CONTRAST ENHANCEMENT

Aziz M. Qaroush

Birzeit University

First Semester 2015/2016

Outline

- What is Image Enhancement ?
- Background
- Intensity Transformation Functions
  - Negatives
  - Log and Inverse Log
  - Power-Law
  - Piecewise Transformation
  - Gray-level and Bit Slicing
- Histogram Processing
  - Equalization
  - Specification
  - Local Processing

What is a Digital Image? (cont…)

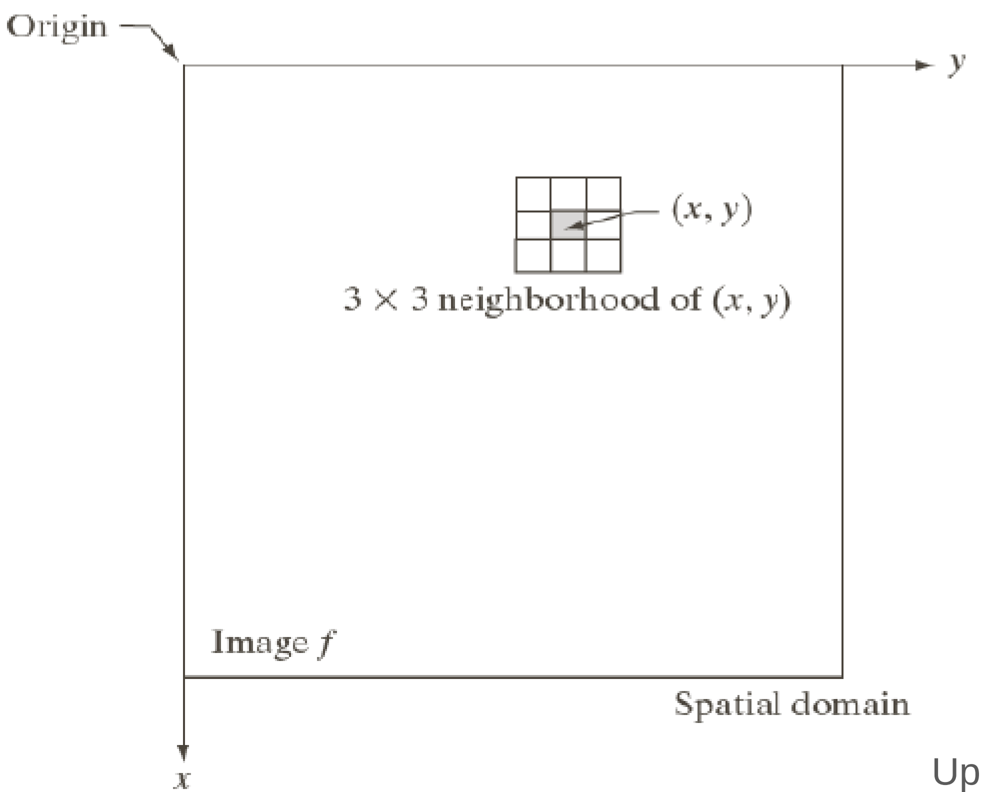
- Spatial domain of the image is the set of pixels composing the image
- Enhancement in the spatial domain involves direct operation on the pixel intensities
- This can be expressed mathematically as

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

  - f(x,y) is the input image
  - g(x,y) is the output image
  - T[ ] is an operator defined over some neighborhood of (x,y)
- **Important**
  - Keep in mind that g(x,y) may take any value from the set of available gray levels only. Thus, when mapping we should assign the mapped value to the closest level

What is a Digital Image? (cont…)

- Defining the neighborhood around (x,y)
  - Use a square/rectangular subimage that is centered at (x,y)
- Operation
  - Move the center of the subimage from pixel to pixel and apply the operator T at each location (x,y) to compute the output g(x,y)



Point Processing

- The simplest form of the operator T is when the neighborhood size is 1x1 pixels. Accordingly, g(x,y) is only dependent on the value of f at (x,y)
- In this case, T is called the gray-level or intensity transformation function that can be represented as

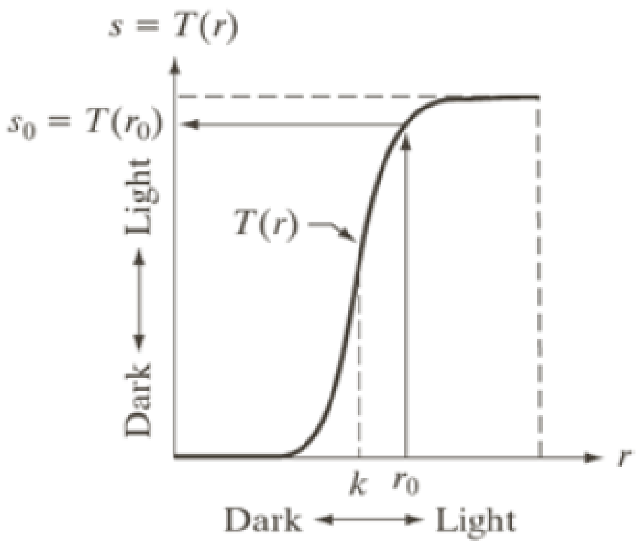
$$s = T(r)$$

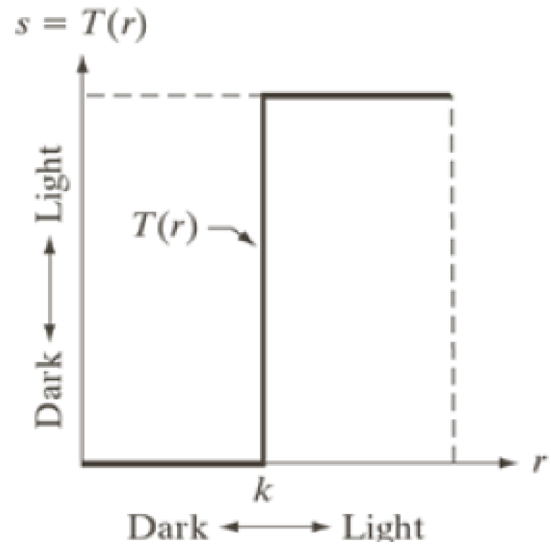
  - s is a variable denoting g(x,y)
  - r is a variable denoting f(x,y)

- This is kind of processing is referred as point processing

Point Processing - Example

- Intensity transformation function examples





T(r) performs contrast stretching by mapping levels less than k to narrow range while those above k are mapped to wider range

T(r) reduces the number of levels in the image to two

Point Processing Example - Thresholding

- Thresholding transformations are particularly useful for segmentation in which we want to isolate an object of interest from a background

$$s = \begin{cases} 1.0 & r > \text{threshold} \\ 0.0 & r <= \text{threshold} \end{cases}$$

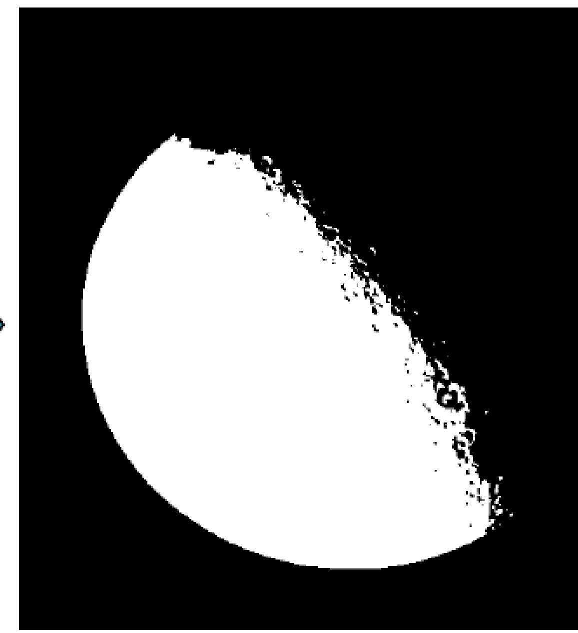


Image enhancement: Contrast Enhancement

- Contrast is an important factor in any subjective evaluation of image quality.
- Contrast is created by the difference in luminance reflected from two adjacent surfaces. In other words, contrast is the difference in visual properties that makes an object distinguishable from other objects and the background.
- In visual perception, contrast is determined by the difference in the colour and brightness of the object with other objects.
- Contrast enhancements are typically performed as a contrast stretch followed by a tonal enhancement, although these could both be performed in one step.

Contrast Enhancement - Gray Level Transformations

- **Contrast Enhancement done by pixel value mapping called Gray Level Transformations**
- Mapping can be performed by mathematical substitution or lookup tables
- Some common functions are
  - Linear (negative/identity)
  - Logarithmic (log/inverse log)
  - Power law (nth power/nth root)
  - Piecewise-Linear Transformations

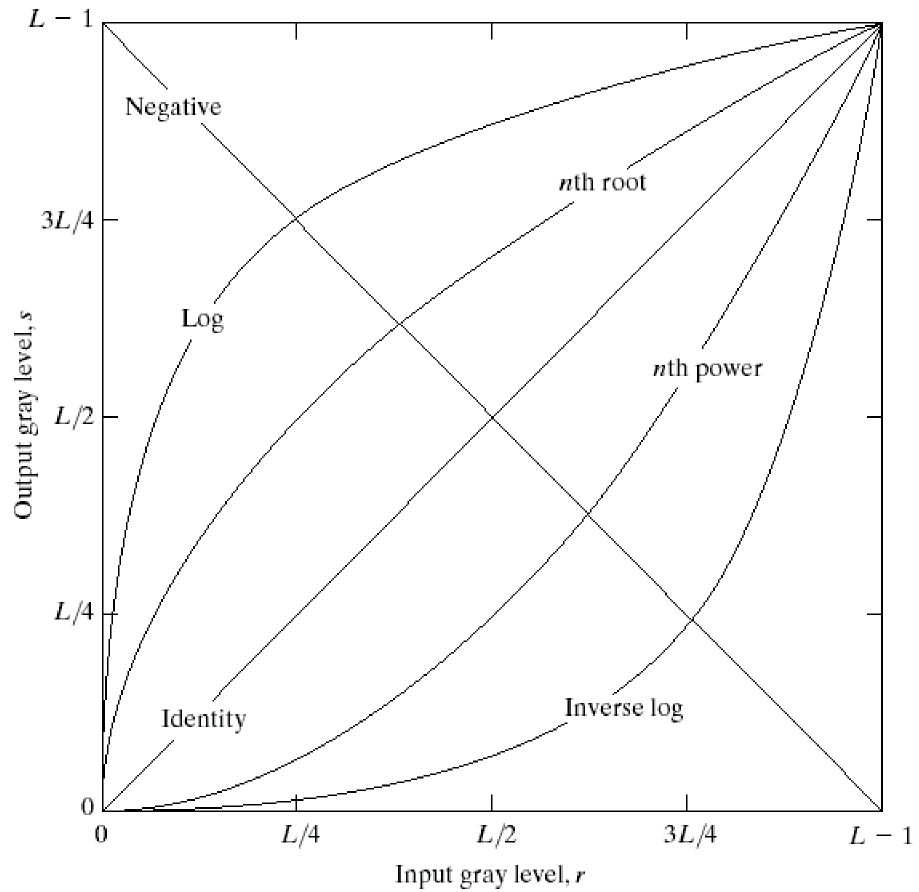


Image Negatives

- Can be performed by using

$$s = L - 1 - r$$

  - where L-1 is the maximum intensity value

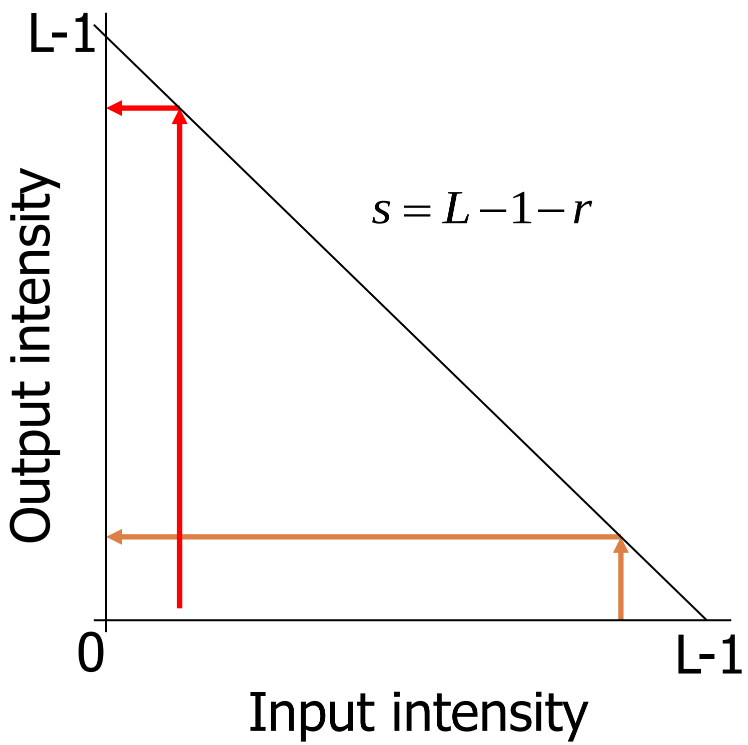
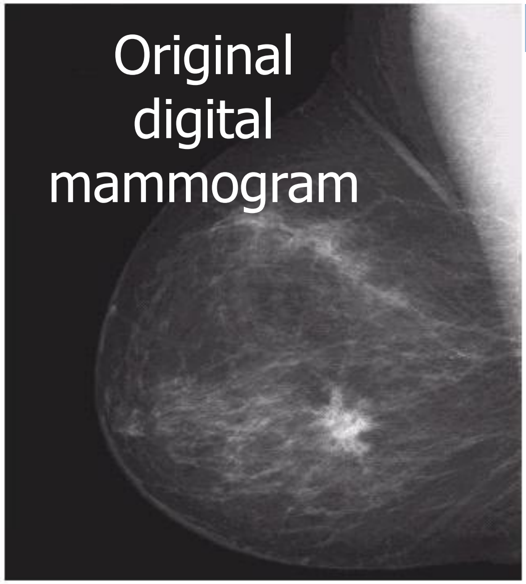
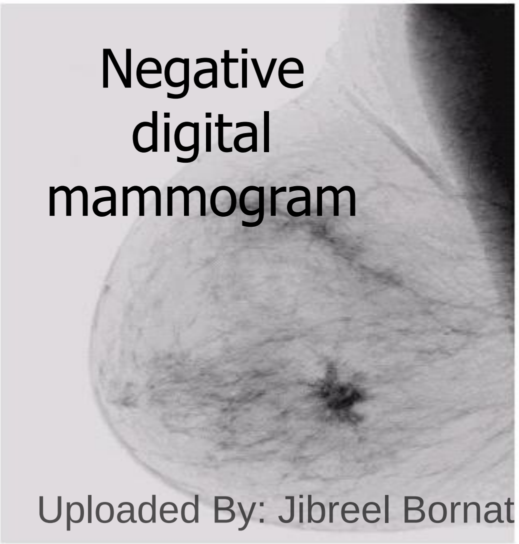


Image Negatives - Example





Log and inverse Log Transformations

- The general form of the log transformation

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

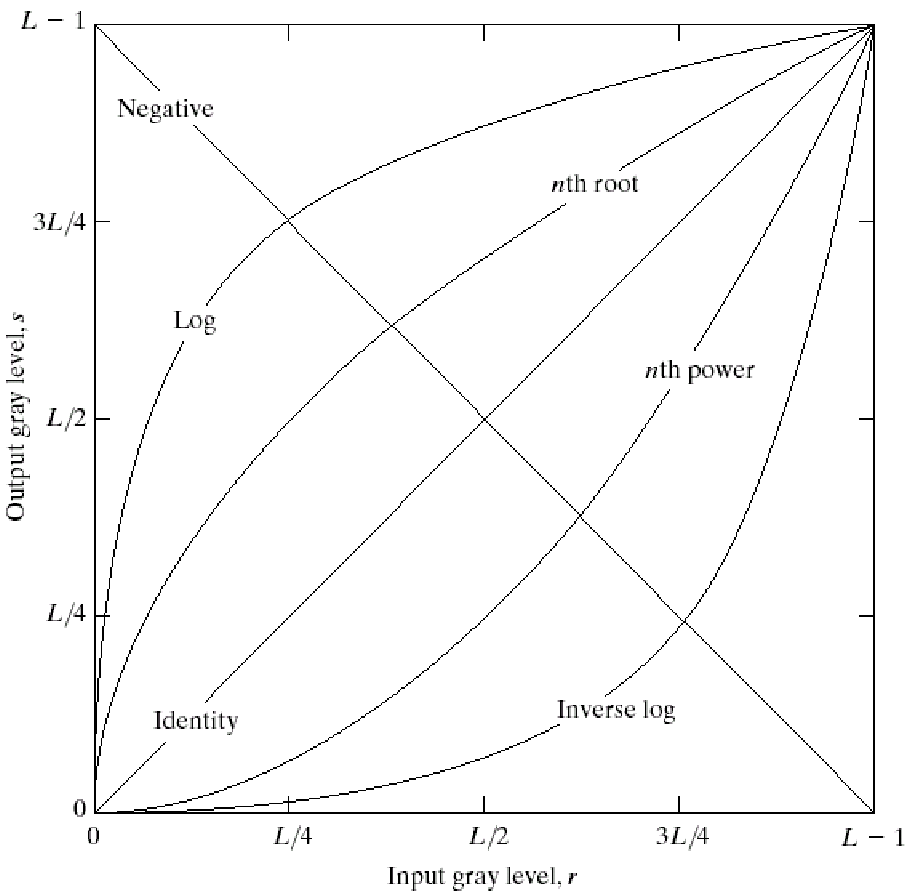
  - b is the base
  - Maps narrow range of low intensity levels to wider range and wide range of high intensity levels to narrower range
  - Usually used to expand the values of dark pixels and compress the higher level values
- The general form of the inverse log

$$s = b^{cr} - 1$$

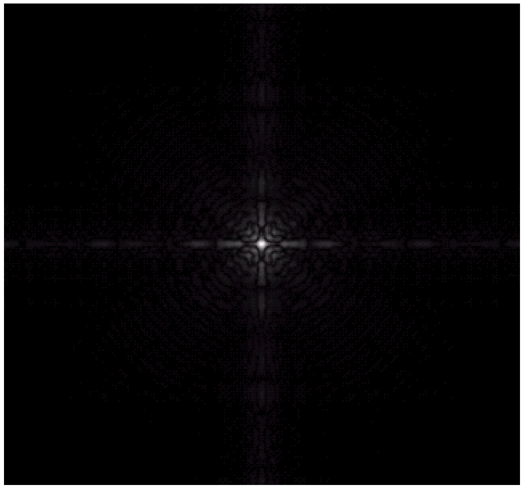
  - Its operation is the opposite of the log transformation

Log Transformation Example

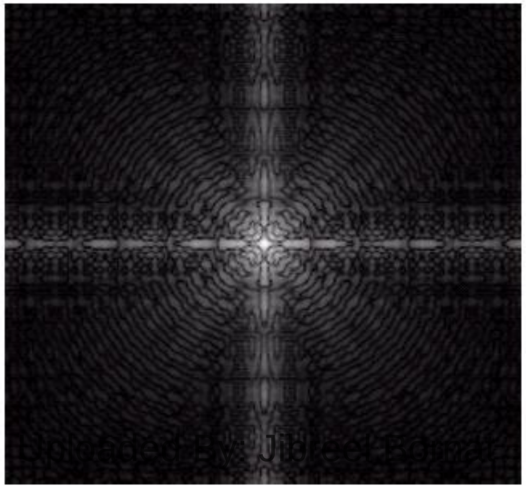
- It is very important in mapping wide dynamic ranges into narrow ones
- Fourier spectrum values in the range $[0, 1.5\text{x}10^6]$ transformed to $[0, 255]$ using log transformation



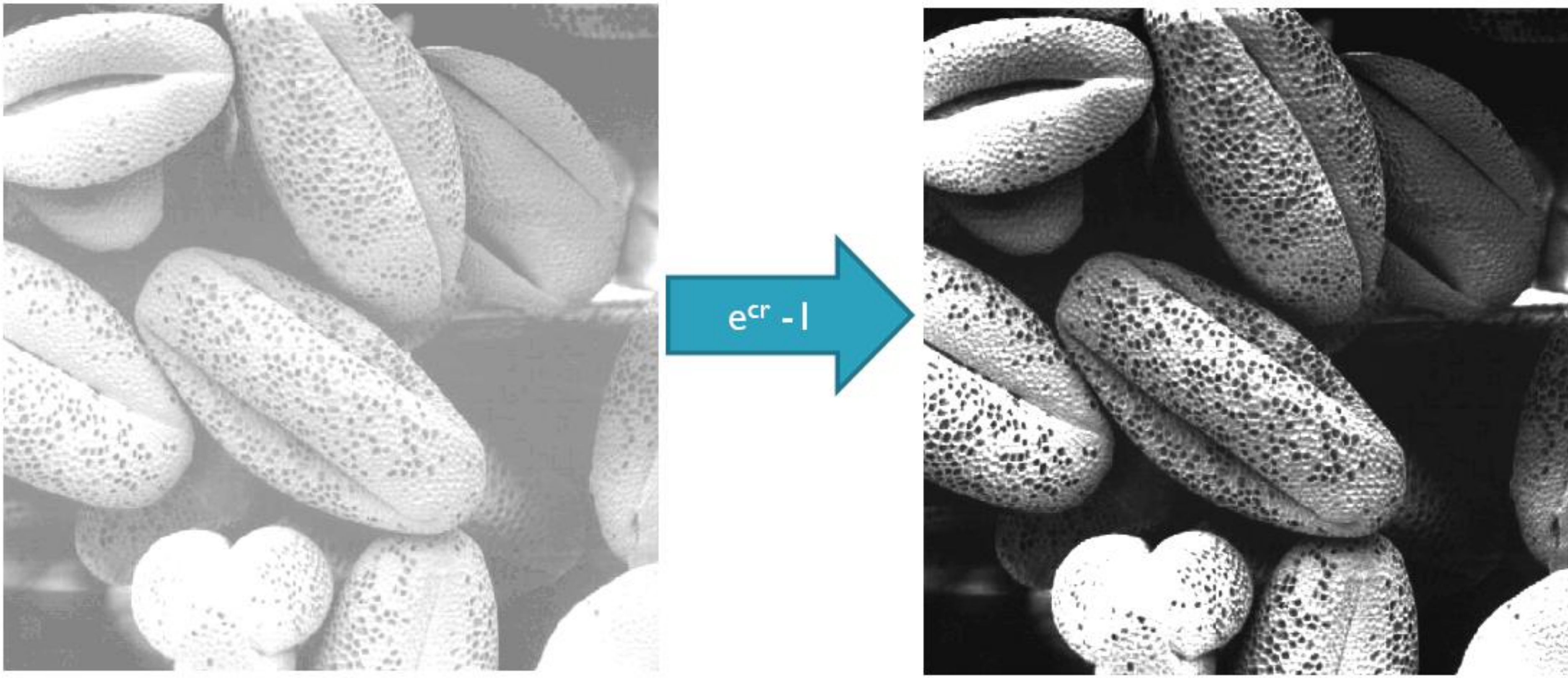
Fourier spectrum



Log Tr. of Fourier spectrum



Inverse Log Transformation Example



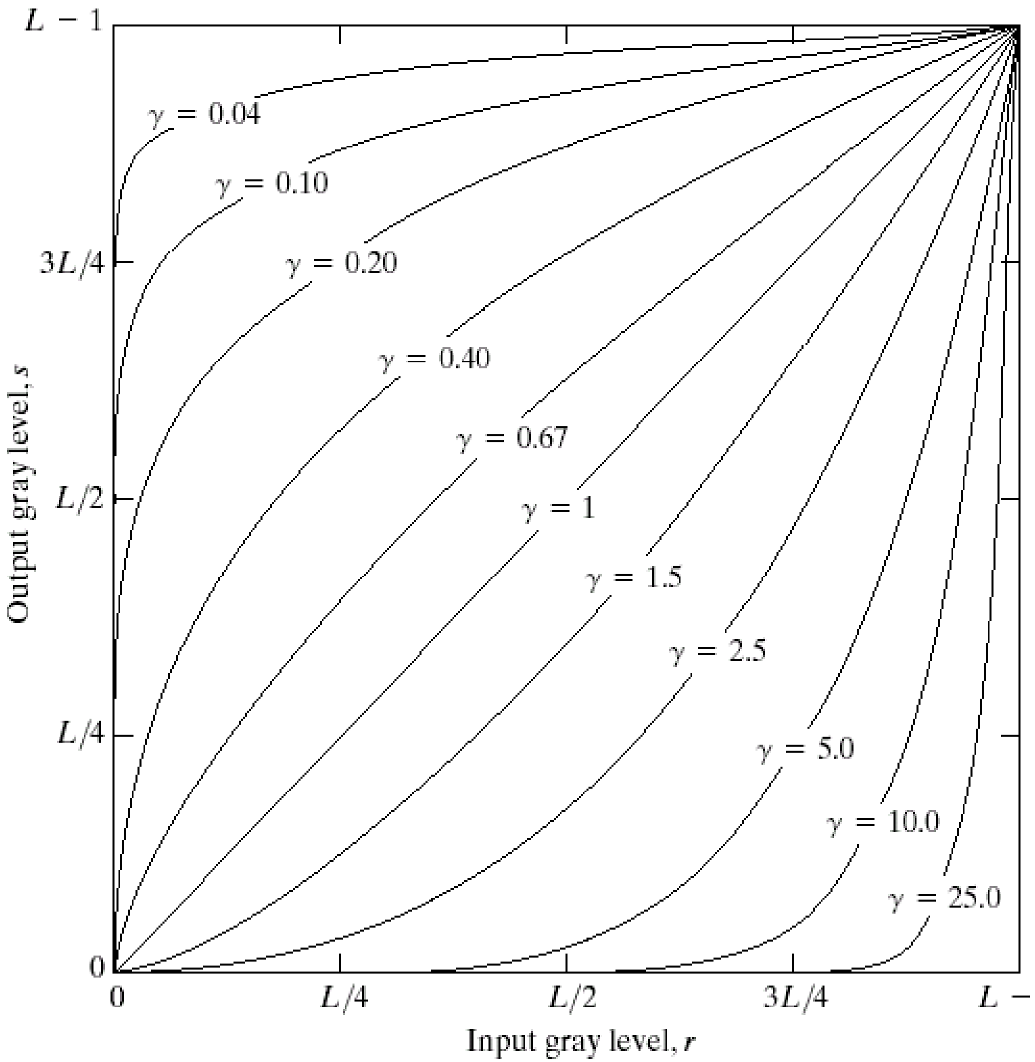
Power-Law transformations

- The general form

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

- Power law is similar to log when gamma < 1 and similar to inverse log when gamma > 1
- Like the logarithmic transform, they are used to change the dynamic range of an image. However, in contrast to the logarithmic operator, they enhance high intensity pixel values.

Power-Law transformations



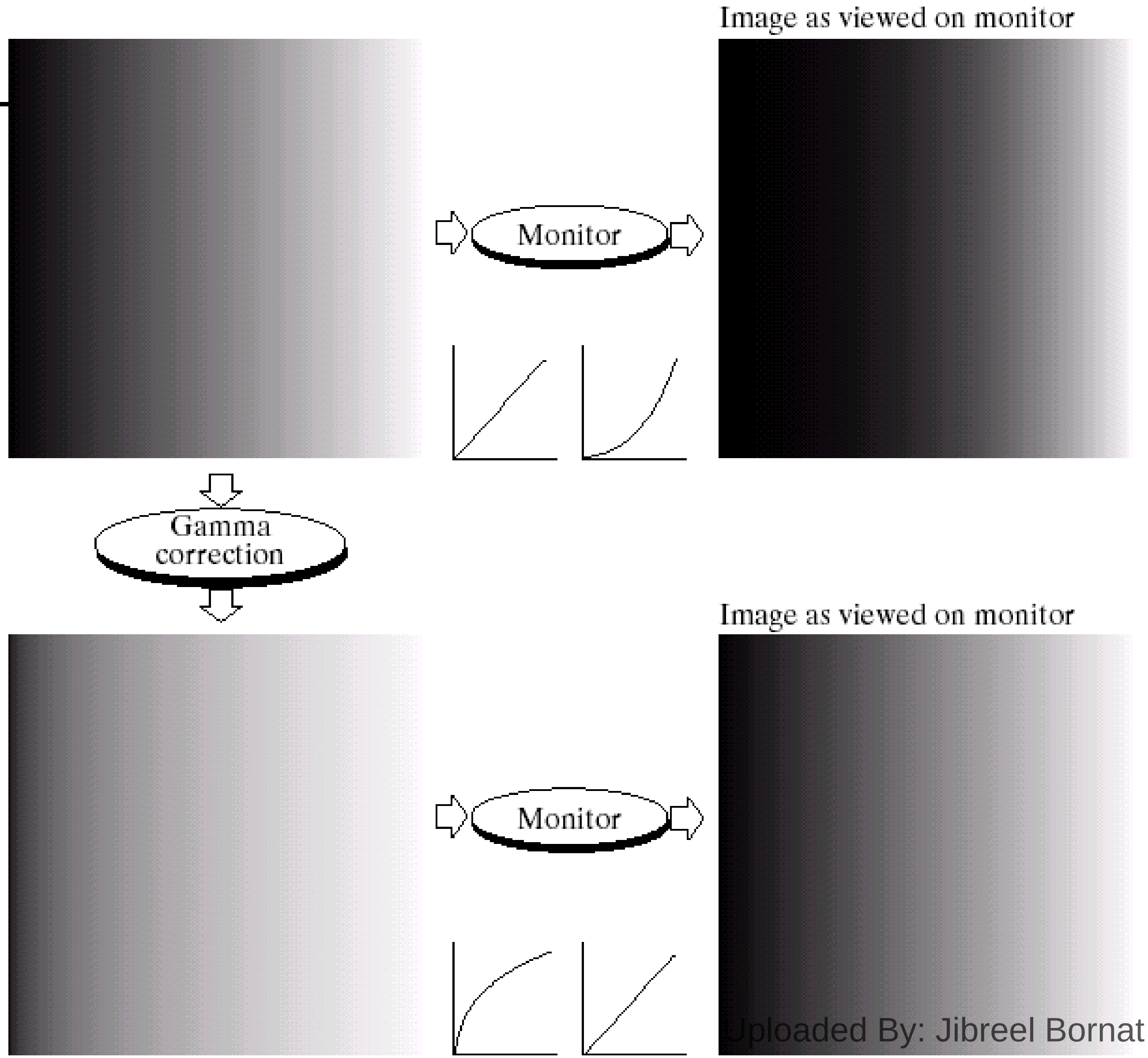
power-law curves with fractional values of γ map a narrow range of dark input values to a wider range of output values, and opposite for higher values of input levels

Power-Law transformations

- Gamma-correction
  - Display devices have intensity-to-voltage response that is a power functions. Thus, images tend to be darker when displayed. Correction is needed using nth root before feeding the image to the monitor
  - Solution –display image after gamma correction to value that represents “average” of the types of monitors and computer systems to be used to display the image.

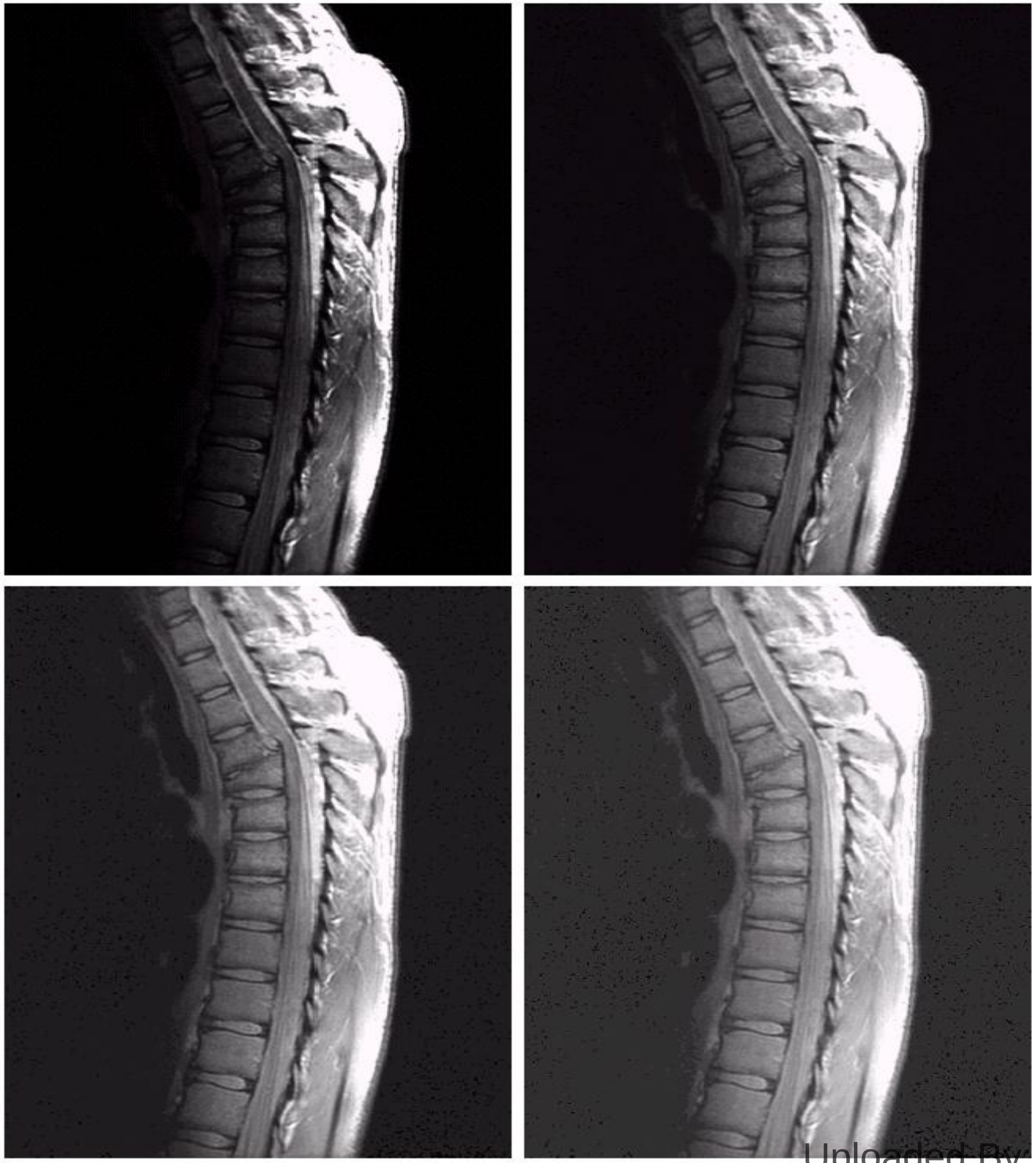
Power-Law transformations

Gamma-correction Example



Power-Law transformations

- Gamma-correction application



a b
c d

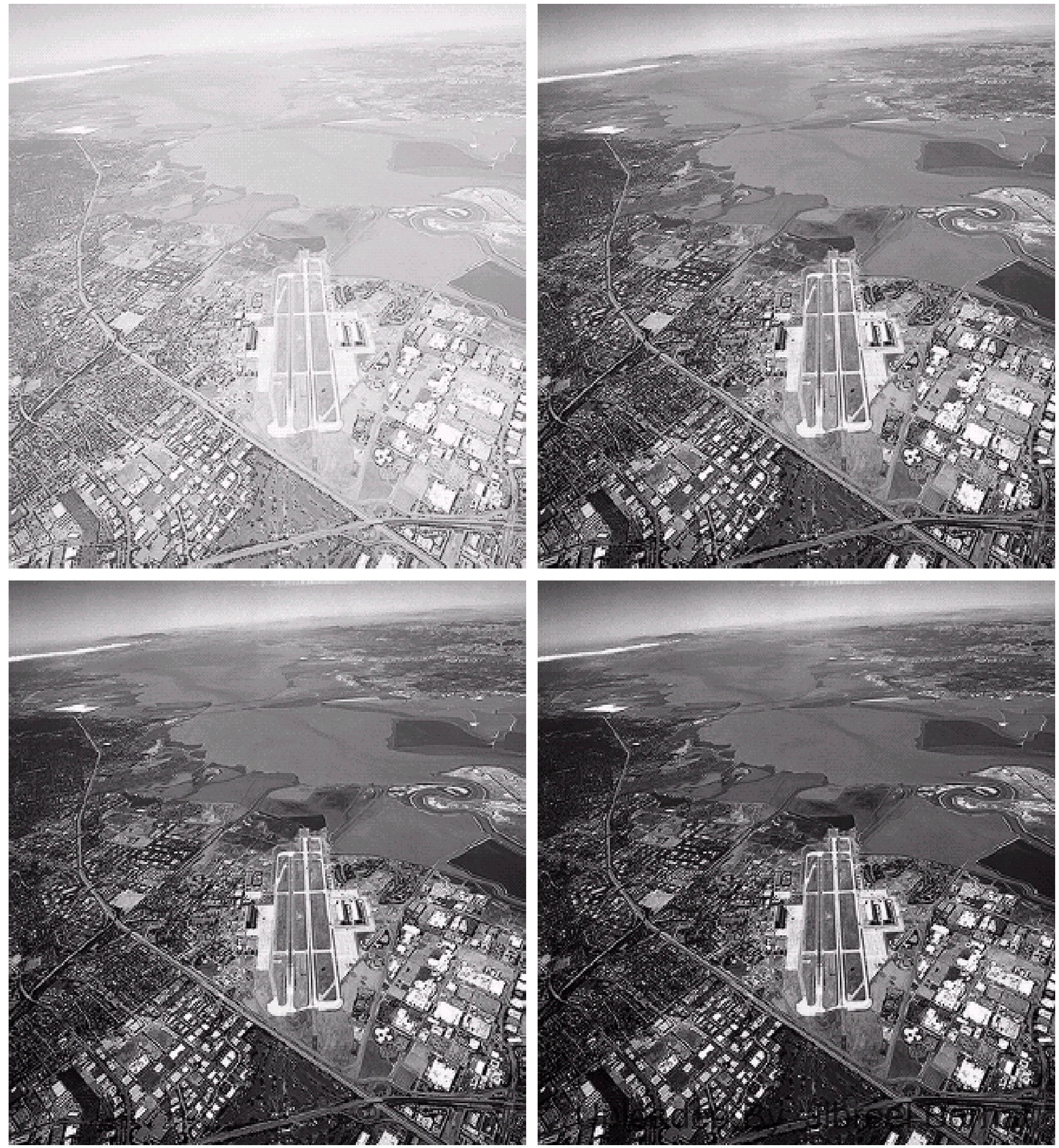
FIGURE 3.8 (a) Magnetic resonance (MR) image of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image for this example courtesy of Dr. David R. Pickens, Department of Radiology and Radiological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

Power-Law transformations

- Gamma-correction application

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.)

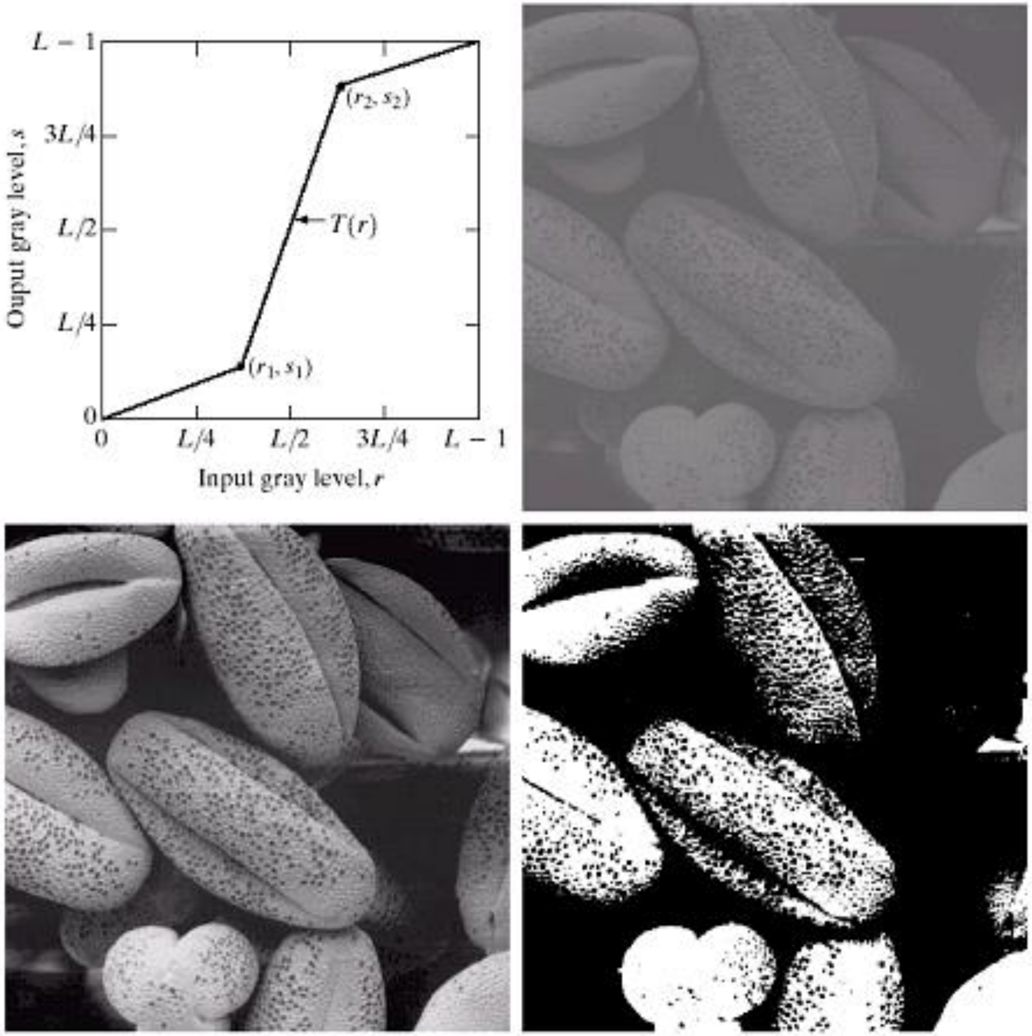


Piecewise-Linear Transformations

- Can represent arbitrarily complex functions to achieve different results
  - **Contrast stretching**
  - **Gray-level Slicing**
  - **Bit-plane Slicing**

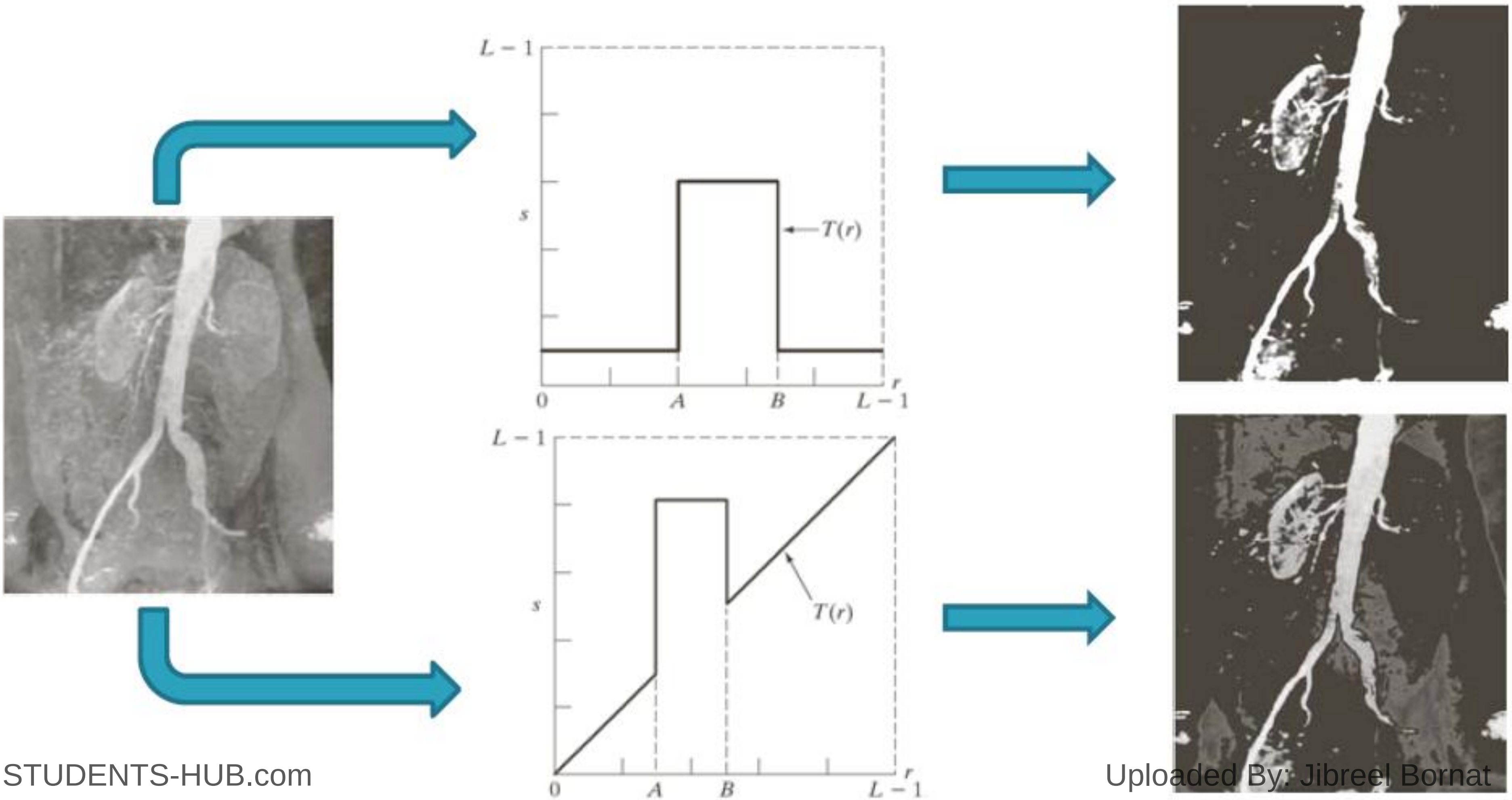
Contrast Stretching

- Improve the contrast by `stretching' the range of intensity values it contains to span a desired range of values
  - $r1 \leq r2$ and $s1 \leq s2$ to preserve the order of gray levels
  - The result depends on the values of r1, r2, s1, and s2



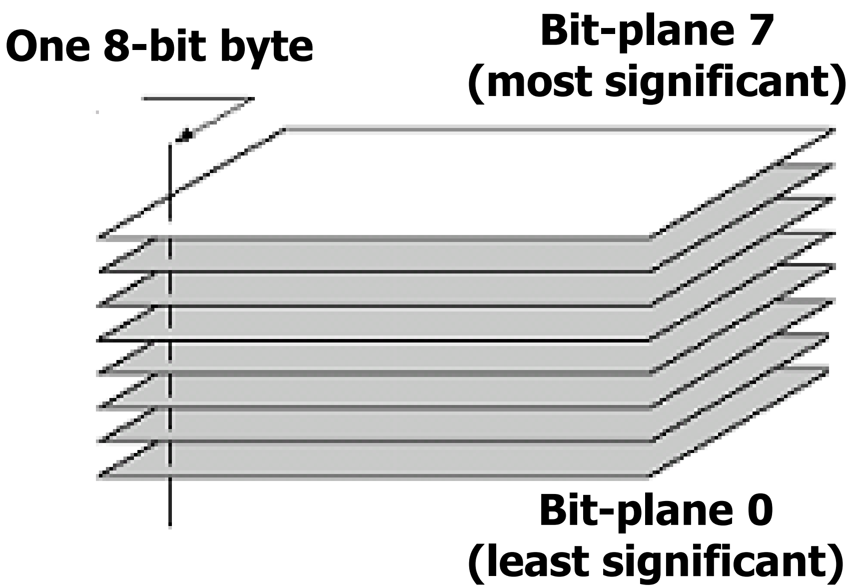
Gray-level Slicing

- Used to highlight specific range of gray levels
- Two approaches

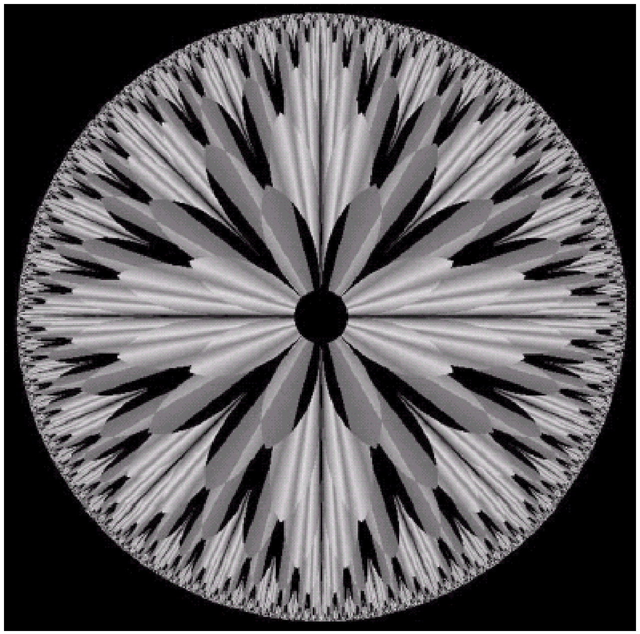


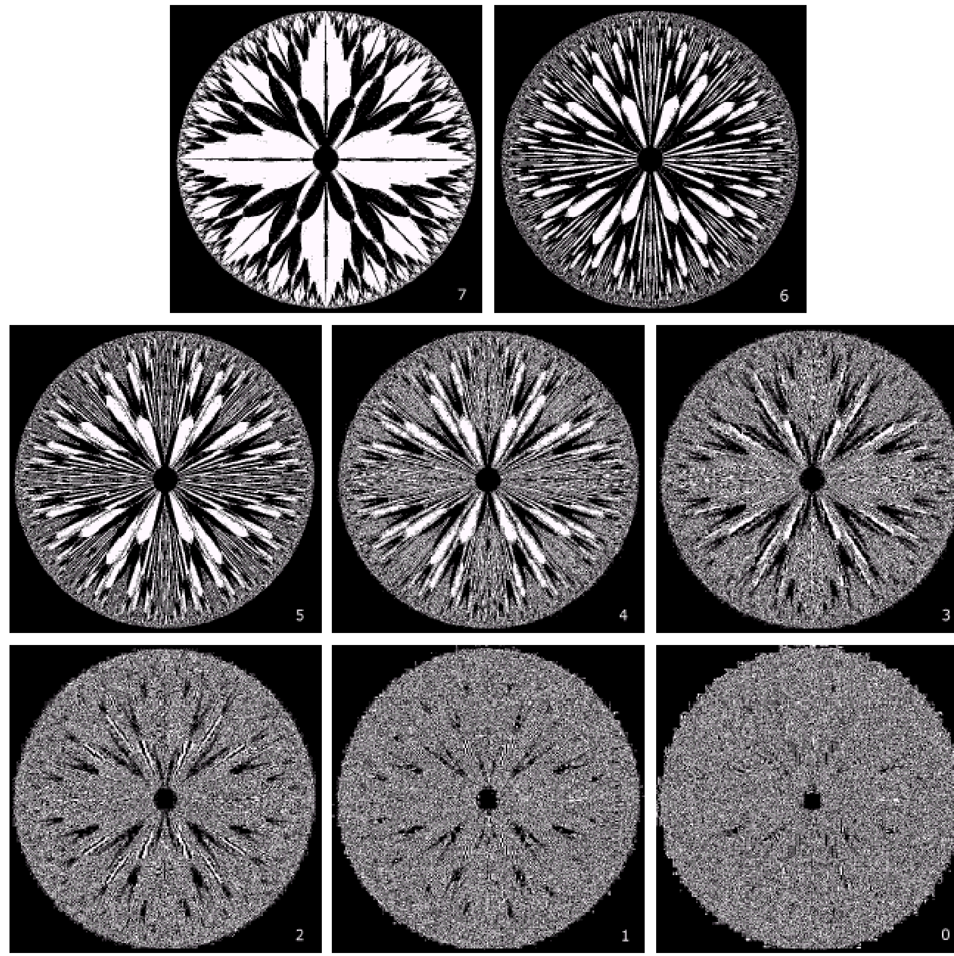
Bit-plane Slicing

- Highlight the contribution of specific bits to the appearance of the image
- Each pixel value is represented by a set of bits
- Lower bits correspond to fine details while higher bits correspond to the global visual content
- Useful in image compression !



Bit-plane Slicing - example





Bit-plane Slicing - example

Bit-plane Slicing - example

Planes 7 & 6

Planes 7,6,5

Planes 7,6,5,4

Histogram Processing

- For an image with gray levels in [0,L-1] and MxN pixels, the histogram is a discrete function given by

$$h(r_k) = n_k$$

  - 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

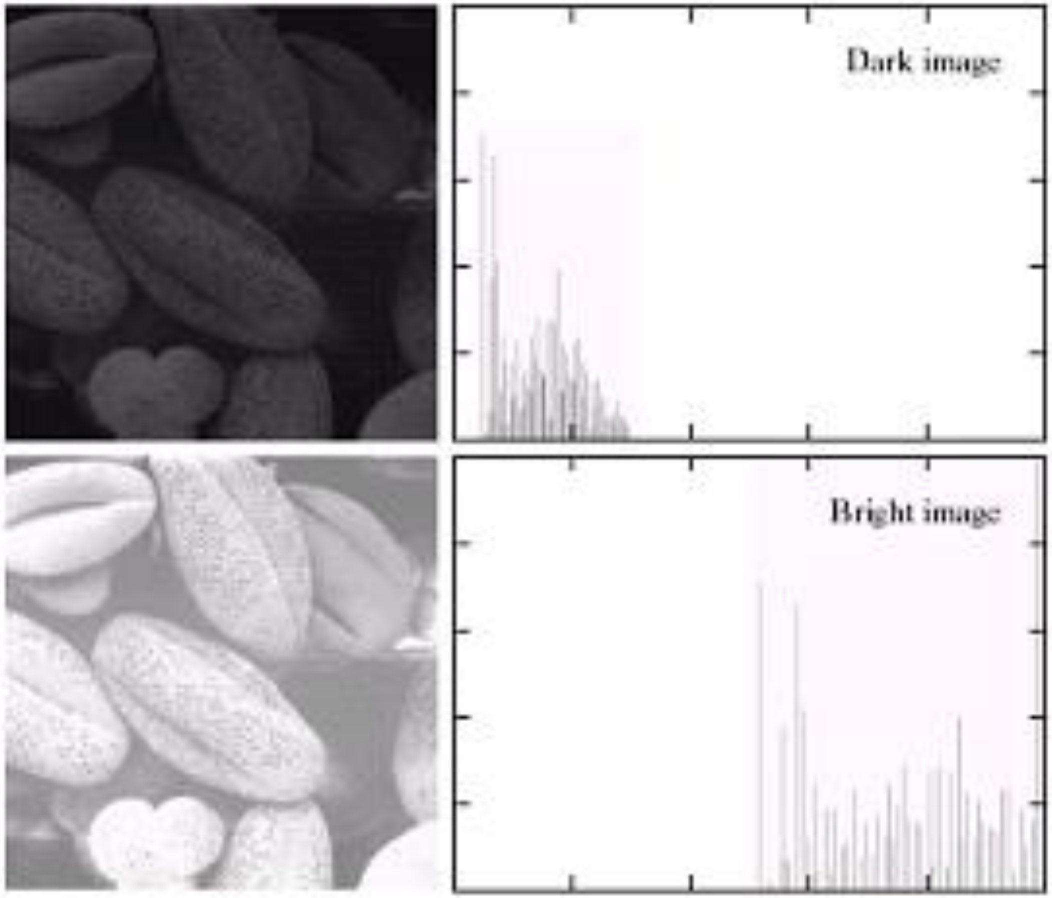
- It is a common practice to normalize the histogram function by the number of pixels in the image by

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

  - The normalized histogram can be used as an estimate of the probability density function of the image
  - Histograms are widely used in image processing: enhancement, compression, segmentation …

Histogram Processing

- For enhancement, histograms can be used to infer the type of image quality: dark, bright, low or high contrast



Dark image

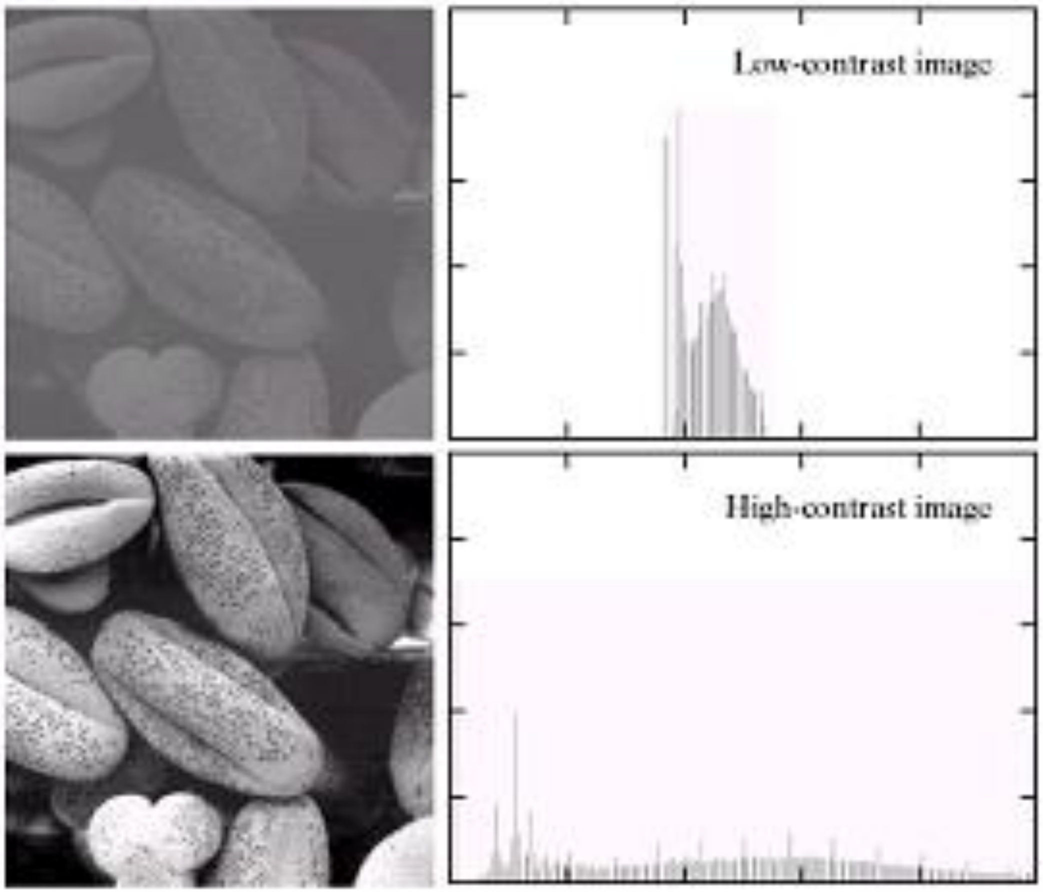
Components of histogram are concentrated on the low side of the gray scale.

Bright image

Components of histogram are concentrated on the high side of the gray scale.

Histogram Processing

- For enhancement, histograms can be used to infer the type of image quality: dark, bright, low or high contrast



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 higher than the others

Histogram Equalization

- It is quite acceptable that high contrast images have flat histograms (uniform distribution)
- 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
- Histogram equalization attempts to transform the original histogram into a flat one for the goal of better contrast

Histogram Equalization

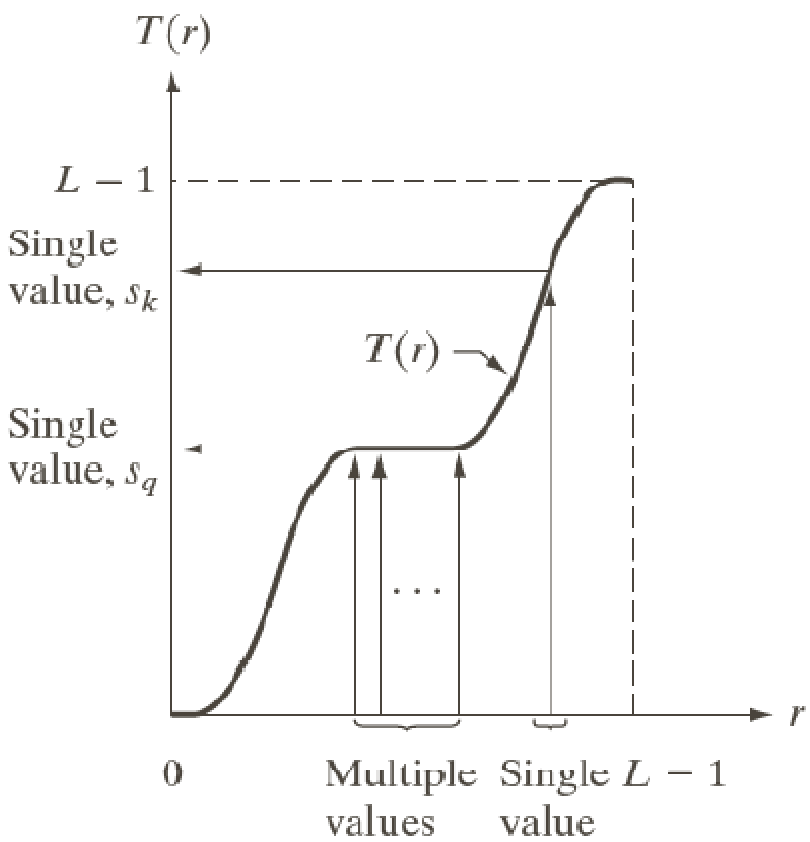
- Let r be a continuous variable that represents the intensity values in the range [0,L-1], then a valid transformation function for enhancement purposes s = T(r) should satisfy
  - T(r) is monotonically increasing in the interval $0 \le r \le L-1$
    - Preserves the increasing order from black to white in the output image thus it won't cause a negative image
  - T(r) is bounded by [0,L-1] for all values of r
    - Guarantees that the output gray levels will be in the same range as the input levels.
  - The inverse transformation function that maps s back to r

$$r = T^{-1}(s)$$

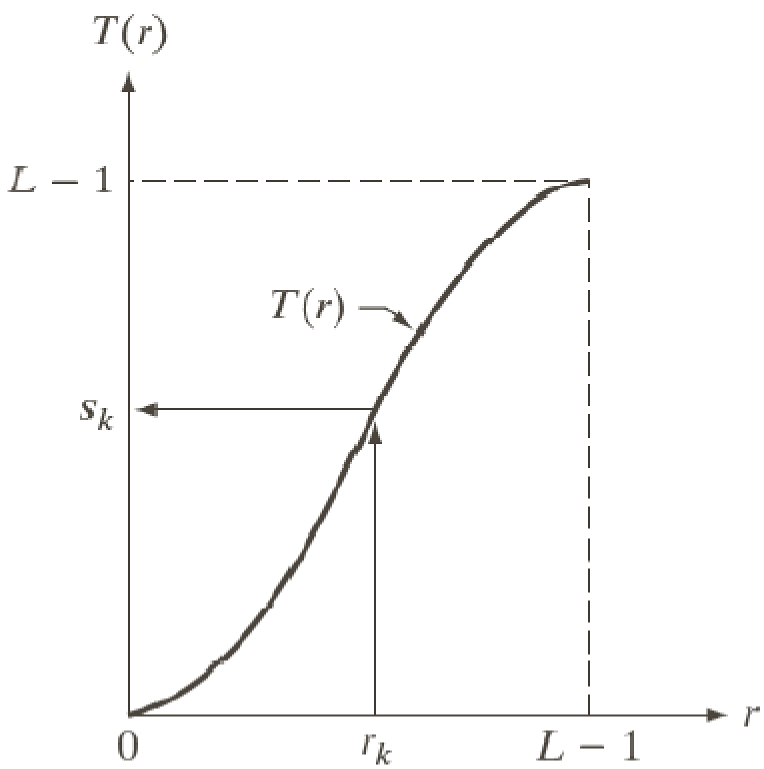
    - requires that T(r) to be strictly monotonically increasing

Histogram Equalization

- Examples of transformation functions



Monotonically Increasing



Strictly monotonically increasing

Histogram Equalization

- Consider the gray level intensity represented by r as a random variable in the interval [0,L-1]
- If a random variable ***r*** is transformed by a monotonic transformation function ***T(r)*** to produce a new random variable ***s,***
- Then probability density function of ***s*** can be obtained from knowledge of ***T(r)*** and the probability density function of ***s***, as follows:

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

  - Note that the probability density function (pdf or shortly called density function) of random variable x is defined as the derivative of the cdf

$$p(x) = \frac{dF(x)}{dx}$$

Histogram Equalization

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

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

  - Where w is a dummy variable of integration
  - 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)
  - When the upper limit is r = L-1 the integral evaluates to 1. thus condition (b) satisfied also.

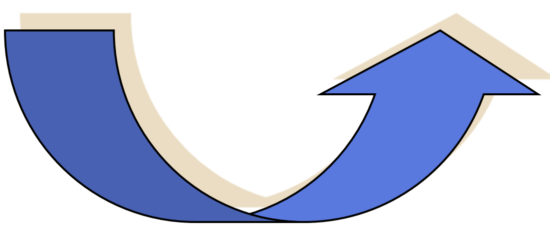
Histogram Equalization

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

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

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

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



Substitute and yield

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

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

Histogram Equalization

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

yields a random variable s characterized by a uniform probability function

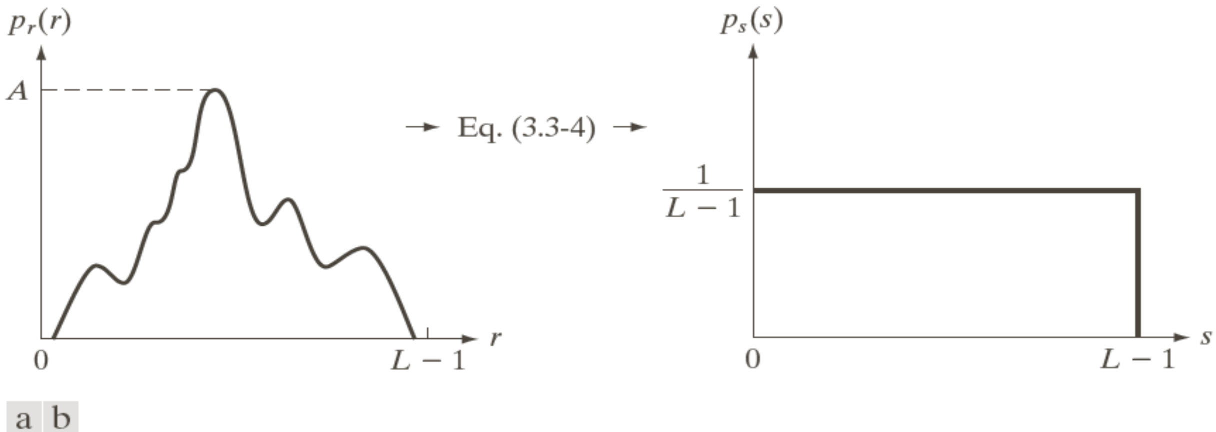


FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying the transformation in Eq. (3.3-4) to all intensity levels, r. The resulting intensities, s, have a uniform PDF, independently of the form of the PDF of the r's.

Histogram Equalization

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

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

- The discrete version of transformation

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

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

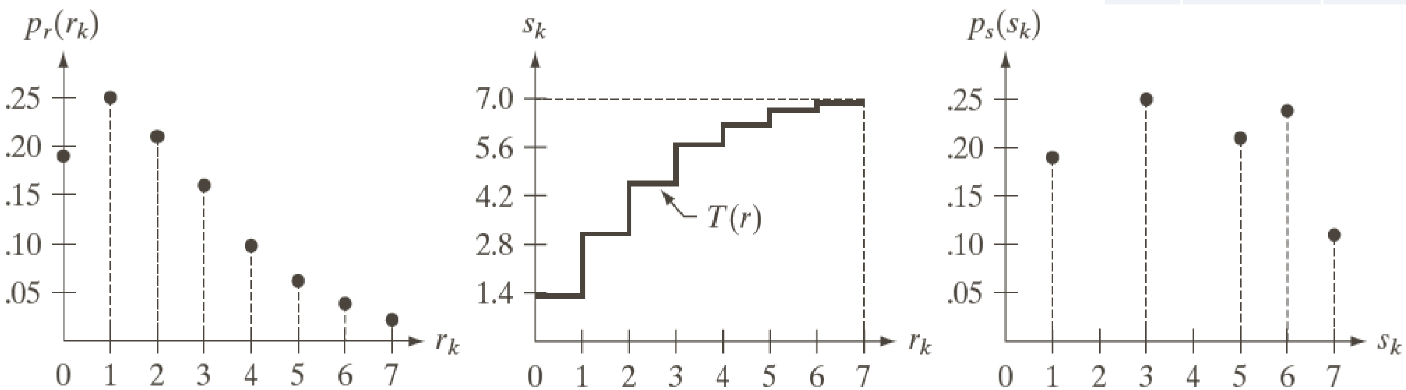
Thus, an output image is obtained by mapping each pixel with level r_k in the input image into a corresponding pixel with level s_k in the output image

Histogram Equalization

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_k	Value	Appr.	Ps(s)
s0	1.33	1	
s1	3.08	3	790/4096
s2	4.55	5	
s3	5.67	6	1023/4096
s4	6.23	6	
s5	6.65	7	850/4096
s6	6.86	7	(656+329)/4096
s7	7.00	7	(245+122+81)/4096



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

a b c

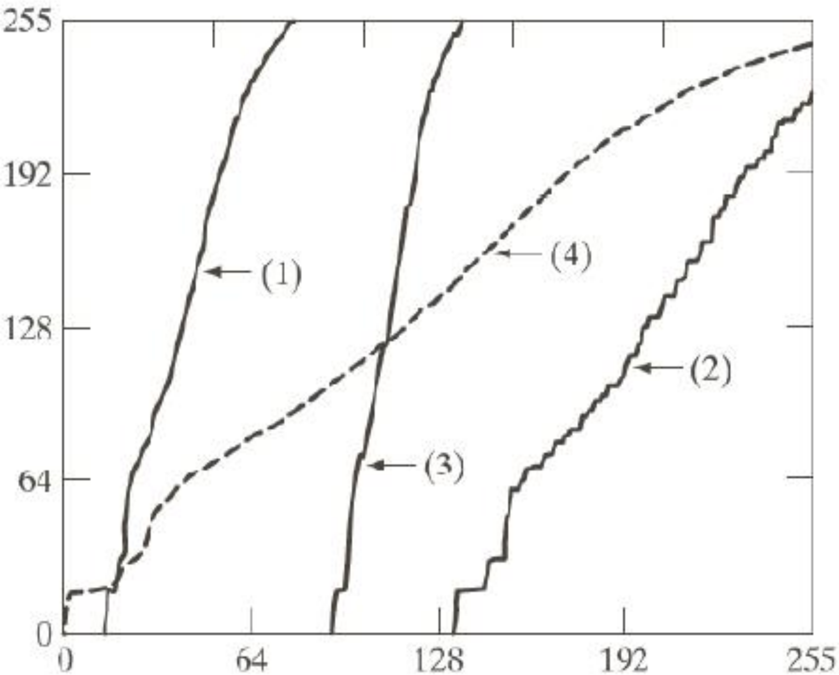
FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

Histogram Equalization

- 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

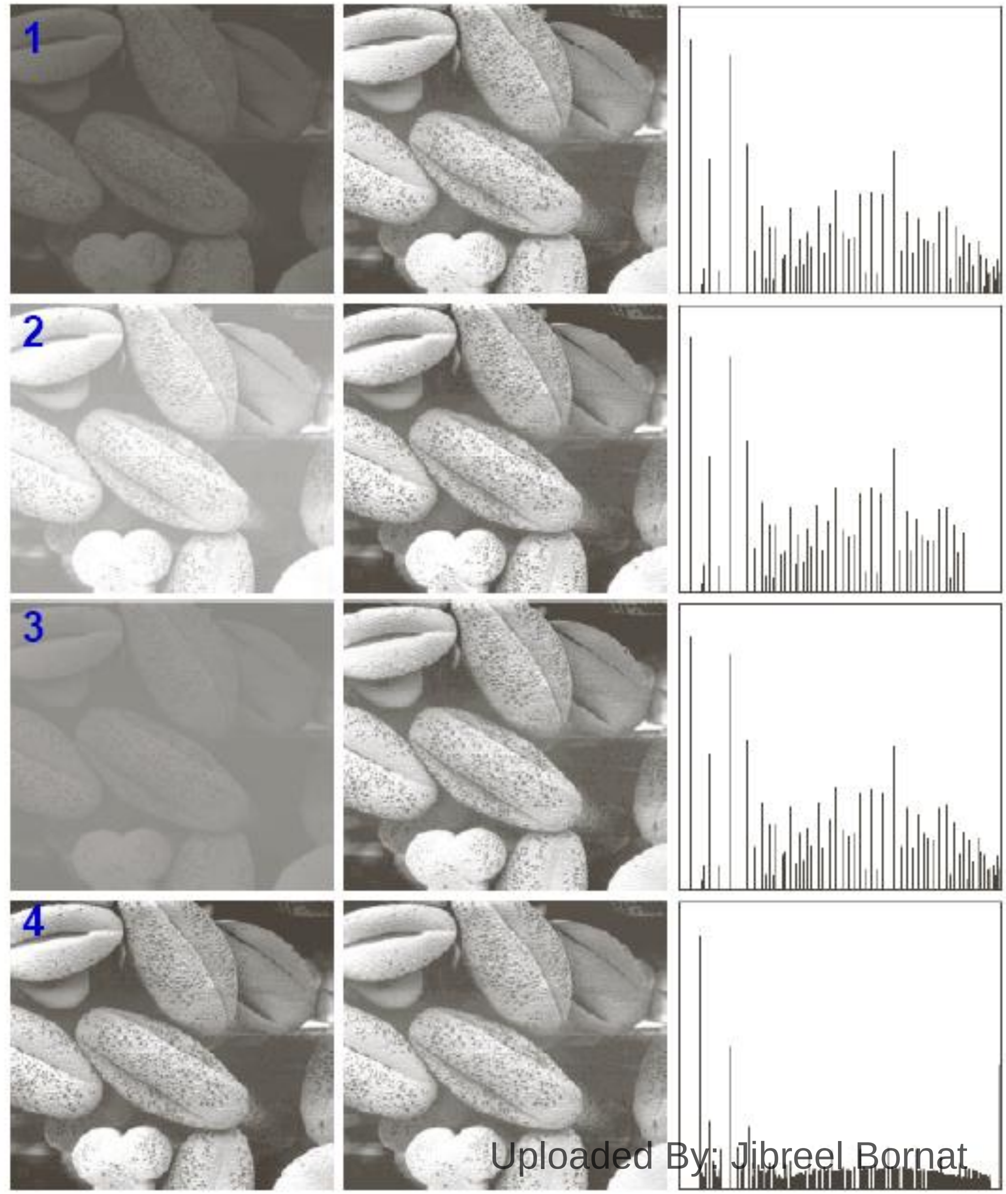
Histogram Equalization

- **Example**



Transformation for the processed images

Original | Equalized | Output Histogram



Histogram Specification

- Histogram equalization has a disadvantage which is that it can generate only one type of output image.

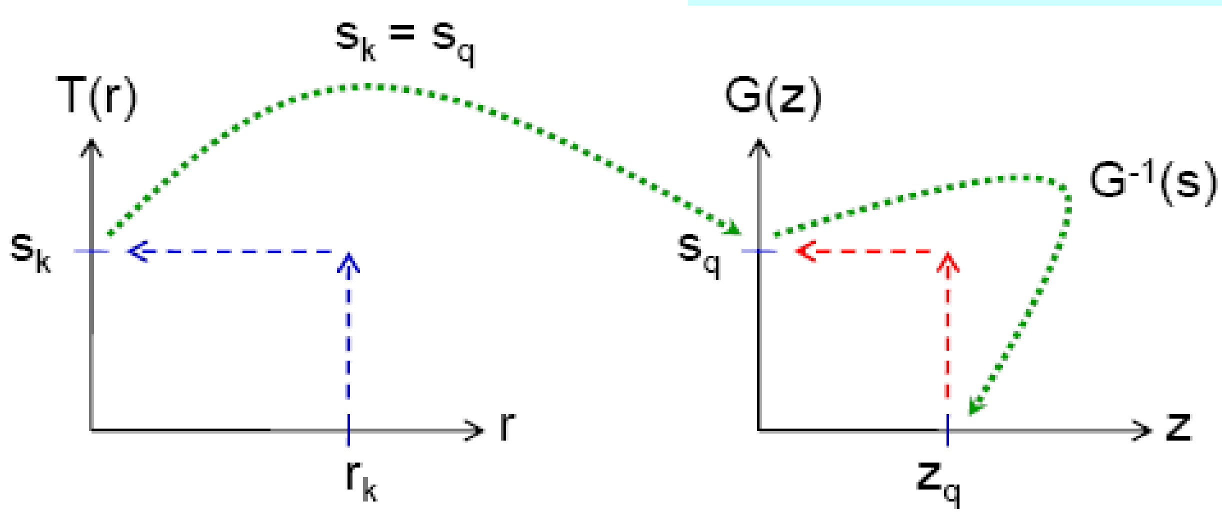
- With Histogram Specification, we can specify the shape of the histogram that we wish the output image to have.

- We can use the method used in deriving the transformation function of histogram equalization to find transformation function for the desired histogram, however
  - This requires the availability of $p_s(s)$ in mathematical form and the ability to express s in terms of r
  - It doesn't have to be a uniform histogram
  - Histogram specification is a trial-and-error process
  - There are no rules for specifying histograms, and one must resort to analysis on a case-by-case basis for any given enhancement task.

Histogram Specification

- Alternatively
  - Let $p_r(r)$ and $p_z(z)$ denote the original and desired histograms and assume that there exist two transformation functions $s = T(r) \quad and \quad s = G(z)$

    This implies that we can find the mapping from r to s by knowing the inverse of G(z) $z = G^{-1}(s) = G^{-1}(T(r))$



  - This is a simple operation if $G^{-1}(s)$ can be obtained !!!!

Histogram Specification

- **The algorithm**
  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

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

Histogram Specification

- **The algorithm**

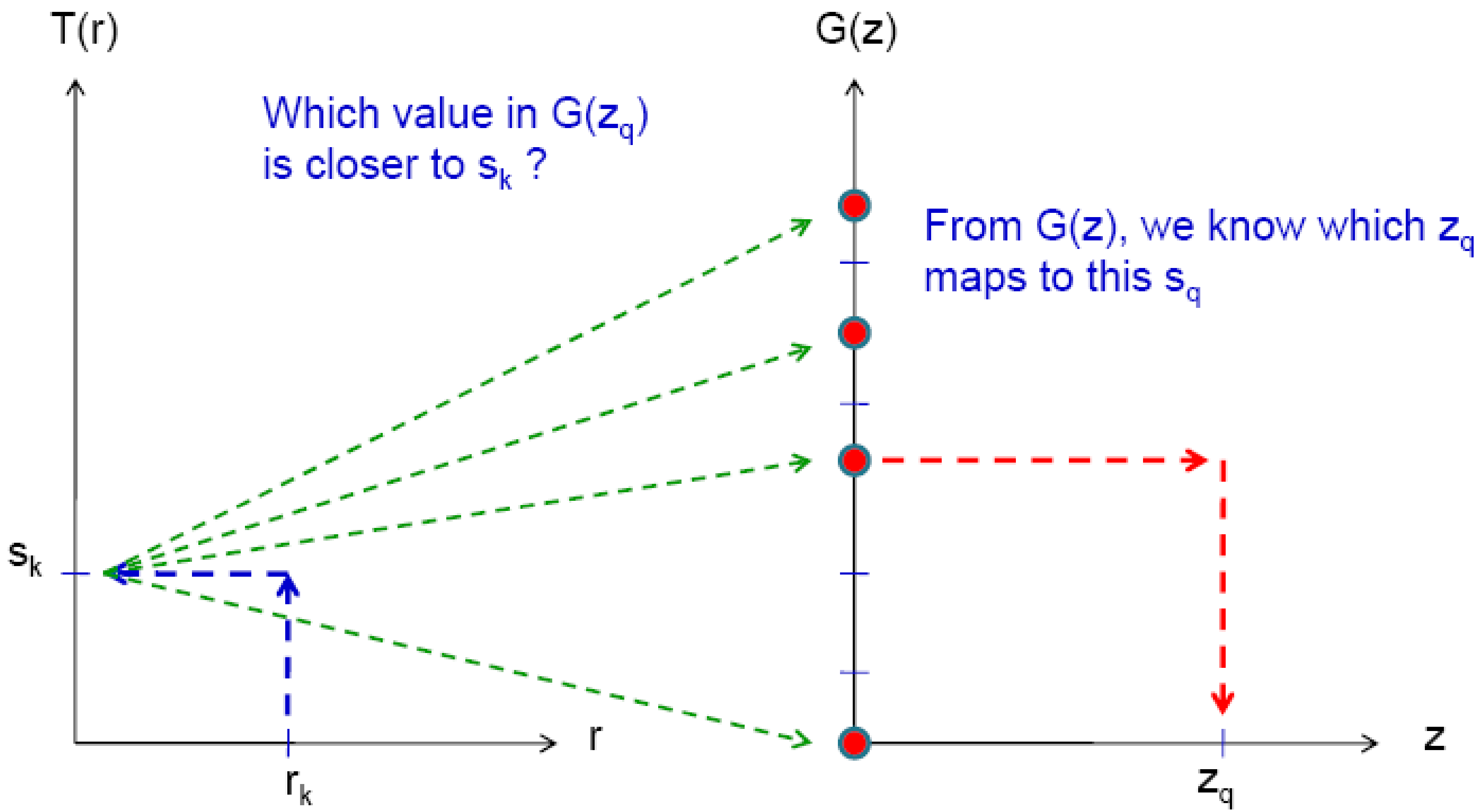
3. Obtain the inversed transformation function G^{-1} (mapping from s to z)

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

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

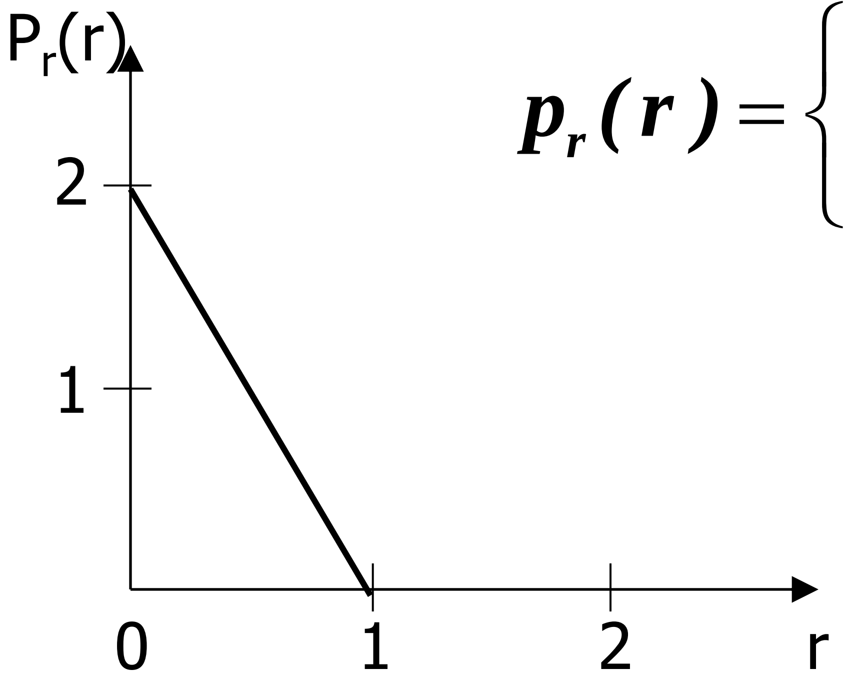
Histogram Specification

- Illustration



Example

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

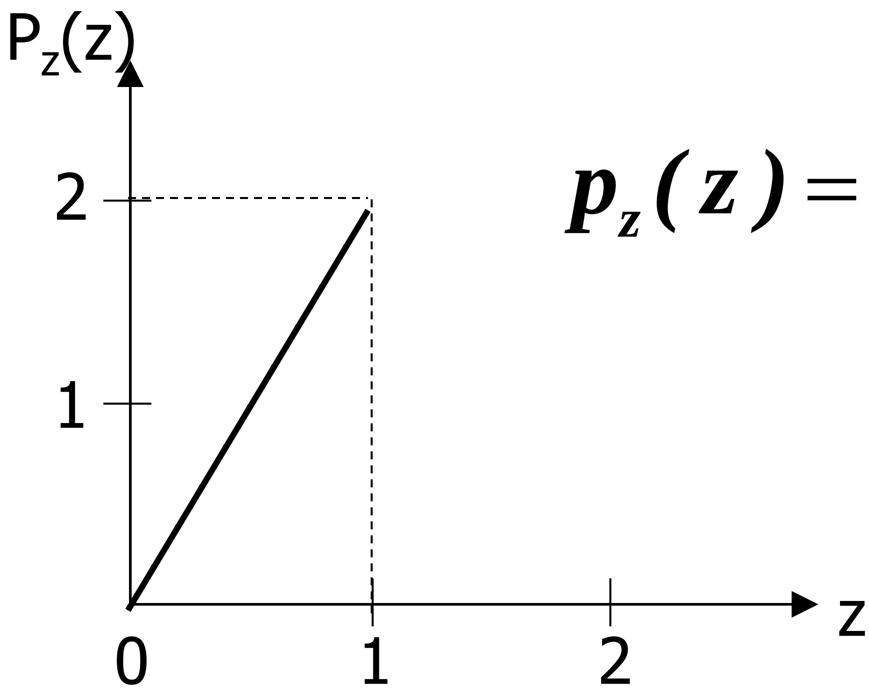


$$p_r(r) = \begin{cases} -2r + 2 & ; 0 \le r \le 1 \\ 0 & ; \text{elsewhere} \end{cases}$$

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

Example

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

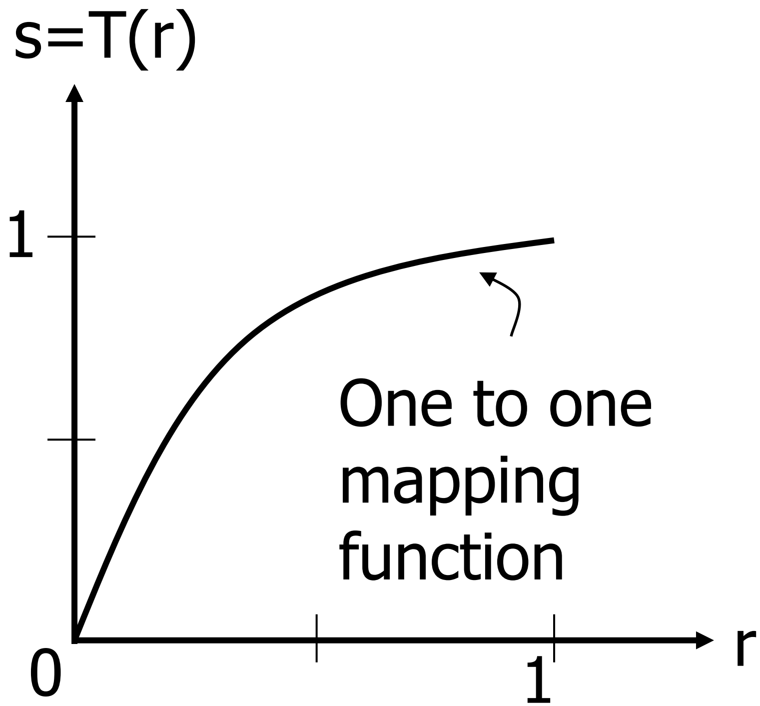


$$p_z(z) = \begin{cases} 2z & ; 0 \le z \le 1 \\ 0 & ; \text{elsewhere} \end{cases}$$

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

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

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

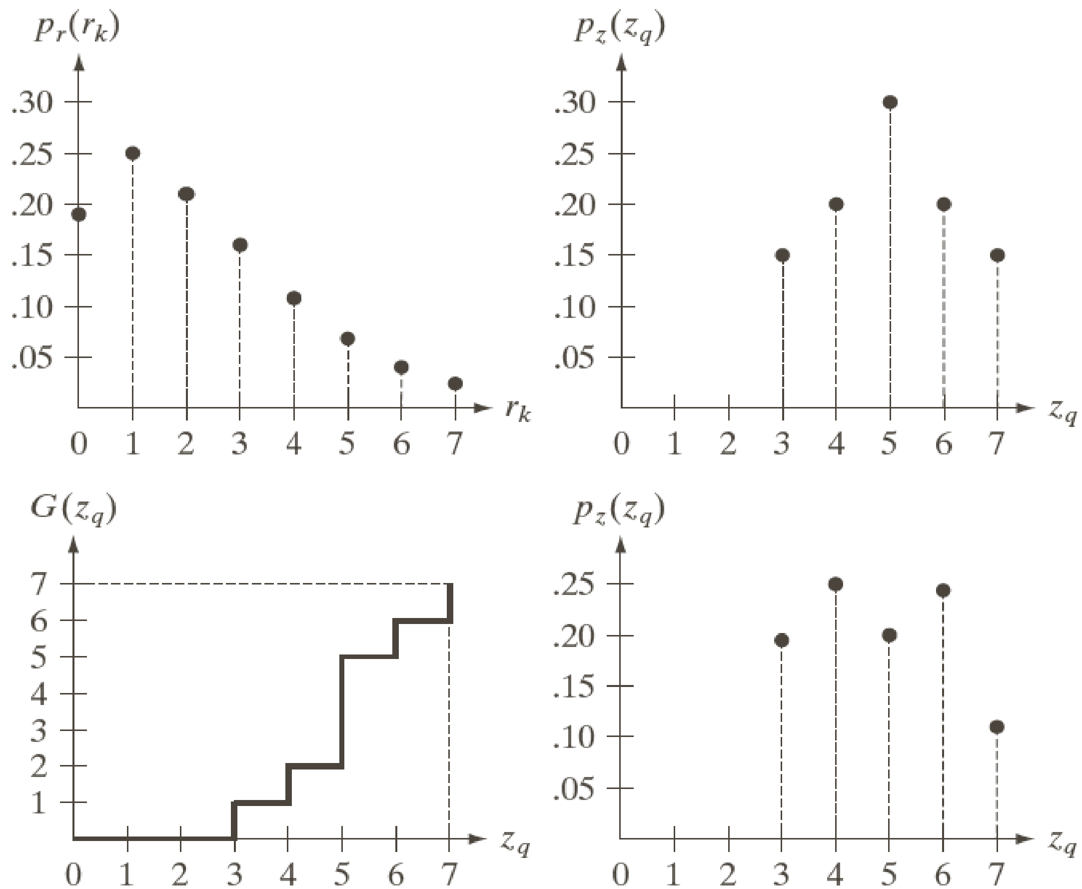
$$= \sum_{j=0}^{k} \frac{n_j}{n} \qquad k = 0,1,2,\ldots,L-1$$

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

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

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

Histogram Specification

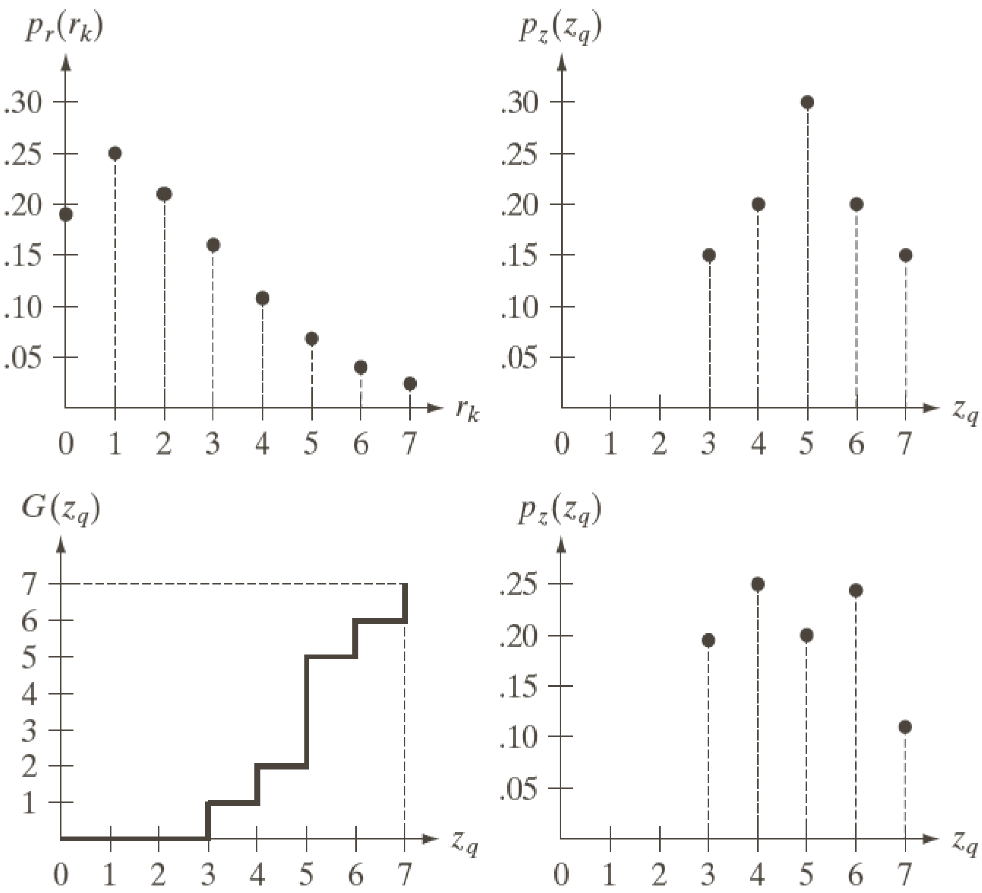


a	b
c	d

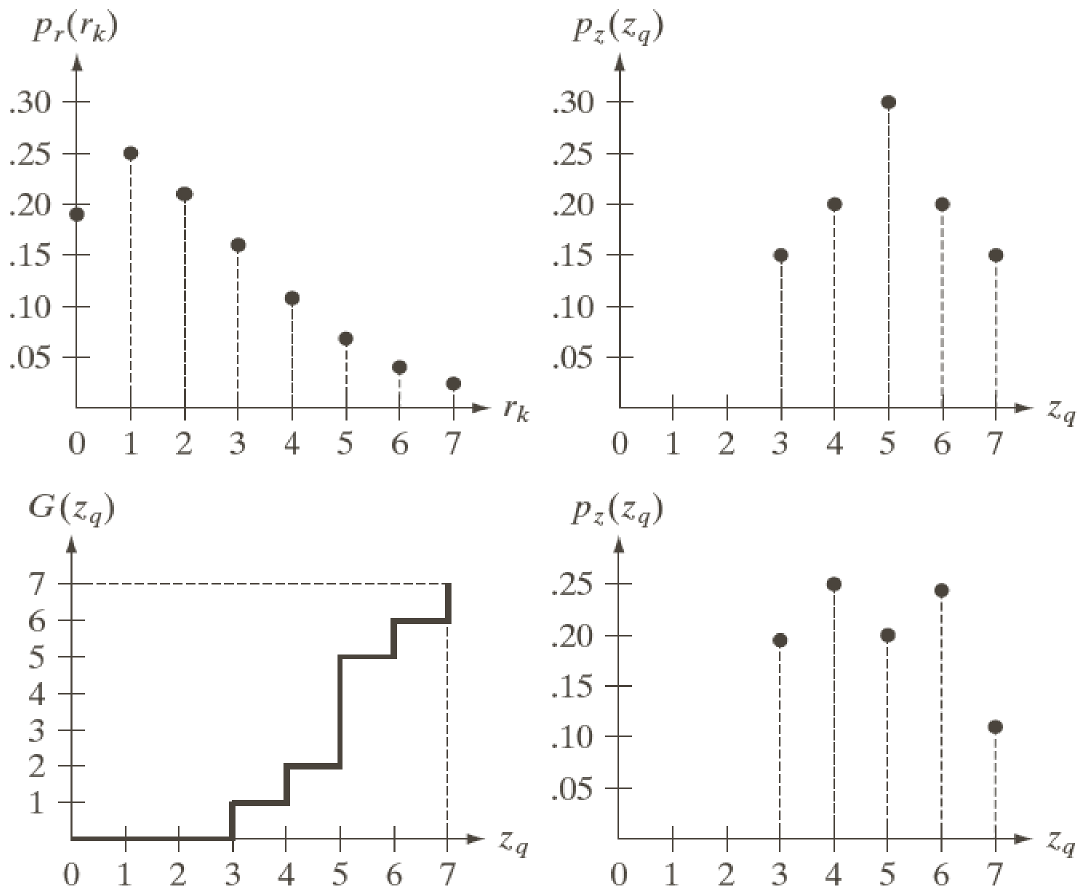
FIGURE 3.22 (a) Histogram of a 3-bit image. (b) Specified histogram. (c) Transformation function obtained from the specified histogram. (d) Result of performing histogram specification. Compare (b) and (d).

Histogram Specification

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

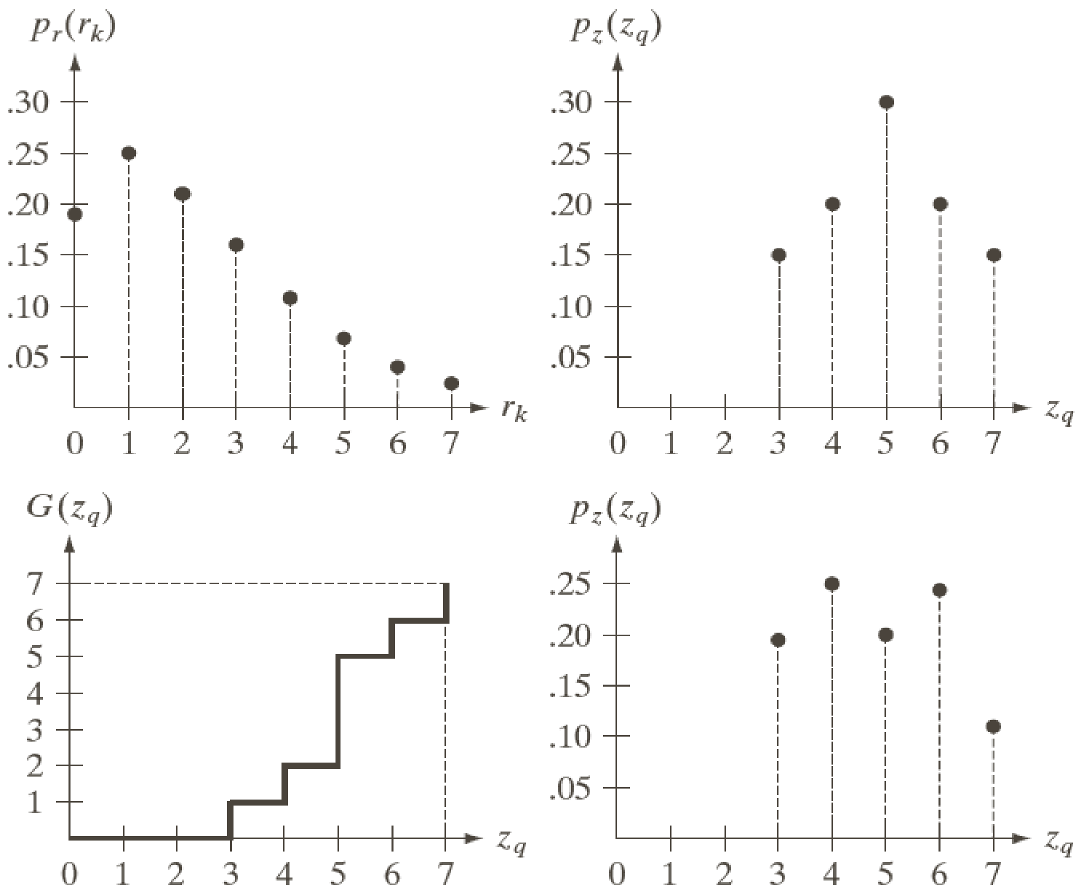


Histogram Specification



Zq	Specified Pz(Zq)
z0	0.00
z1	0.00
z2	0.00
z3	0.15
z4	0.20
z5	0.30
z6	0.20
z7	0.15

Histogram Specification



Z_k	Value	Appr.
z0	0.00	0
z1	0.00	0
z2	0.00	0
z3	1.05	1
z4	2.45	2
z5	4.55	5
z6	5.95	6
z7	7.00	7

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

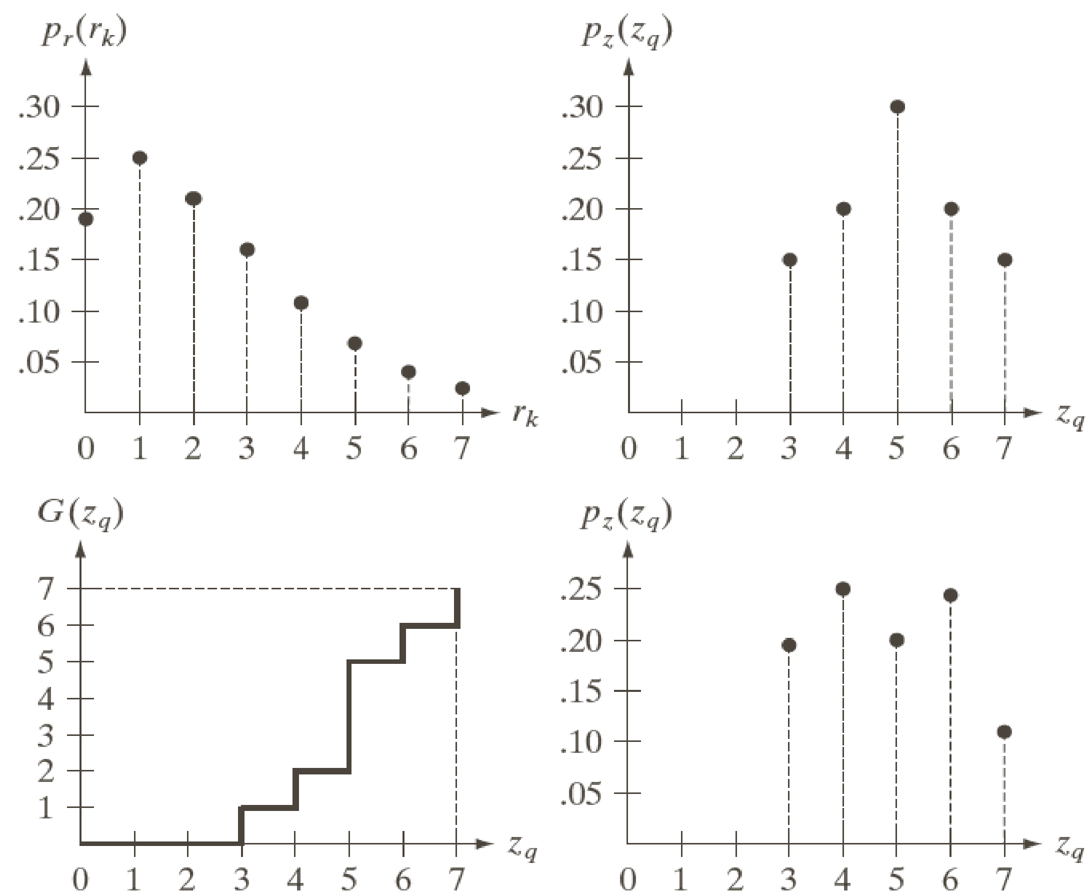
Histogram Specification

s_k	Appr.
s0	1
s1	3
s2	5
s3	6
s4	6
s5	7
s6	7
s7	7

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

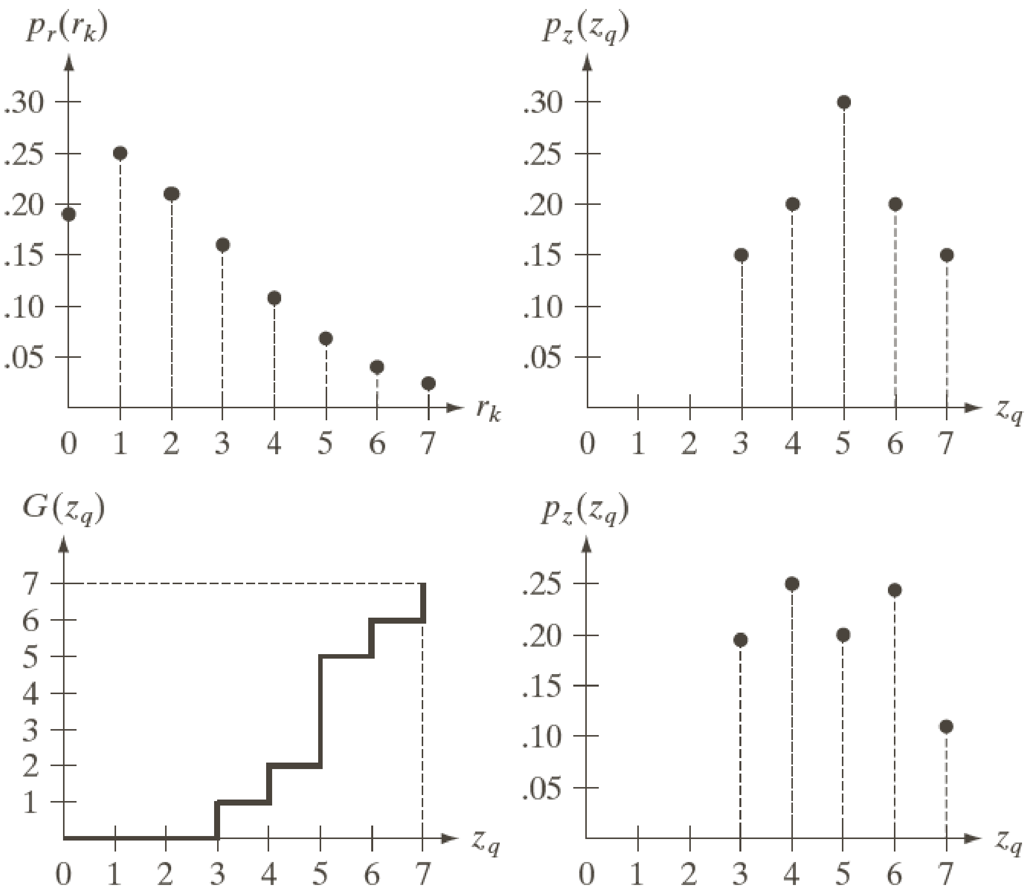
Z_k	Appr.
z0	0
z1	0
z2	0
z3	1
z4	2
z5	5
z6	6
z7	7

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



Find the smallest value of z_q so that the value $G(z_q)$ is closest to s_k

Histogram Specification



r_k	Nk
r0	790
r1	1023
r2	850
r3	656
r4	329
r5	245
r6	122
r7	81

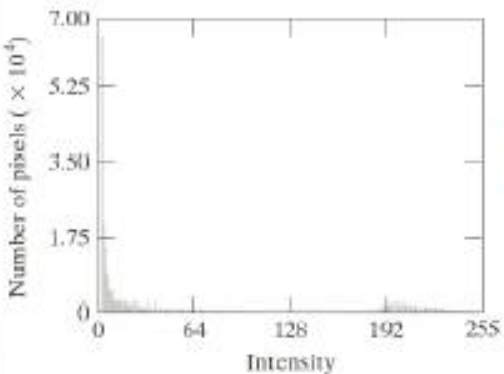
s_k	Appr.
s0	1
s1	3
s2	5
s3	6
s4	6
s5	7
s6	7
s7	7

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

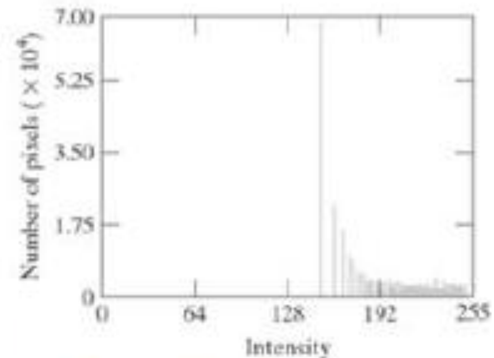
Zq	Specified Pz(Zq)	Actual Pz(Zq)
z0	0.00	0.00
z1	0.00	0.00
z2	0.00	0.00
z3	0.15	790/4096=0.19
z4	0.20	1023/4096=0.25
z5	0.30	850/4096=0.21
z6	0.20	(656+329)/4096=0.24
z7	0.15	(245+122+81)/4096=0.11

Histogram Specification

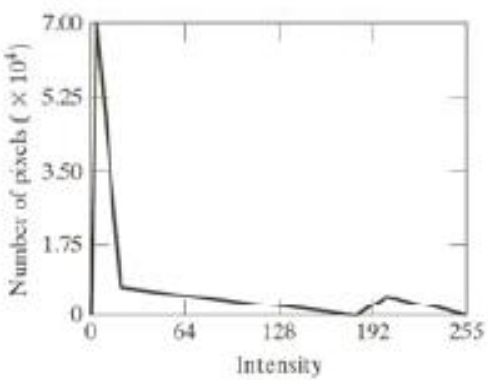
Original Image



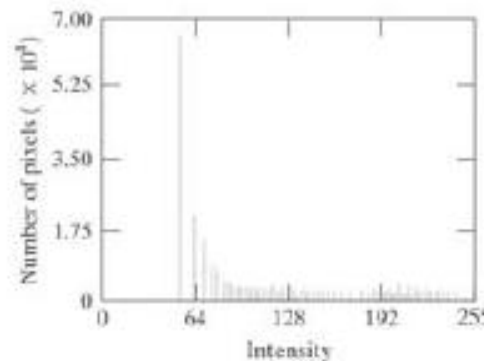
Histogram Equalization



Desired Histogram



Histogram Specification

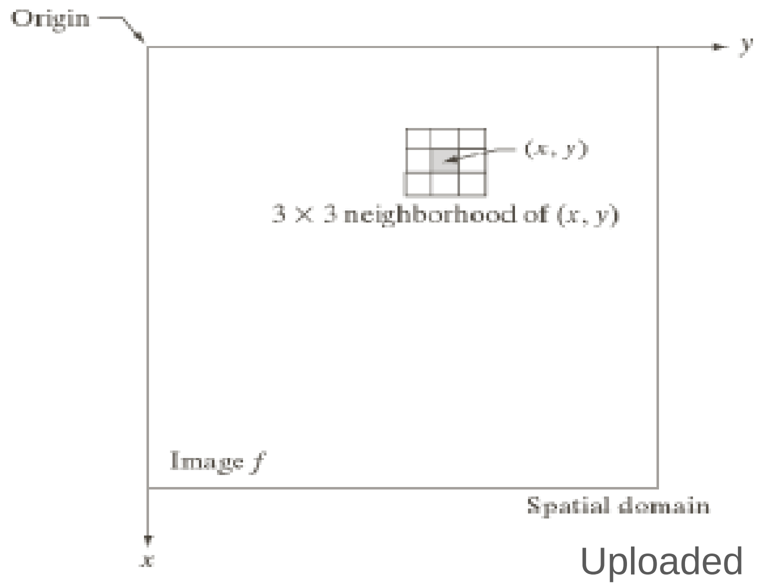


Local Histogram Processing

- Both histogram equalization and specification methods discussed earlier are considered global
- In other words, pixels are modified using a transformation function that is defined using all pixels in the image
- Such methods are suitable for overall enhancement and may not be suitable in situations where we want to enhance small areas in the image whose pixel count contributes less to the global transformation function

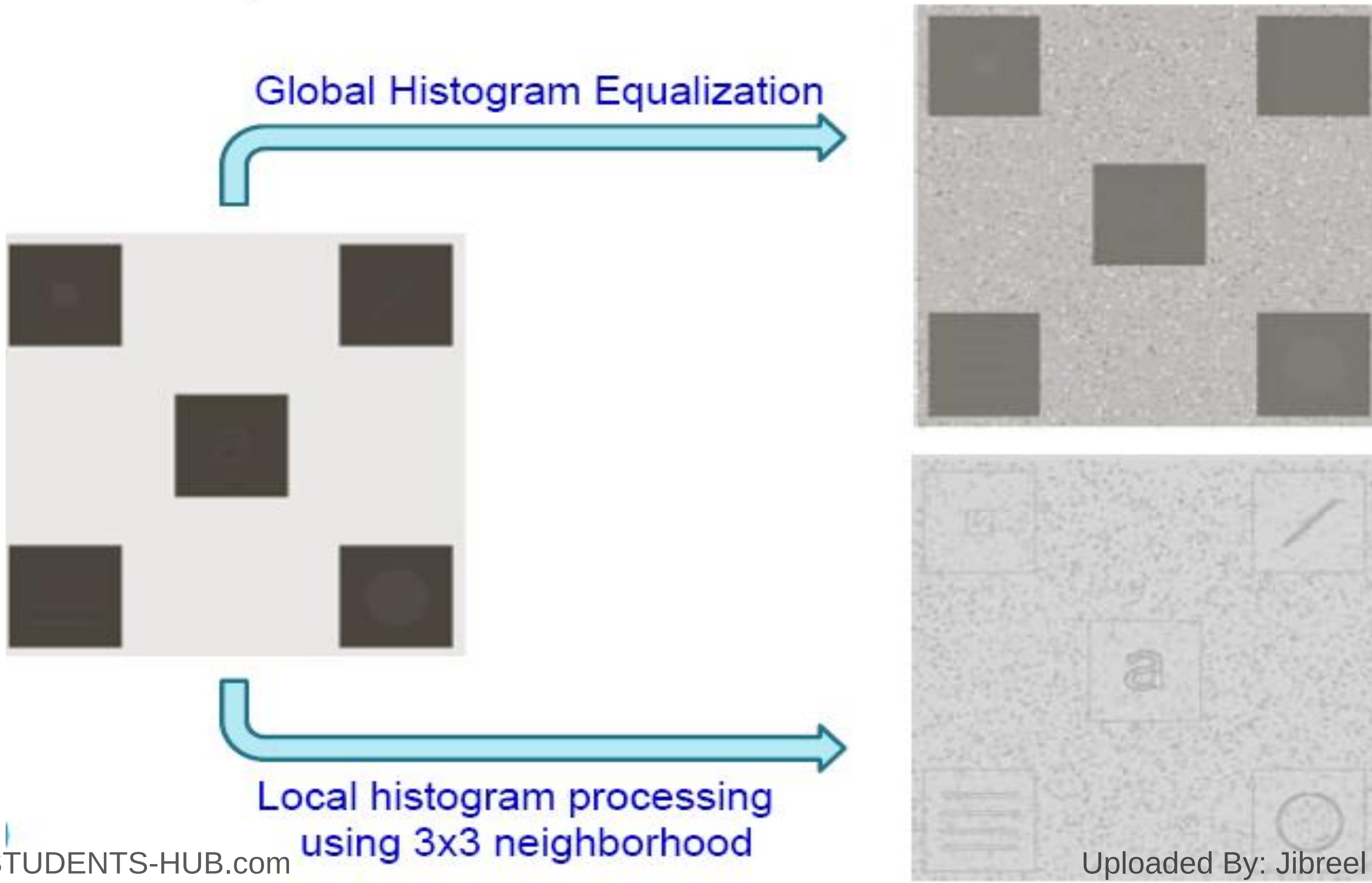
Histogram Specification

- **Solution:** Consider local processing such that for each pixel (x,y) in the image
  - Define a small neighborhood of size mxn that is centered around the pixel
  - Use pixels inside the neighborhood to construct the transformation function
  - Use the computed function map the pixel at (x,y)
  - Repeat for all pixels

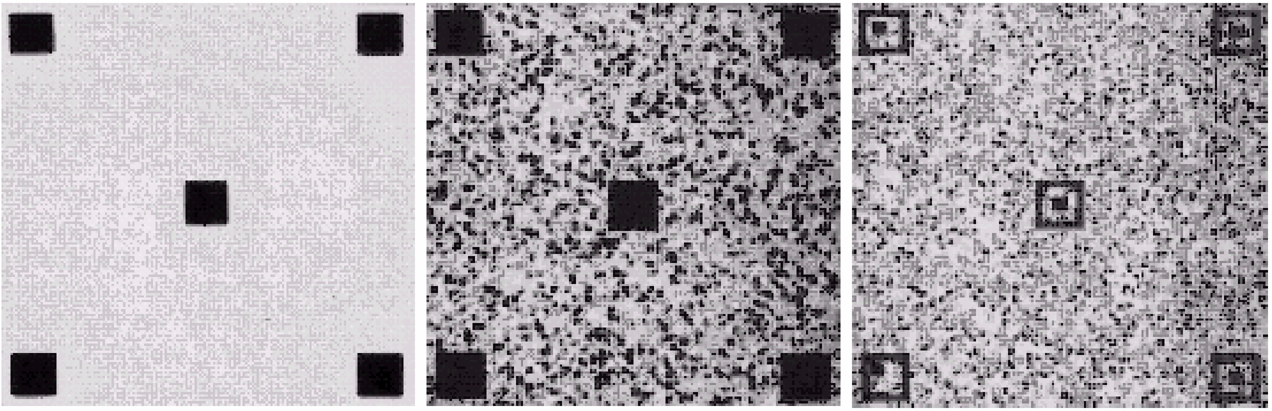


Local Histogram Processing

- **Example**



Local Enhancement



(a) (b) (c)

a) Original image (slightly blurred to reduce noise)
b) global histogram equalization (enhance noise & slightly increase contrast but the construction is not changed)
c) local histogram equalization using 7x7 neighborhood (reveals the small squares inside larger ones of the original image.

- define a square or rectangular neighborhood and move the center of this area from pixel to pixel.
- at each location, the histogram of the points in the neighborhood is computed and either histogram equalization or histogram specification transformation function is obtained.
- another approach used to reduce computation is to utilize nonoverlapping regions, but it usually produces an undesirable checkerboard effect.

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.