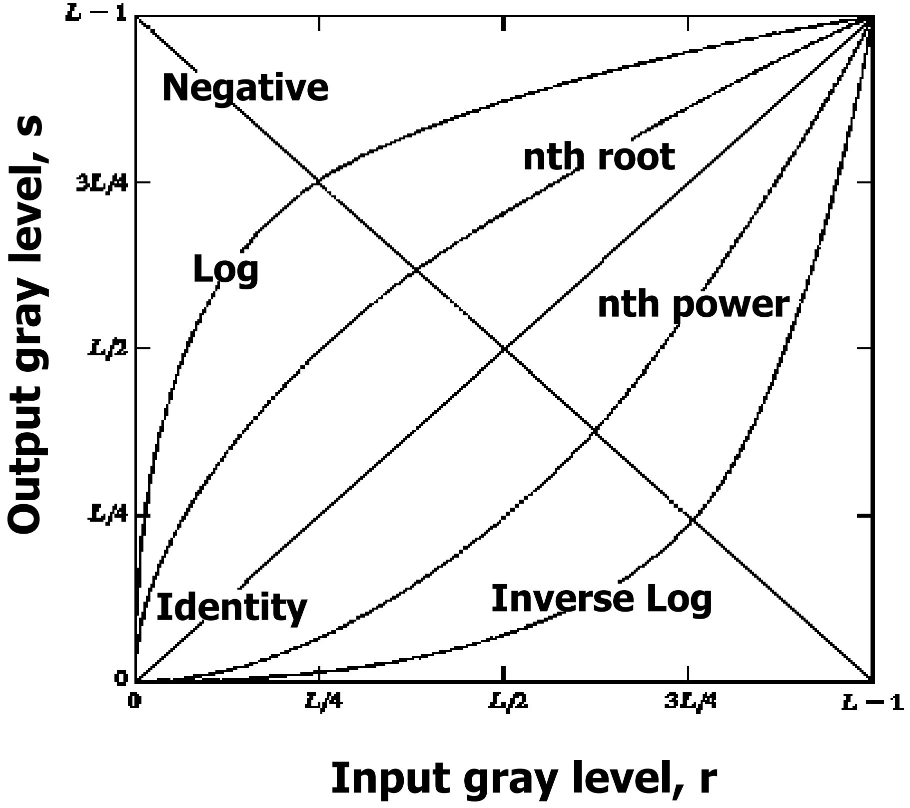
- A 4x4 image is given
- The image is transformed using the point transform shown
- Find the pixel values of the output image

17	64	128	128
15	63	132	133
11	60	142	140
11	60	142	138



3 - Basic Gray-level Transformation Functions

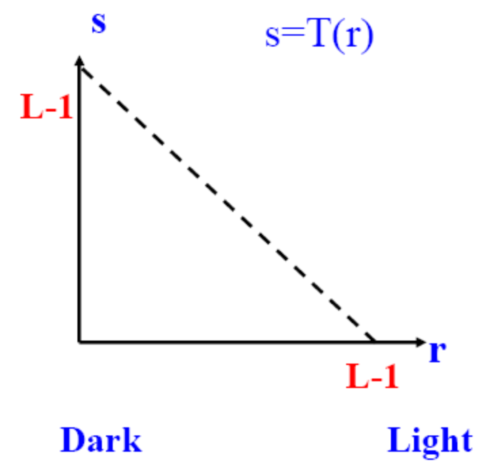
Simplest of all image enhancement Techniques



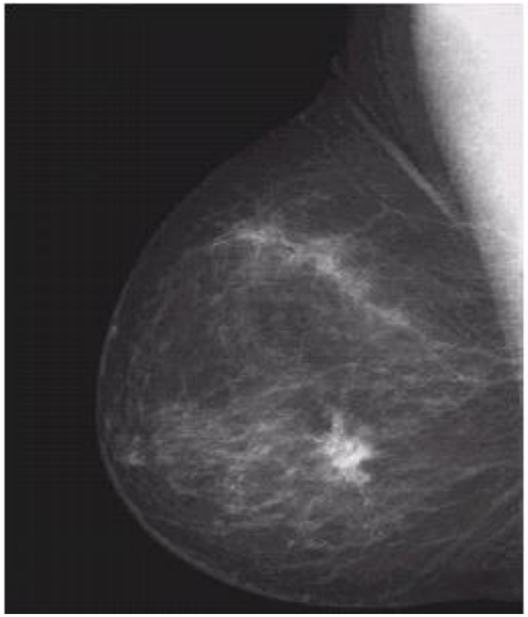
- Linear function
  - Negative and identity transformations
- Logarithm function
  - Log and inverse-log transformation
- Power-law function
  - n^{th} power and n^{th} root transformations

Image Negative

- Reversing the intensity levels of an image produces the equivalent of a photographic negative.
- Suited for enhancing white or gray detail embedded in dark regions especially when black areas are dominant in size.
- Has applications in medical imaging.



Original Digital mammogram

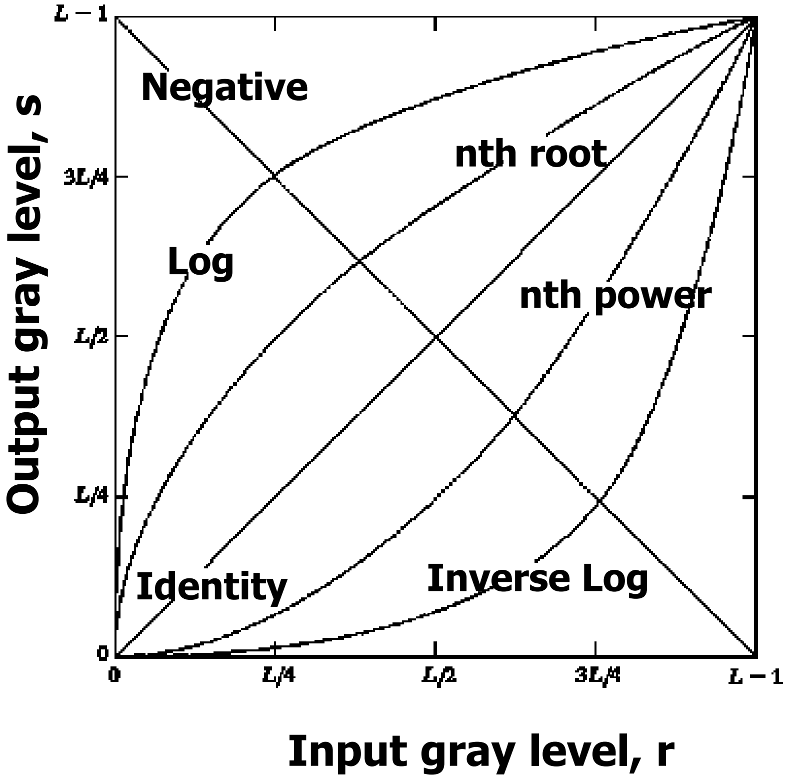


Negative image obtained using negative transformation

Log Transformation

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

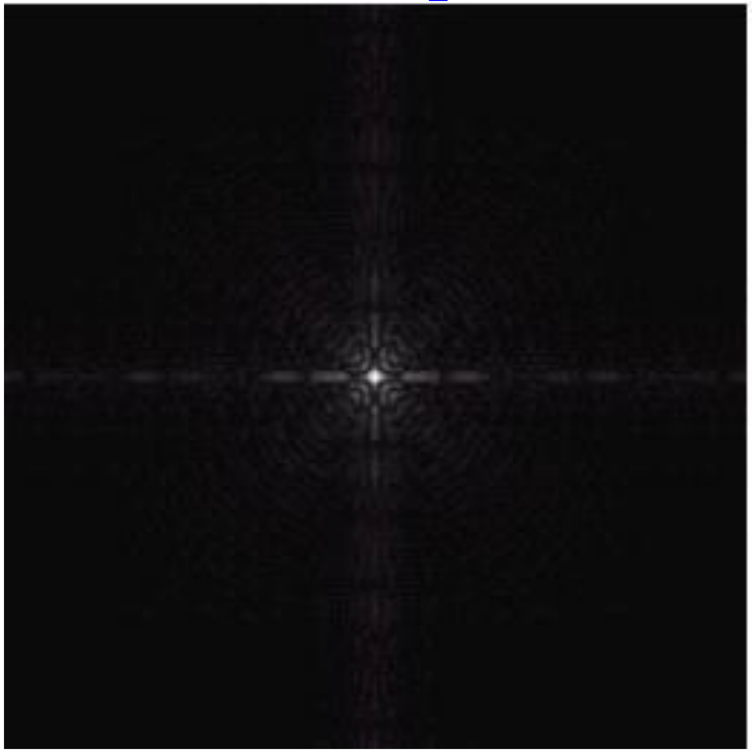
- Maps a narrow range of low gray-level input image into a wider range of output levels.
- The opposite is true of higher values of input levels



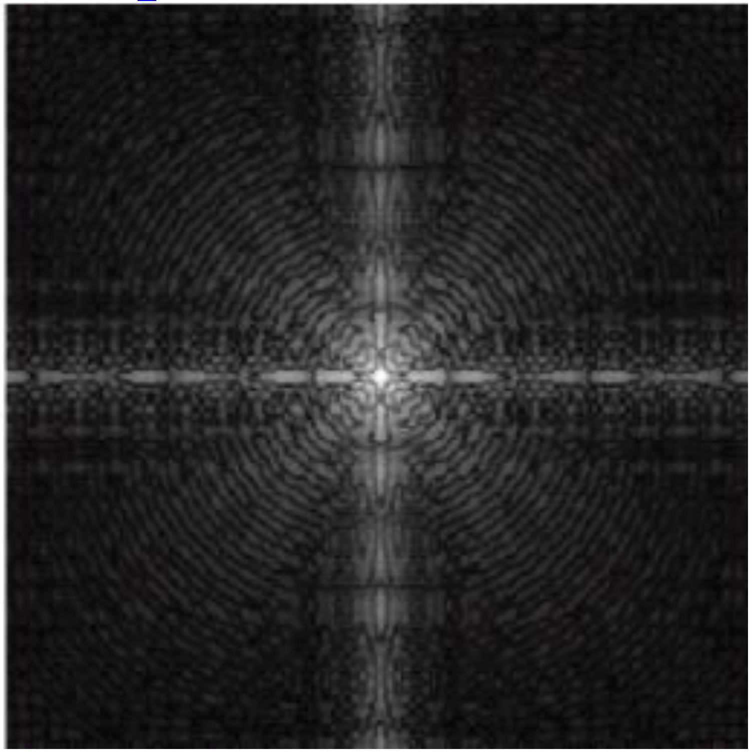
- Used to expand the values of dark pixels in an image while compressing the higher-level values.
- Opposite is true for inverse log transformation

Log Transformation

- Log function compresses the dynamic range of images with large variations in pixel values
- Classical example: Fourier Spectrum

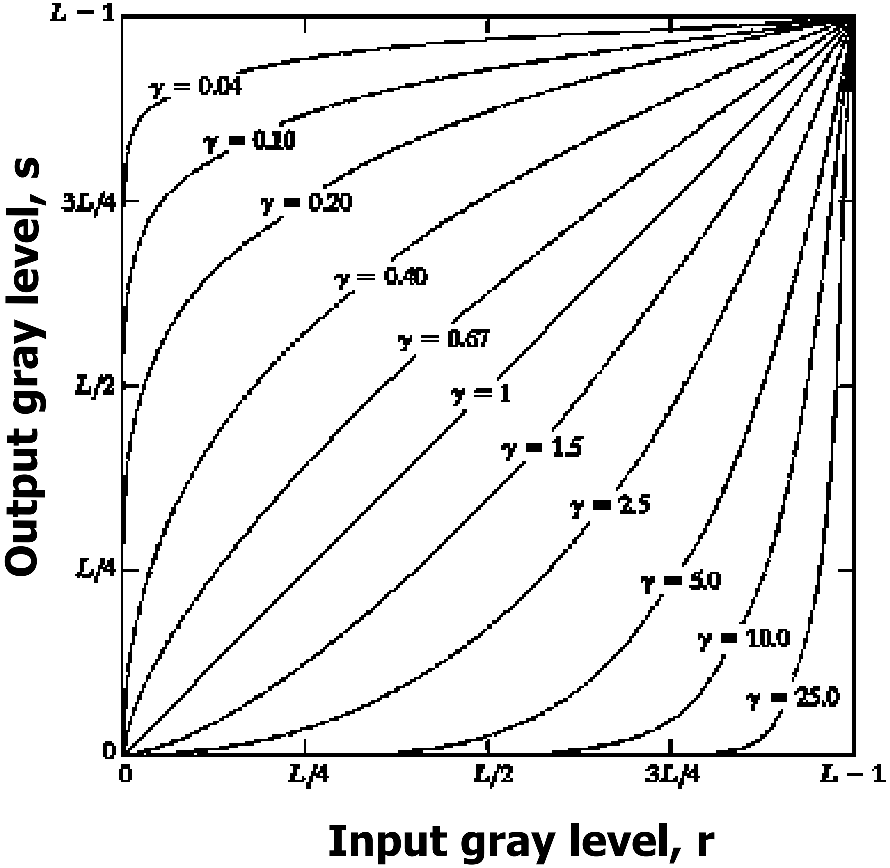


Fourier Spectrum
with values in the range = 0 to 10^6



Applying log transformation
with c=1 and range = 0 to 6.2

Power-law Transformation

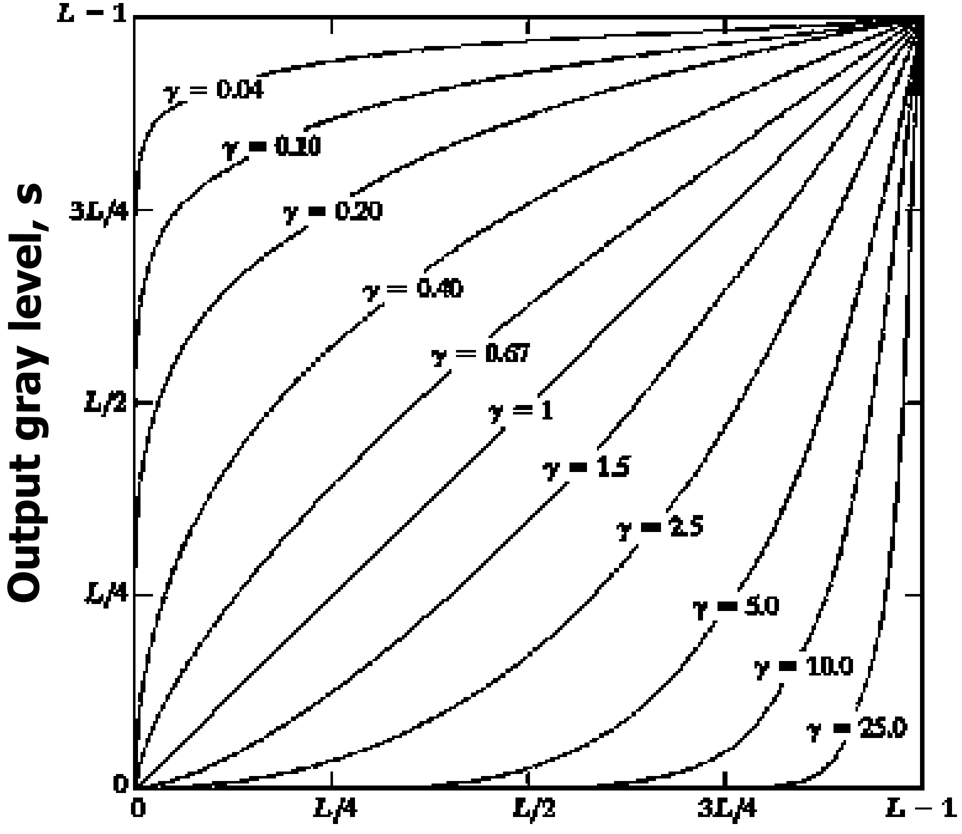


Plots of $s = cr^{\gamma}$ for various values of γ
(c = 1 in all cases)

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

- Power-law curves with fractional values of γ 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.
- $c = \gamma = 1 \Rightarrow$ Identity function

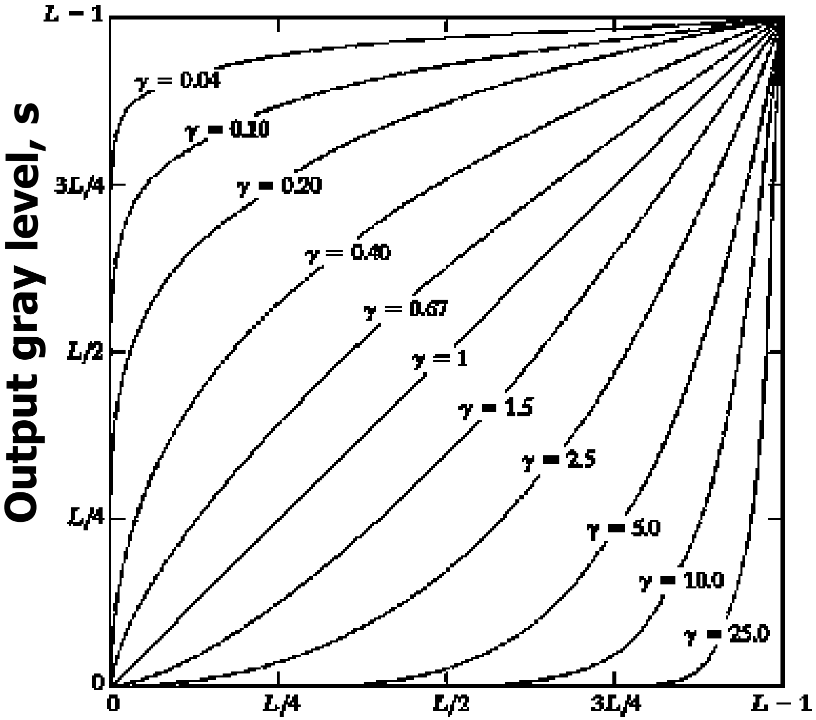
Gamma Correction



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

- A variety of devices used for image capture, printing and display respond according to a power law
- By convention, the exponent in the power-law equation is referred to as *gamma*
- The process used to correct this power-low response phenomena is called *gamma correction*

Gamma Correction



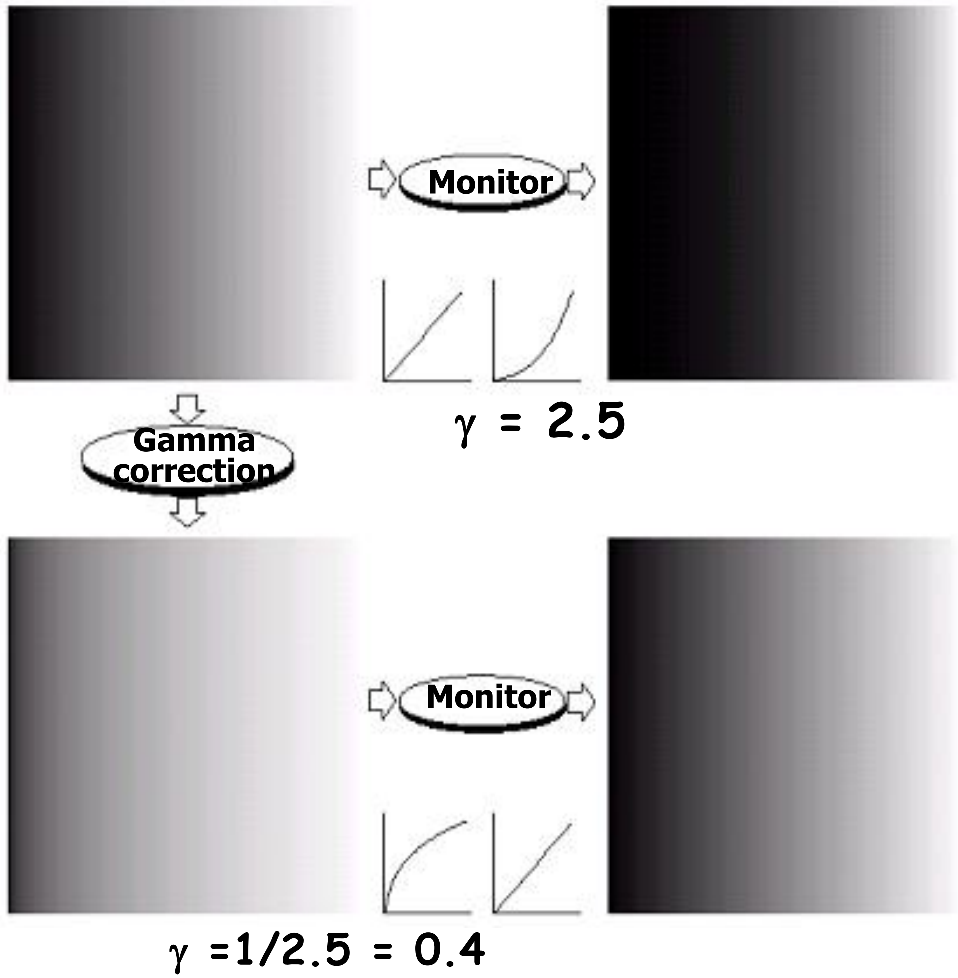
Input gray level, r

Plots of $s = cr^{\gamma}$ **for various values of** γ

(c = 1 in all cases)

- CRT devices have an intensity-to-voltage response that is a power function, with γ varying from 1.8 to 2.5
- w.r.t. the curve for $\gamma = 2.5$ in the figure, such display systems would tend to produce images that are darker than intended
- Gamma correction is done by preprocessing the image before inputting it to the monitor with $s = cr^{1/\gamma}$

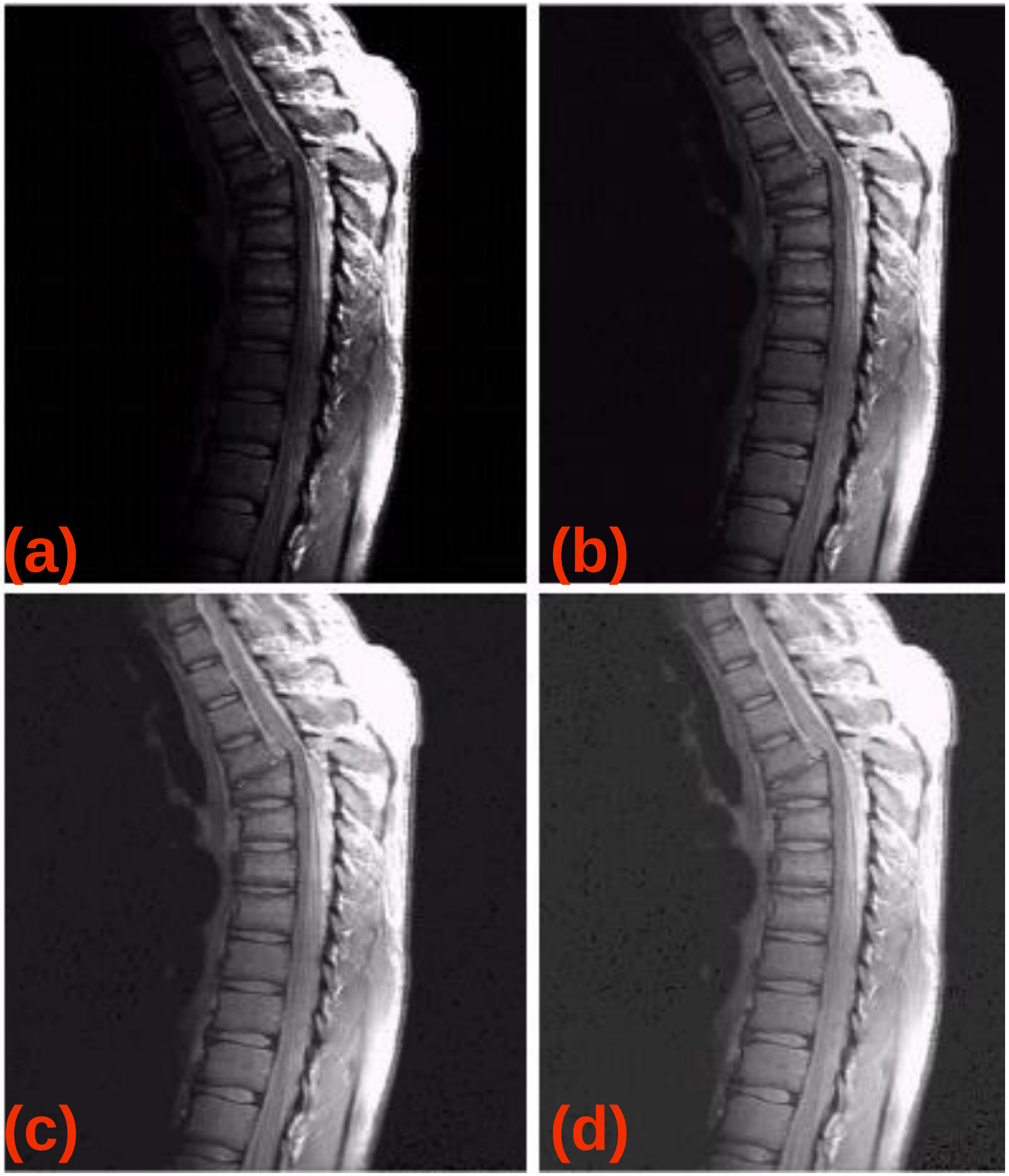
Gamma Correction



- A simple gray-scale linear wedge input to a monitor
- The output of the monitor appears darker than the input
- Gamma correction is done by performing the transformation $s = r^{1/2.5}$
- Gamma corrected input image produces an output that is close in appearance to the original image

- Similar analysis → other imaging devices (scanner & printer)
- Only difference → device dependent value of gamma

Power-law Transformation: Example



(a) A magnetic resonance image of an upper thoracic human spine with a fracture dislocation and spinal cord impingement

- The picture is predominately dark
- An expansion of gray levels are desirable ⇨ needs $\gamma < 1$

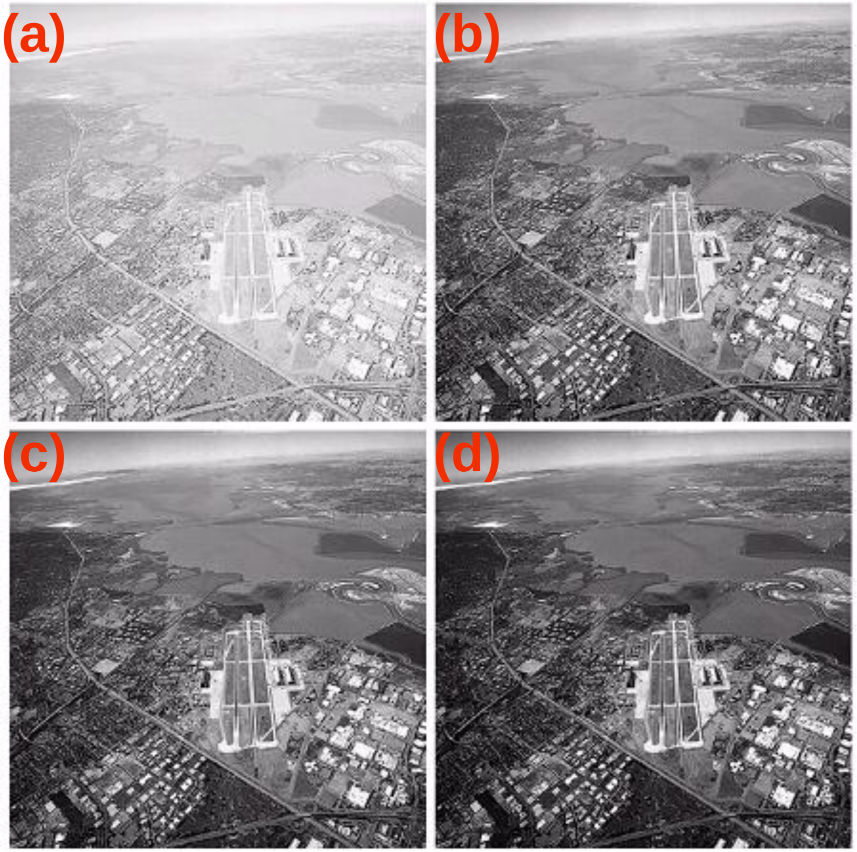
(b) Result after power-law transformation with $\gamma = 0.6$, c=1

(c) Transformation with $\gamma = 0.4$, c=1 (best result)

(d) Transformation with $\gamma = 0.3$, c=1 (under acceptable level)

Another Example

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



(a) image has a washed-out appearance, it needs a compression of gray levels ⇨ 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 detail is lost)

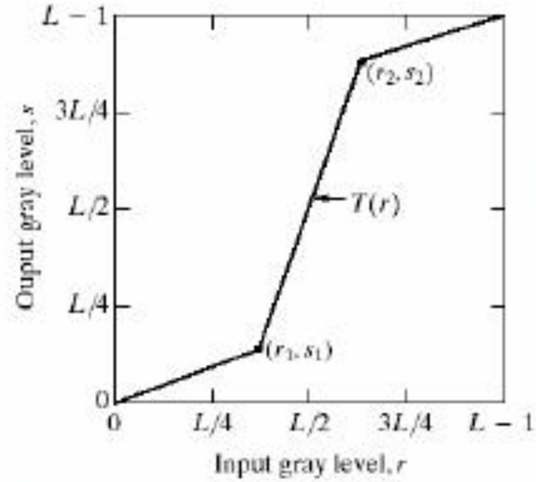
Piecewise-Linear Transformation Functions

- **Contrast Stretching**
- **Gray-level Slicing**
- **Bit-plane Slicing**

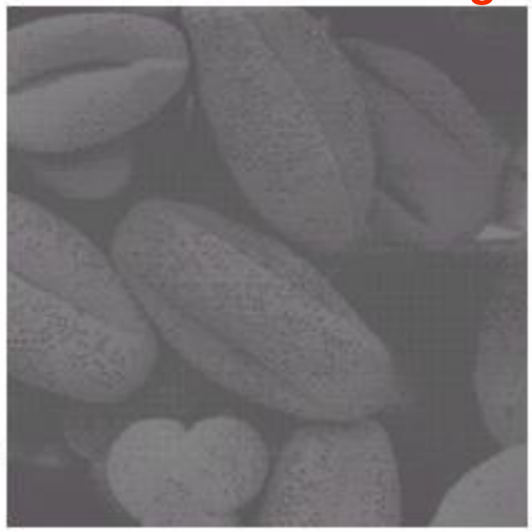
- Advantage:
  - The form of piecewise functions can be arbitrarily complex
- Disadvantage:
  - Their specification requires considerably more user input

Contrast Stretching

Transformation function



Low contrast image

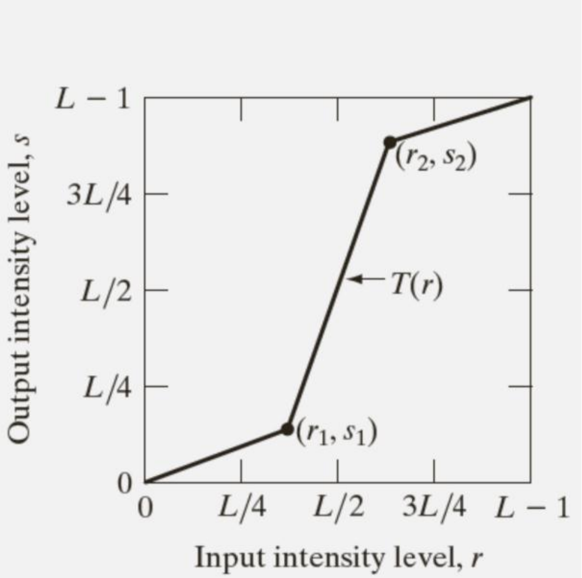


Contrast Stretching

Thresholding

- Increase the dynamic range of the gray levels in the image
- A low-contrast image results from:
  - poor illumination,
  - lack of dynamic range in the imaging sensor, or
  - even wrong setting of a lens aperture of image acquisition
- The location of points (r_1,s_1) and (r_2,s_2) control the shape of the transformation function

Contrast Stretching



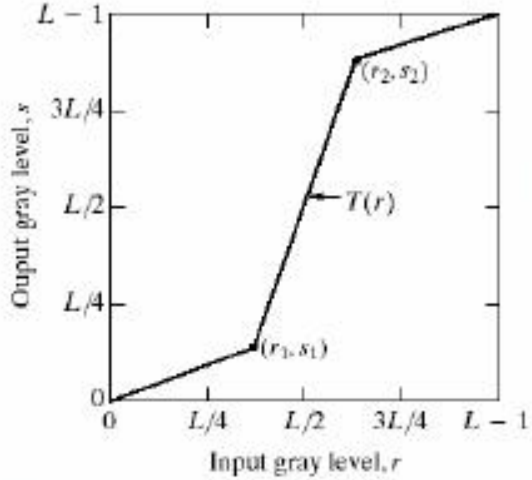
- $r_1 = s_1$ and $r_2 = s_2$ → linear transformation function produces no changes in intensity levels
- $r_1 = r_2$ and $s_1 = 0$ and $s_2 = L - 1$ → thresholding function
- Intermediate values of (r_1, s_1) and (r_2, s_2) produce various degrees of spread in the intensity levels of output image, thus affecting its contrast

$$s = \begin{cases} \alpha r, & 0 \le r \le r_1 \\ \beta(r - r_1) + s_1, & r_1 \le r \le r_2 \\ \gamma(r - r_2) + s_2, & r_2 \le r \le L-1 \end{cases}$$

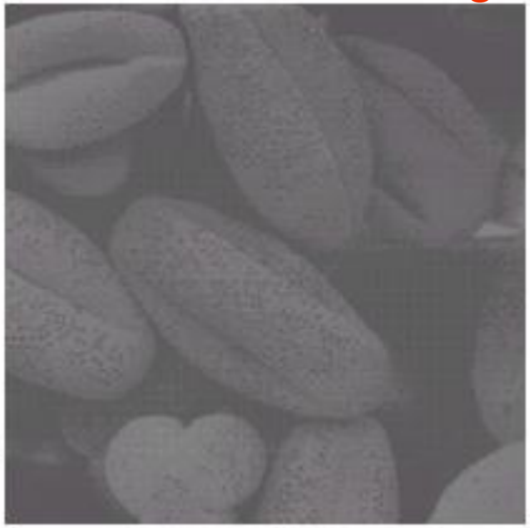
α, β, and γ are the slope of the respective curves.

Contrast Stretching

Transformation function



Low contrast image



Contrast Stretching

Thresholding

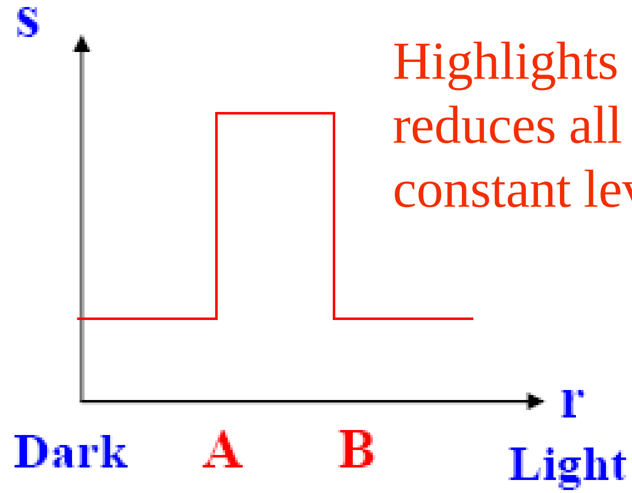
- **Contrast Stretching:**
  $(r_1, s_1) = (r_{min}, 0)$ and
  $(r_2, s_2) = (r_{max}, L\text{-}1)$
- **Thresholding:**
  $(r_1, s_1) = (m, 0)$ and
  $(r_2, s_2) = (m, L\text{-}1)$,
  where m is the mean intensity level in the image

Gray-level Slicing

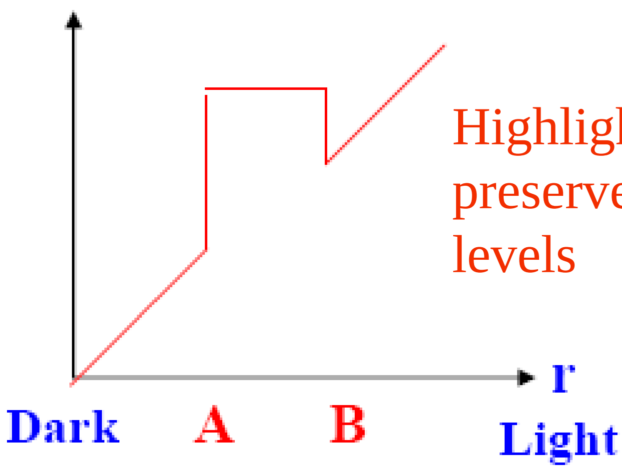
- Highlights a specific range of gray-levels in an image
- Application: enhancing flaws in X-ray images

Two basic methods:

1. Display a high value for all gray levels in the range of interest and a low value for all other gray levels
2. Brighten the desired range of gray levels but preserves the background and gray level tonalities

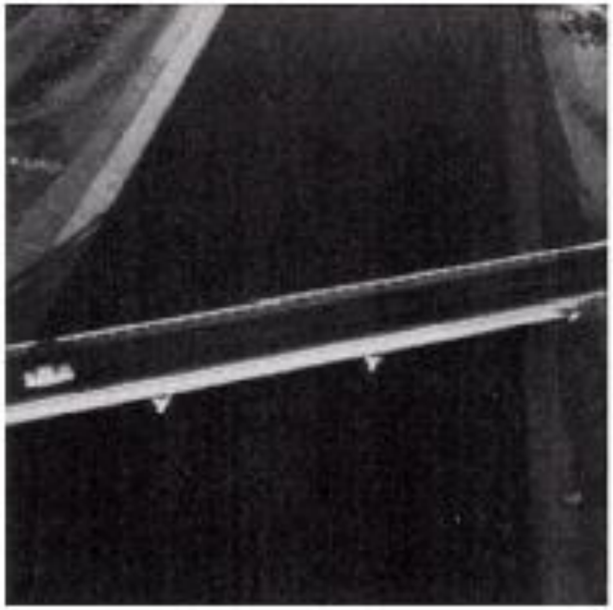


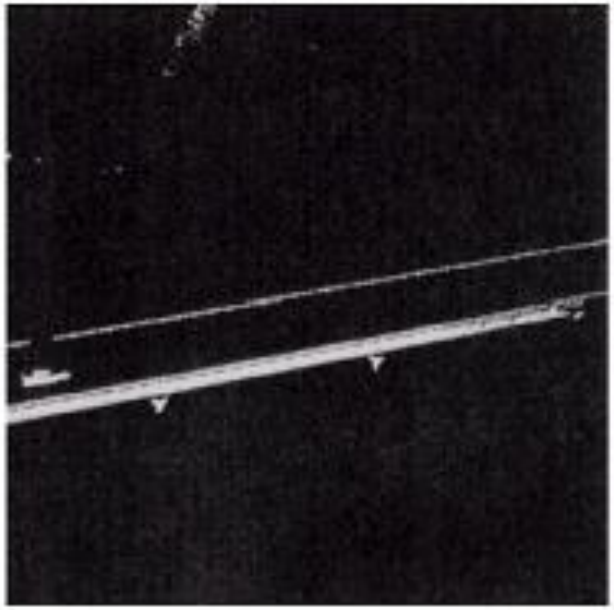
Highlights a range and reduces all others to a constant level



Highlights a range but preserves all other levels

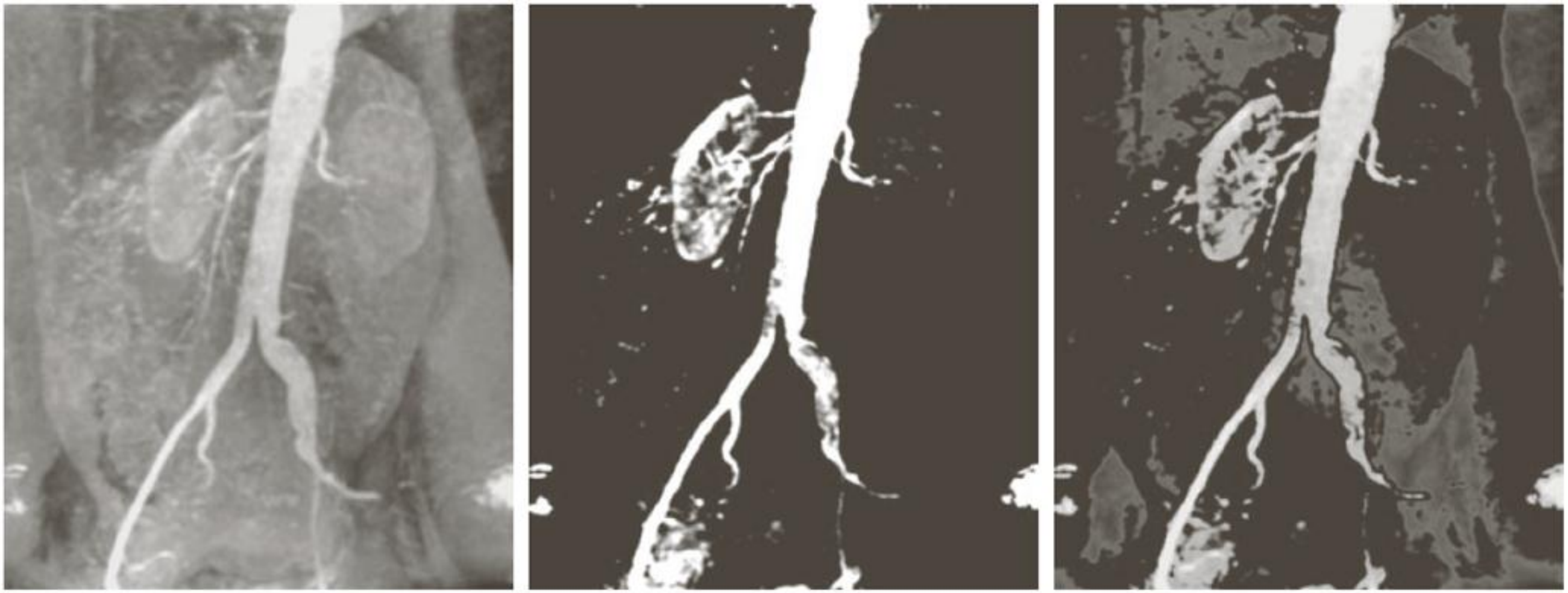
Gray-level Slicing





Result of using first method of gray-level slicing

Gray-level Slicing

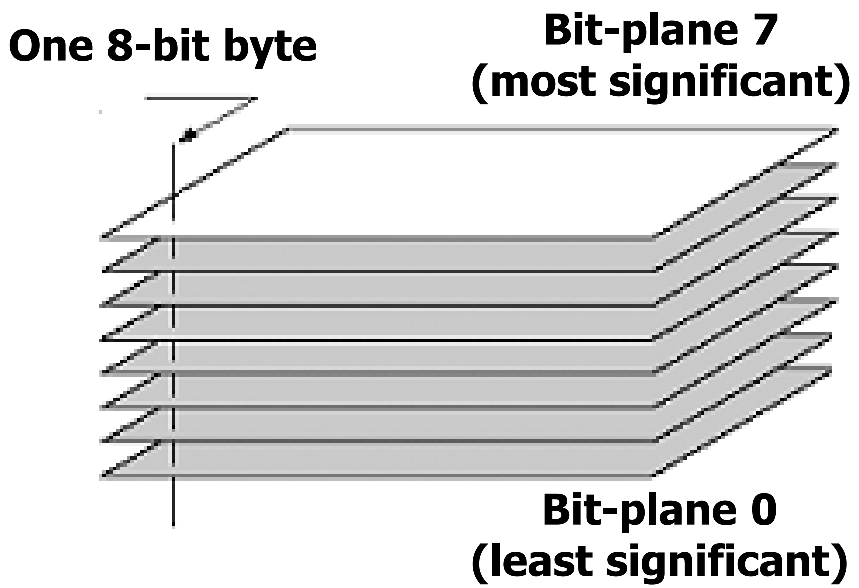


a b c

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.)

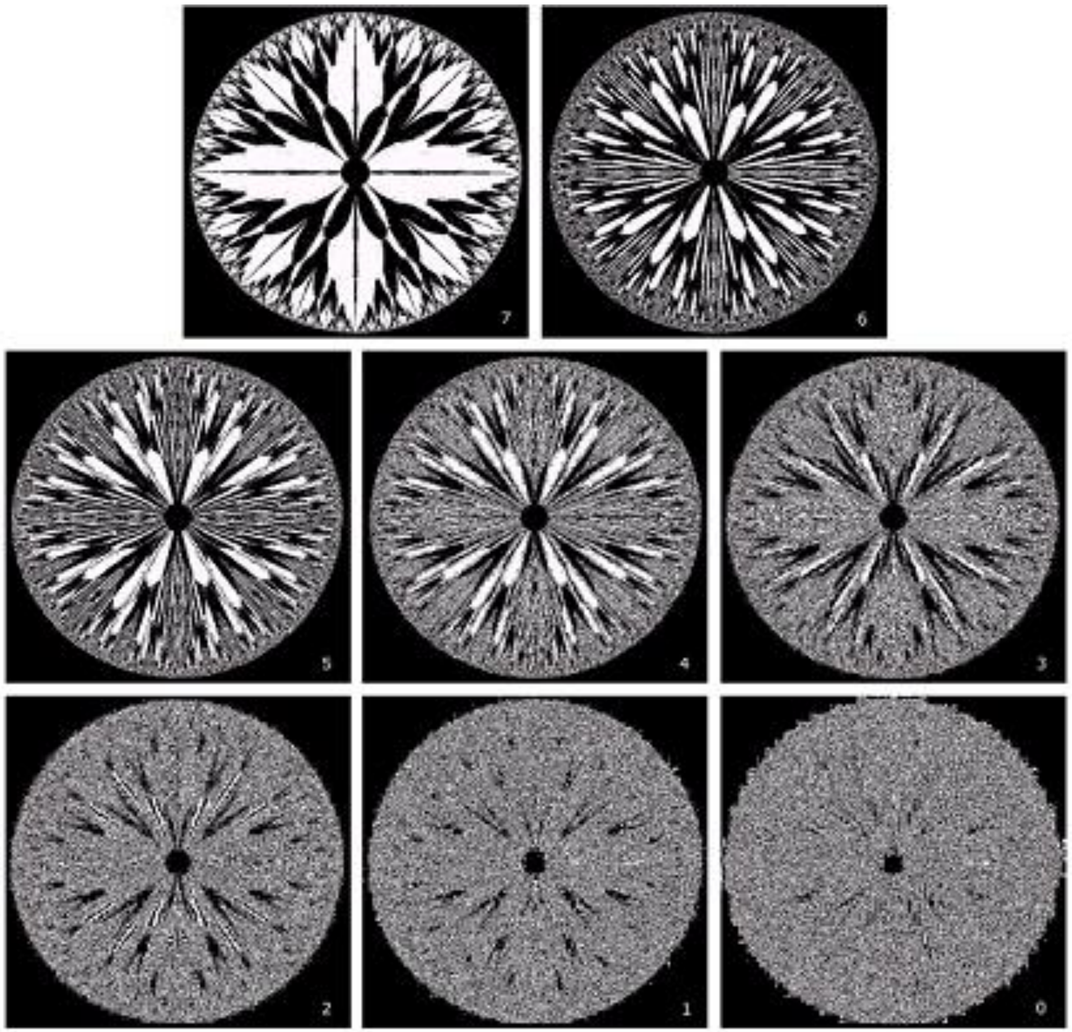
Bit-plane Slicing

- 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



- This type of decomposition is useful for image compression

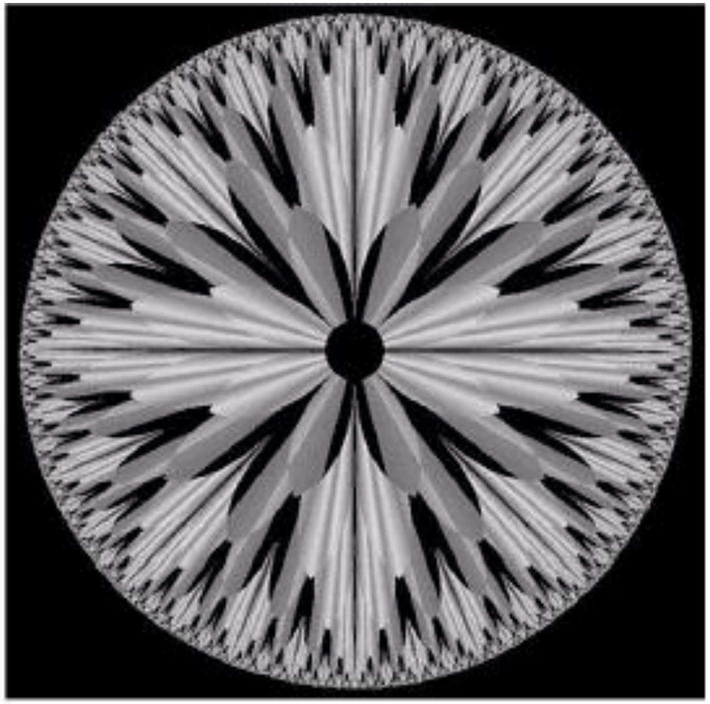
Bit-plane Slicing
Example: 8 bit planes



Bit-plane 7	Bit-plane 6	
Bit-plane 5	Bit-plane 4	Bit-plane 3
Bit-plane 2	Bit-plane 1	Bit-plane 0

- Higher order bit planes of an image carry a significant amount of visually relevant details.
- Lower order planes contribute more to fine (often imperceptible) details.

Bit-plane Slicing: Example



An 8-bit fractal 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 128 and 255 to 255

Let particular pixel of grayscale image has value 212. So, its binary value will be 11010100. So, its 1st bit is 0, 2nd is 0, 3rd is 1, 4rth is 0, 5th is 1, 6th is 0, 7th is 1, 8th is 1. In this manner, we will take these 8 bit of all pixel and will draw 8 binary image. We have to do this to all the pixels and generate new images.

Bit-plane Slicing: Another 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.

Bit-plane Slicing: Compression

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).

FIGURE 3.14 (a) An 8-bit gray-scale image of size 500 × 1192 pixels.

- This type of decomposition is useful for image compression, in which fewer than all planes are used in reconstructing an image

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
- Histogram is a discrete function formed by counting the number of pixels that have a certain gray level in the image
- Popular tool for real time image processing
  - Simple to calculate in s/w and also lend themselves to economic h/w implementations

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 k^{th} 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

Normalized Histogram

- Dividing each of histogram at gray level $\boldsymbol{r_k}$ by the total number of pixels in the image, $\boldsymbol{n}$

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

for $k = 0,1,\ldots, L\text{-}1$

- $\boldsymbol{p(r_k)}$ gives an estimate of the probability of occurrence of gray level $\boldsymbol{r_k}$
- The sum of all components of a normalized histogram is equal to 1
- Total number of pixels $n = M \times N$

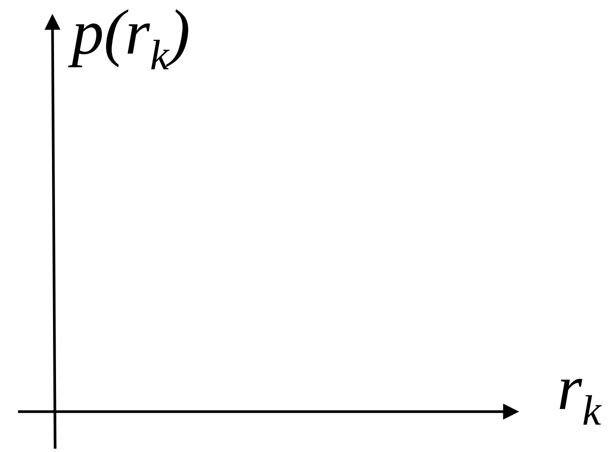
Histogram Processing

- Example: an image with gray levels between 0 and 7 is given below.

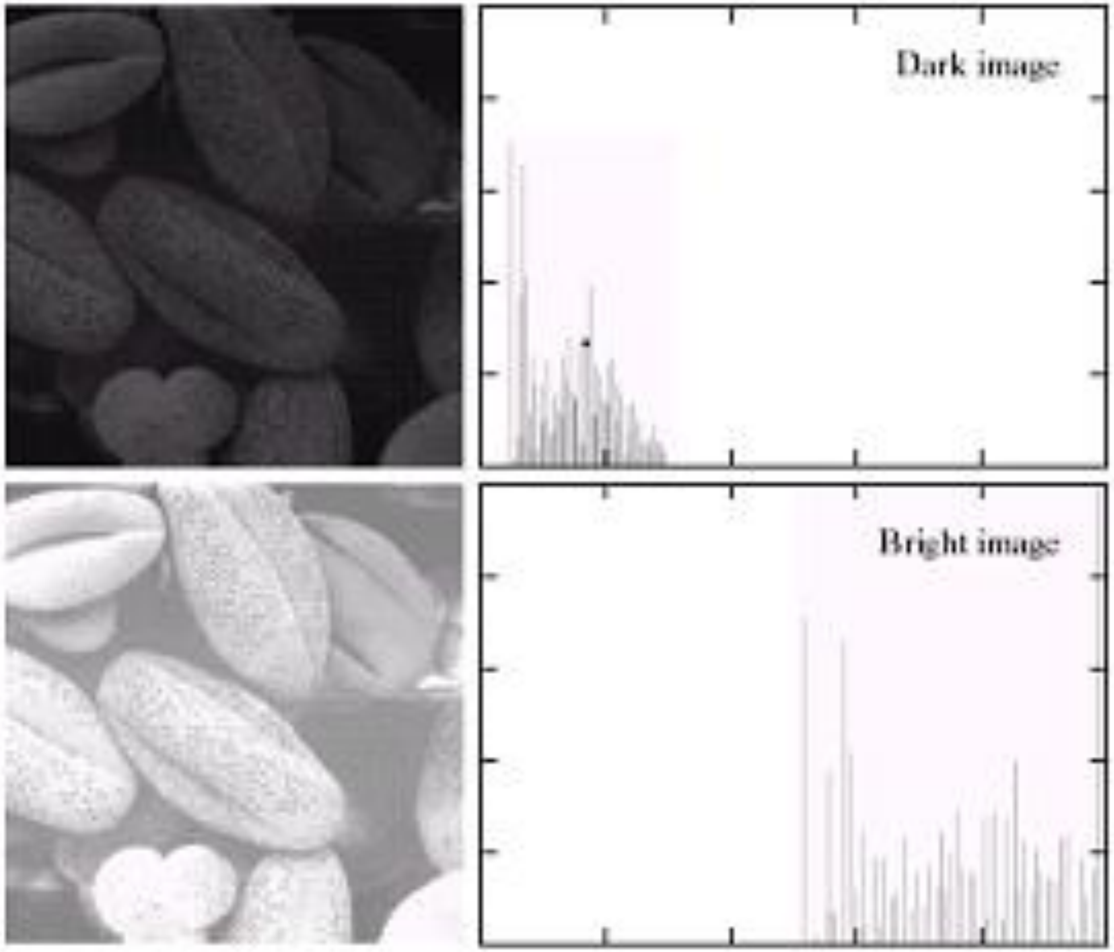
Find the histogram of the image.

1	6	2	2
1	3	3	3
4	6	4	0
1	6	4	7

0: 1/16	4: 3/16
1: 3/16	5: 0/16
2: 2/16	6: 3/16
3: 3/16	7: 1/16



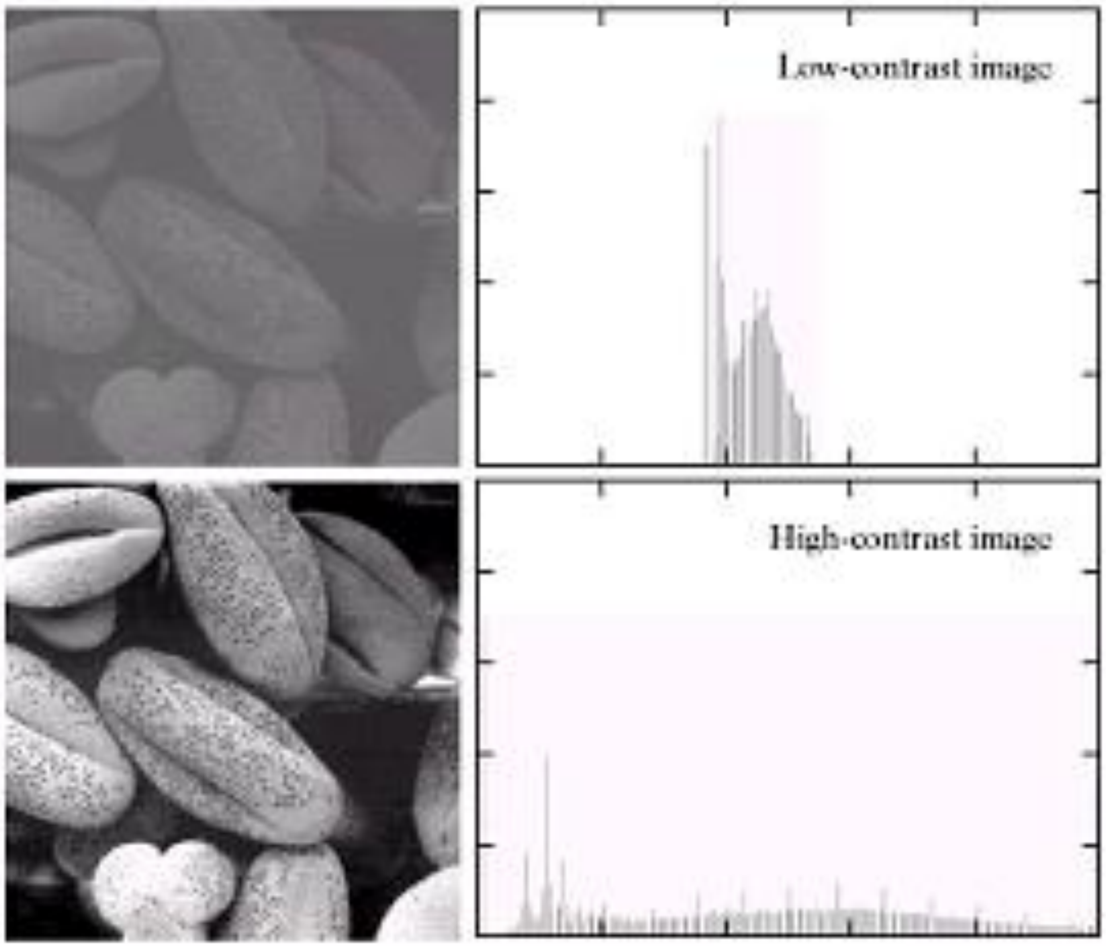
Histogram Example 1



Dark Image:
Components of histogram are concentrated on the low side of the gray scale

Light Image:
Components of histogram are concentrated on the high side of the gray scale

Histogram Example 1 (cont…)



Low-Contrast Image: Histogram is narrow and centered toward the middle of the gray scale

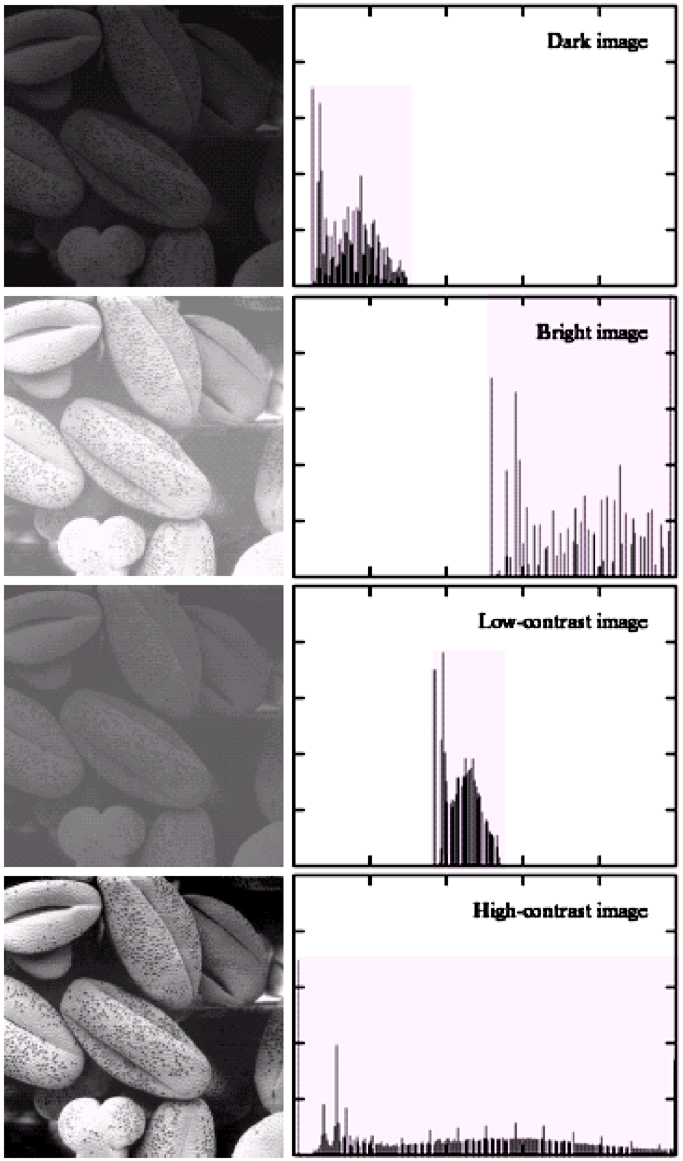
High-Contrast Image: Histogram covers broad range of the gray scale and the distribution of pixels is not too far from uniform (very few vertical lines being much higher than the others)

Histogram Example 1 (cont…)

A selection of images and their histograms

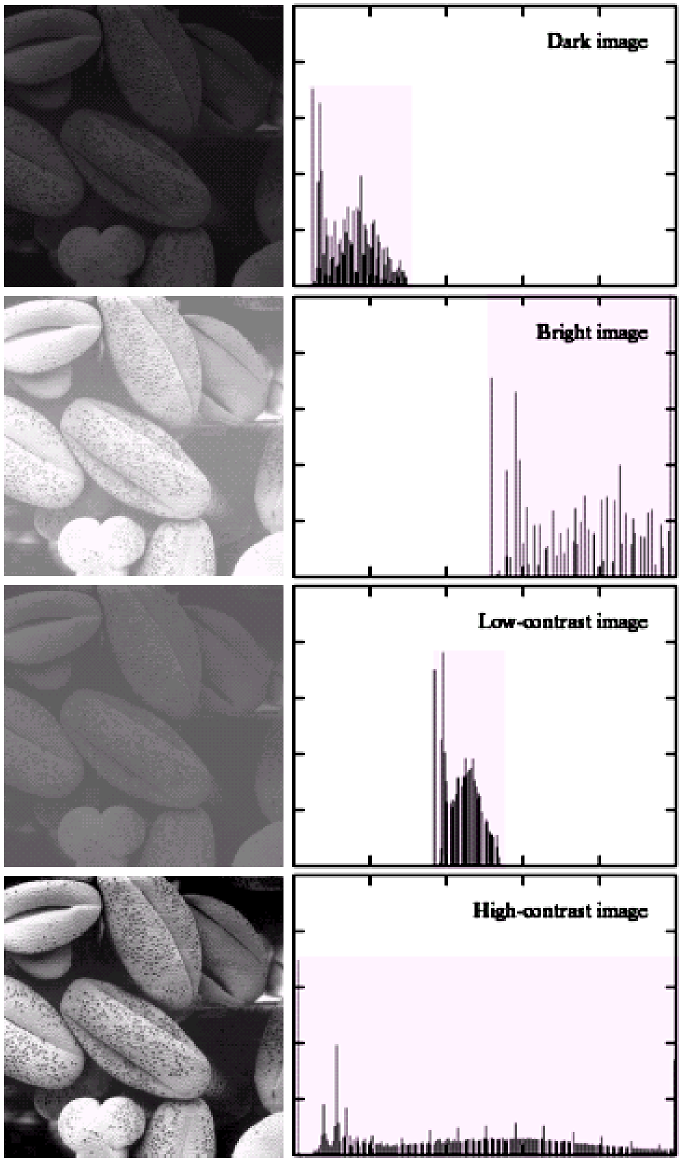
Notice the relationships between the images and their histograms

Note that the high contrast image has the most evenly spaced histogram



Histogram Example 1 (cont…)

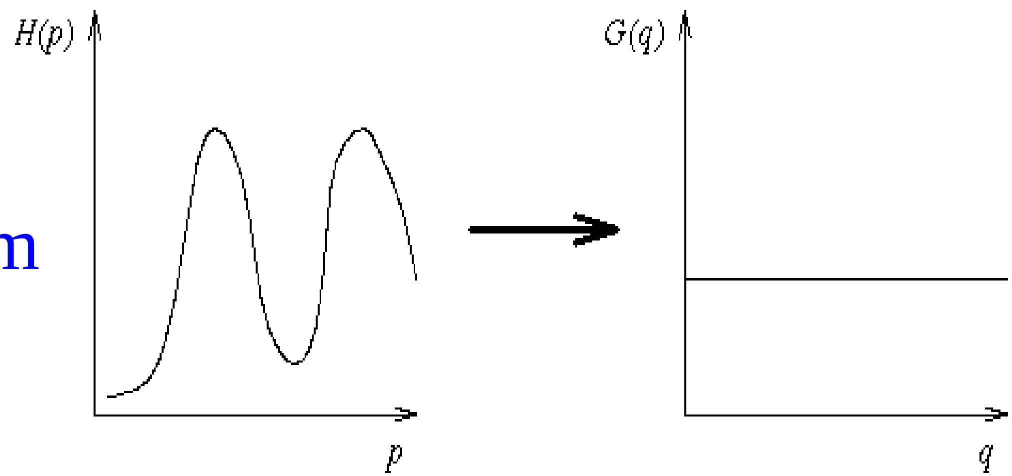
- Observation from these histograms
  - As the low-contrast image's histogram is narrow and centered toward the middle of the gray scale, if we distribute the histogram to a wider range the quality of the image will be improved



Histogram Equalization

- Increases dynamic range of an image
- Enhances contrast of image to cover all possible grey levels
- Ideal histogram = flat (equalized)
  – same no. of pixels at each grey level
- *Ideal no. of pixels* at each grey level = $i = \dfrac{M \times N}{L}$

Typical histogram



Ideal histogram

Histogram Equalization

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

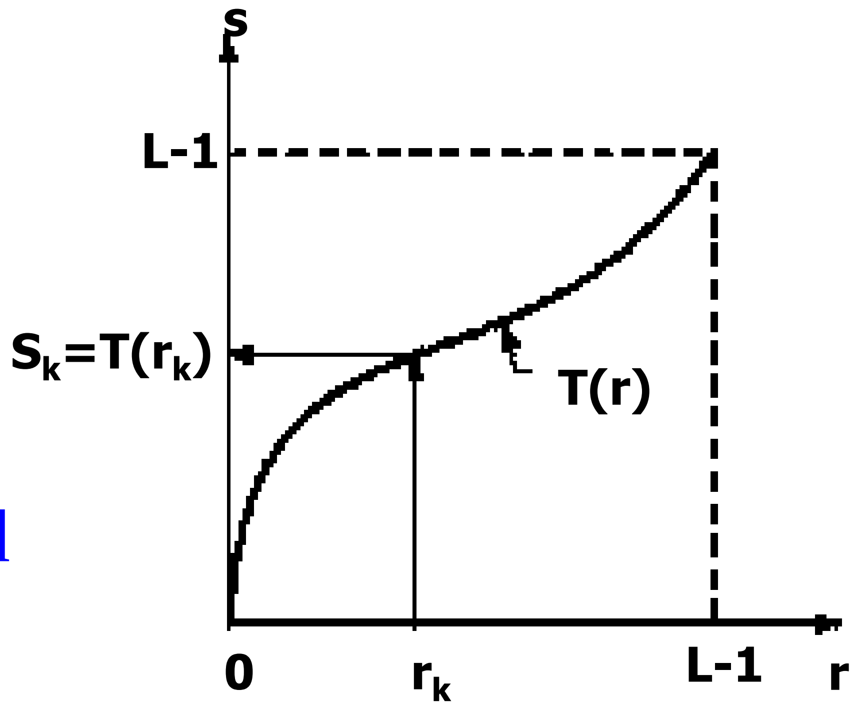
- Goal: find a non-linear transformation $s=T(r)$, to be applied to each pixel of the input image $f(x,y)=r$, such that a uniform distribution of gray levels in the entire range results for the output image $g(x,y)=s$.

Histogram Equalization

- To get the mapping of the form $s = T(r)$, analyze ideal, continuous case first, assuming
  - $0 \le r \le L\text{-}1$ and $0 \le s \le L\text{-}1$
  - Assume that

    (a). $T(r)$ is single-valued and monotonically increasingly in the interval $0 \le r \le L\text{-}1$; and

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



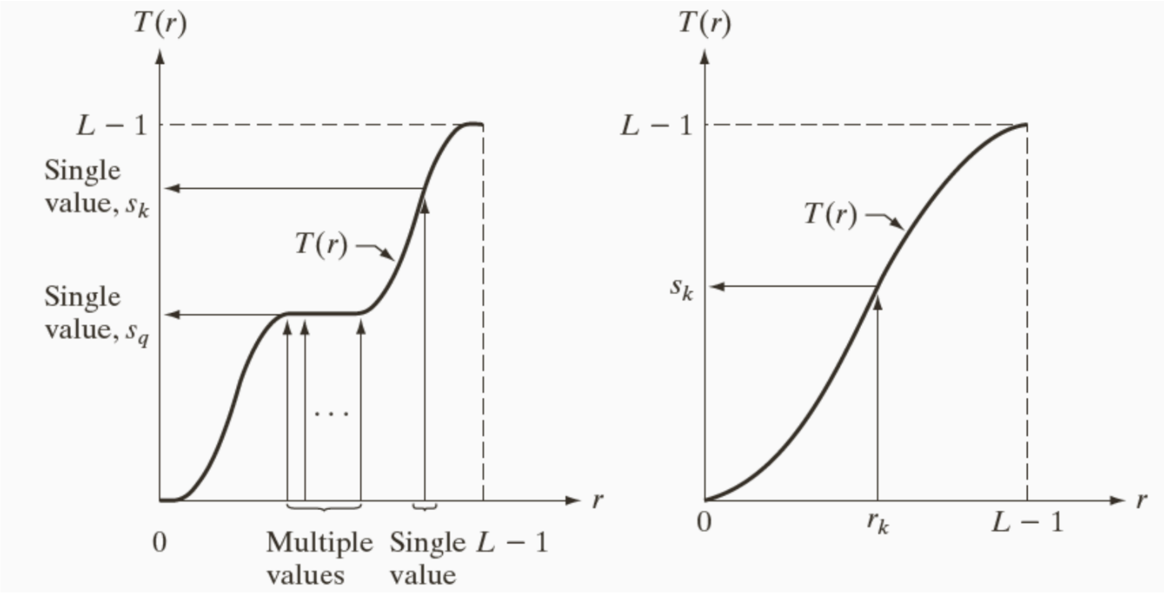
A gray level transformation function satisfying two conditions

Meaning of Two Conditions of *T*(*r*)

- 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
- Not monotonically increasing could result in at least a section of the intensity range being inverted, this produces some inverted gray levels in the output image
- $0 \leq T(r) \leq L\text{-}1$ for $0 \leq r \leq L\text{-}1$ guarantees that the output gray levels will be in the same range as the input levels
- The inverse transformation from s back to r is

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

Monotonically Increasing Function



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.

- A function $T(r)$ is strictly monotonically increasing if $T(r_2) > T(r_1)$ for $r_2 > r_1$, and vice versa.

Probability Density Function

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

- PDF (probability density function) is one of the fundamental descriptors of a random variable

Random Variables

- 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:

$$p_y(y) = p_x(x)\left|\frac{dx}{dy}\right|$$

absolute value

- This equation is valid if ***T*(*x*)** is an increasing or decreasing monotonic function

Applied to Image

- If $p_r(r)$ and $T(r)$ are known and $T^{-1}(s)$ satisfies condition (a) then $p_s(s)$ can be obtained using a formula :

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

where
- $p_r(r)$ denote the PDF of random variable r
- $p_s(s)$ denote the PDF of random variable s

The PDF of the transformed variable *s* is determined by the PDF of the input image and by the chosen transformation function.

Transformation Function

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

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

where *w* is a dummy variable of integration

The right side is recognized as the cumulative distribution function of random variable *r*

Note: $\boldsymbol{T(r)}$ depends on $\boldsymbol{p_r(r)}$

Cumulative Distribution Function

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

- 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)
- Similarly, the integral of a probability density function for variables in the range [0, *L*-1] also is in the range [0, *L*-1]
  - Area under a PDF curve always is 1
  - Thus, CDF satisfies the condition (b) also
- We can use CDF as a transformation function

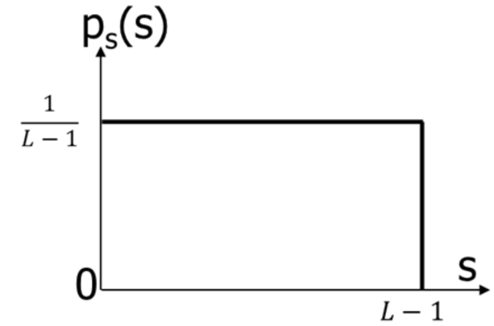
Finding $p_s(s)$ from given $T(r)$

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

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

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

$$= (L-1)\,p_r(r)$$



Substitute and yield

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

$$= p_r(r)\left|\frac{1}{(L-1)\,p_r(r)}\right|$$

$$= \frac{1}{(L-1)} \quad \text{where } 0 \le s \le L-1$$

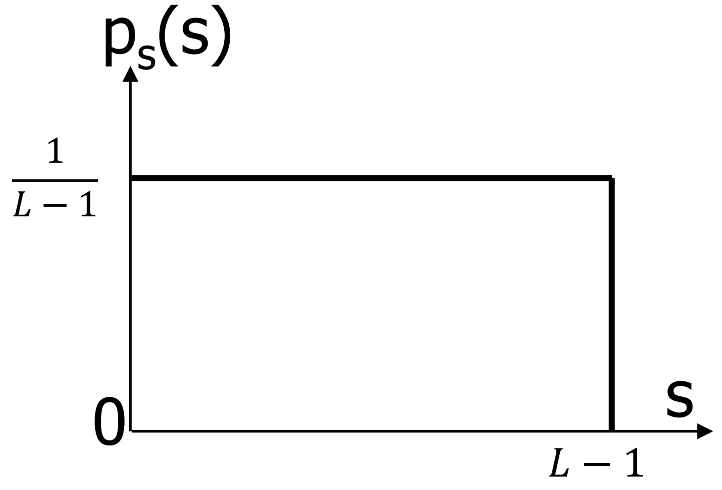
$p_s(s)$

- $p_s(s)$ is always **a uniform probability density function**, independent of the form of $p_r(r)$

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

yields

a random variable s characterized by a uniform probability function



Discrete Transformation Function

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

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

- The discrete version of transformation

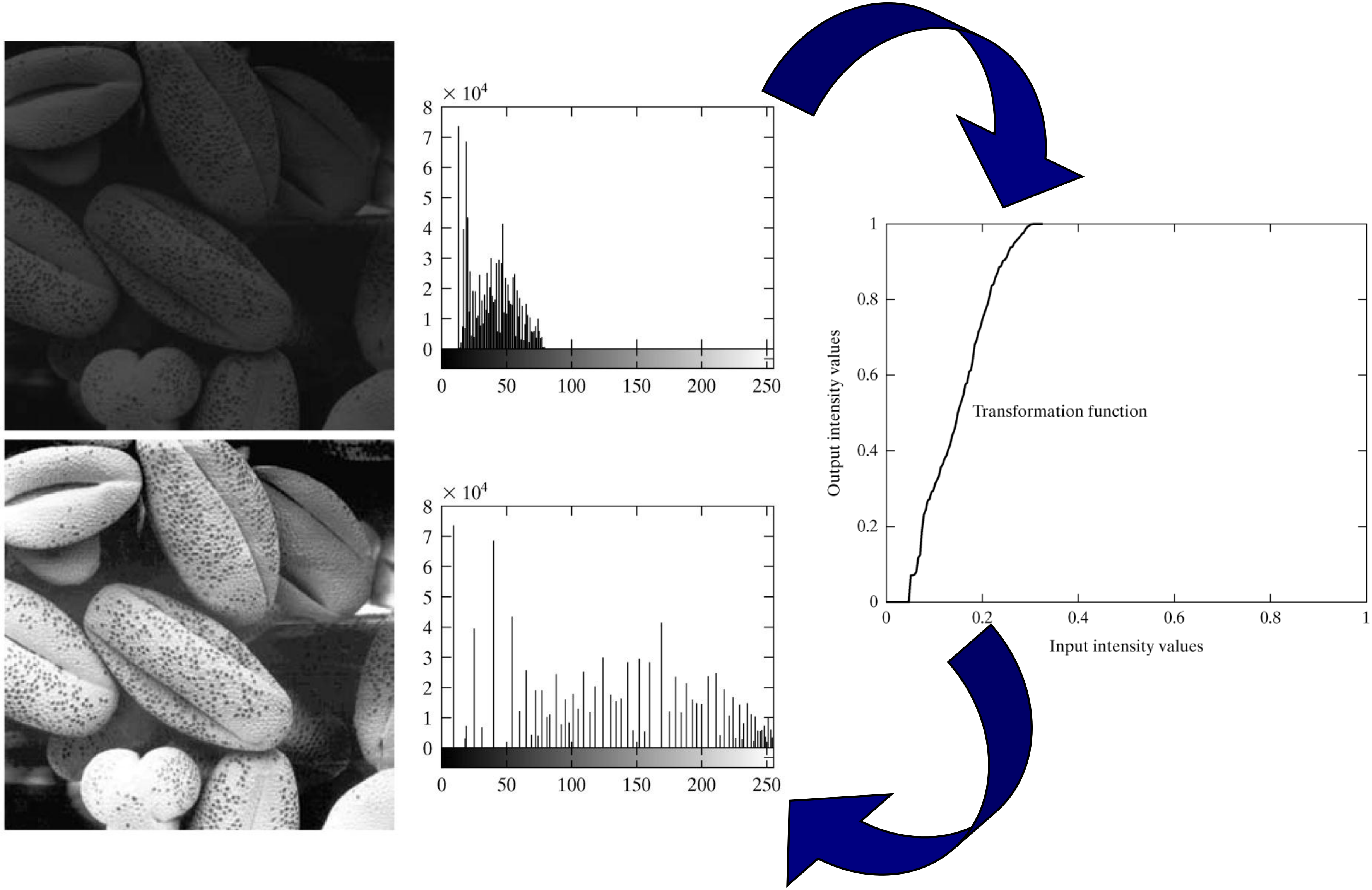
$$s_k = T(r_k) = (L-1)\sum_{j=0}^{k} p_r(r_j)$$

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

Histogram Equalization

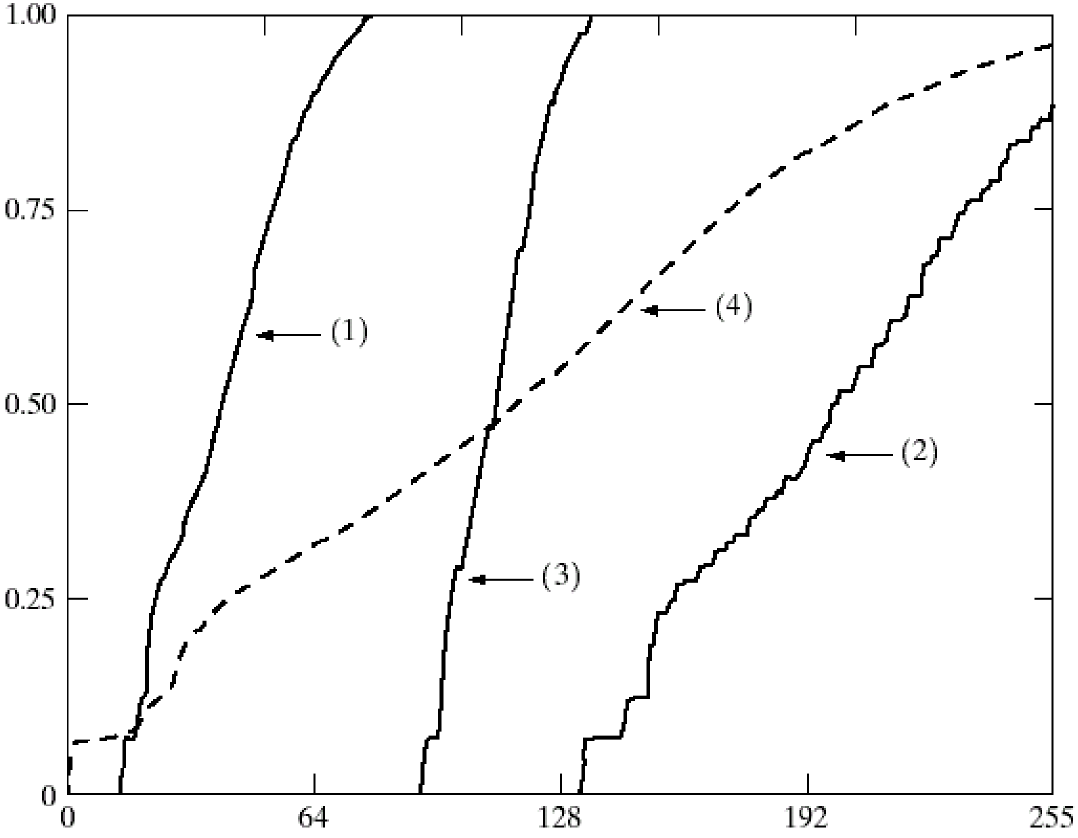
- An output image is obtained by mapping each pixel with level r_k in the input image into a corresponding pixel with level s_k in the output image: Histogram Equalization or Histogram Linearization
- In discrete space, it cannot be proved in general that this discrete transformation will produce the discrete equivalent of a uniform probability density function, which would be a uniform histogram
- The equation has a general tendency of spreading the histogram of the input image so that the levels of the histogram-equalized image will span a fuller range of the gray scale
- Advantage: fully automatic method

Equalization Transformation Function

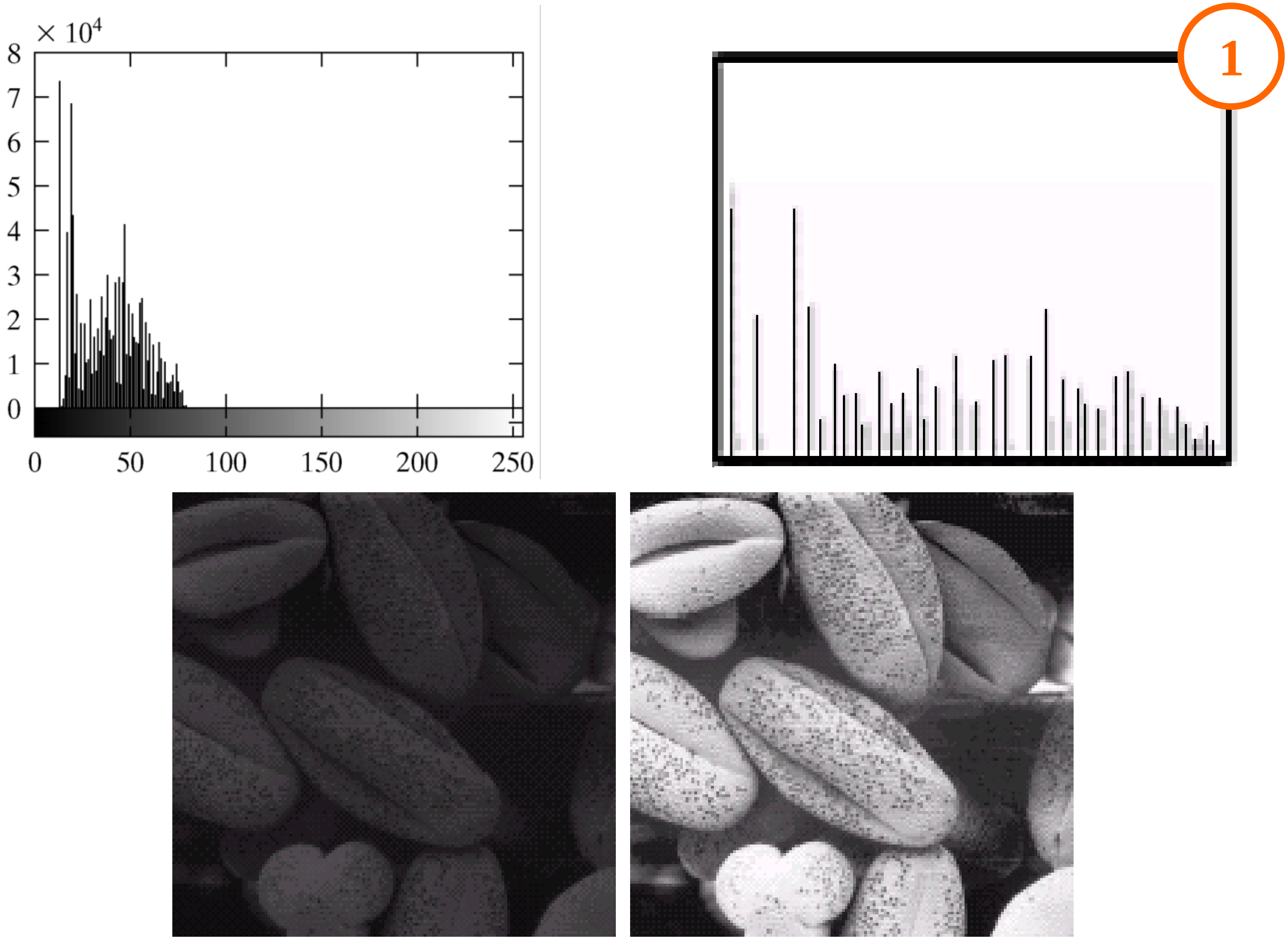


Equalization Example 1 Transformation Functions

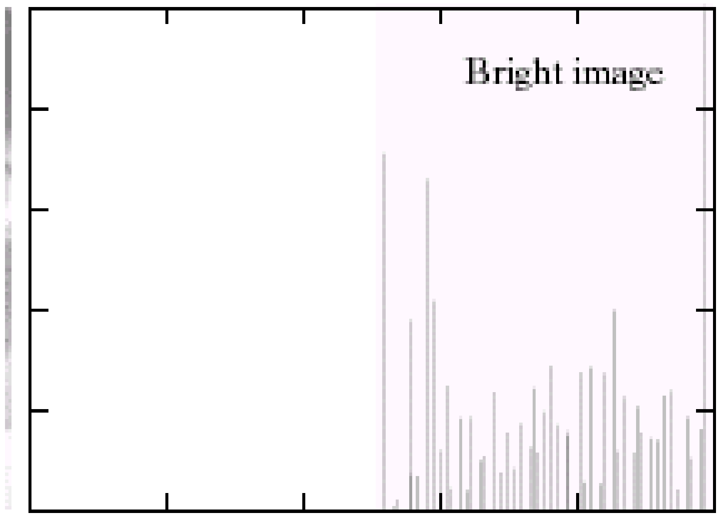
The functions used to equalise the images in the previous example

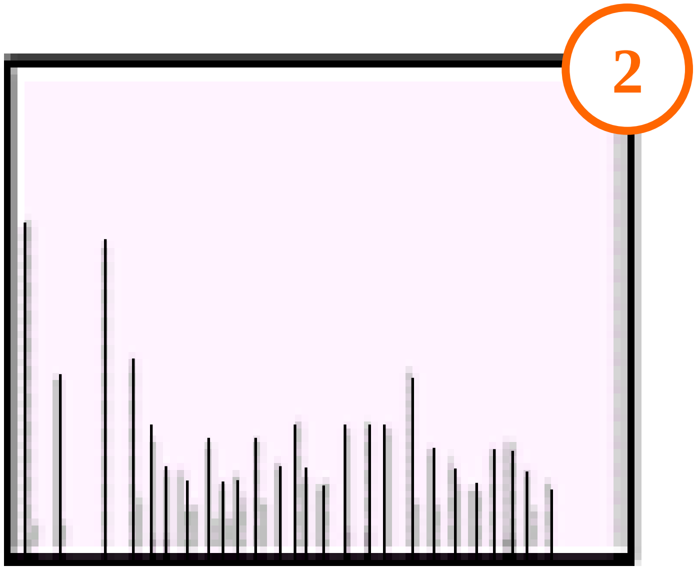


Equalization Example 1 (Cont..)

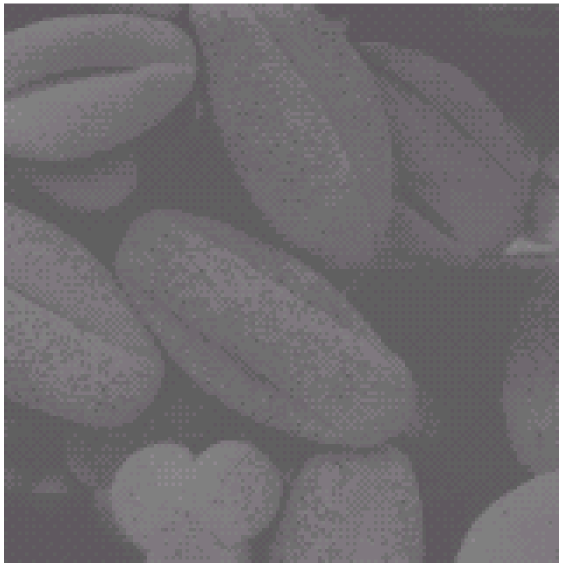


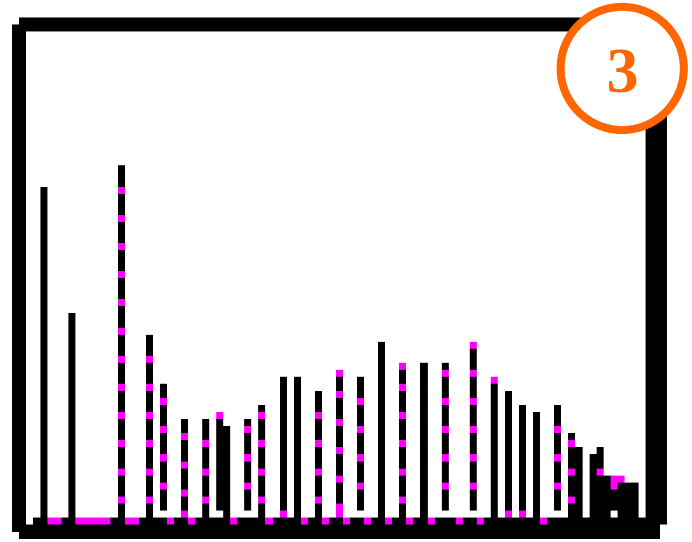
Equalization Example 1 (Cont..)



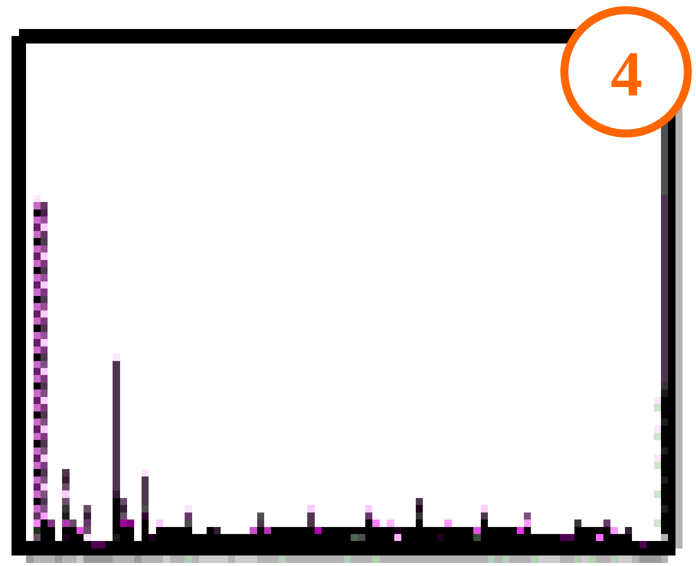


Equalization Example 1 (Cont..)

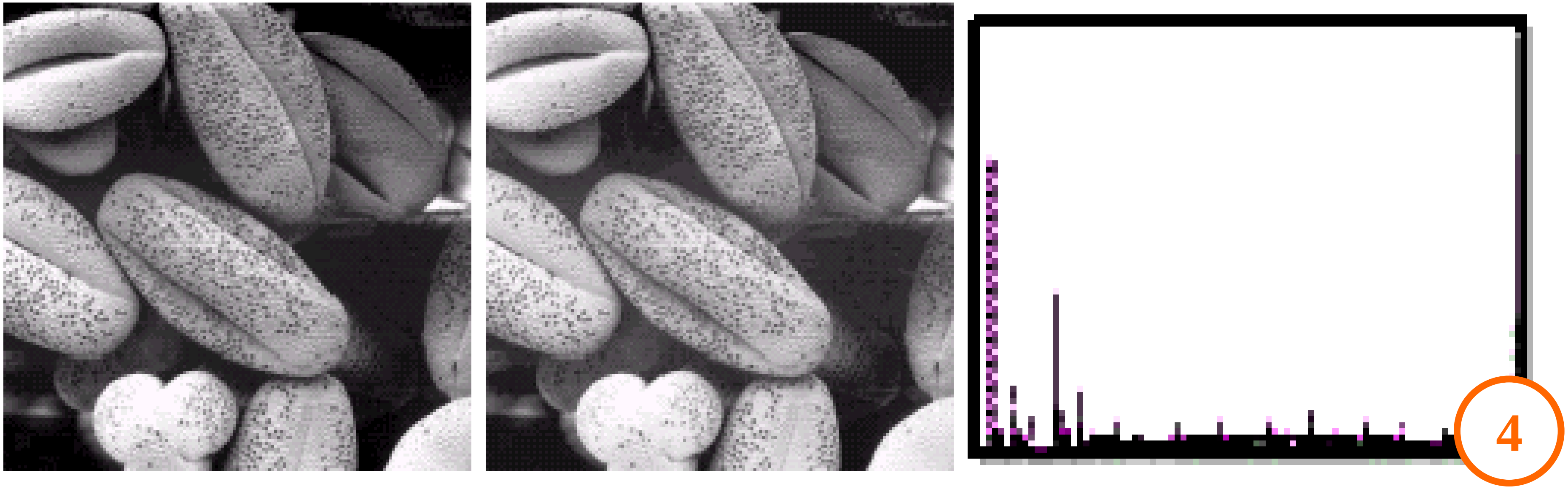








Equalization Example 1 (Cont..)



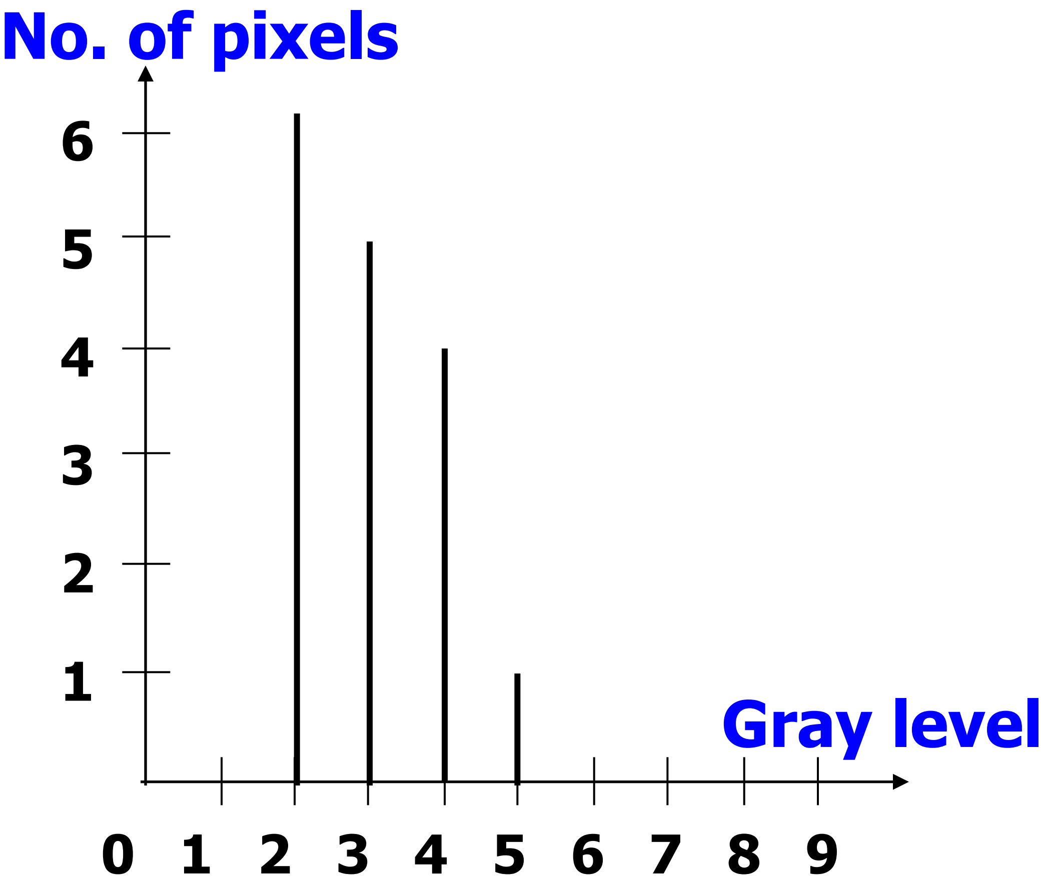
The quality is not improved much because the original image already has a broaden gray-level scale

Histogram Equalization: Example 2

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

4x4 image

Gray scale = [0,9]



Histogram

Histogram Equalization: Example 2

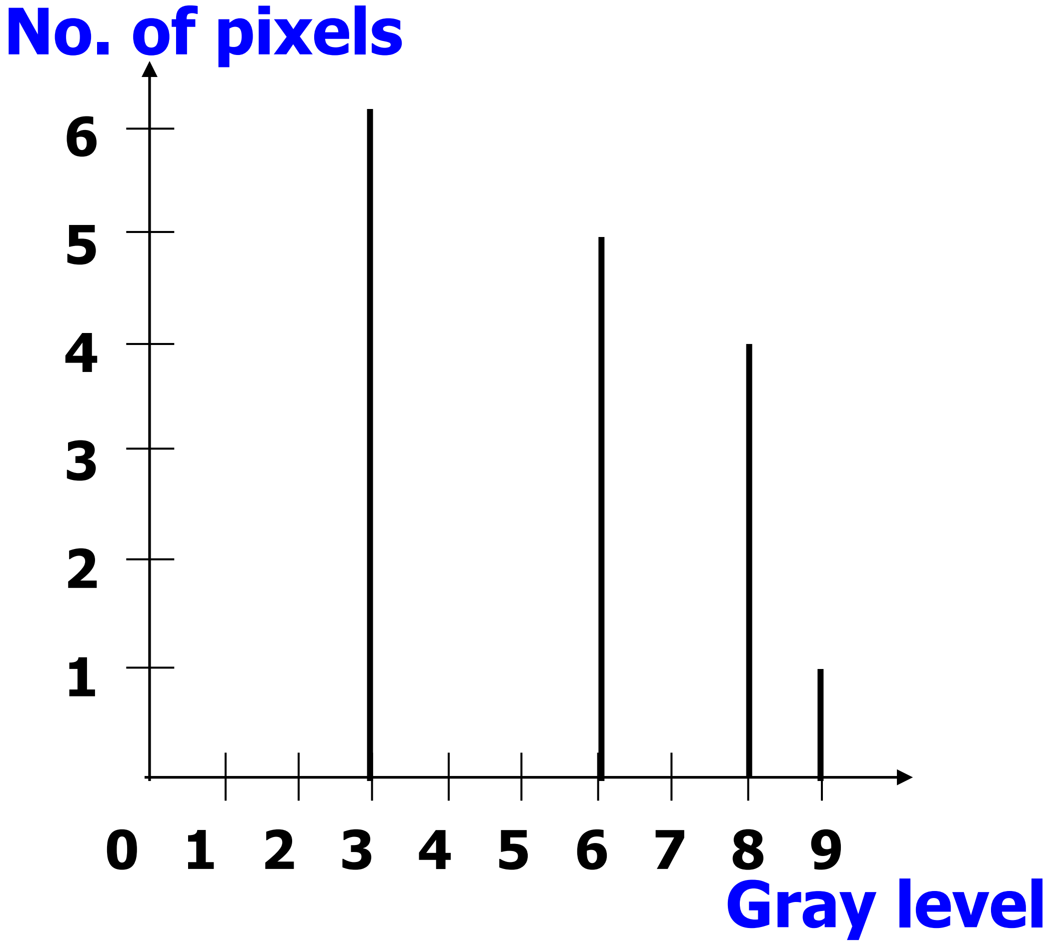
Gray Level (r)	0	1	2	3	4	5	6	7	8	9
No. of pixels	0	0	6	5	4	1	0	0	0	0
$\sum_{j=0}^{k} n_j$	0	0	6	11	15	16	16	16	16	16
$\sum_{j=0}^{k} \frac{n_j}{n}$	0	0	6 / 16	11 / 16	15 / 16	16 / 16	16 / 16	16 / 16	16 / 16	16 / 16
$9 \times \sum_{j=0}^{k} \frac{n_j}{n}$	0	0	3.3 ≈3	6.1 ≈6	8.4 ≈8	9	9	9	9	9

Histogram Equalization: Example 2

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

Output image

Gray scale = [0,9]



Histogram equalization

Histogram Equalization: Example 3

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

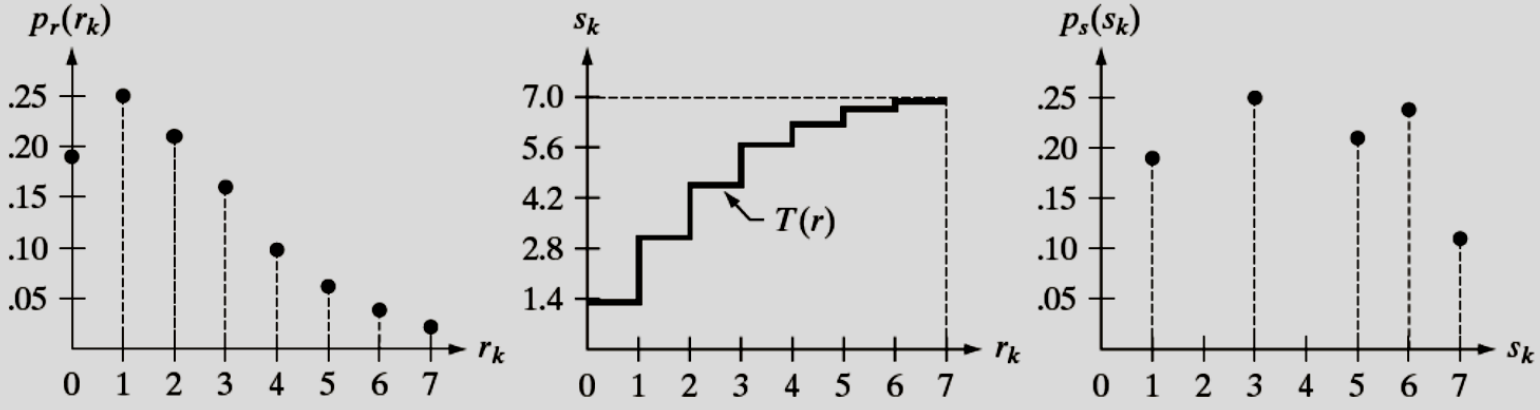
TABLE 3.1 Intensity distribution and histogram values for a 3-bit, 64×64 digital image.

$S_0 = 7p_r(r_0) = 1.33$

$S_1 = 7p_r(r_0) + 7p_r(r_1) = 3.08$

$S_2 = 4.55$, $S_3 = 5.67$, $S_4 = 6.23$

$S_5 = 6.65$, $S_6 = 6.86$, $S_7 = 7.00$



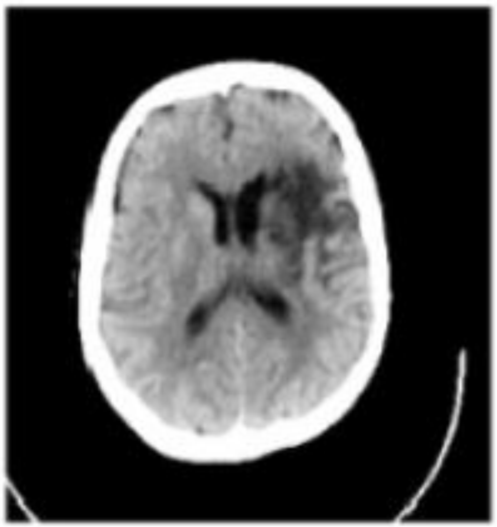
a b c

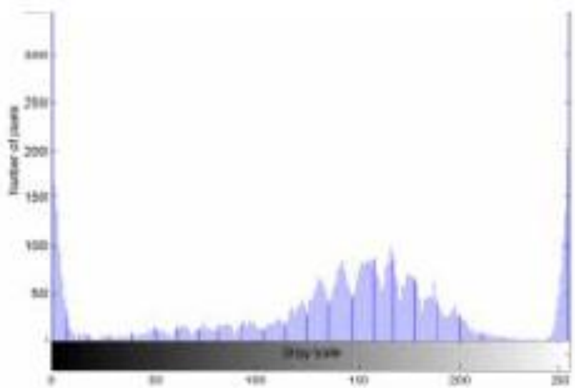
FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

A Note on Histogram Equalization

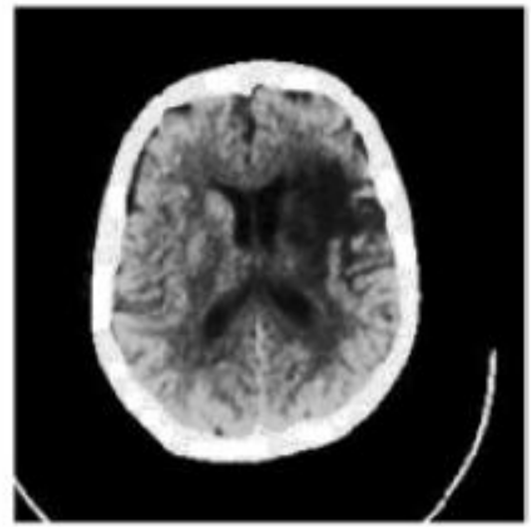
- It is clearly seen that
  - Histogram equalization distributes the gray level to reach the maximum gray level (white)
  - If the cumulative numbers of gray levels are slightly different, they will be mapped to little different or same gray levels as we may have to approximate the processed gray level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

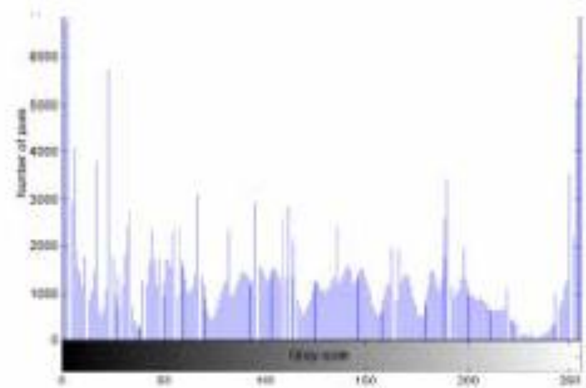
Histogram Equalization (HE): Example 4





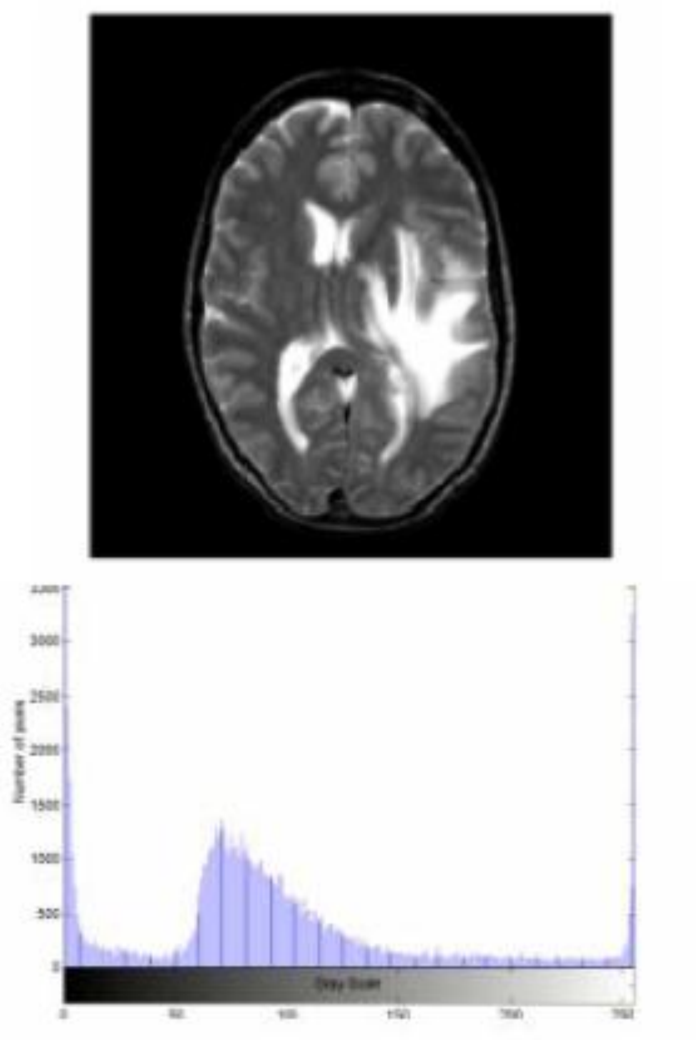
Before HE



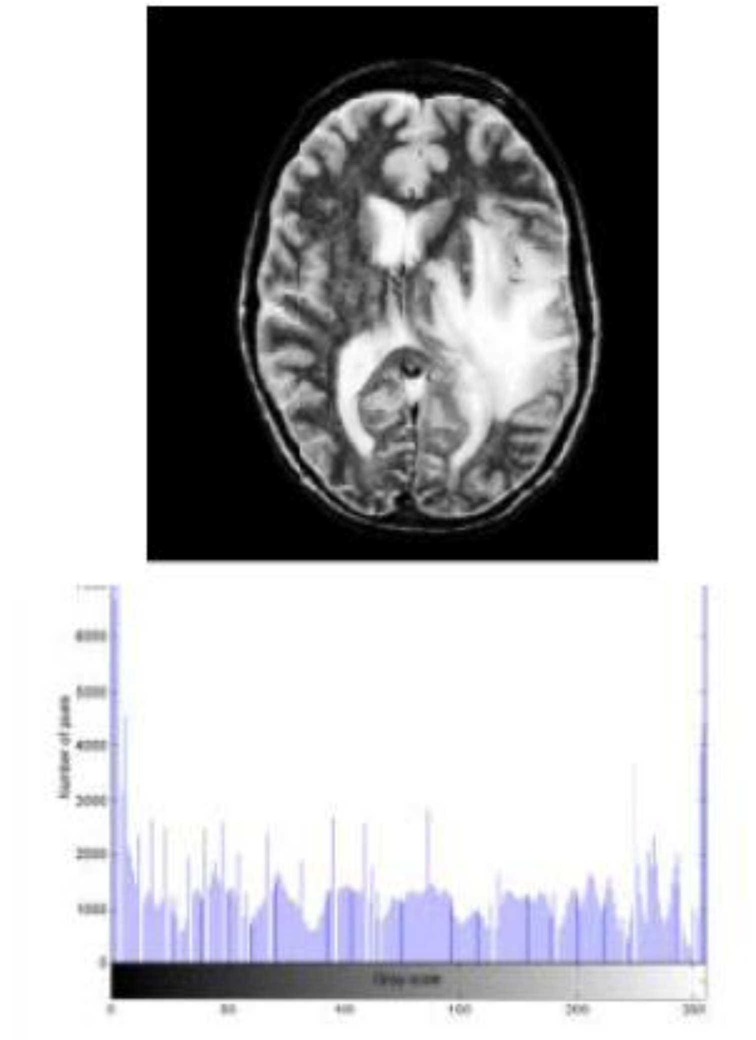


After HE

Histogram Equalization (HE): Example 5



Before HE



After HE

Histogram Matching (Specification)

- Histogram equalization has a disadvantage which is that it can generate only one type of output image.
- With Histogram Specification, we can specify the shape of the histogram that we wish the output image to have.
- It doesn’t have to be a uniform histogram

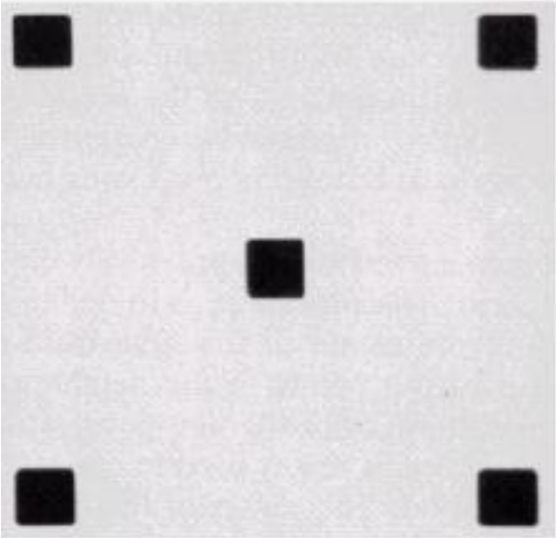
Global Vs Local Enhancement

- Histogram processing: global approach suitable for overall enhancement
  - Pixels are modified by a transformation function based on the gray-level content of an entire image
- Sometimes, we may need to enhance details over small areas in an image, which is called a local enhancement
  - Pixels in these areas may have negligible influence on the computation of a global transformation; and hence it is not necessary to get the desired local enhancement
- For local enhancement, transformation should be based on gray-level distribution in the neighborhood of every pixel

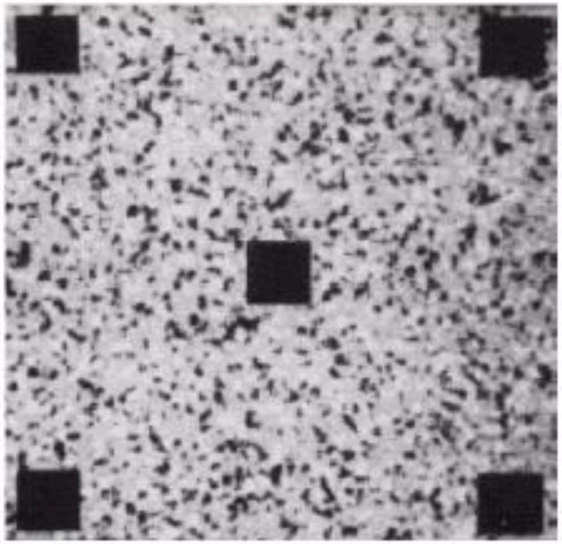
Local Histogram Processing

- Local histogram processing:
  - Define a neighborhood and move its center from pixel to pixel
  - At each location the histogram of the points in the neighborhood is computed and a histogram equalization transformation function is obtained
  - The gray level of the pixel centered in the neighborhood is mapped with this transformation function
  - The center of the neighborhood is moved to the next pixel and the procedure repeated

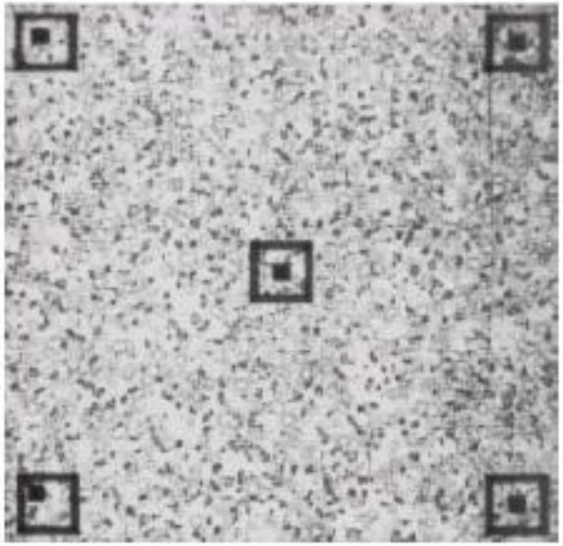
Example: Local Histogram Processing



Original image
(slightly noisy)

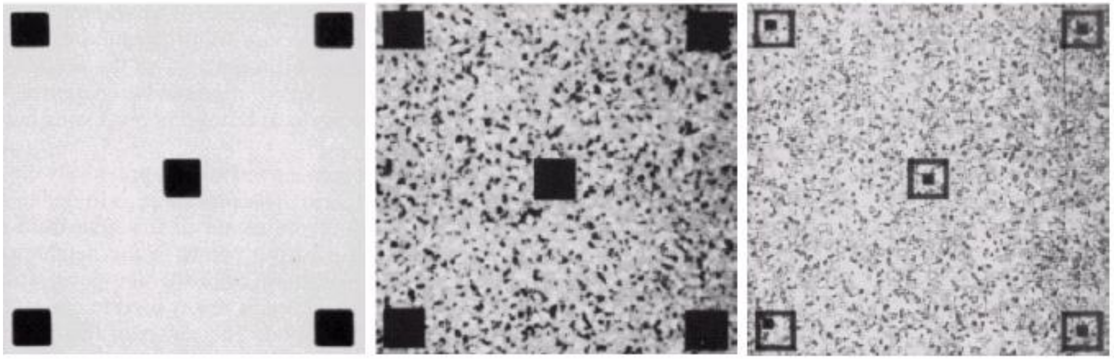


Global Histogram Equalization
(enhanced noise & slight increase contrast but does not reveal any new significant detail)



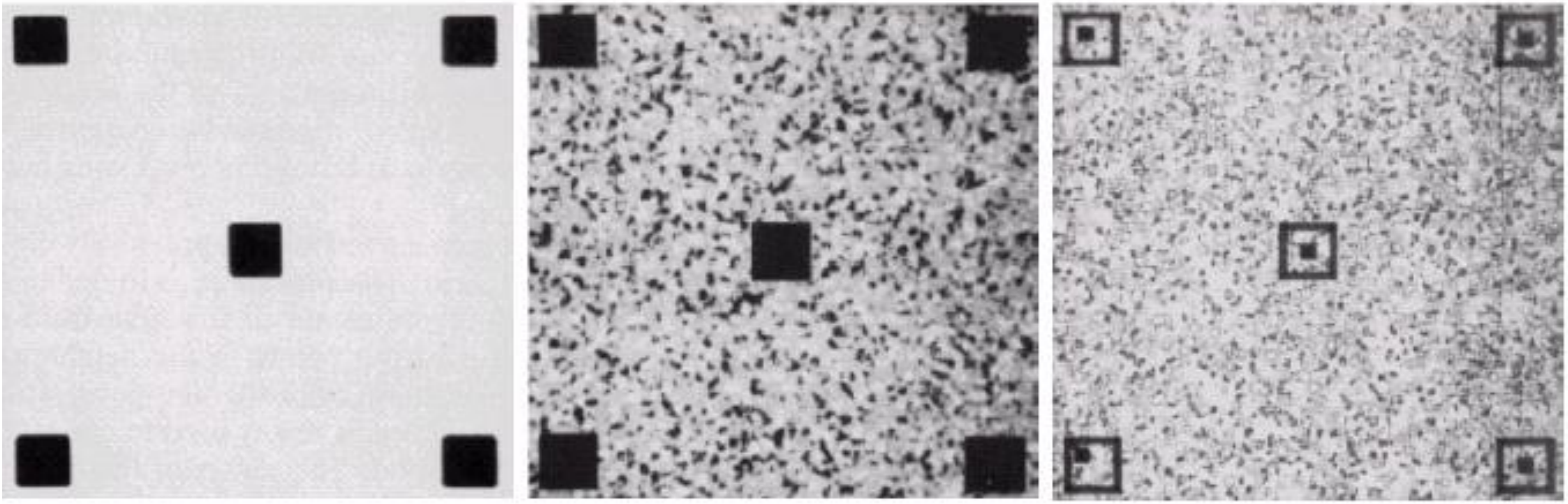
Local Histogram Equalization using a 7 x 7 neighborhood about each pixel
(reveals the small squares inside larger ones of the original image)

Explanation of the Result



- 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

Local Enhancement



- Approaches to reduce computation
  - Updating the histogram obtained in the previous location with the new data introduced at each motion step
    - It is possible since only one new row/column of the neighborhood changes during a pixel-to-pixel translation of the region
  - To utilize nonoverlapping regions, but it usually produces an undesirable checkerboard effect

Use of Histogram Statistics for Image Enhancement

- Mean of gray levels in an image: a measure of darkness, brightness of the image
- Variance of gray levels in an image: a measure of average contrast

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

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

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

$$\sigma^2 = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) - m]^2$$

Use of Histogram Statistics for Image Enhancement

- Two uses of the mean and variance for enhancement purposes
  - Global mean and variance are measured over an entire image and are used primarily for gross adjustments of overall intensity and contrast
  - Local mean and variance are used as the basis for making changes that depend on image characteristics in a predefined region about each pixel in the image

Use of Histogram Statistics for Image Enhancement

- Local mean and variance

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

- where
  - (x, y): coordinates of a pixel in an image
  - S_{xy}: neighborhood (sub image of specified size), centered at (x, y)
  - $r_{s,t}$: gray level at coordinates (s, t) in the neighborhood
  - $p(r_{s,t})$: neighborhood normalized histogram component corresponding to that value of gray level

Use of Histogram Statistics for Image Enhancement

- Local enhancement method

$$g(x,y) = \begin{cases} E\,.f(x,y) \text{ if } m_{S_{xy}} \le k_0 m_G, \quad k_1\sigma_G \le \sigma_{S_{xy}} \le k_2\sigma_G \\ f(x,y) \; otherwise \end{cases}$$

$m_G : Global\ mean \quad \sigma_G : Global\ variance$

- An enhancement method that can tell the difference between dark and light and, at the same time, is capable of enhancing only the dark or light areas

Use of Histogram Statistics for Image Enhancement

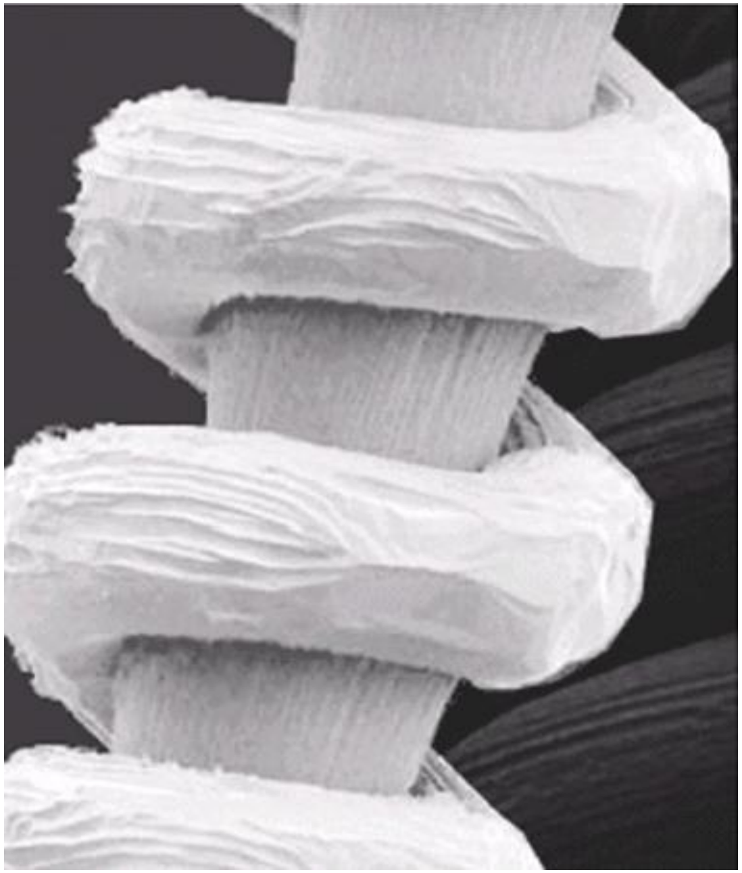
- Local enhancement method

$$g(x,y) = \begin{cases} E \,.\, f(x,y) \text{ if } m_{S_{xy}} \le k_0 m_G, \; k_1 \sigma_G \le \sigma_{S_{xy}} \le k_2 \sigma_G \\ f(x,y) \text{ otherwise} \end{cases}$$

Checks the candidacy of pixel at a point (x, y) for processing	Restricts the lowest values of contrast, we wish to accept	Checks whether the contrast of an area makes it a candidate for enhancement

- Constants are +ve. (Values > 1, to enhance light areas and < 1 for dark areas)
- In case of enhancement of dark areas small value for E is chosen in order to preserve the general visual balance of the image
- Choosing the parameters generally requires a bit of experimentation with a given image/class of images

Local Enhancement



SEM image of a tungsten filament and support

Enhanced SEM image

(note the enhanced area on the right side of the image)

References

- Medical Image Processing course at McMaster University, Canada
- Digital Image Processing course by Dr. Wanasanan Thongsongkrit

  wanasana@eng.cmu.ac.th
- Digital Image Processing course by Dr. Brian Mac Namee

  http://www.comp.dit.ie/bmacnamee/
- R. C. Gonzalez and R. E. Woods, Digital Image Processing, Third Edition, Pearson, 2012.