The First Portrait of a Black Hole & Beyond

Katie Bouman

EHT work presented is joint work with: Michael Johnson, Andrew Chael, Jose Gomez, Kazu Akiyama, Shep Doeleman, and the **entire Event Horizon Telescope Collaboration**

Deep Probabilistic Imaging is joint work with He Sun

Array Optimization is joint work with He Sun and Adrian Dalca

Image Interpolation

- **Interpolation** refers to estimating the values at unknown locations using known data
  - Interpolation is often needed for zooming, shrinking, rotating, etc.
- Nearest neighbor interpolation uses the closest neighboring value
- Bilinear interpolation uses 4 nearest neighbors to estimate the value through solving equations of form:

$$v(x, y) = ax + by + cxy + d$$

  - Note there are 4 unknowns (a, b, c, d), so 4 equations are sufficient
- Bicubic interpolation uses 16 neighbors to solve the system:

$$v(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j$$

Chapter 2: Digital Image Fundamentals

Image Zooming: NN vs Bilinear Interpolation

a	b	c
d	e	f

~~FIGURE 2.25~~ Top row: images zoomed from 128 × 128, 64 × 64, and 32 × 32 pixels to 1024 × 1024 pixels, using nearest neighbor gray-level interpolation. Bottom row: same sequence, but using bilinear interpolation.

Chapter 2: Digital Image Fundamentals
Image Interpolation

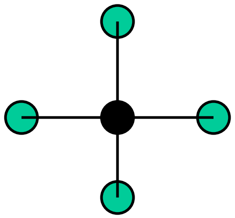
a b c

Bilinear interpolation Bicubic interpolation

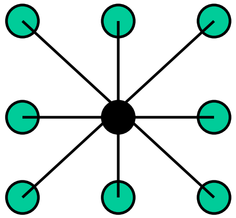
FIGURE 2.27 (a) Image reduced to 72 dpi and zoomed back to its original 930 dpi using nearest neighbor interpolation. This figure is the same as Fig. 2.23(d). (b) Image reduced to 72 dpi and zoomed using bilinear interpolation. (c) Same as (b) but using bicubic interpolation.

Image pixel neighborhoods

Neighbors of a pixel $p=(i,j)$



$N_4(p)=\{(i-1,j),(i+1,j),(i,j-1),(i,j+1)\}$



$N_8(p)=\{(i-1,j),(i+1,j),(i,j-1),(i,j+1),$
$(i-1,j-1),(i-1,j+1),(i+1,j-1),(i+1,j+1)\}$

Adjacency

4-adjacency: p,q are 4-adjacent if p is in the set $N_4(q)$

8-adjacency: p,q are 8-adjacent if p is in the set $N_8(q)$

Note that if p is in $N_{4/8}(q)$, then q must be also in $N_{4/8}(p)$

Common Distance Definitions

Euclidean distance
(2-norm)

$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
2	1	0	1	2
$\sqrt{5}$	$\sqrt{2}$	1	$\sqrt{2}$	$\sqrt{5}$
$2\sqrt{2}$	$\sqrt{5}$	2	$\sqrt{5}$	$2\sqrt{2}$

D_4 distance
(city-block distance)

4	3	2	3	4
3	2	1	2	3
2	1	0	1	2
3	2	1	2	3
4	3	2	3	4

D_8 distance
(checkboard distance)

2	2	2	2	2
2	1	1	1	2
2	1	0	1	2
2	1	1	1	2
2	2	2	2	2

Gigapixel (e.g. 365 GigaPixel)

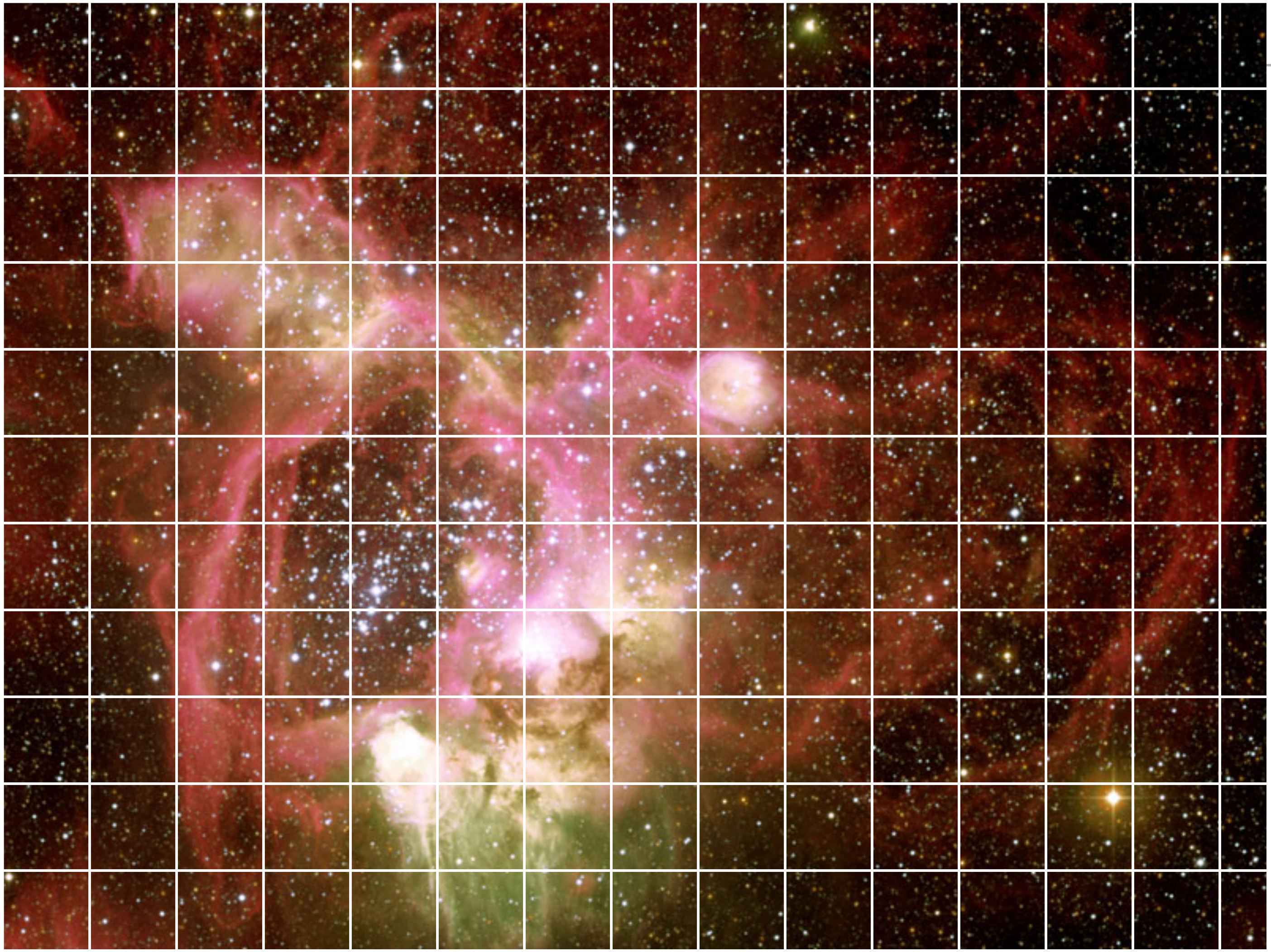
Mega-pel → Giga-pel

Photographers and artists have manually or semi-automatically stitched hundreds of mega-pixel pictures together to demonstrate how a giga-pel picture looks like → **the power of pixels**

http://triton.tpd.tno.nl/gigazoom/Delft2.htm

London: http://btlondon2012.co.uk/pano.html

Block-based Processing



Chapter 3: Intensity Transformations and Spatial Filtering

WHICH ONE LOOKS BETTER?

It makes all the difference whether one sees darkness through the light or brightness through the shadows.

David Lindsay

Chapter 3: Intensity Transformations and Spatial Filtering

- Basic Intensity Transformation Functions
- Histogram Processing
- Spatial Filtering
- Smoothing Spatial Filters
- Sharpening Spatial Filters
- Other Filters
- Combining Spatial Enhancement Methods

Chapter 3: Intensity Transformations and Spatial Filtering

Another example: which one is better for the pilot?

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.)

- Goal: Image enhancement seeks
  - to improve the visual appearance of an image, or
  - convert it to a form suited for analysis by a human or a machine.
- Image enhancement does not, however,
  - seek to restore the image, nor
  - increase its information contents
- Peculiarity:
  - actually, there is some evidence which suggests that a distorted image can be more pleasing than a perfect image!

Major Problem in Image Enhancement:

– the lack of a general standard of **image quality** makes it very difficult to evaluate the performance of different IE schemes.

Thus, Image Enhancement algorithms are mostly application-dependent, subjective and often ad-hoc.

Therefore, mostly subjective criteria are used in evaluating image enhancement algorithms.

Subjective Criteria Used in Image Enhancement

***A. Goodness scale*: how good an image is**

Overall goodness scale	**Group goodness scale**
Excellent (5)	Best (7)
Good (4)	Well above average (6)
Fair (3)	Slightly above average (5)
Poor (2)	Average (4)
Unsatisfactory (1)	Slightly below average (3)
	Well below average (2)
	Worst (1)

the numbers in parenthesis indicate a numerical weight attached to the rating.

***B. Impairment scale*: how bad the degradation is in an image**

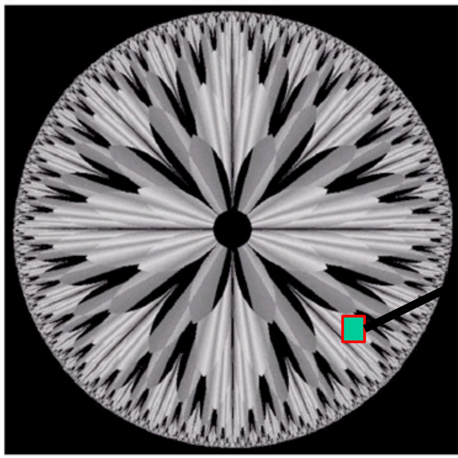
- not noticeable (1)
- just noticeable (2)
- definitely noticeable but only slight impairment (3)
- impairment acceptable (4)
- somewhat objectionable (5)
- definitely objectionable (6)
- extremely objectionable (7)

Bit representation

Eight-bit ones' complement

Binary value	Ones' complement interpretation	Unsigned interpretation
00000000	+0	0
00000001	1	1
⋮	⋮	⋮
01111101	125	125
01111110	126	126
01111111	127	127
10000000	−127	128
10000001	−126	129
10000010	−125	130
⋮	⋮	⋮
11111101	−2	253
11111110	−1	254
11111111	−0	255

Bit plane slicing



42 → | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |

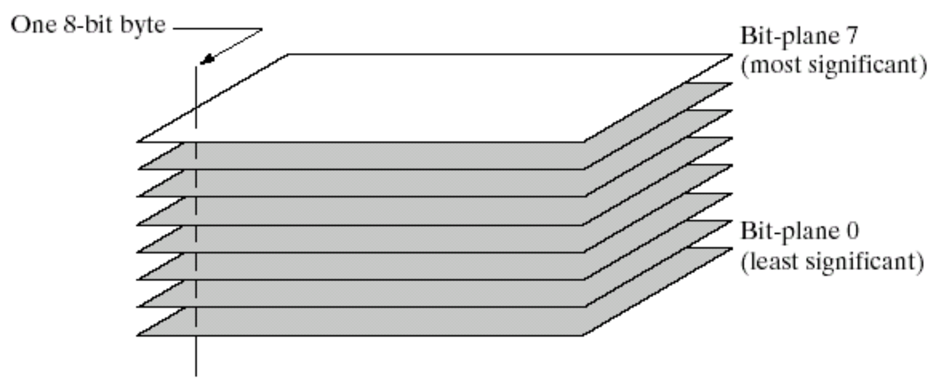


FIGURE 3.12 Bit-plane representation of an 8-bit image.

Bit plane slicing

Thresholded image at intensity value 128

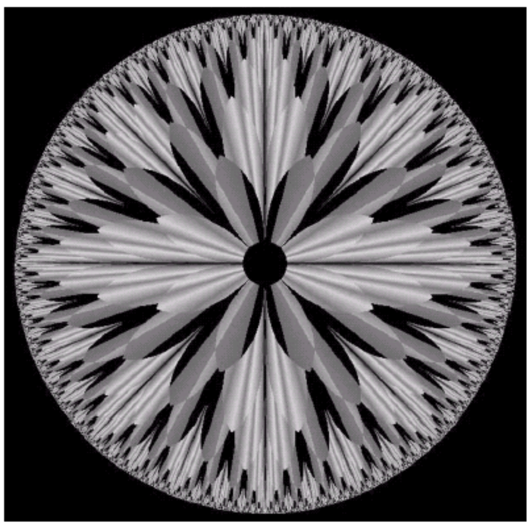


FIGURE 3.13 An 8-bit fractal image. (A fractal is an image generated from mathematical expressions). (Courtesy of Ms. Melissa D. Binde, Swarthmore College, Swarthmore, PA.)

Similar to noise

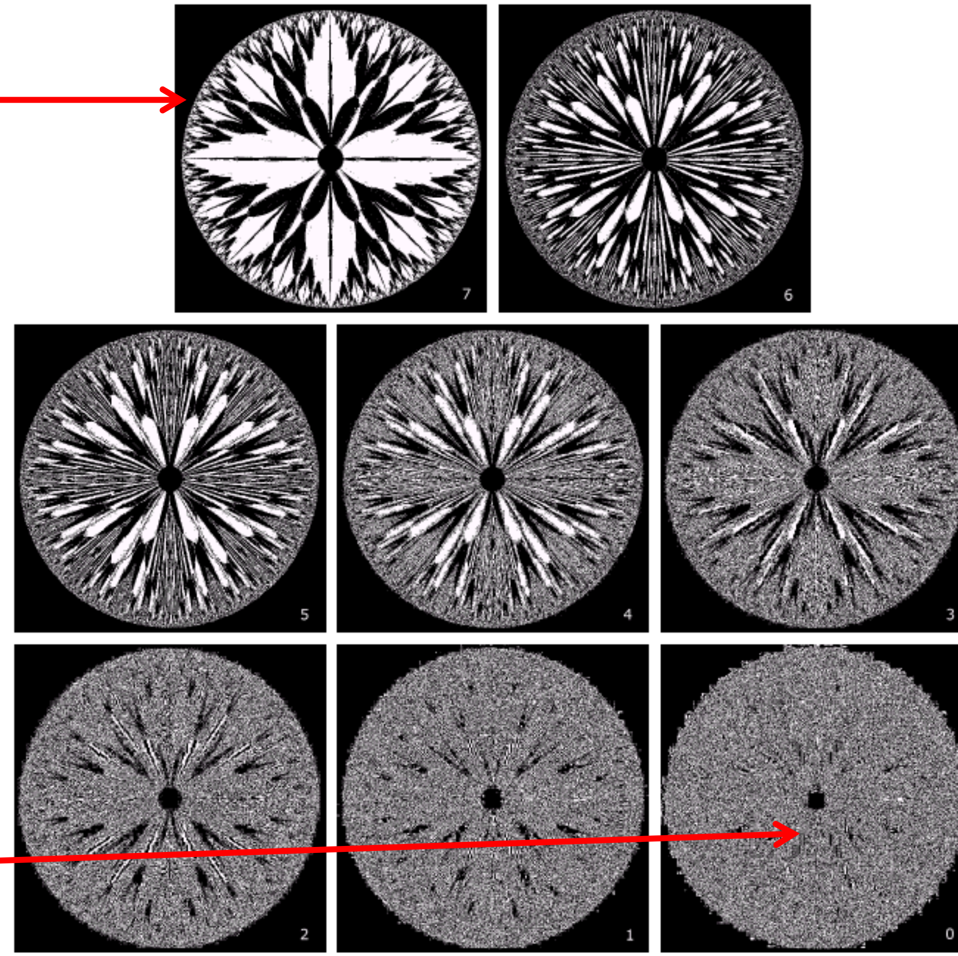
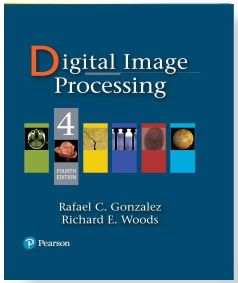
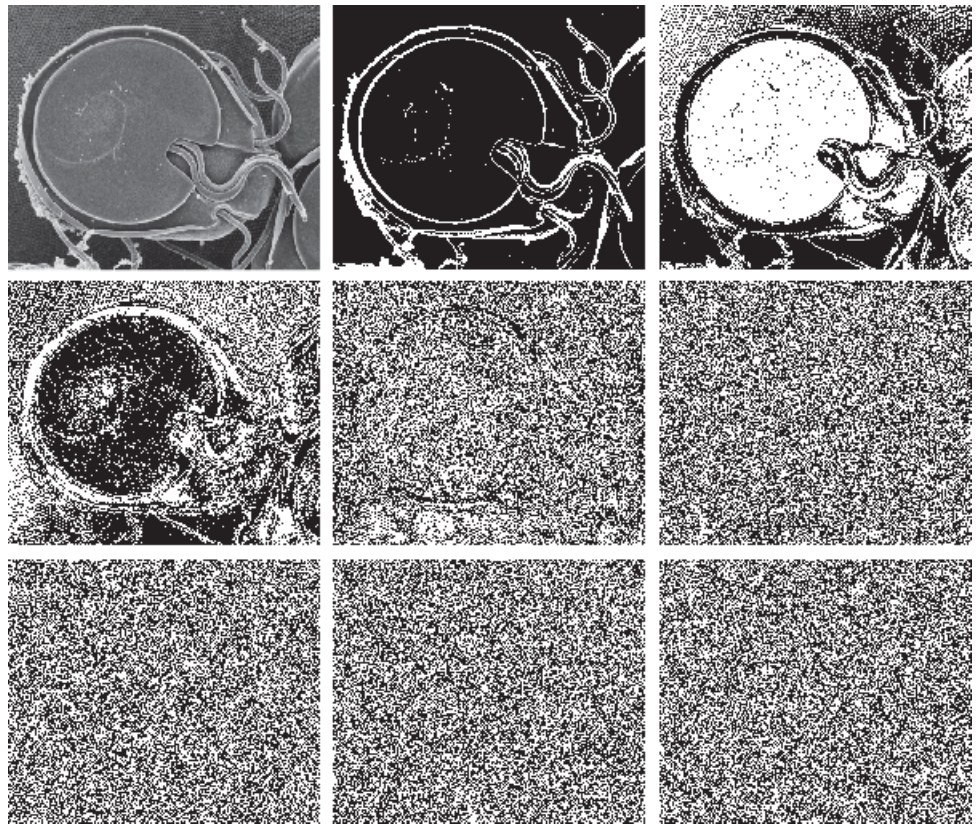


FIGURE 3.14 The eight bit planes of the image in Fig. 3.13. The number at the bottom, right of each image identifies the bit plane.

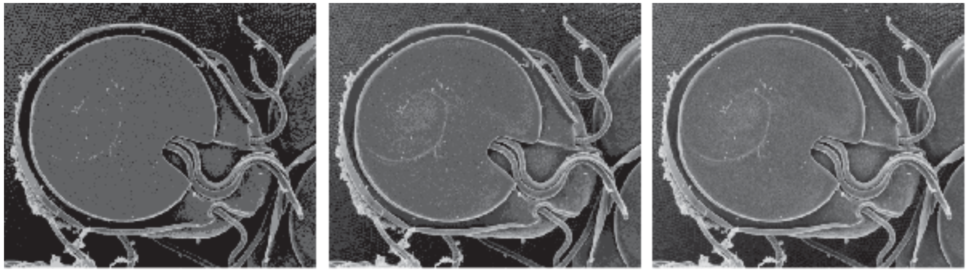


a b c
d e f
g h i

FIGURE 3.14
(a) An 8-bit gray-scale image of size 837 × 988 pixels. (b) through (i) Bit planes 8 through 1, respectively, where plane 1 contains the least significant bit. Each bit plane is a binary image. Figure (a) is an SEM image of a trophozoite that causes a disease called *giardiasis*. (Courtesy of Dr. Stan Erlandsen, U.S. Center for Disease Control and Prevention.)



A Scanning Electron Microscope of a trophozoite (causes giardiasis disease)



a b c

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

Bit plane slicing

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

a b c

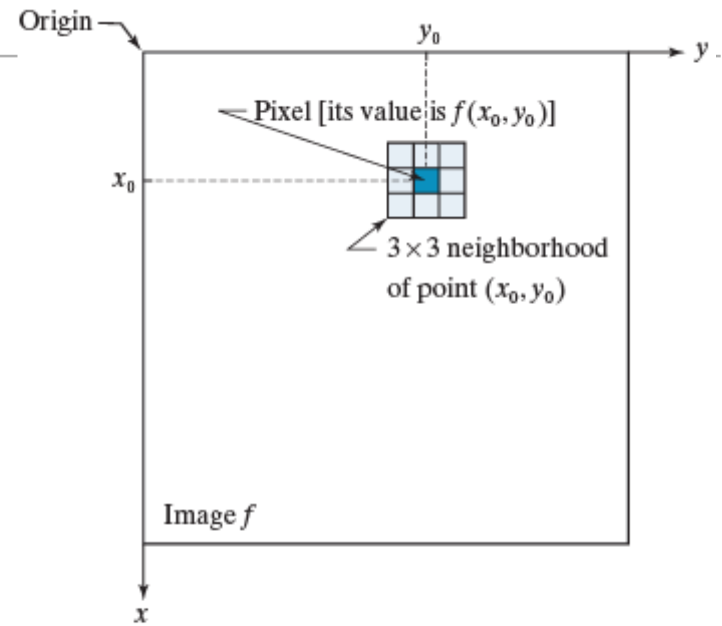
~~FIGURE 3.15~~ Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

Spatial Domain Operation:

$$g(x,y) = T[f_{N(x,y)}(x,y)]$$

$f_{N(x,y)}(x,y)$ is the original image $f(x,y)$ defined over the neighborhood $N(x,y)$ of (x,y); $g(x,y)$ is the output image and $T[.]$ is an operator on $f(x,y)$ defined over a neighborhood $N(x,y)$ of (x,y).

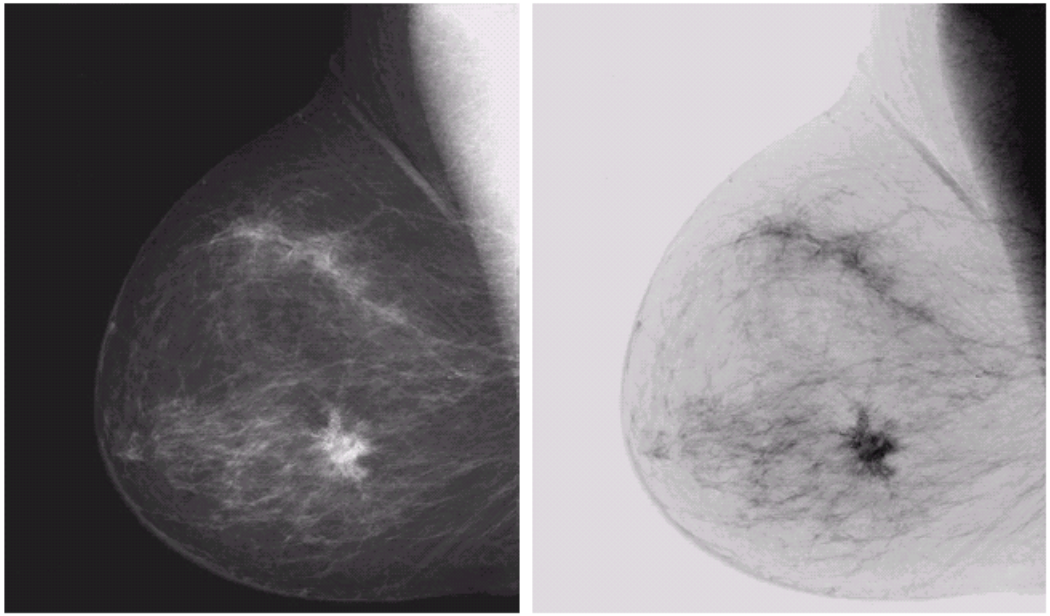
FIGURE 3.1 A 3×3 neighborhood about a point (x_0, y_0) in an image. The neighborhood is moved from pixel to pixel in the image to generate an output image. Recall from Chapter 2 that the value of a pixel at location (x_0, y_0) is $f(x_0, y_0)$, the value of the image at that location.



Special case: when the neighborhood N(x,y) contains 1 pixel, then $T[.]$ is called intensity transformation function.

Chapter 3: Intensity Transformations and Spatial Filtering

Image Negative



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.)

$$g(x,y) = T[f_{N(x,y)}(x,y)] = L - 1 - f(x,y);\ N(x,y) = (x,y)$$

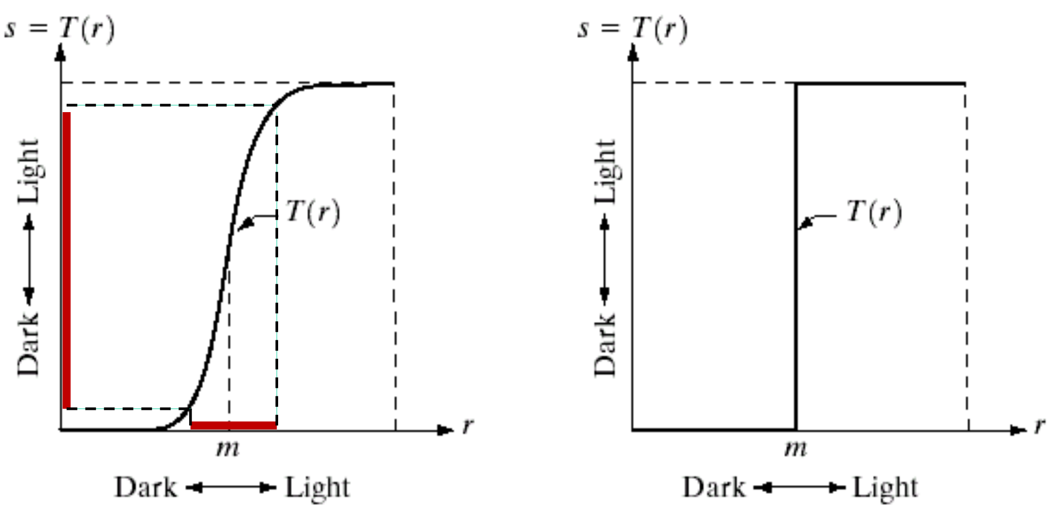
f(x,y) is the input grey level at pixel (x,y),
g(x,y) is the output grey level at (x,y), and
L-1 is the maximum value of g(x,y).

e.g. for L = 256
if f = 5 ➔ g = 250
if f = 250 ➔ g = 5

Lack of contrast

Operation: Contrast Stretching

Intensity Transformation: s = T(r)



a b

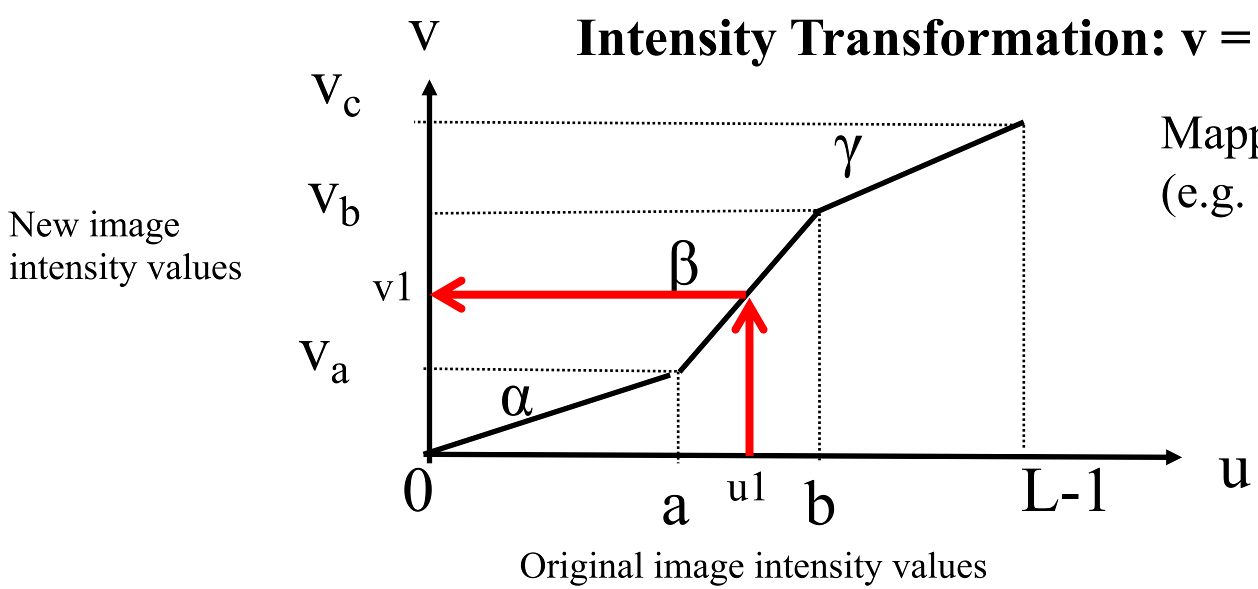
FIGURE 3.2 Gray-level transformation functions for contrast enhancement.

A special case of contrast stretching is illustrated above (bi-level output) and is called ***thresholding***.

Operation: Contrast Stretching

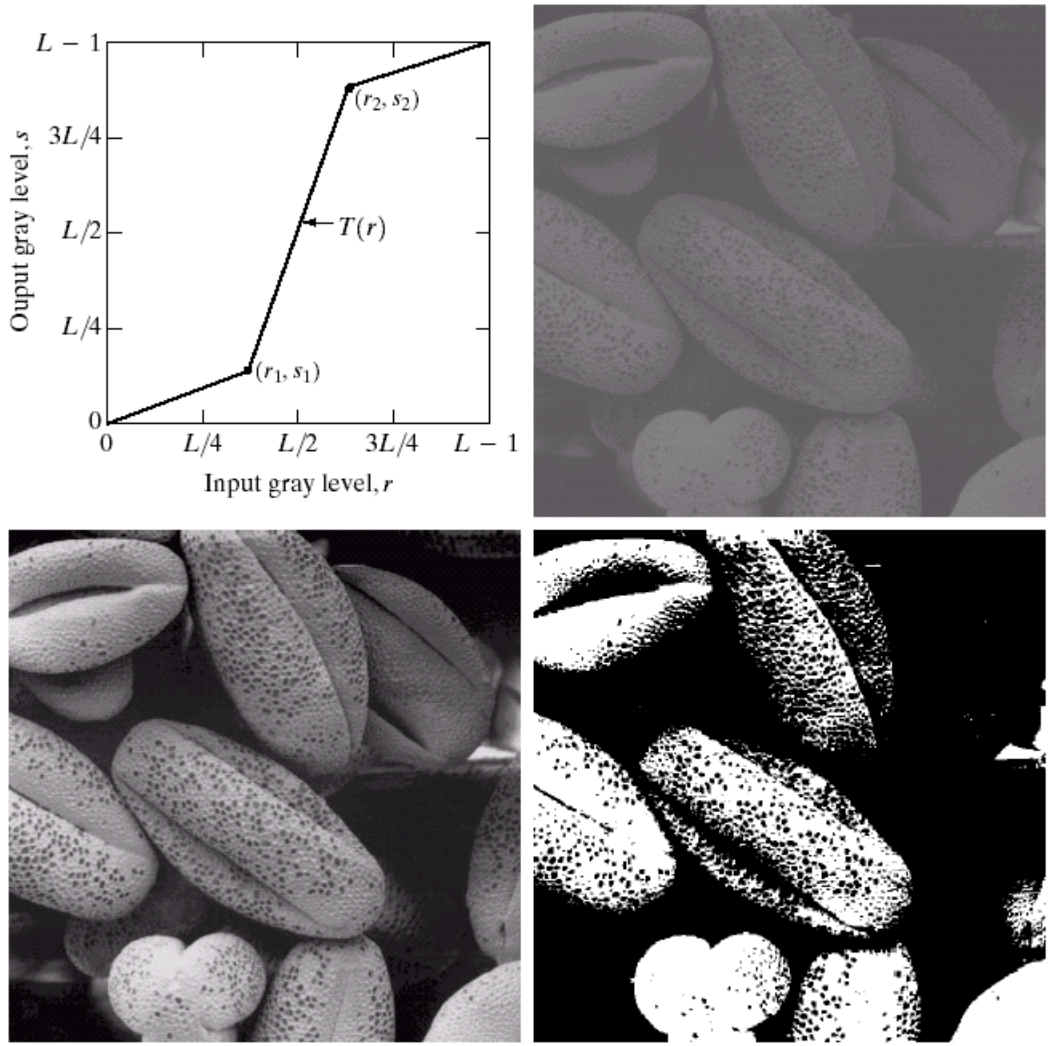
Poor contrast is the most common defect in images and is caused by reduced and/or nonlinear amplitude range or poor lighting conditions.

A typical contrast stretching transformation is shown below (examples are given later):



Mapping function from u to v (e.g. u1 → v1)

Contrast Stretching: Piecewise-Linear Transformation



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.)

A scanning electron microscope image of pollen magnified 700 times.

Thresholded image

Contrast Stretching: Piecewise-Linear Transformation

A special case of piecewise-linear transformation: Intensity-level slicing

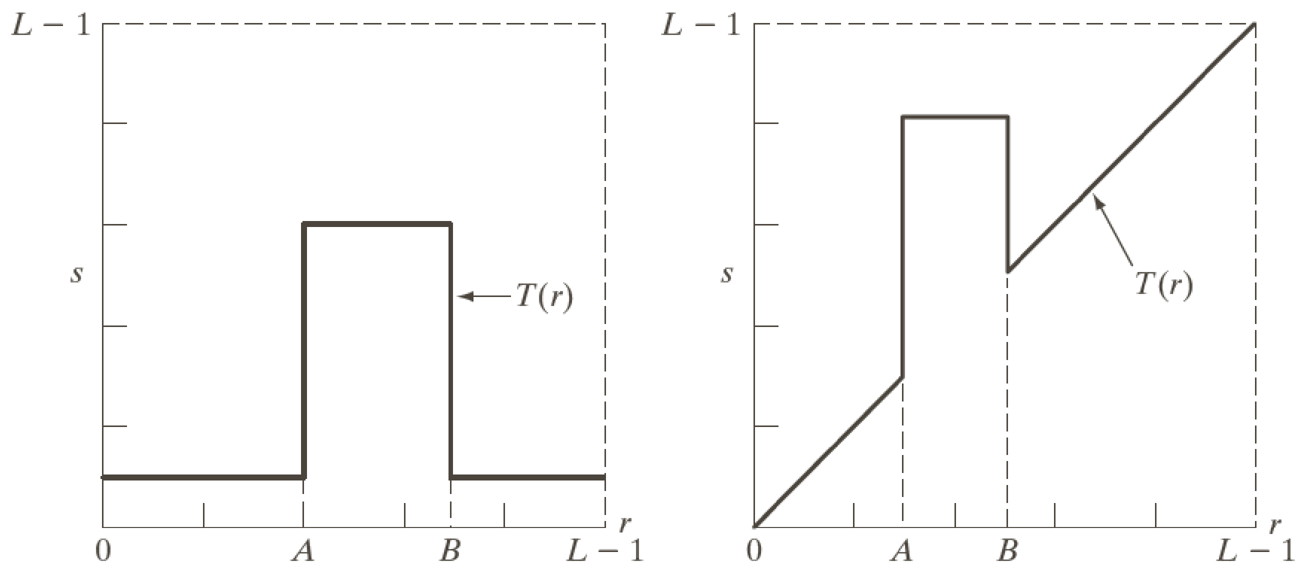


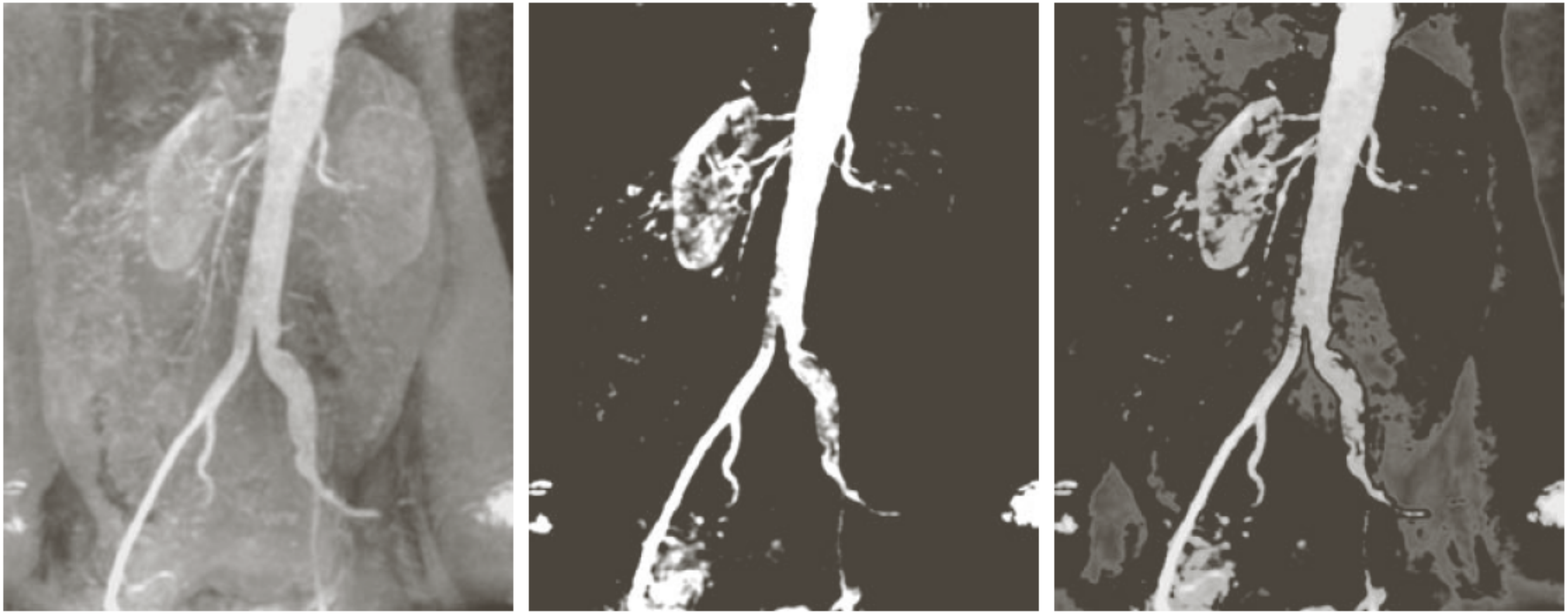
FIGURE 3.11 (a) This transformation highlights intensity range [A, B] and reduces all other intensities to a lower level. (b) This transformation highlights range [A, B] and preserves all other intensity levels.

Purpose: enhancing certain features in the image, e.g.

- masses of water in a satellite image
- flaws in an X-ray image

Chapter 3: Intensity Transformations and Spatial Filtering

Contrast Stretching: Piecewise-Linear Transformations



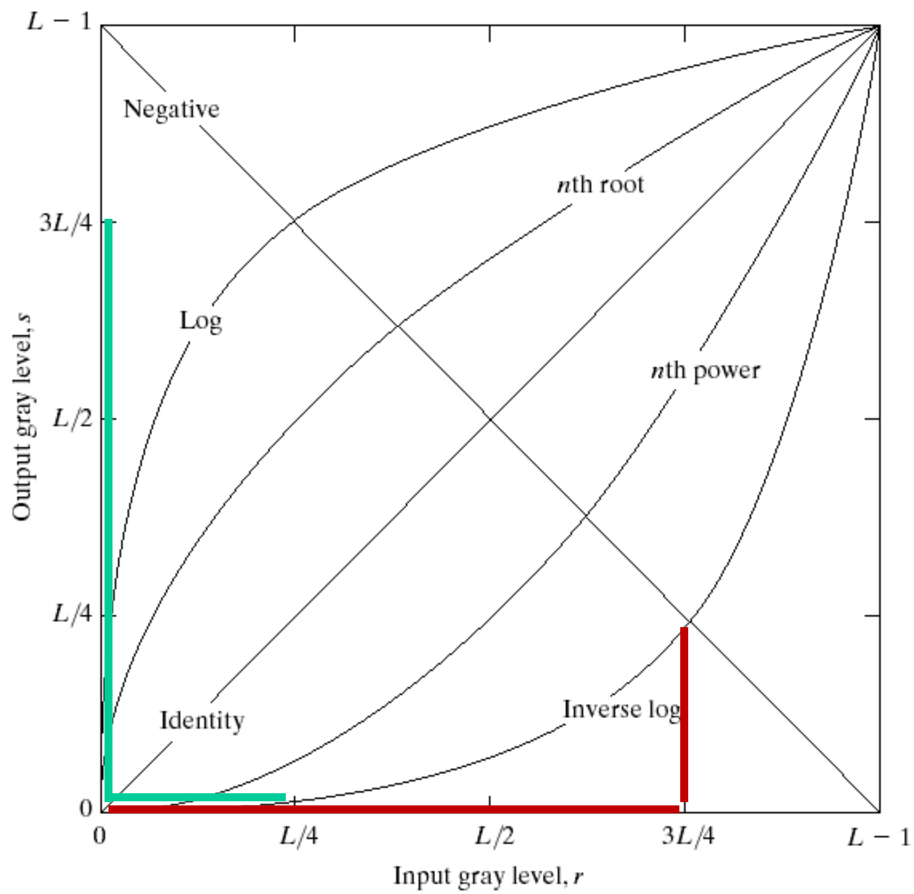
a b c

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.)

Chapter 3: Intensity Transformations and Spatial Filtering

Contrast Stretching: Basic Grey Level Transformations

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



$s = T[r]$

Transform function T[.]

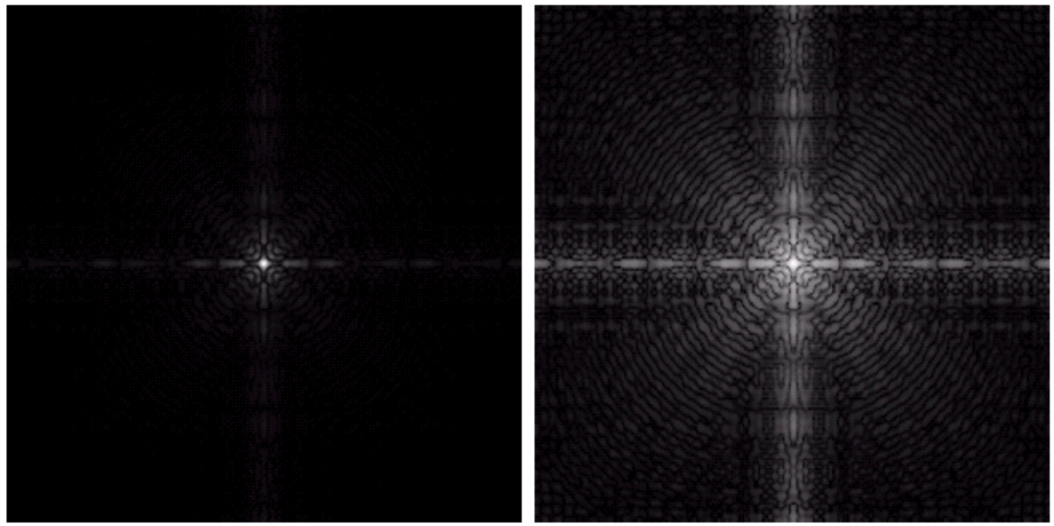
Global image transformation

Chapter 3: Intensity Transformations and Spatial Filtering

Basic Grey Level Transformations: Log Transformation

a b

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



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

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

This transformation maps a narrow range of low gray-level input values into a wider range of output levels.

Purpose: Expand values of dark pixels in an image while compressing higher level values.

Inverse log will do the opposite

Chapter 3: Intensity Transformations and Spatial Filtering

Contrast Stretching: Power-Law transformations

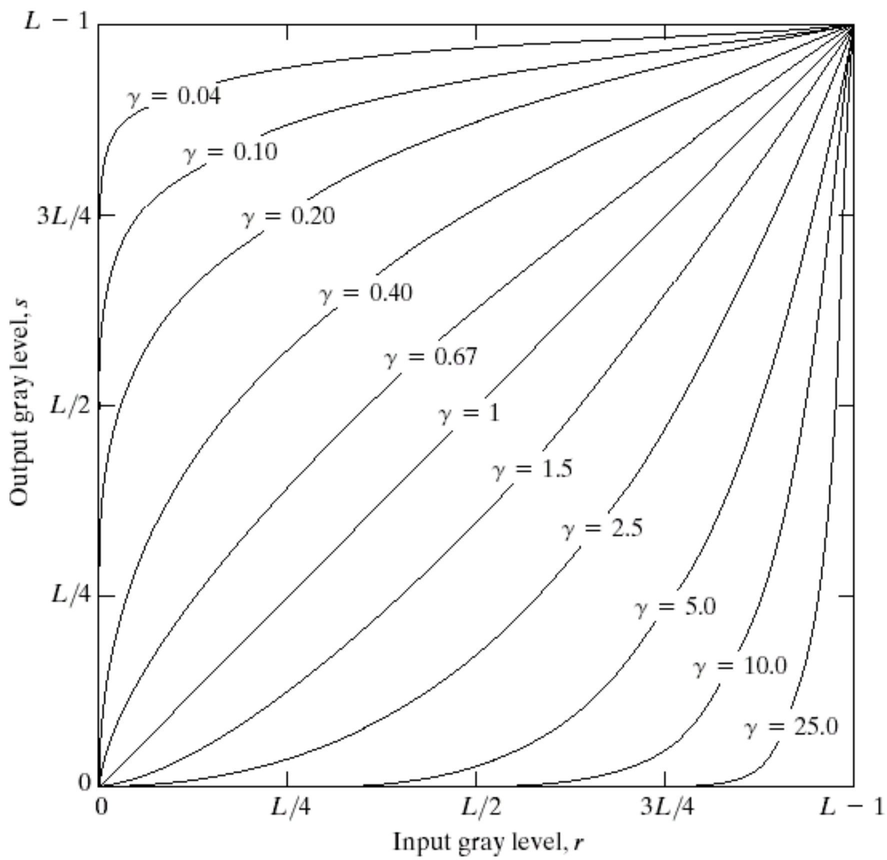


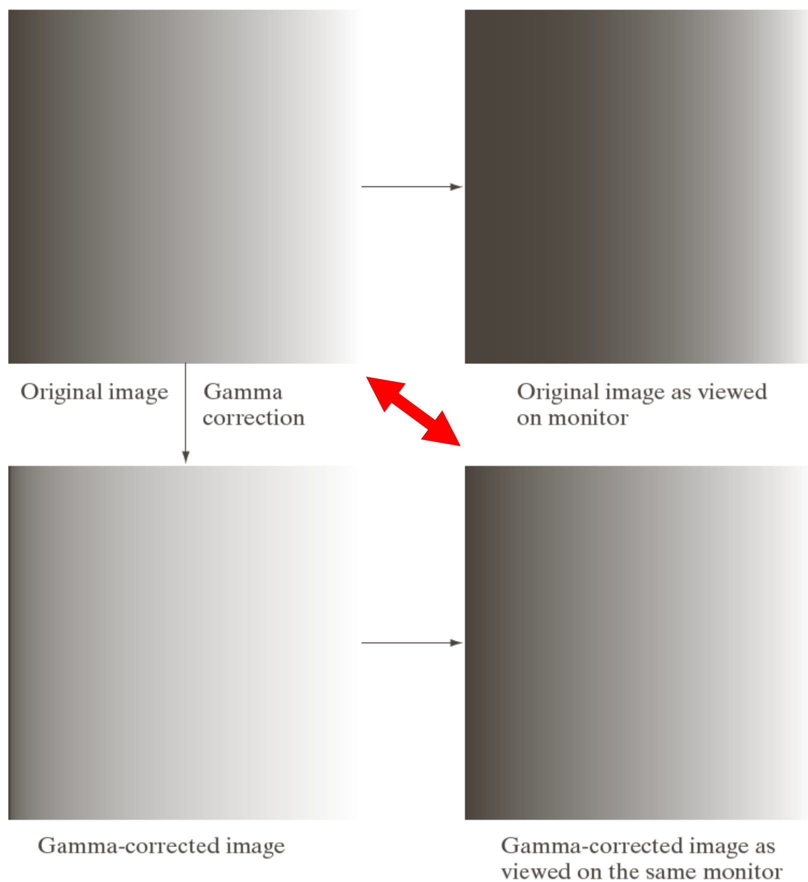
FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

EQ (3.5) $s = cr^{\gamma}$ ⟵ Gamma correction

(c and γ are positive constants, here $c = 1$)

Chapter 3: Intensity Transformations and Spatial Filtering

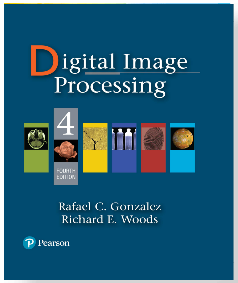
Power-law Transformation: Gamma Correction



a b
c d

[redacted]
(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).

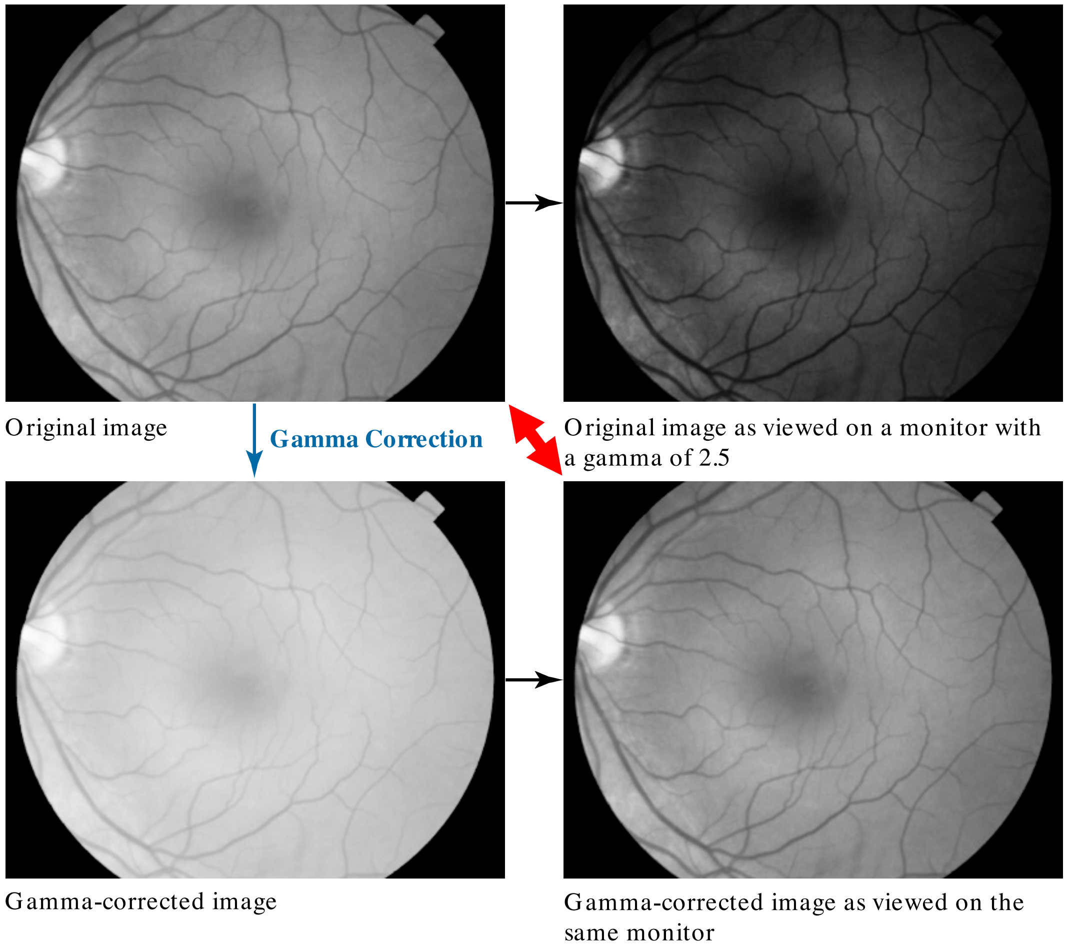
Application: image display in monitors



Chapter 3
Intensity Transformations & Spatial Filtering

a b
c d

FIGURE 3.7
(a) Image of a human retina.
(b) Image as as it appears on a monitor with a gamma setting of 2.5 (note the darkness).
(c) Gamma-corrected image.
(d) Corrected image, as it appears on the same monitor (compare with the original image). (Image (a) courtesy of the National Eye Institute, NIH)

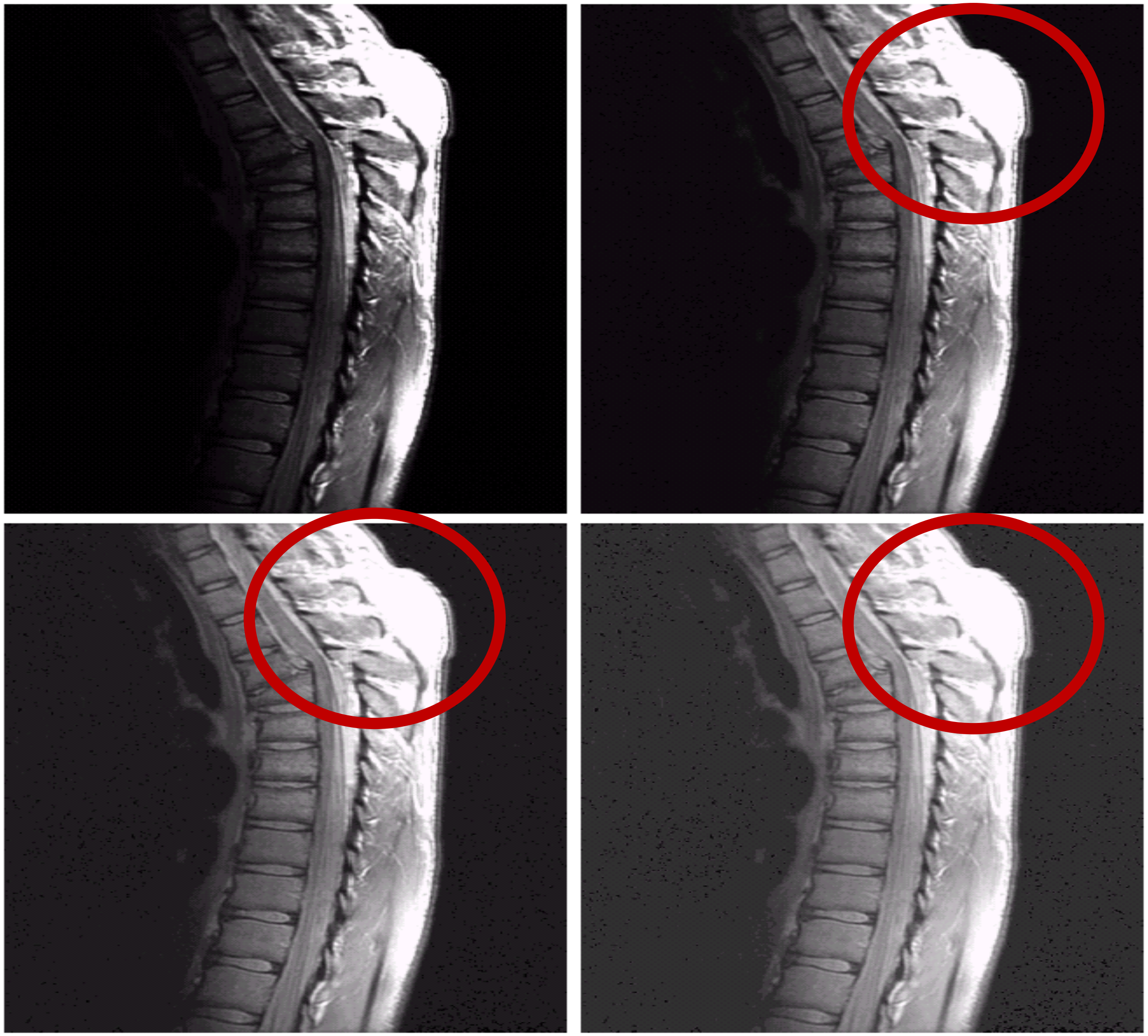


Power-Law Transformation for Gamma-correction of a retina image as displayed on a monitor

Chapter 3: Intensity Transformations and Spatial Filtering

Power-Law Transformation for MR Image Enhancement

Application: Magnetic Resonance image (MR) of a fractured human spine enhanced by gamma values of 0.6, 0.4 and 0.3



a b
c d

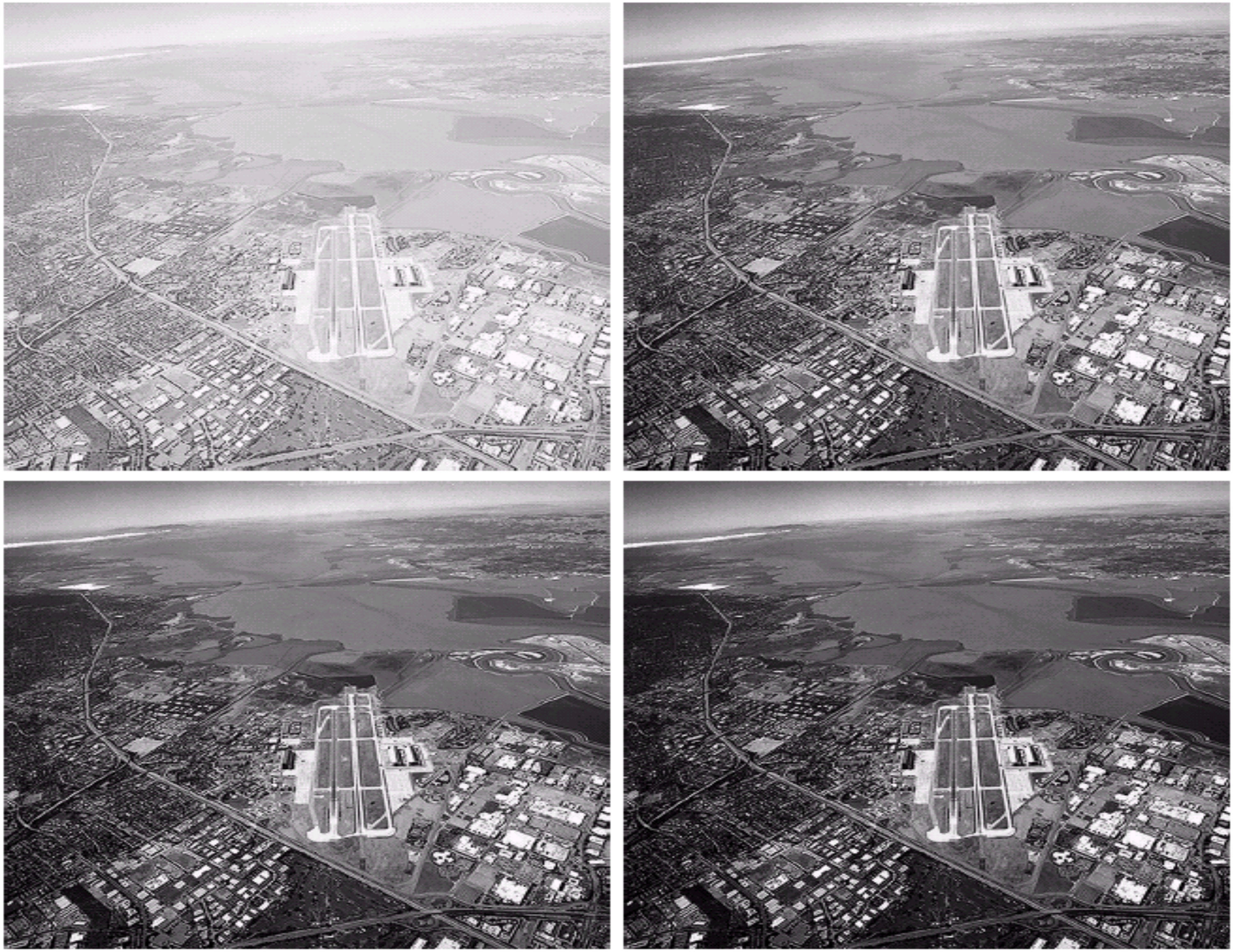
FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine (the region of the fracture is enclosed by the circle). (b)–(d) Results of applying the transformation in Eq. (3-5) with $c = 1$ and $\gamma = 0.6, 0.4$, and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department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

Details in the image are enhanced

Chapter 3: Intensity Transformations and Spatial Filtering: Power-Law Transformation for Aerial Image

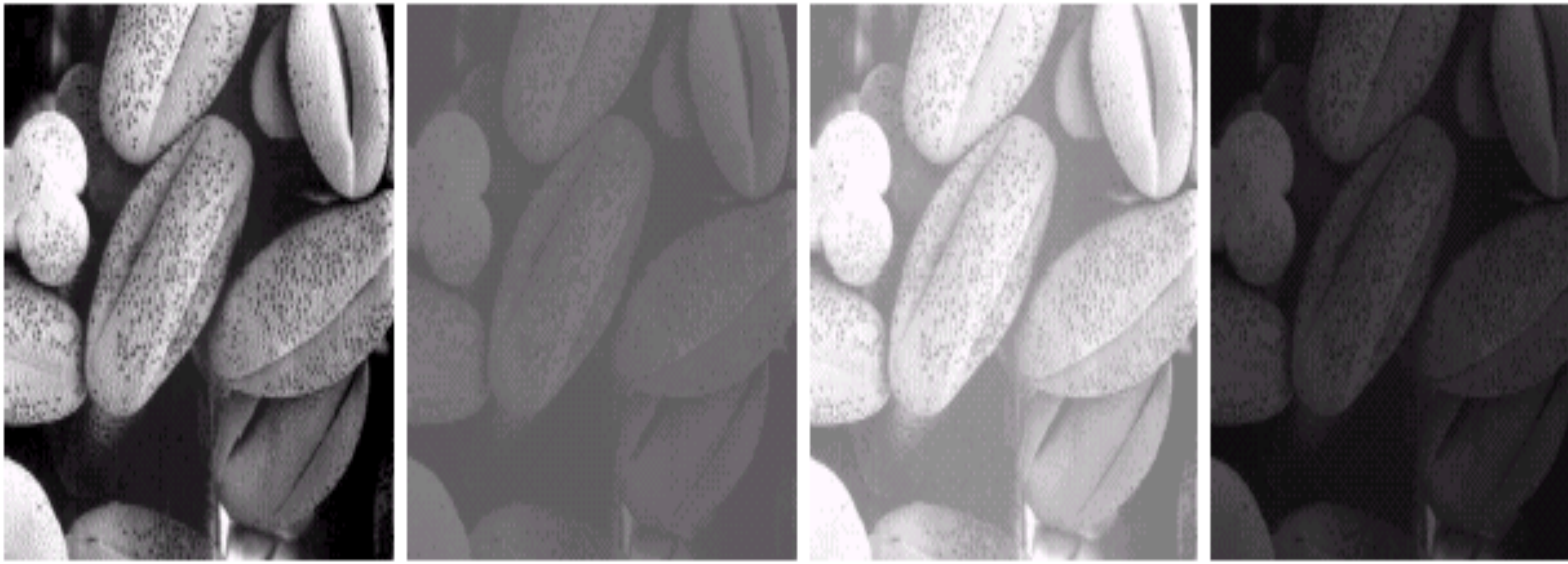
a	b
c	d

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



Chapter 3: Intensity Transformations and Spatial Filtering

Image Intensity Histogram



What is the difference between these scanning electron microscope images of pollen magnified 700 times?

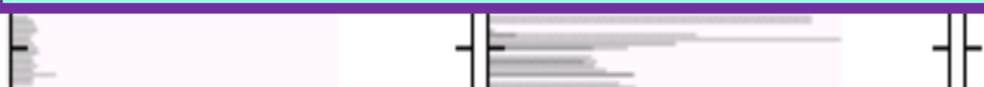


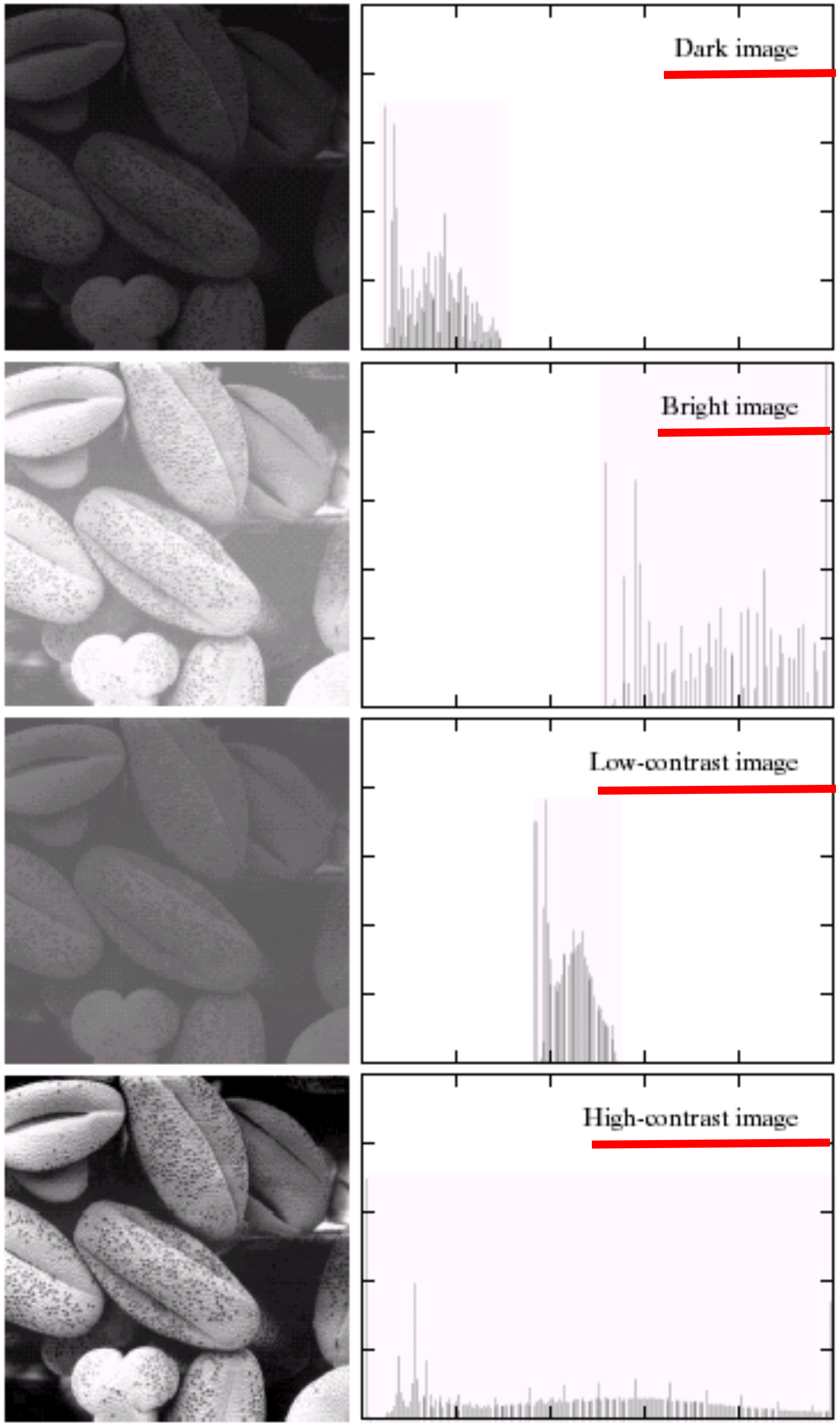
Image Intensity Histogram

The **histogram** of an image with gray levels in the range [0,L-1] is a discrete function $h(r_k) = n_k$

r_k: the k-th gray level
n_k: number of pixels in the image having gray level r_k.

Probability Density Function (PDF) of image intensity values:
Normalized histogram → $p(r_k) = n_k / n$

A scanning electron microscope image of pollen magnified 700 times.



Histogram Processing

Histogram processing re-scales an image so that the enhanced image histogram follows some desired form.

The modification can take on many forms:

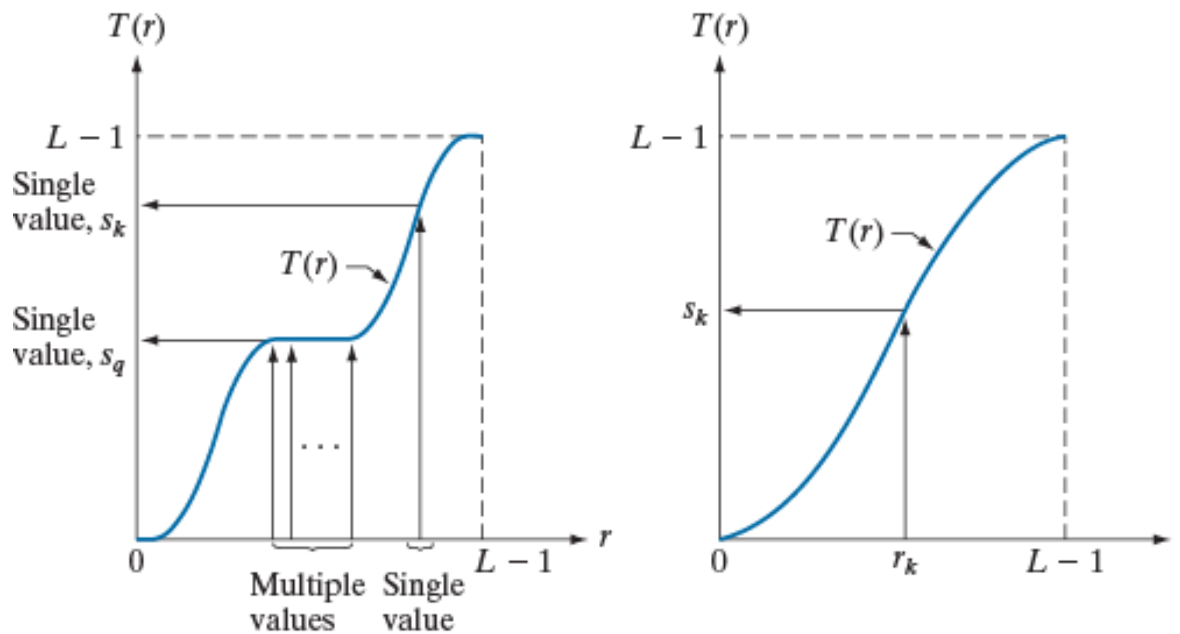
- histogram equalization, or
- histogram shaping (histogram transformation to match a desired shape)
  - e.g. exponential or hyperbolic histogram

Assumptions on histogram equalization:

(a) T(r) is a (strictly*) monotonic increasing function in the interval [0, L-1]; and
(b) T(r) belongs to [0, L-1] for r in [0, L-1]

a b

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



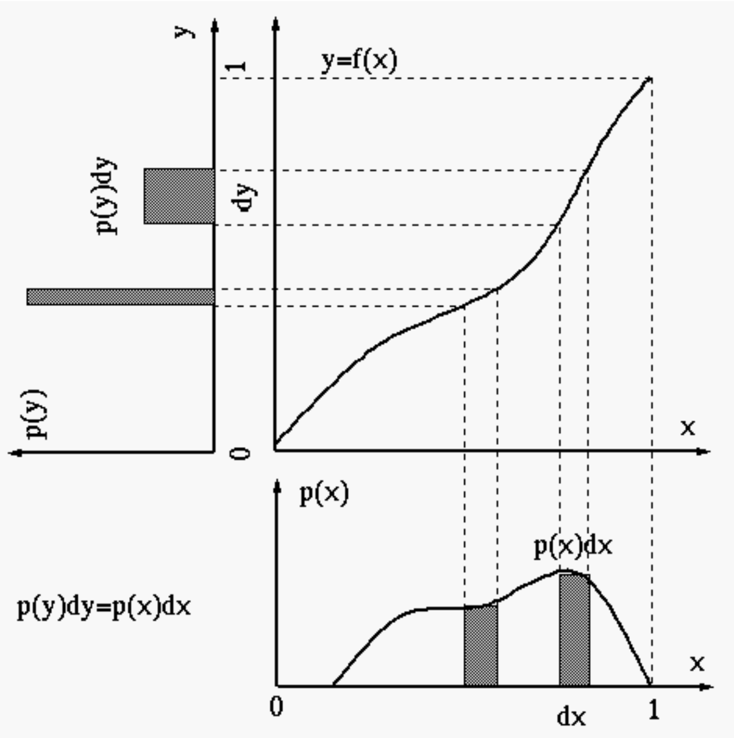
*strict monotonicity is required when the inverse $r=T^{-1}(s)$ is used

Histogram Equalization

- Histogram equalization transforms image gray levels in such a way that the histogram of the resulting image is equalized, i.e., it becomes a constant:

$$h[i] = \text{constant}, \quad \text{for all } i$$

- The goal of histogram equalization:
  - equally use all available gray levels;



- This figure shows that for any given mapping function $y = f(x)$ between the input and output images, the following holds:

$$p(y)dy = p(x)dx$$

- i.e., the number of pixels mapped from x to y is unchanged.

Histogram Equalization

- For the output image to have an equalized histogram, *p(y)* = c (constant) *and c = 1* if the gray levels are assumed to be in the range 0 and 1 *(0<x<1, 0<y<1*). Then we have:

$$c\,dy = p(x)dx, \quad \text{or}\; c\,\frac{dy}{dx} = p(x) \text{ and c = 1}$$

- i.e., the mapping function for histogram equalization is:

$$y = \int_0^x p(u)du = F(x) - F(0) = F(x)$$

- where

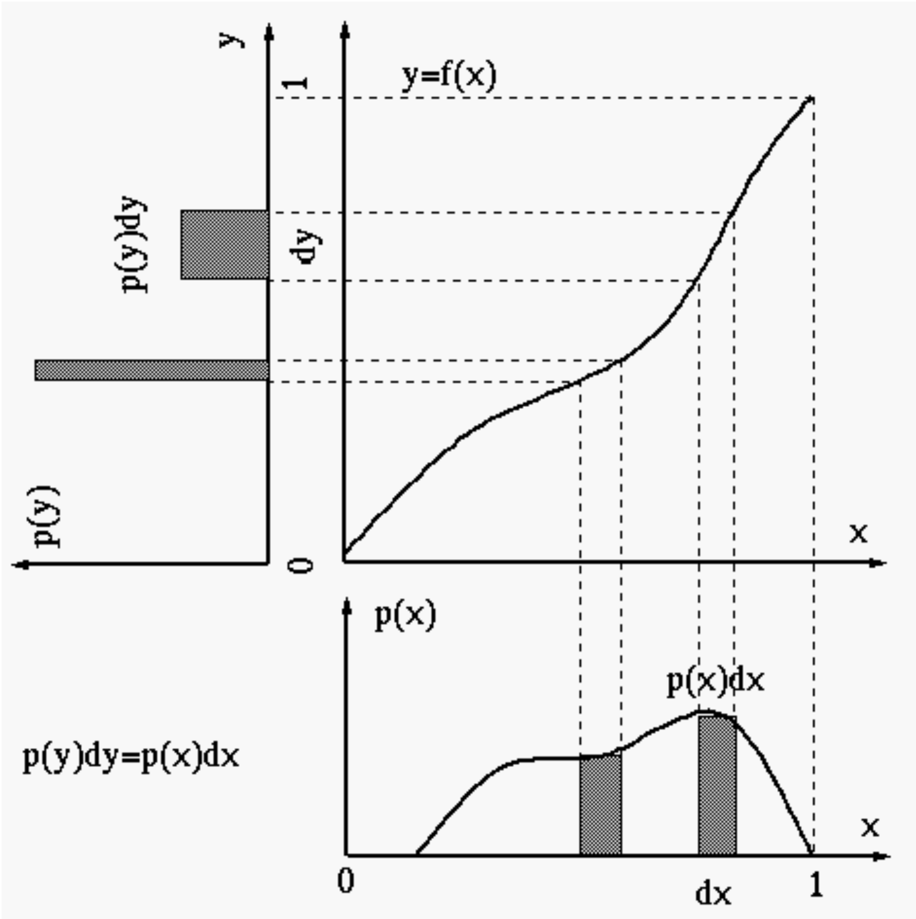
$$F(x) = \int_0^x p(u)du, \qquad F(0) = 0$$

- F(x) is the cumulative (probability) distribution function (CDF) of the input image, which is monotonically increasing. p(x) is the PDF of image x.

Histogram Equalization

Intuitively, histogram equalization is realized by the following:

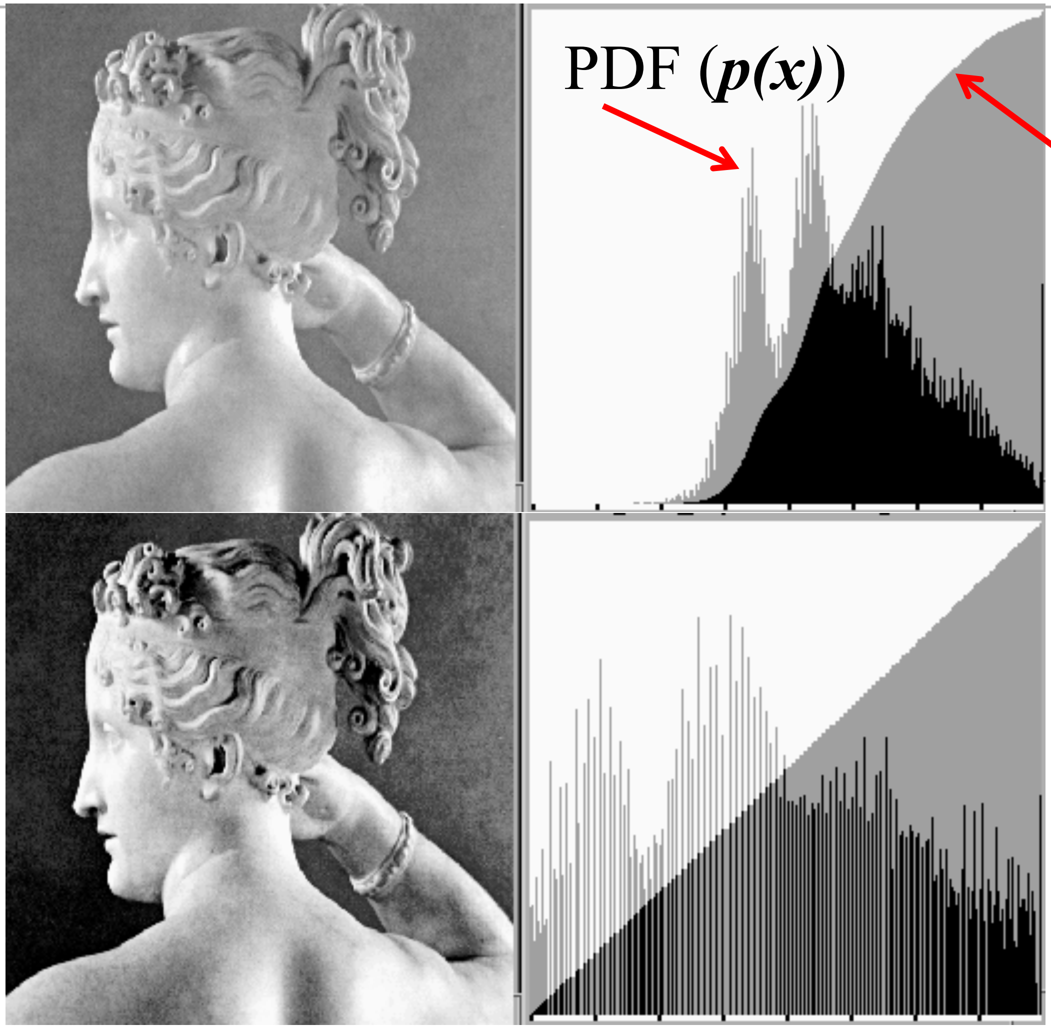
- If *p(x)* is high, *y=F(x)* has a steep slope, *dy* will be wide, causing *p(y)* to be low in order to keep *p(y)dy = p(x)dx*;
- If *p(x)* is low, *y=F(x)* has a shallow slope, *dy* will be narrow, causing *p(y)* to be high to keep p*(y)dy = p(x)dx*



Histogram Equalization

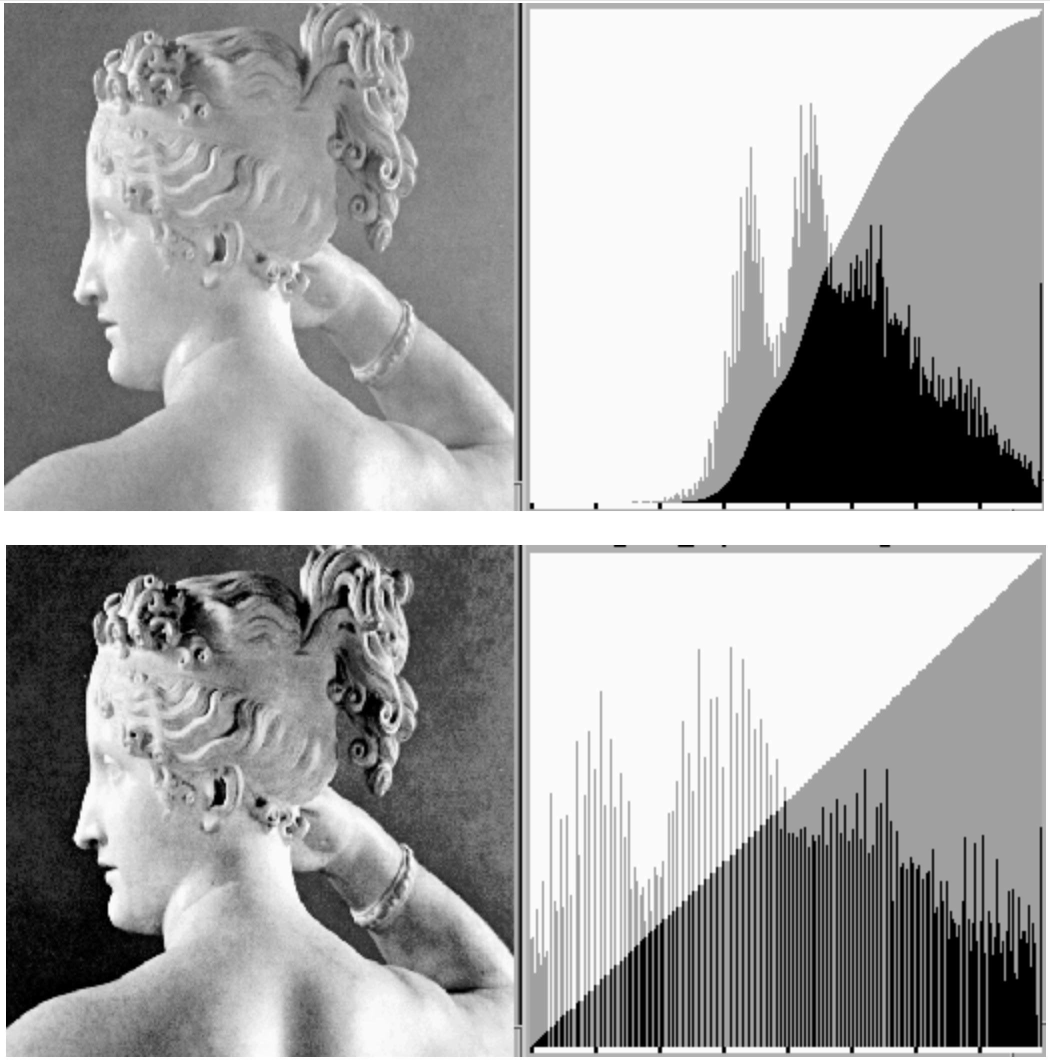
Original image

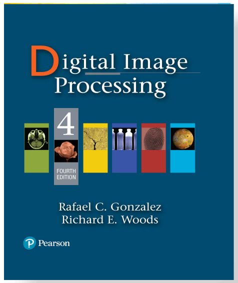
Histogram equalized image

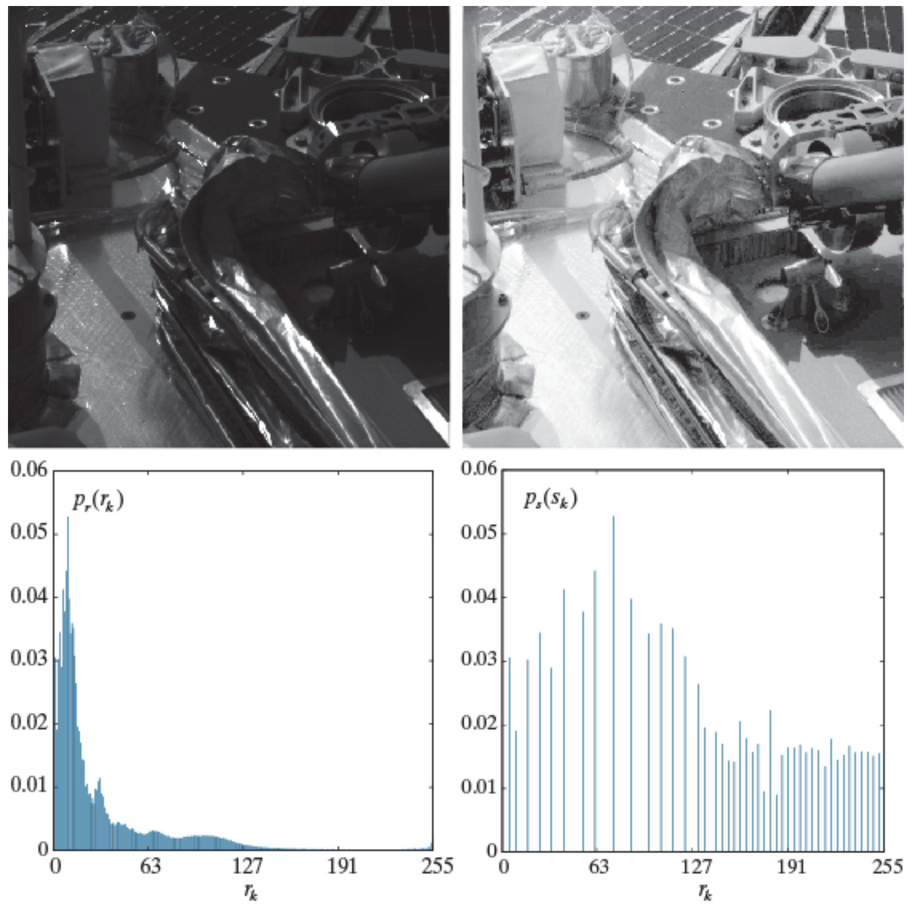


Histogram Equalization

- In the following example, the histogram of a given image is equalized. Although the resulting histogram may not look constant, the cumulative histogram is a linear ramp indicating that the image histogram has indeed been equalized. **The density histogram is not guaranteed to be constant because the pixels of the same gray level cannot be separated to satisfy a constant distribution.**







a b
c d

FIGURE 3.22 (a) Image from Phoenix Lander. (b) Result of histogram equalization. (c) Histogram of image (a). (d) Histogram of image (b). (Original image courtesy of NASA.)

Another example of histogram equalization

Histogram Equalization

For discrete gray levels, the gray level of the input *x* takes one of the *L* discrete values and the continuous mapping function

$$y = F(x) = \int_0^x p(u)du$$

becomes discrete:

$$y' = F(x) = \sum_{i=0}^{x} P_i$$

where P_i is the probability for the gray level of any given pixel to be $(0<i<L)$:

$$P_i = \frac{n_i}{\sum_{i=0}^{L-1} n_i} = \frac{n_i}{N} \quad \text{and} \quad \sum_{i=0}^{L-1} P_i = 1$$

Histogram Equalization

- The resulting function *y'* is in the range $0 \leq y' \leq 1$ and it needs to be converted to the gray levels $0 \leq y \leq L-1$ by one of the following two ways:

$$y = \lfloor y'(L-1) + 0.5 \rfloor$$

$$y = \lfloor \frac{y' - y'_{min}}{1 - y'_{min}}(L-1) + 0.5 \rfloor$$

- where $\lfloor x \rfloor$ is the floor, or the integer part of a real number *x*, and adding *0.5* is for proper rounding. Note that while both conversions map $y'_{max} = 1$ to the highest gray level *L - 1*, the second conversion also maps y'_{min} to 0 to stretch the gray levels of the output image to occupy the entire dynamic range $0 \leq y \leq L-1$

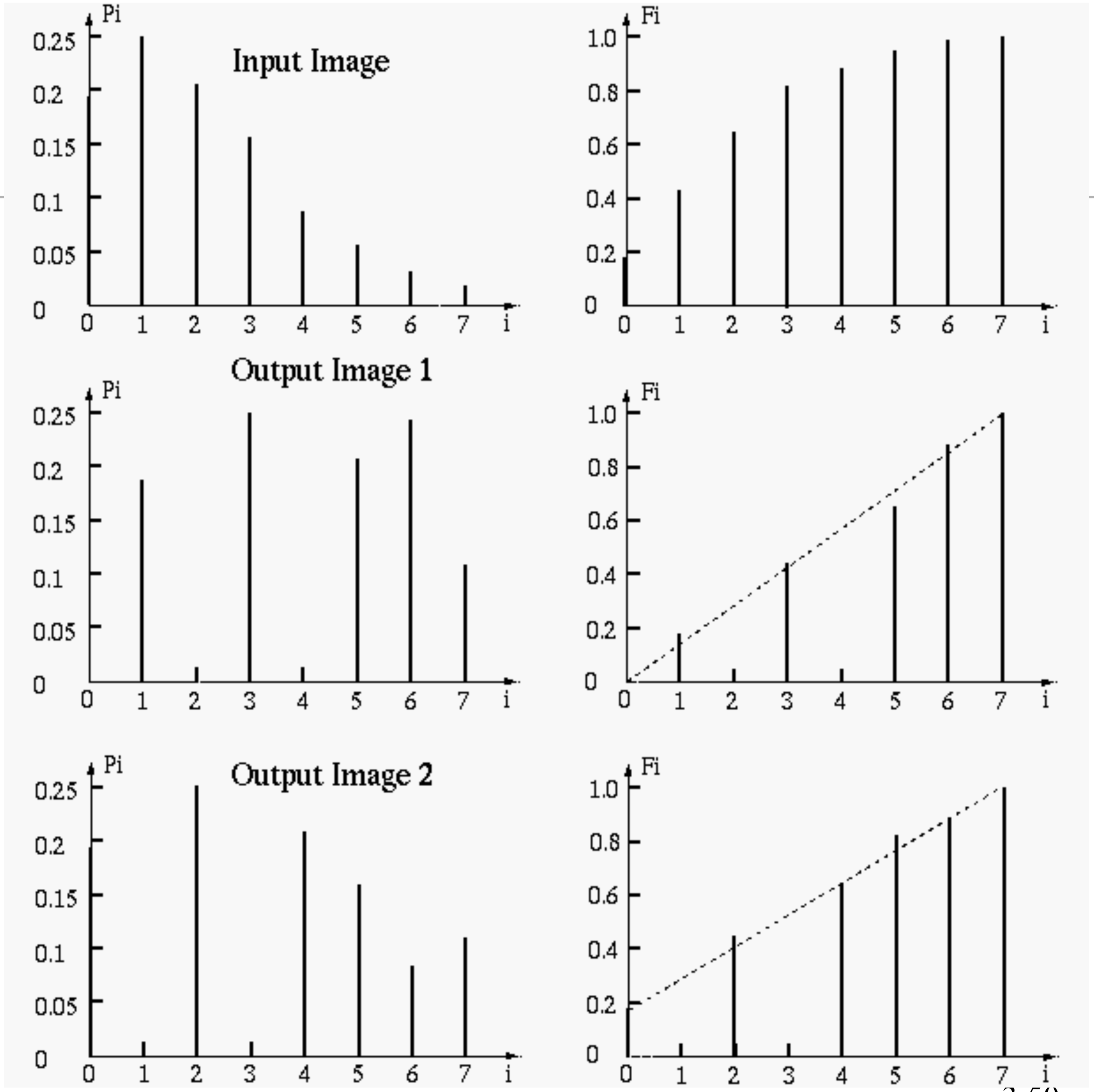
Histogram Equalization

Example: Assume the images have N=4096 pixels in L=8 gray levels. The following table shows the equalization process corresponding to the two conversion methods above:

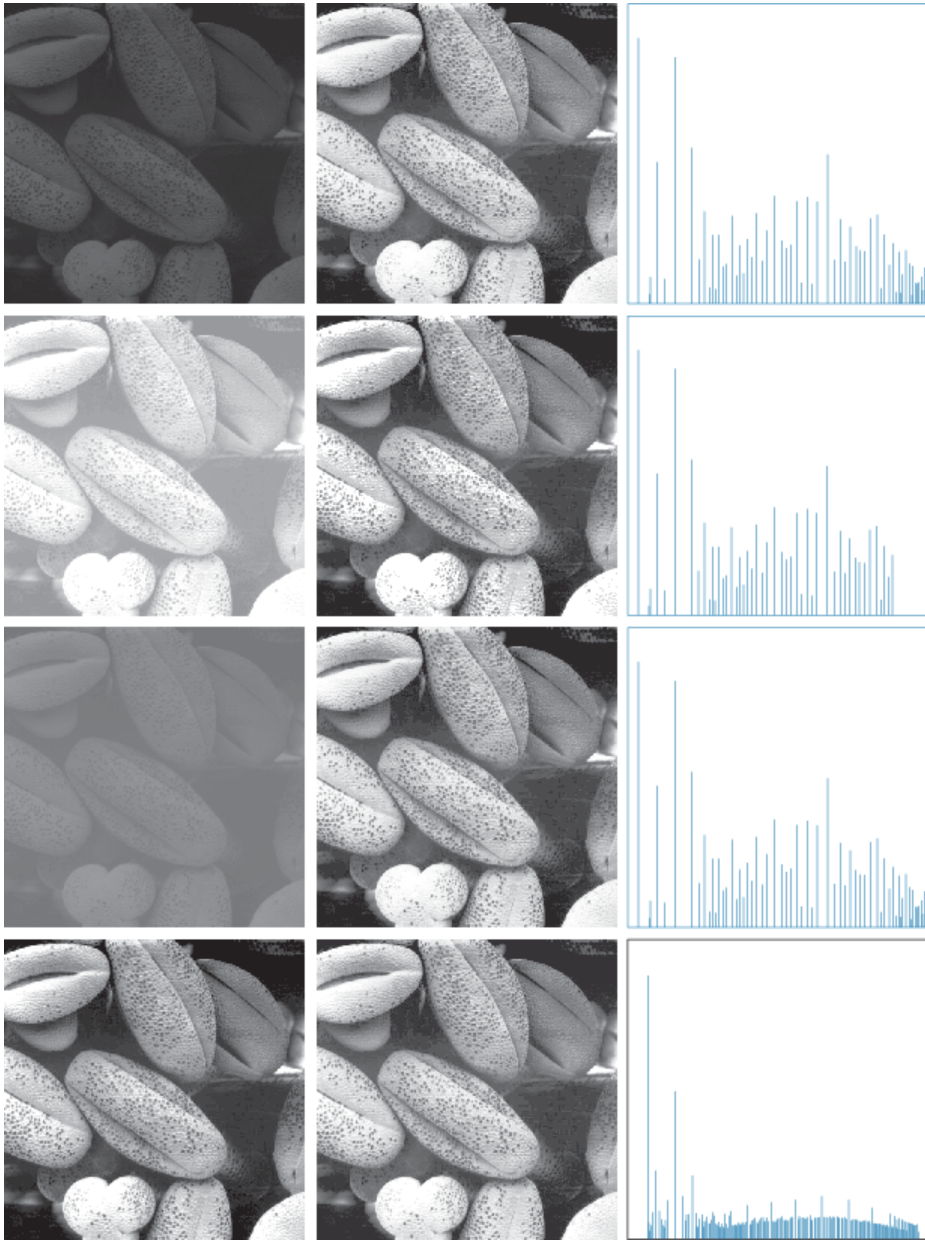
x_i	n_i	$P_i = n_i/N$	$y' = F_i$	y_j^1	P_j^1 PDF	F_j^1 CDF	y_j^2	P_j^2 PDF	F_j^2 CDF
0/7	790	0.19	0.19	1/7	0.19	0.19	0/7	0.19	0.19
1/7	1023	0.25	0.44	3/7	0.25	0.44	2/7	0.25	0.44
2/7	850	0.21	0.65	5/7	0.21	0.65	4/7	0.21	0.65
3/7	656	0.16	0.81	6/7			5/7	0.16	0.81
4/7	329	0.08	0.89	6/7	0.24	0.89	6/7	0.08	0.89
5/7	245	0.06	0.95	7/7			7/7		
6/7	122	0.03	0.98	7/7			7/7		
7/7	81	0.02	1.00	7/7	0.11	1.00	7/7	0.11	1.00
	4096								

Histogram Equalization

PDF CDF



Histogram Equalization

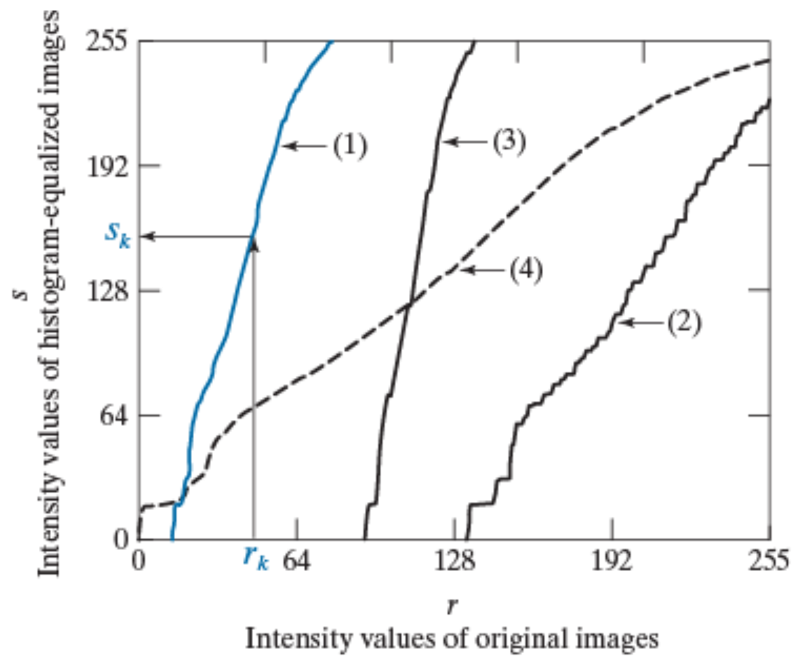


Observe that the histogram equalized images are very similar

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

Histogram Equalization

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



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

Sample Exam Question

- Suppose you have the following 8x8 image with 3-bit representation:
- The aim is to modify the gray-scale of this image such that the histogram of the processed image is as close as possible to constant – in the range between 0 and 7.

a) How is this process called?

b) Determine a transformation function that will achieve this aim.

c) Determine the transformed image and its histogram.

0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
1	1	1	2	2	2	3	4
1	1	1	2	2	2	3	4
1	1	1	2	2	3	3	5
1	1	1	2	2	3	3	5
1	1	2	2	2	3	4	6
1	1	2	2	2	3	4	7

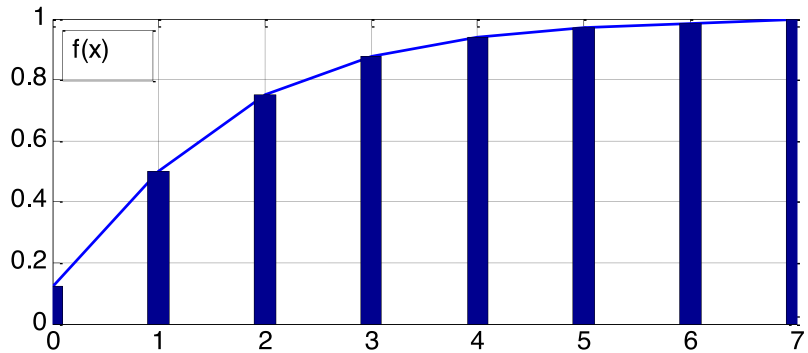
Solution:

a) Histogram Equalization

b) The histogram, *H*, PDF, *p* and CDF can be computed easily, as follows:

$$H(x) = [8,24,16,8,4,2,1,1] \Rightarrow p(i) = [0.125, 0.375, 0.25,\ 0.125, 0.0625,\ 0.0313,\ 0.0156,\ 0.0156]$$

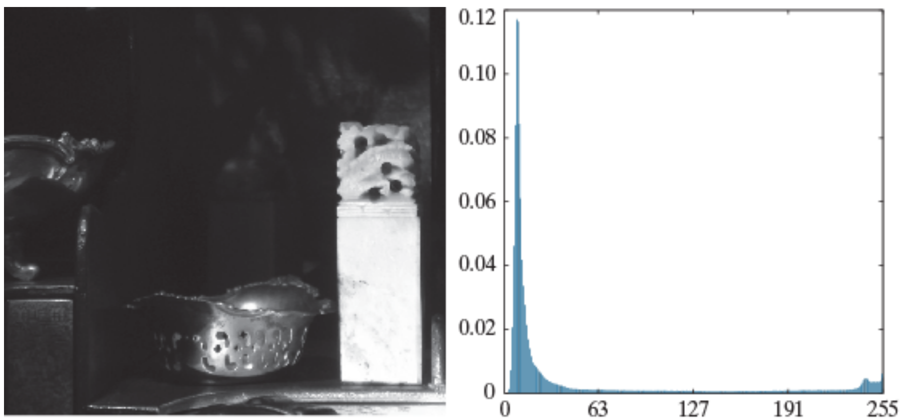
$$y' = f(x) = \sum_{i=0}^{x} p(i) = [0.125,\ 0.5, 0.75, 0.875, 0.9375, 0.9688, 0.9844, 1]$$



Since the resulting function is in the range of $0 \le y' \le 1$ it needs to be converted to the 3-bit grey levels

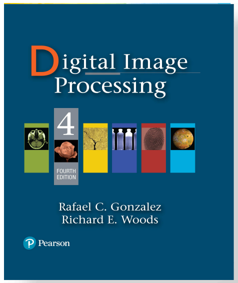
$$0 \le y \le 7 \quad \text{by} \quad y = \lfloor y'(8-1) + 0.5 \rfloor = [1, 4, 5, 6, 7, 7, 7, 7]$$

Problems with histogram equalization and the need for histogram specification, see example below:



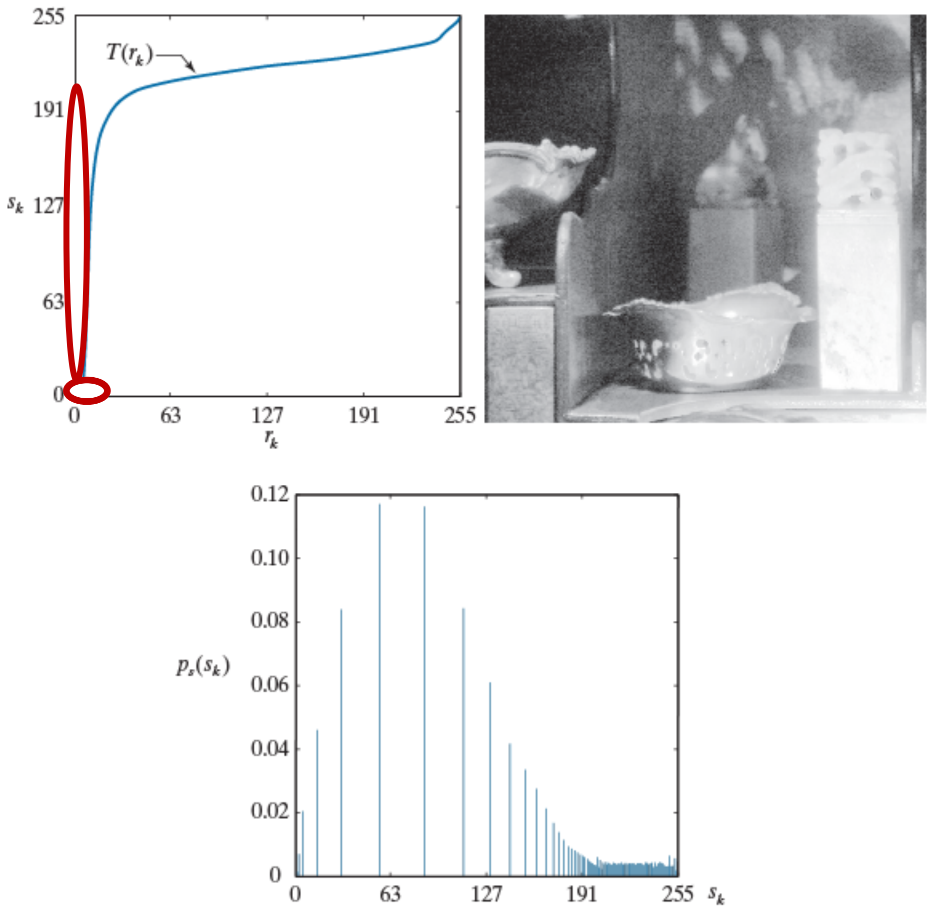
a b
FIGURE 3.24 (a) An image, and (b) its histogram.

Let us try to equalize this histogram



a b
c

FIGURE 3.25
(a) Histogram equalization transformation obtained using the histogram in Fig. 3.24(b).
(b) Histogram equalized image.
(c) Histogram of equalized image.



What we achieved:
(a) object in the dark area is now visible, thanks to the equalization, but
(b) the resulting image is too noisy! Why?

Look at what happened to the narrow range of darkest end of the image histogram (considered the noisiest due to imaging sensor limitations)

It was expanded into a much higher range of intensity values in the output image

Conclusion: Consider histogram specification, instead

Histogram Matching (Specification)

- Automatic enhancement (histogram equalization) may not be the best thing to do in all applications.
- We may wish to map an input image r to an output image z with a given arbitrary histogram $p_z(z)$. So, how?
- This problem can be solved through histogram matching or histogram specification.
- In histogram matching we are given:
  - an input image *r* with histogram $p_r(r)$ *and*
  - a pre-specified histogram of the target image z, $p_z(z)$.
- The problem is to match $p_r(r)$ to $p_z(z)$, in other words, we want to create an image *z* from image *r* that has a histogram $p_z(z)$.
- That is, how to map image r to image z?

Histogram Matching

Let r and z be two random variables with continuous intensities and $p_r(r)$ and $p_z(z)$ their respective PDFs.

We perform the following steps:

- Let s be a random variable satisfying:

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

- That is, s is the image obtained from image r through histogram equalization!
- Define another random variable z, satisfying:

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

Chapter 3: Intensity Transformations and Spatial Filtering

Histogram Matching

- So, s = G(z) is a new image obtained from image z (again) by histogram equalization.
- Since s = G(z) (by design), G(z) = T(r)

$$r \xrightarrow{T(r)} s \xleftarrow{G(z)} z \text{ (unknown, but its } p_z(z) \text{ is given)}$$

$$s \xrightarrow[G^{-1}(z)]{} z$$

Therefore, z must satisfy

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

Histogram Matching

To summarize:

An image z whose intensity levels have a specified PDF $p_z(z)$ can be obtained from a given image r by following these steps:

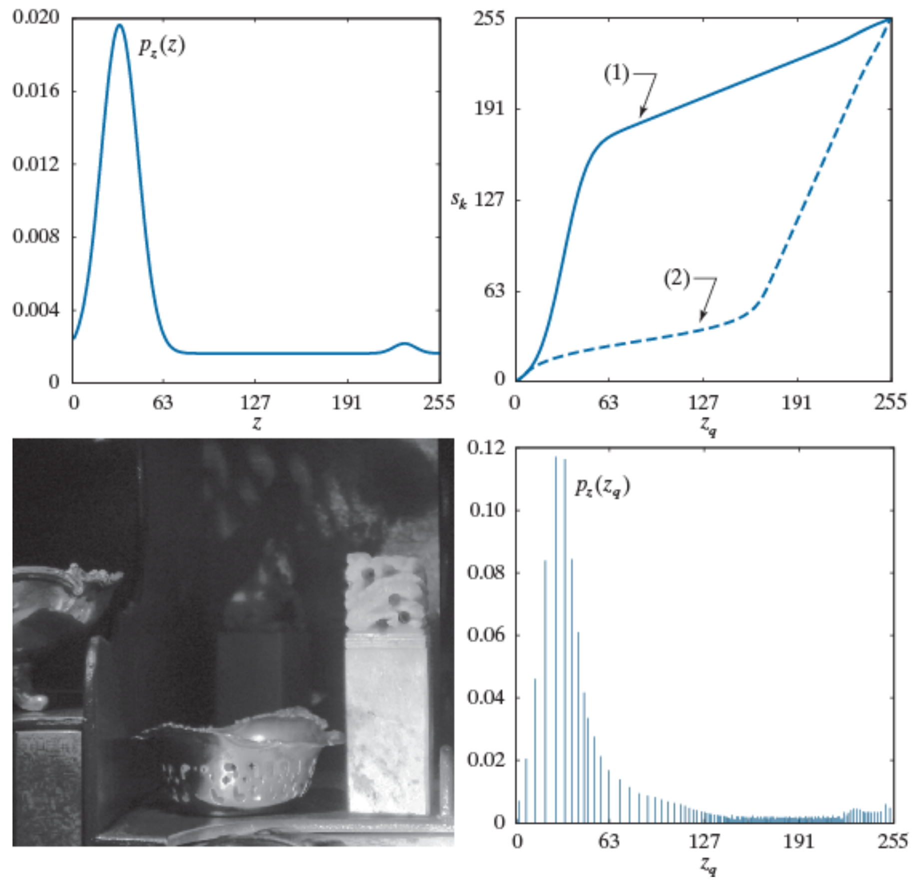
- Obtain $p_r(r)$ from the original image r and use $s = T(r)$ to obtain image s
- Use the specified PDF $p_z(z)$ to obtain $G(z)$
- Find the inverse transformation $z = G^{-1}(s)$
- Obtain the output image z as follows:
  - For each pixel with value s in the equalized image s, perform the inverse mapping $z = G^{-1}(s)$ to obtain the corresponding pixel in the output image z.
  - When all pixels have been processed, the PDF of the output image will be equal to the specified PDF $p_z(z)$.

Let us try to apply a smoother transition levels in the dark regions

a b
c d

FIGURE 3.26 Histogram specification. (a) Specified histogram. (b) Transformation $G(z_q)$, labeled (1), and $G^{-1}(s_k)$, labeled (2). (c) Result of histogram specification. (d) Histogram of image (c).

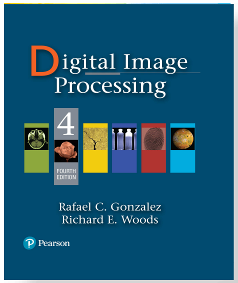
Specified histogram $p_z(z)$

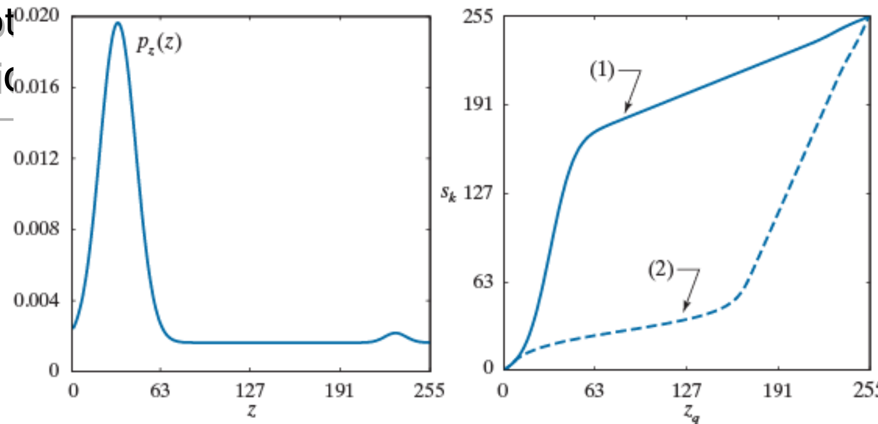


(1) $G(z)$
(2) $G^{-1}(z)$

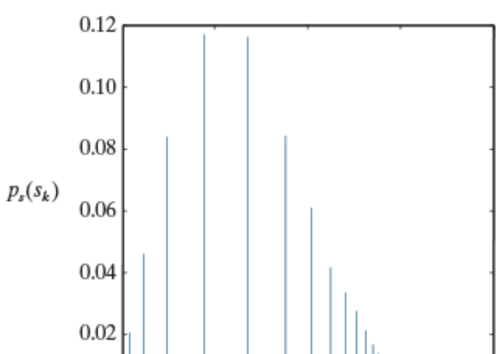
$G(z)$ is the function that equalizes $p_z(z)$ to get the intermediate image s

$G^{-1}(s)$ obtains image z from image s



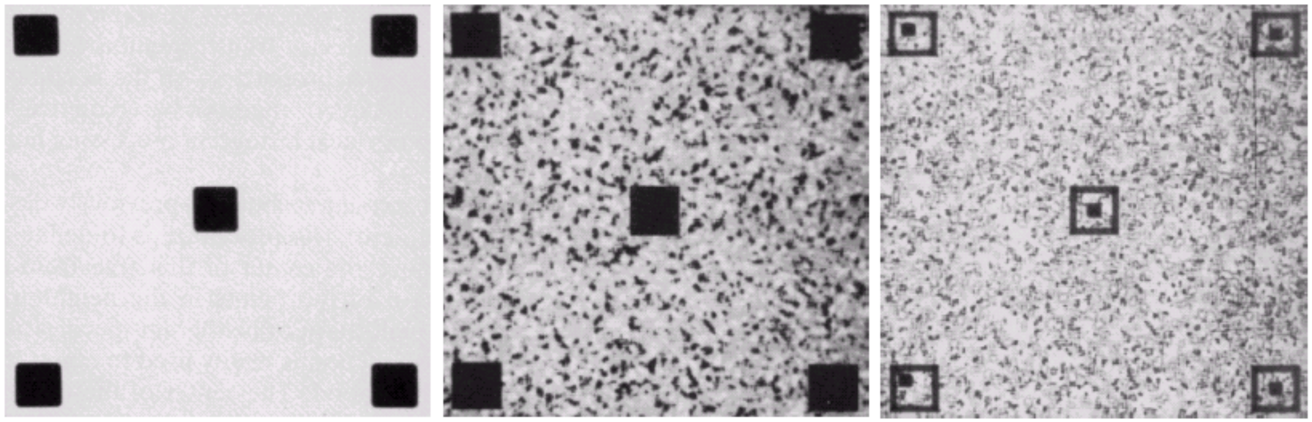


Compare now the two images!



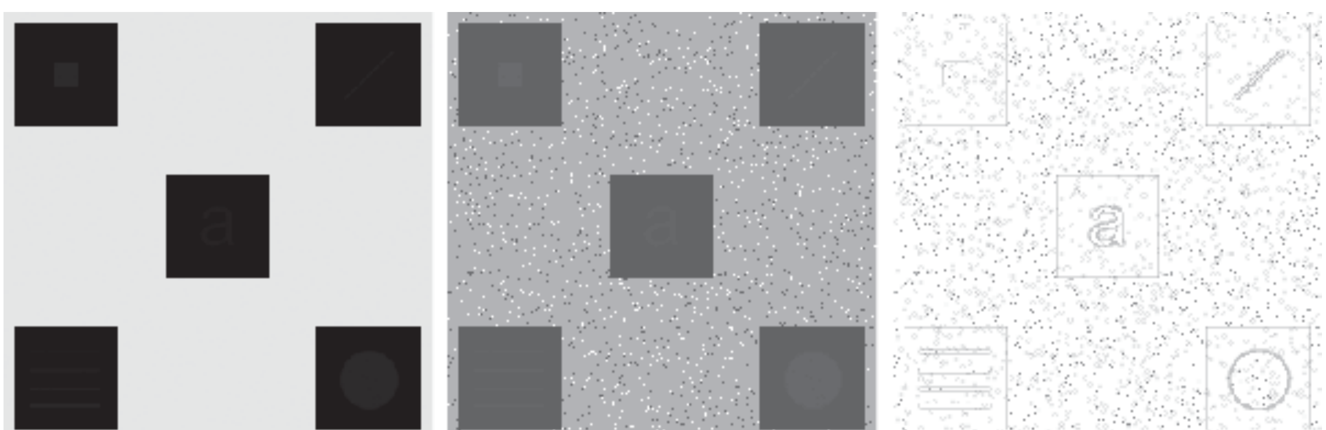
Chapter 3: Intensity Transformation and Spatial Filtering

Localized Histogram Equalization



a b c

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



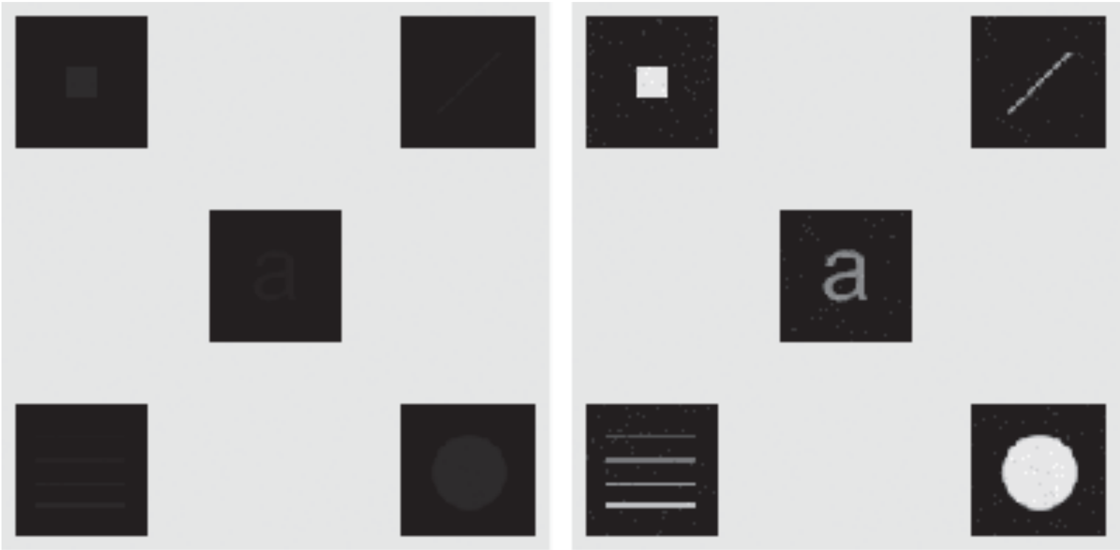
a b c

FIGURE 3.32
(a) Original image. (b) Result of global histogram equalization. (c) Result of local histogram equalization.

An even more sophisticated example (details later)

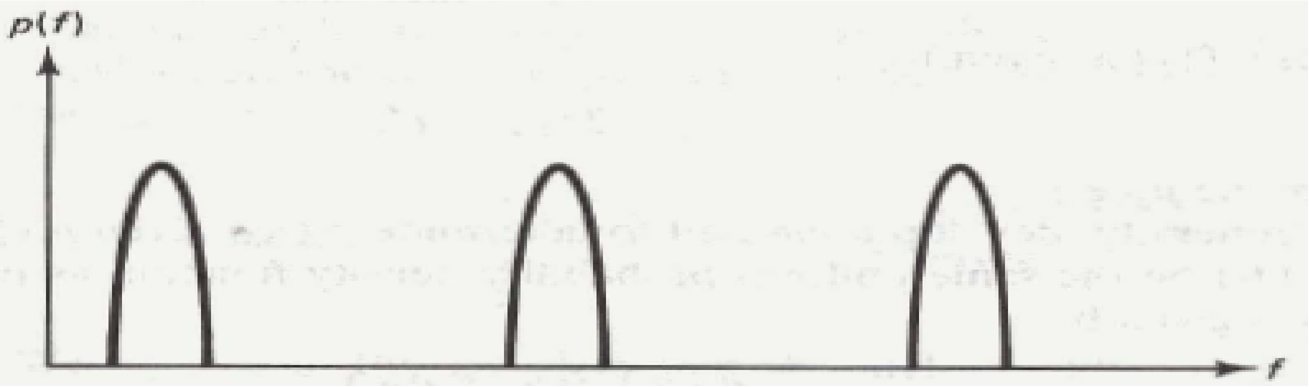
a b

FIGURE 3.33 (a) Original image. (b) Result of local enhancement based on local histogram statistics. Compare (b) with Fig. 3.32(c).



Sample Exam Question

- Let $f(x, y)$ be an image of 256x256 pixels. The histogram of f, $p(f)$ is sketched below.



A- What can you say about the problem of $f(x, y)$?

B- Roughly sketch a transformation function which is likely to improve the contrast of the image.

Solution:

A-There is a lack of contrast in the image since all gray level values are concentrated on 3 limited regions. To enhance the contrast, the following transformation can be used:

B-