Image Enhancement

Laurent Najman

laurent.najman@esiee.fr

ESIEE Paris

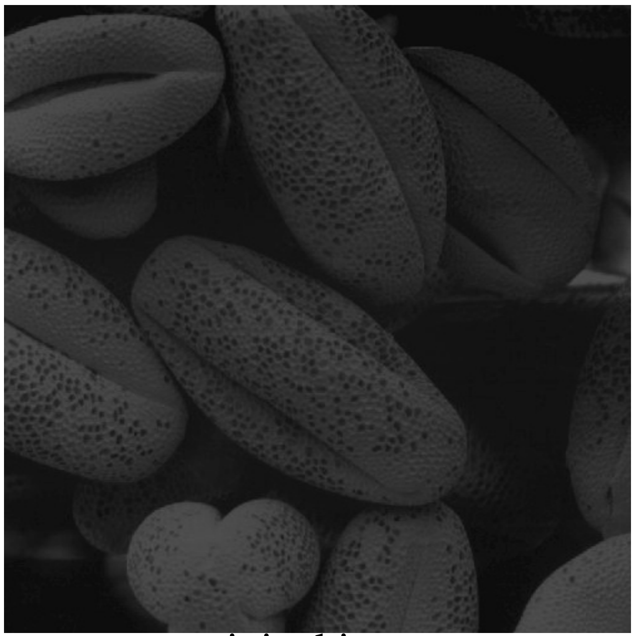
Université Paris-Est, Laboratoire d'Informatique Gaspard-Monge, Équipe A3SI

Book

- Digital Image Processing, Second Edition
  - authors: Rafael C. Gonzalez and Richard E.Woods
  - editor: Prentice Hall

What is image enhancement ?

We try to improve an image so that it looks **subjectively** better!



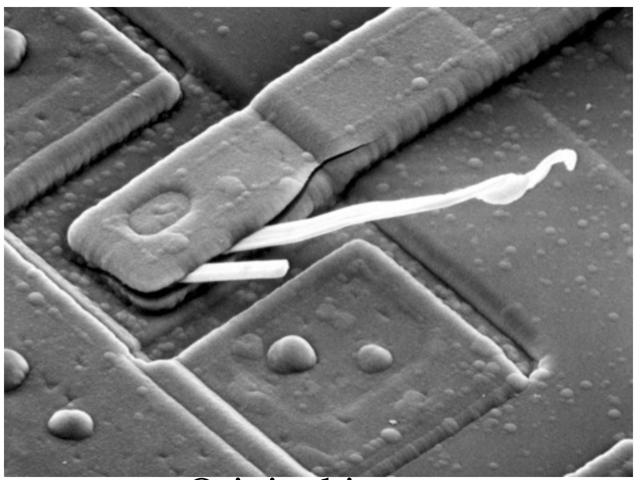
original image

enhanced image

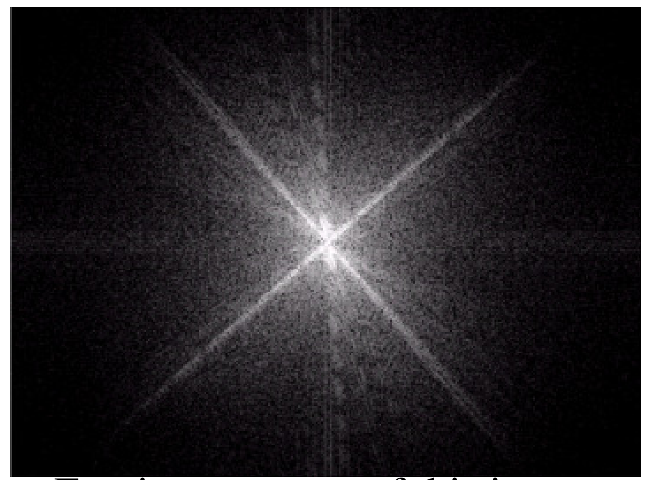
Which one of these images looks “better”?

Domain Methods

- Spatial Domain
  - refers to the aggregate of pixels composing an image
  - direct manipulation of the pixels
- Frequency Domain
  - Fourier transfrom is the key
  - Consists of variations on low- and high-pass filtering



Original image



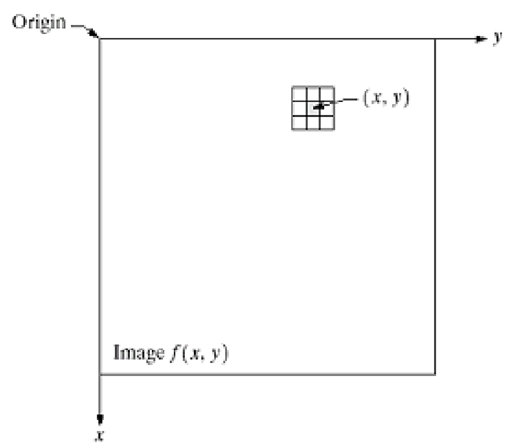
Fourier spectrum of this image

Spatial Domain – Background

- Spatial domain: aggregate of pixels/voxels composing an image.
- Spatial domain method: procedures that operate directly on these pixels.
- We use masks (say, 3×3) for enhancement techniques: mask processing or filtering

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

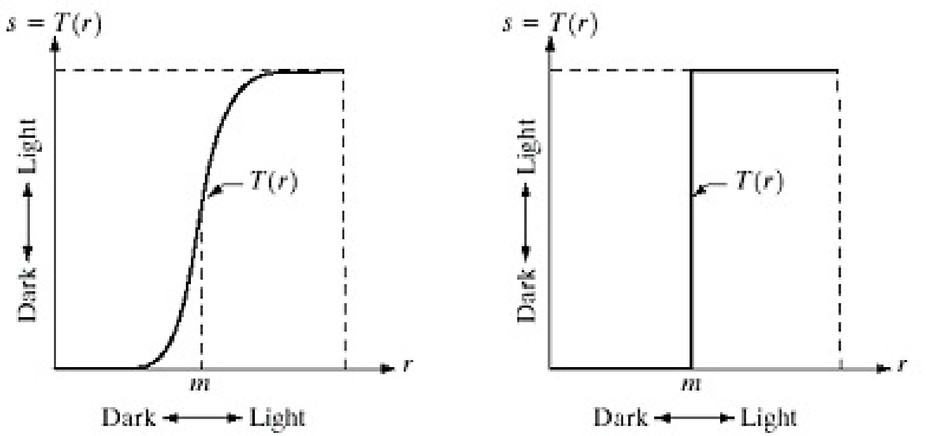
FIGURE 3.1 A 3×3 neighborhood about a point (x, y) in an image.



Intensity Mapping

$$s = T(r)$$

- T is of size 1×1
- T is a gray-level transformation function
- gray level in the range $[0, L-1]$



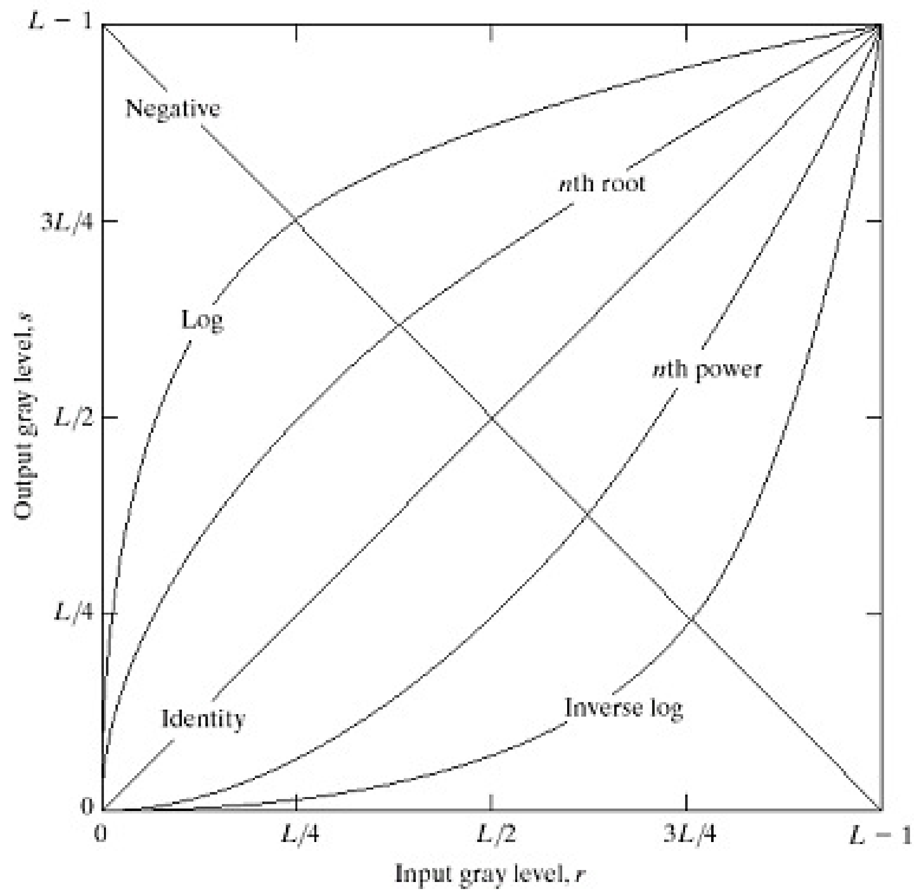
a b

FIGURE 3.2 Gray-level transformation functions for contrast enhancement.

Basic Gray Level Transformations

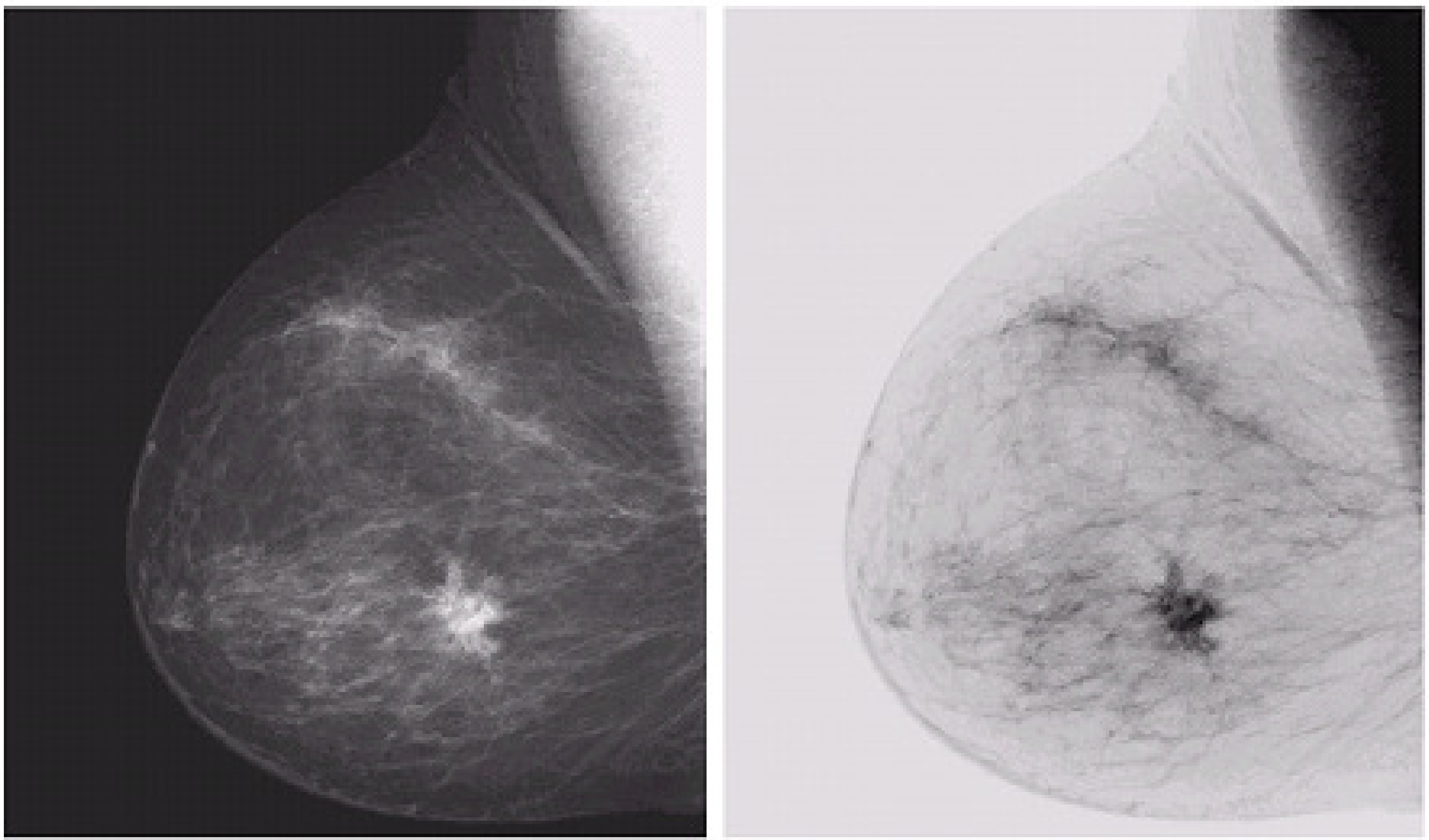
- linear
- logarithmic: $s = c\,\log(1 + r)$
- power-law: $s = c\,r^{\gamma}$

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



Application of Linear Transforation

Negative image: $s = L - 1 - r$

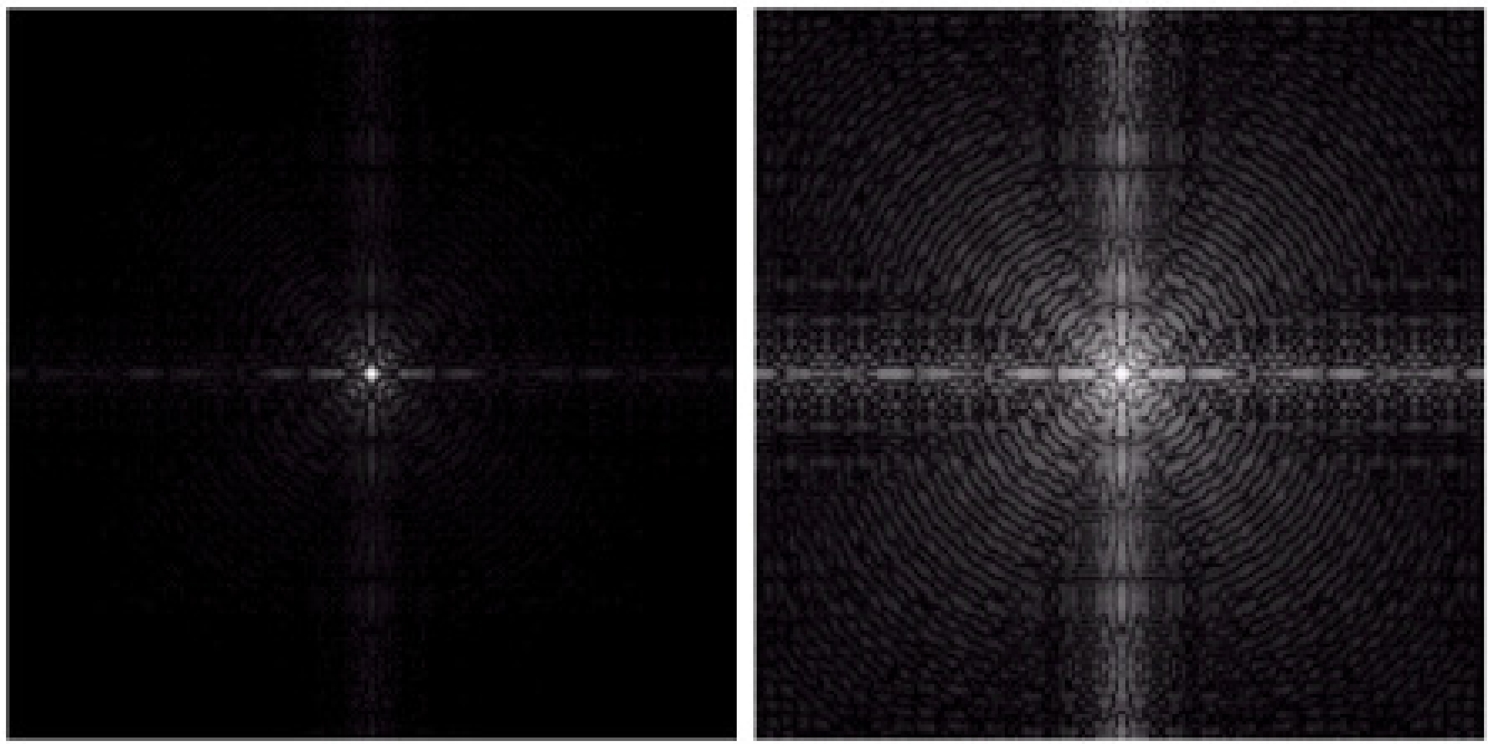


a b

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.)

Log Transformations (1)

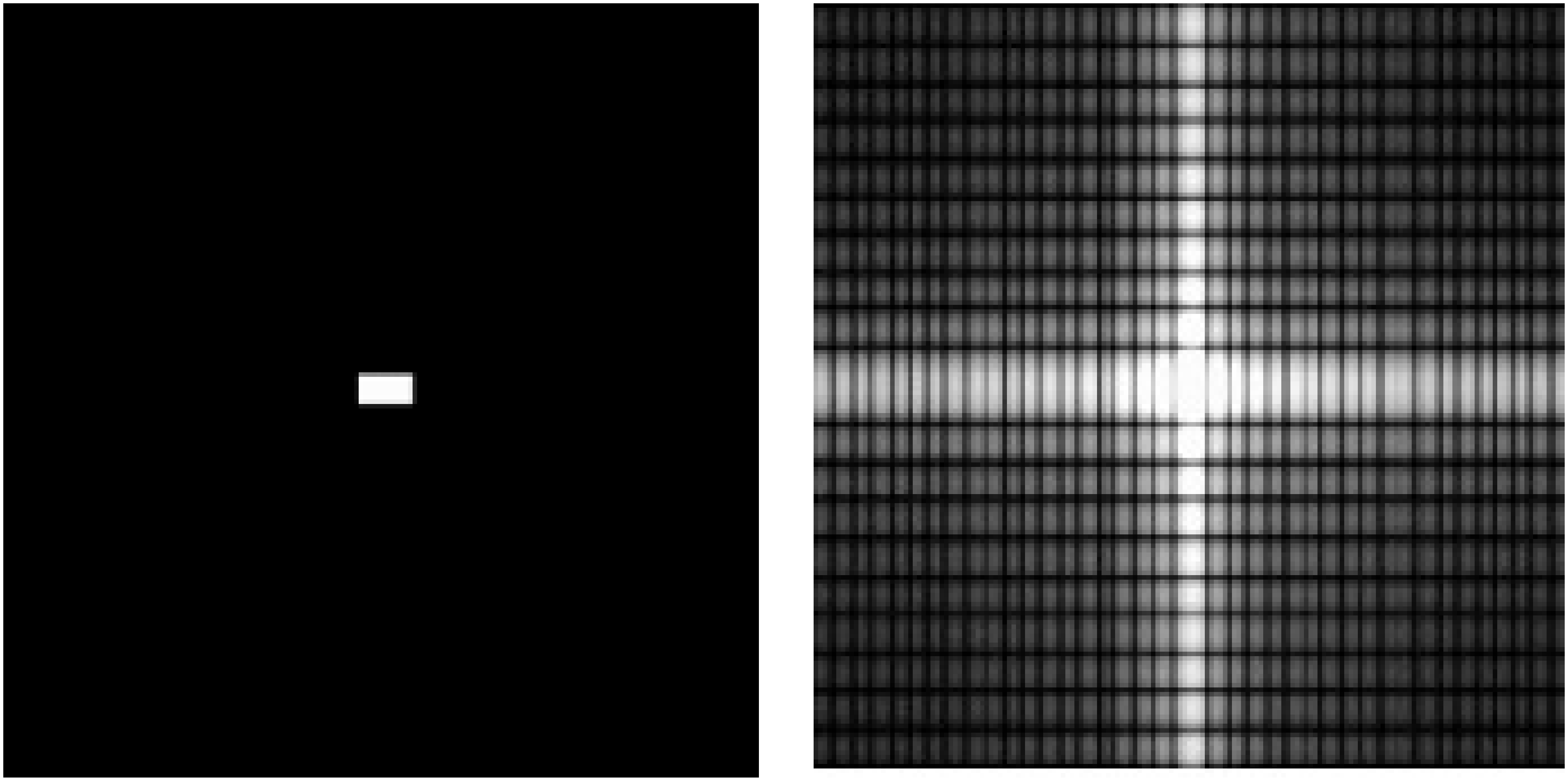
$$s = c\ log(+1 + r)$$



a b

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

Log Transformations (2)



Power-law Transformation

$$s = cr^\gamma$$

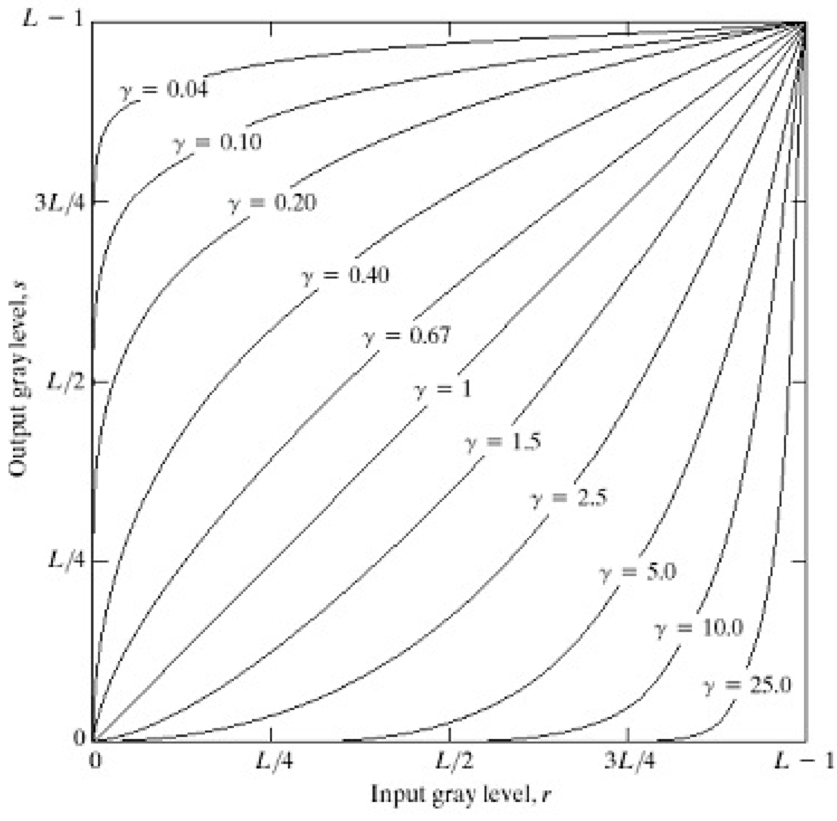
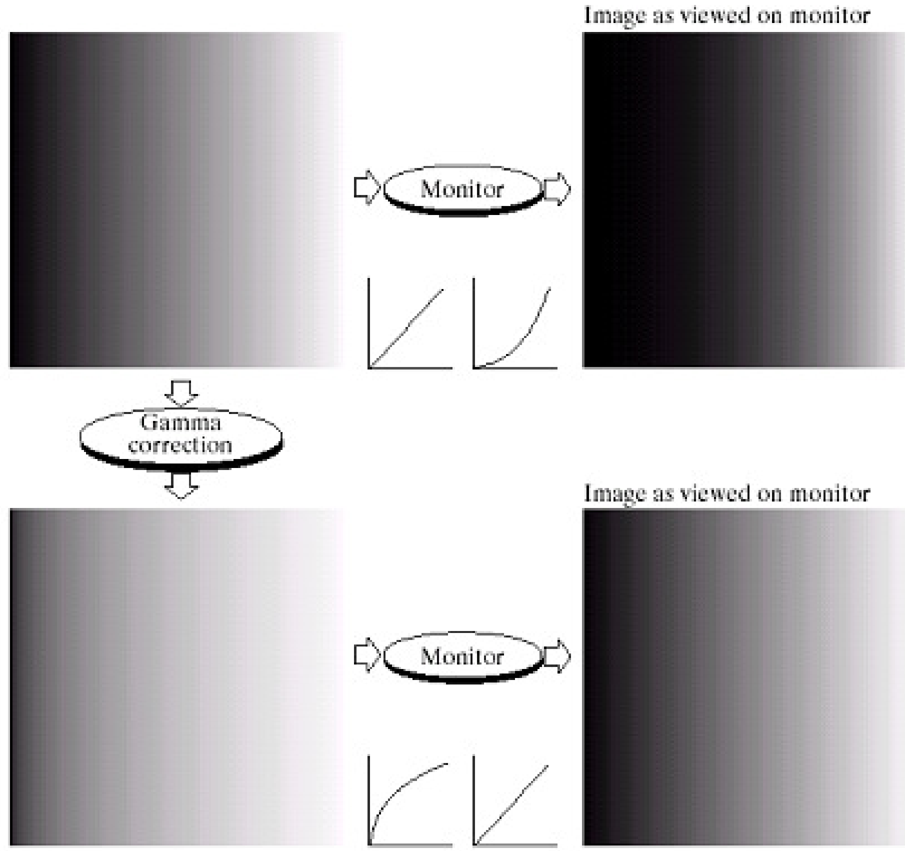


FIGURE 3.6 Plots of the equation $s = cr^\gamma$ for various values of γ ($c = 1$ in all cases).

CRT Monitor: Gamma Correction

- Gamma Correction: transformation to display an image accurately on a computer screen



a b
c d

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

Application of Power-law Transformation

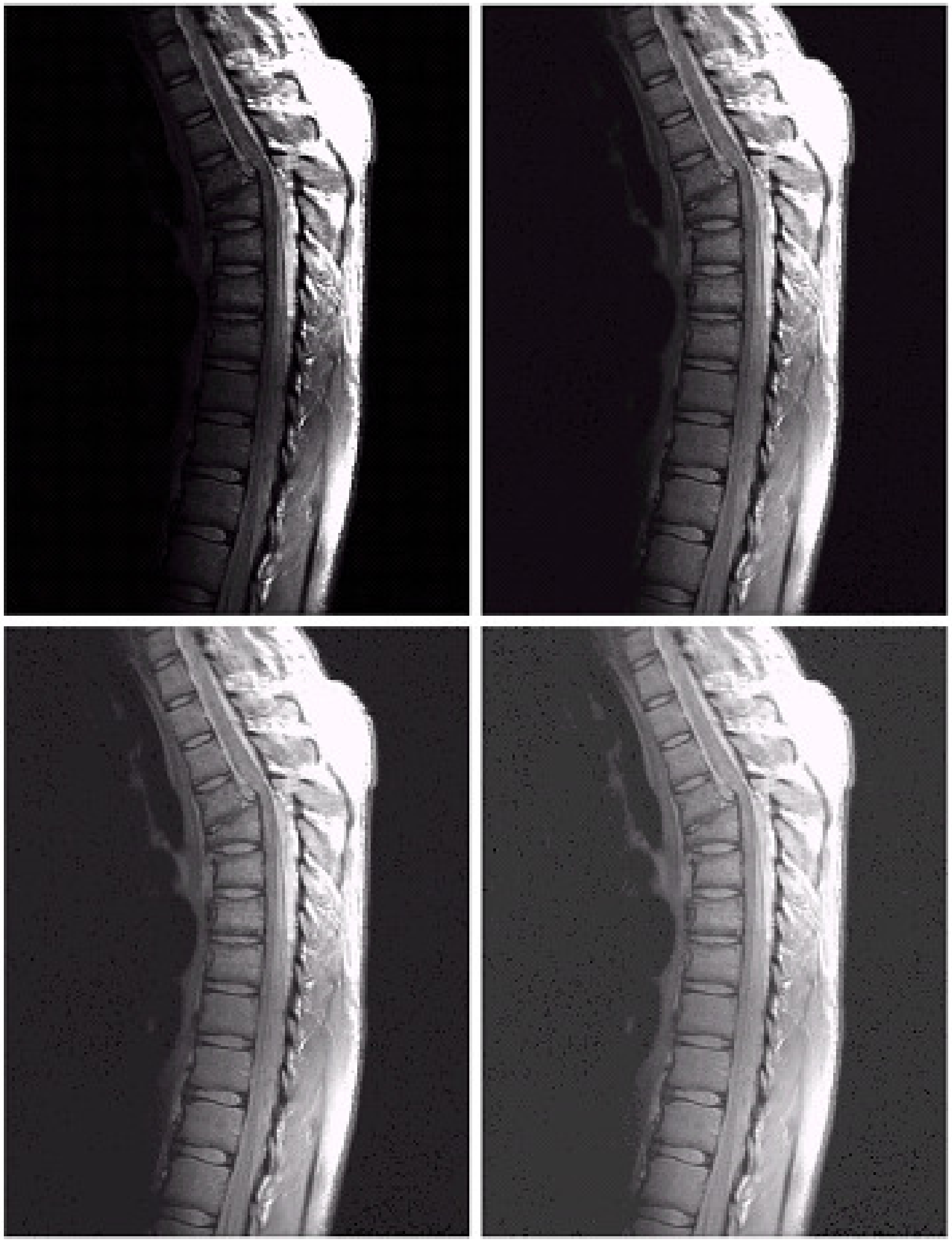
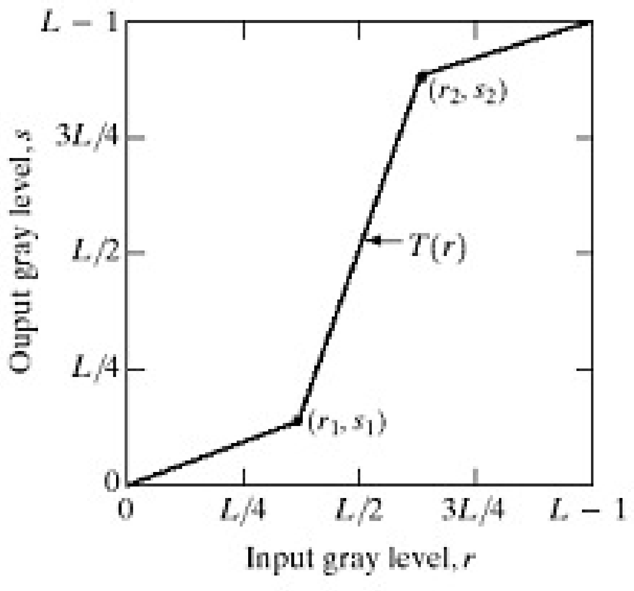
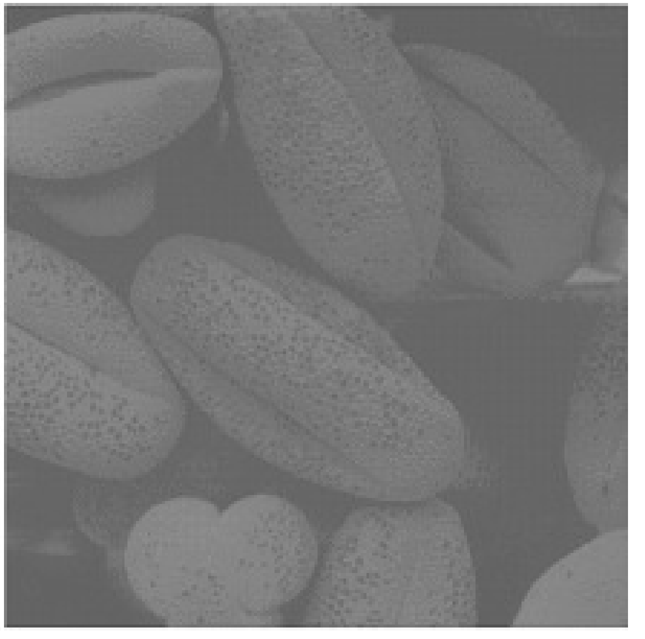


FIGURE 3.8 (a) Magnetic resonance (MR) image of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4$, and 0.3, respectively. (Original image for this example courtesy of Dr. David R. Pickens, Department of Radiology and Radiological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

Contrast Stretching

- Piecewise-linear transformations
  - gray level in the range $[0, L-1]$
  - (r_1, s_1) and (r_2, s_2) control the shape of the transformation

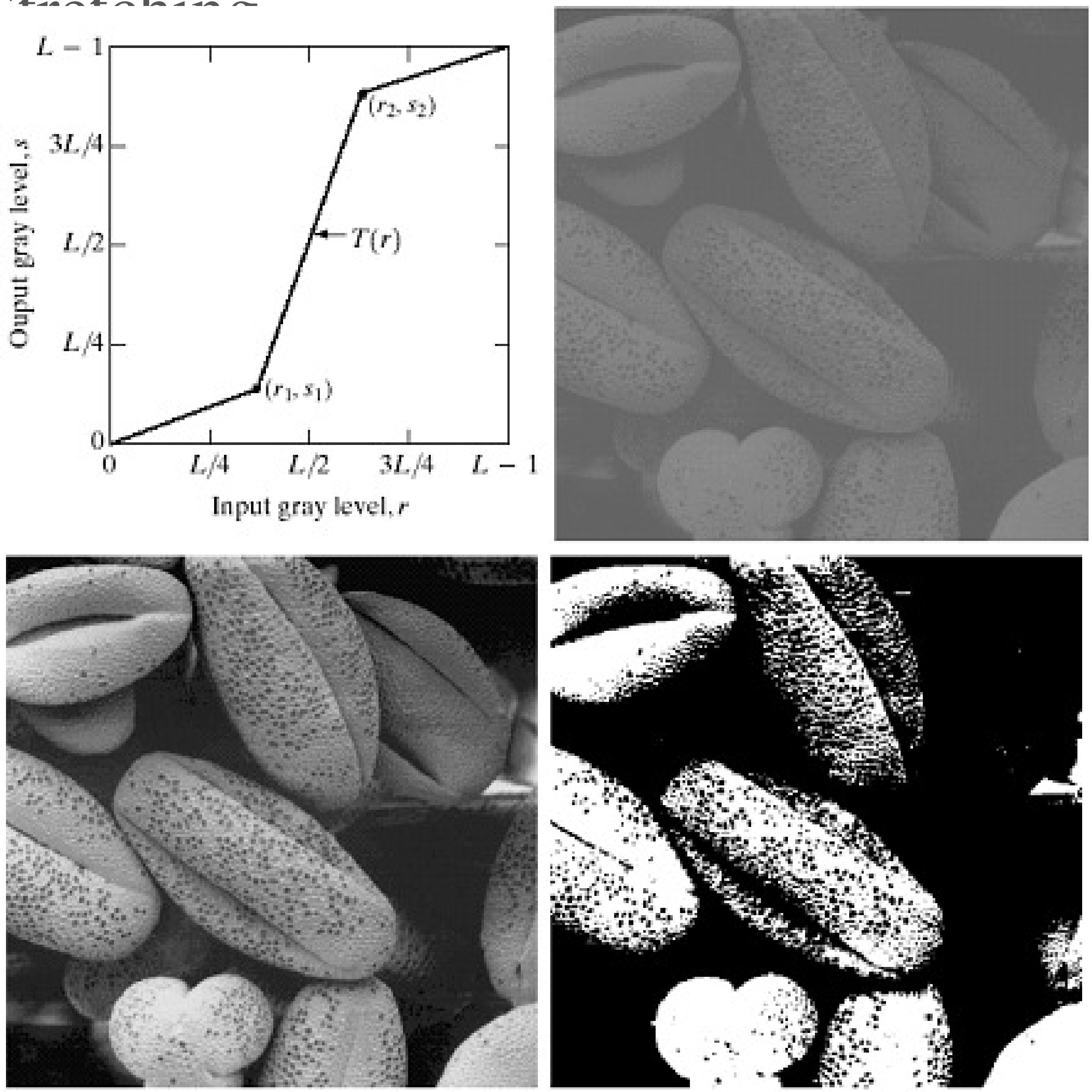




Contrast Stretching

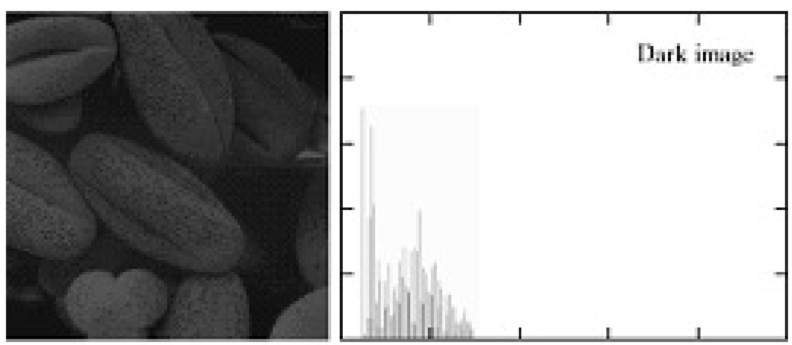
- Three cases:
  - $r_1 = s_1$ and $r_2 = s_2$: linear function
  - $r_1 = r_2$, $s_1 = 0$ and $s_2 = L - 1$: thresholding function (binary image)
  - intermediate value
- In general: $r_1 \leqslant r_2$, $s_1 \leqslant s_2$
- Stretch value linearly

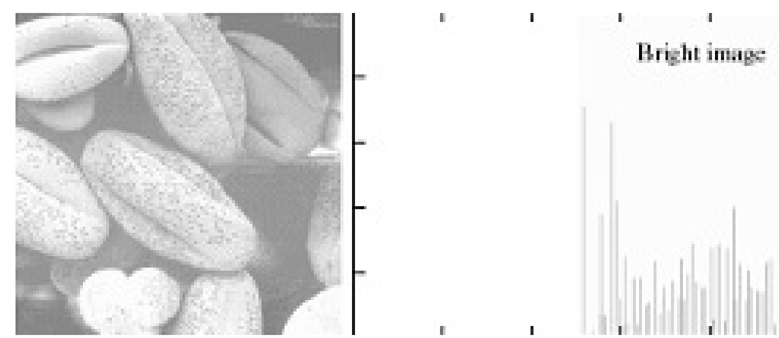
Contrast Stretching



Histogram

- Histogram: discrete function $h(r_k) = n_k$
  - r_k: k^{th} gray level
  - n_k: # pixels in the image having gray level r_k
  - gray level in the range $[0, L-1]$
  - n: total number of pixels





Histogram normalization

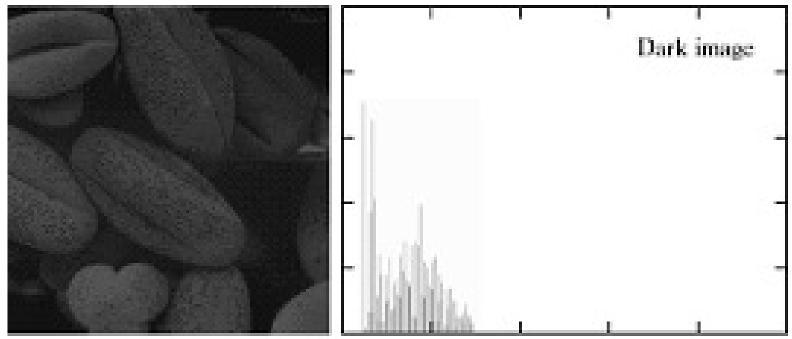
$$norm(r_k) = (r_k - r_{\min})\frac{V_{\max} - V_{\min}}{r_{\max} - r_{\min}} + V_{\min}$$

where

- $r_{\min}$ is the lowest gray-value
- $r_{\max}$ is the highest gray-value
- $V_{\min}$ is the new desired lowest gray value
- $V_{\max}$ is the new desired highest gray value

Histogram Equalization (1)

- we make all gray values in an image equally probable



- probability density function
  - $p_r(r)$: probability density function (PDF) of random variable r
  - $p_r(r)dr$: # pixels with gray level values in the range $[r, r + dr]$

Histogram Equalization (2)

- $s = T(r)$ with $0 \leqslant r \leqslant 1$
  - r has been normalized between 0 and 1
- assumptions:
  - $T(r)$ is single-valued
  - monotonically increasing
  - $0 \leqslant T(r) \leqslant 1$ for $0 \leqslant r \leqslant 1$
  - $r = T^{-1}(s)$ with $0 \leqslant s \leqslant 1$

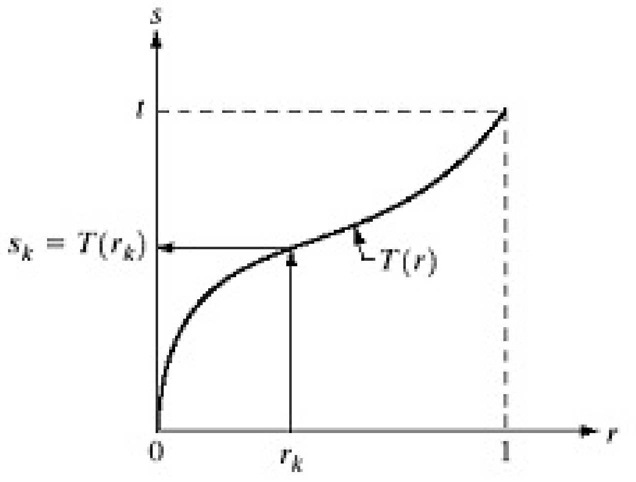


FIGURE 3.16 A gray-level transformation function that is both single valued and monotonically increasing.

Histogram Equalization (3)

- $p_r(r)$ and $p_s(s)$ PDF of two random variables
- $s = T(r)$ with $0 \leqslant r \leqslant 1$

$$p_r(r)\,dr = p_s(s)\,ds$$

We want $p_r(r)$ transformed into $p_s(s)$ to look like constant

Proove that $s = T(r) = \int_0^r p_r(w)dw$ does the job.

Histogram Equalization (4)

Hence

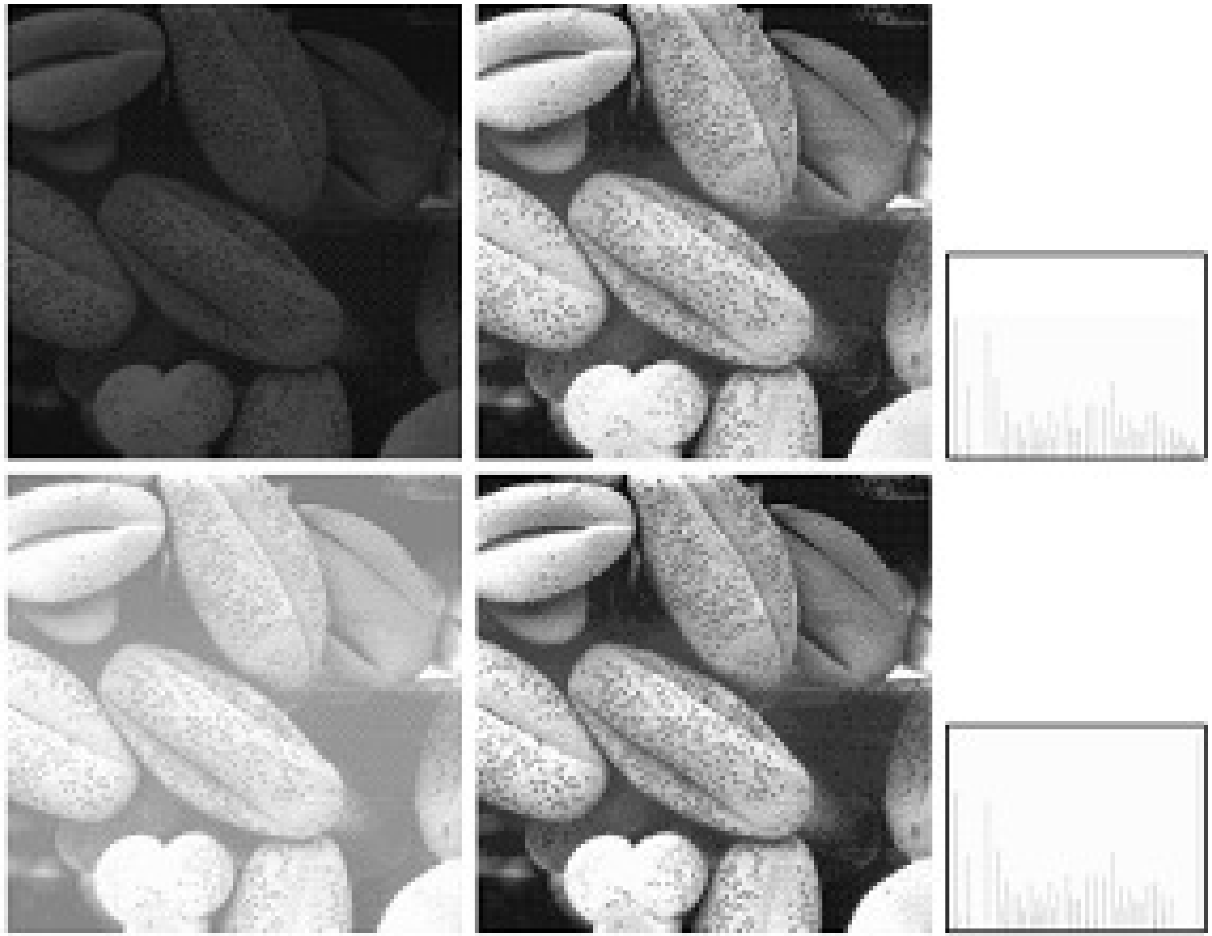
$$s = \int_0^r p_r(r)\, dr = T(r)$$

- we deal with image: discrete value
  - summation instead of integrals
  - probabilities instead of PDF
- probability of occurence of gray level r_k is:
  - $p_r(r_k) = \frac{n_k}{n}$ with $k = 0, 1, ..., L-1$

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

$$s_k = \sum_{j=0}^{k} \frac{n_j}{n} \quad \text{with} \quad k = 0, 1, ..., L-1$$

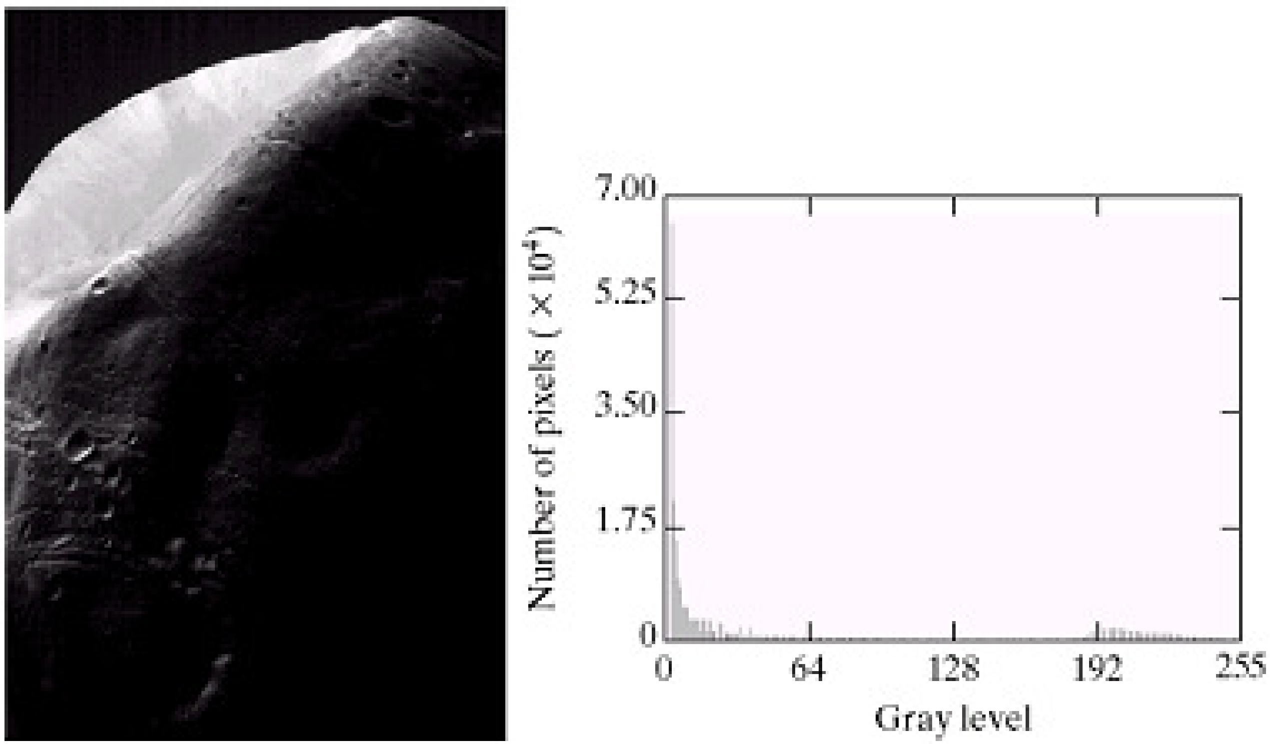
Histogram Equalization – Example (5)



Histogram Equalization (6)

- Why Histogram Equalization does not produce flat histograms ?

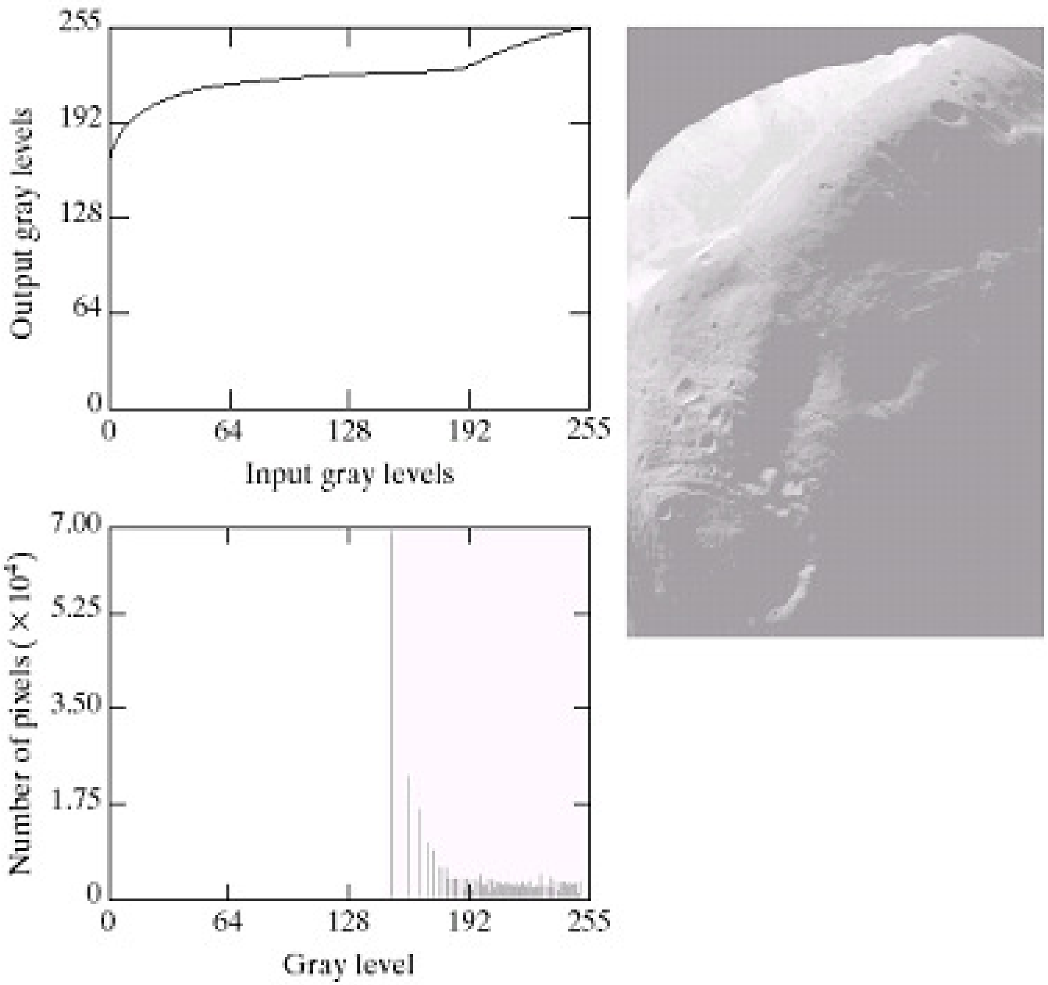
Histogram Equalization



a b

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

Histogram Equalization (8)

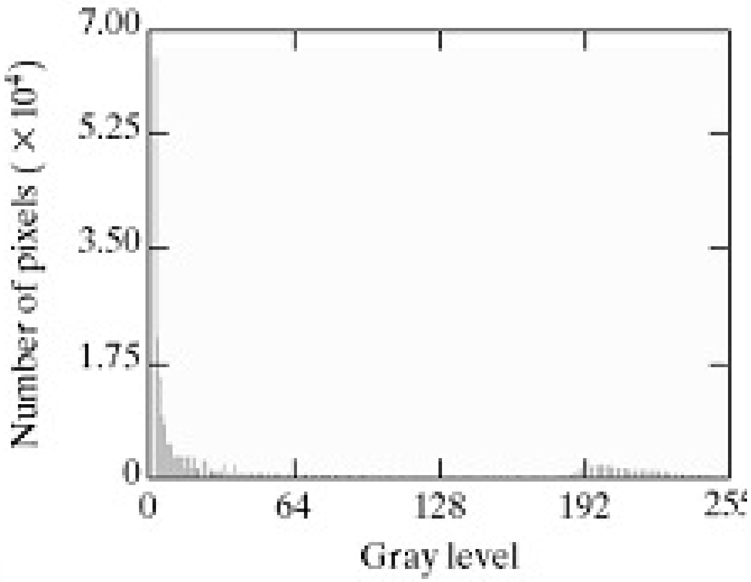


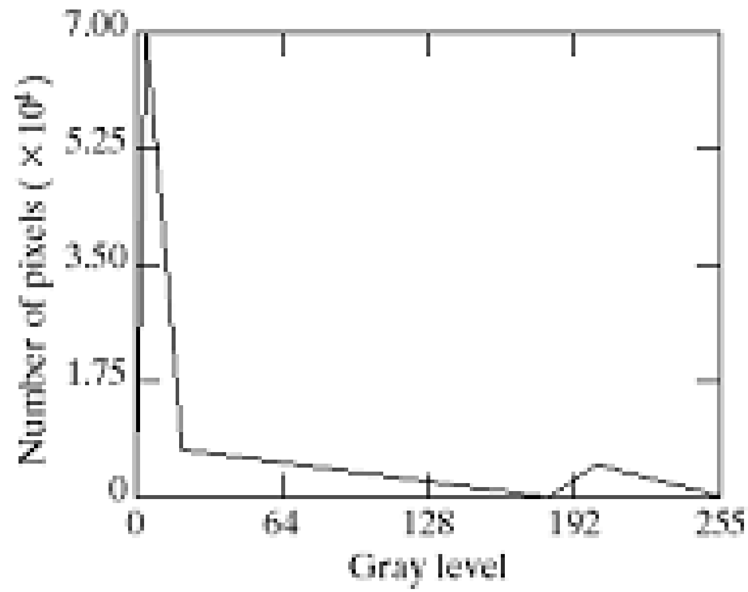
a b
c

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

Histogram Matching (1)

- we specify the shape of the histogram that we wish the processed image to have



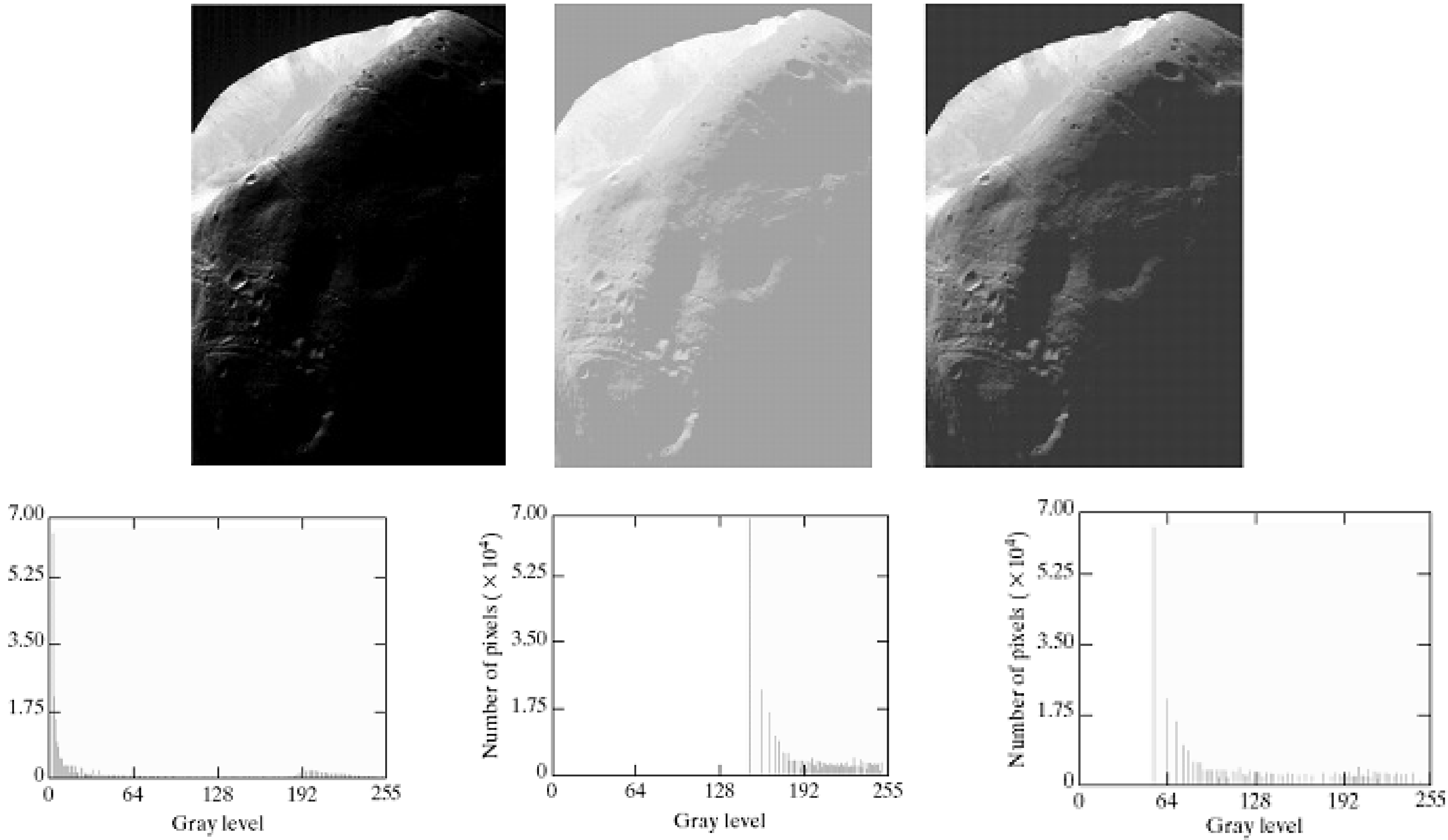


Histogram Matching (2)

- Let assume we want to have the PDF $p_z(z)$
  - $s = T(r) = \int_0^r p_r(w)\, dw$
  - $G(z) = \int_0^z p_r(t)\, dt = s$
- $G(z) = T(r)$

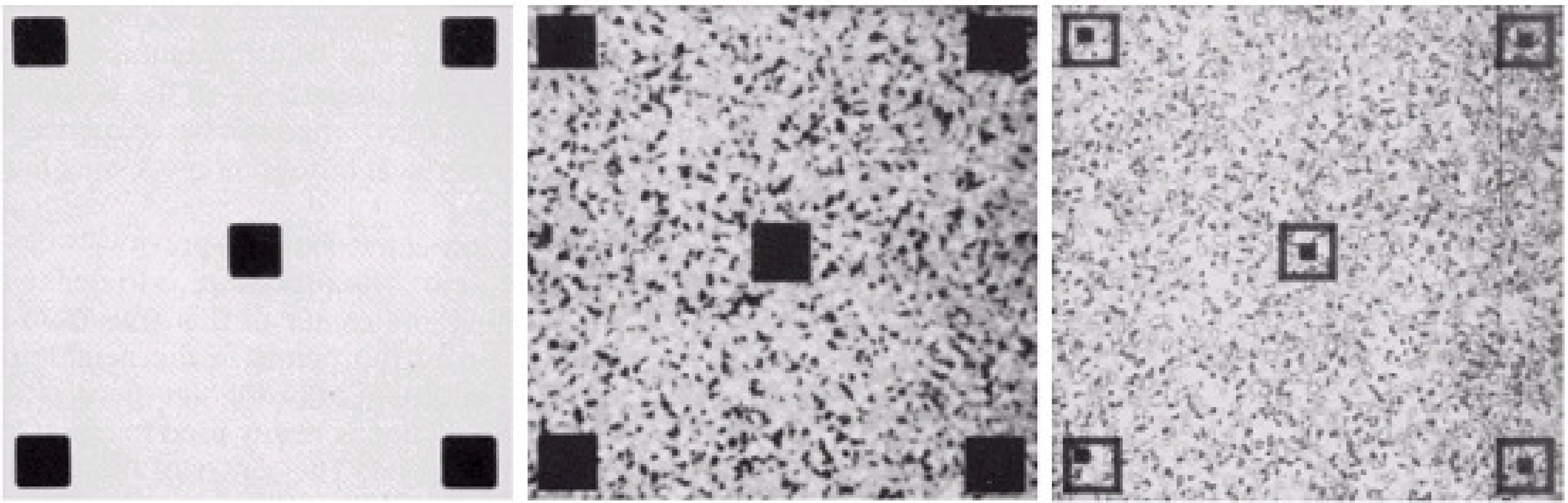
  (1) $$z = G^{-1}(s) = G^{-1}[T(r)]$$
- Algorithm:
  - (1) Obtain $T(r)$
  - (2) Compute $G(z)$
  - (3) Compute $G^{-1}(z)$
  - (4) Obtain output image by applying eq. (1)

Histogram Matching (3)



Local Enhancement

- local histogram equalization
  - using a $N \times N$ masks
  - applying the equalization only to the pixel at the center of the mask
  - repeat the process to all the pixel (convolution)



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.

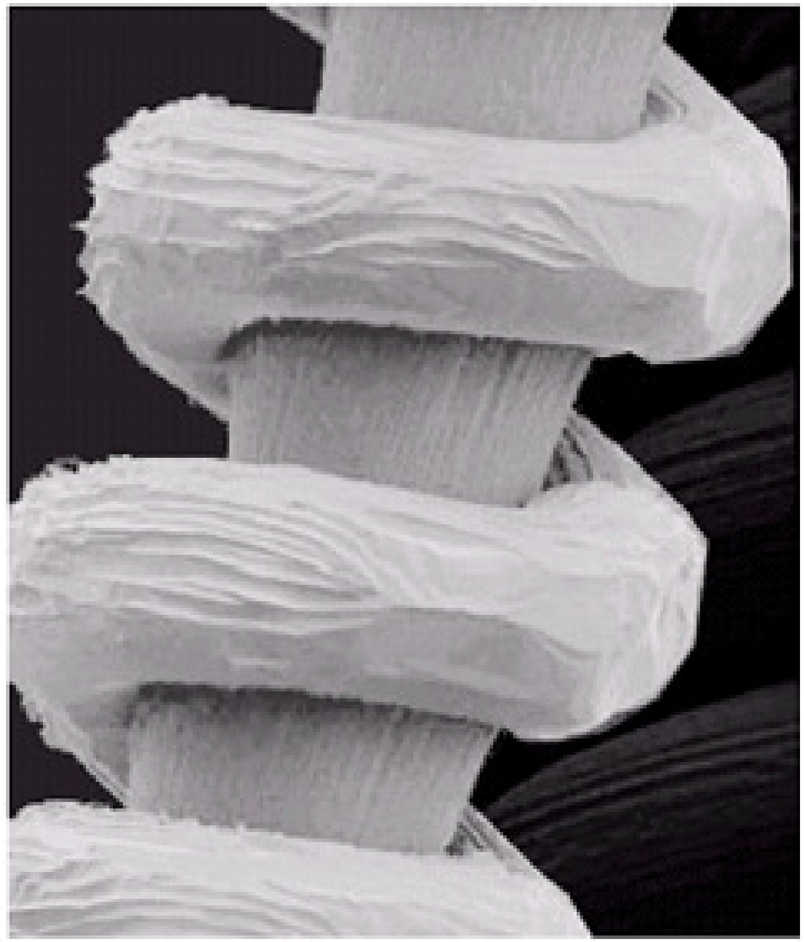
Using Histogram Statistics

- we use some statistical parameters
  - global:
    - $p(r_i) = \frac{n_i}{n}$
    - $m(r) = \sum_{i=0}^{L-1} p(r_i)\, r_i$
    - $\sigma^2(r) = \sum_{i=0}^{L-1} (r_i - m)^2\, p(r_i)$
  - local:
    - $p(r_{s,t})$: neighborhood normalized histogram at coordinates (s,t) using a mask centered at (x,y)
    - $m_{S_{xy}} = \sum_{(s,t)\in S_{xy}} p(r_{s,t})\, r_{s,t}$
    - $\sigma^2(S_{xy}) = \sum_{(s,t)\in S_{xy}} [r_{s,t} - m_{S_{xy}}]^2\, p(r_{s,t})$

Local Statistics – Example (1)

- How to enhance this image?

FIGURE 3.24 SEM image of a tungsten filament and support, magnified approximately 130×. (Original image courtesy of Mr. Michael Shaffer, Department of Geological Sciences, University of Oregon, Eugene).



Local Statistics – Example (2)

Image Analysis!

- What do we want to achieve?
  - We want to enhance dark areas while leaving light areas unchanged
- Can we use local statistic to obtain it?
  - where the image is dark: local mean $\ll$ global mean
  - enhance area with only low contrast: local standard deviation $\ll$ global standard deviation
  - avoiding to enhance constant areas: local standard deviation higher than a fixed minimum value

Local Statistics – Example (3)

Mathematical translation

- $g(x, y) = E.f(x, y)$
  - if $m_{S_{xy}} \leqslant k_0\, M_G$
  - and $k_1\, D_G \leqslant \sigma_{S_{xy}} \leqslant k_2\, D_G$
- $g(x, y) = f(x, y)$ otherwise
- E_0, k_0, k_1, k_2: specified parameters
- M_G: global mean of the input image
- D_G: global standard deviation

Local Statistics – Example (4)

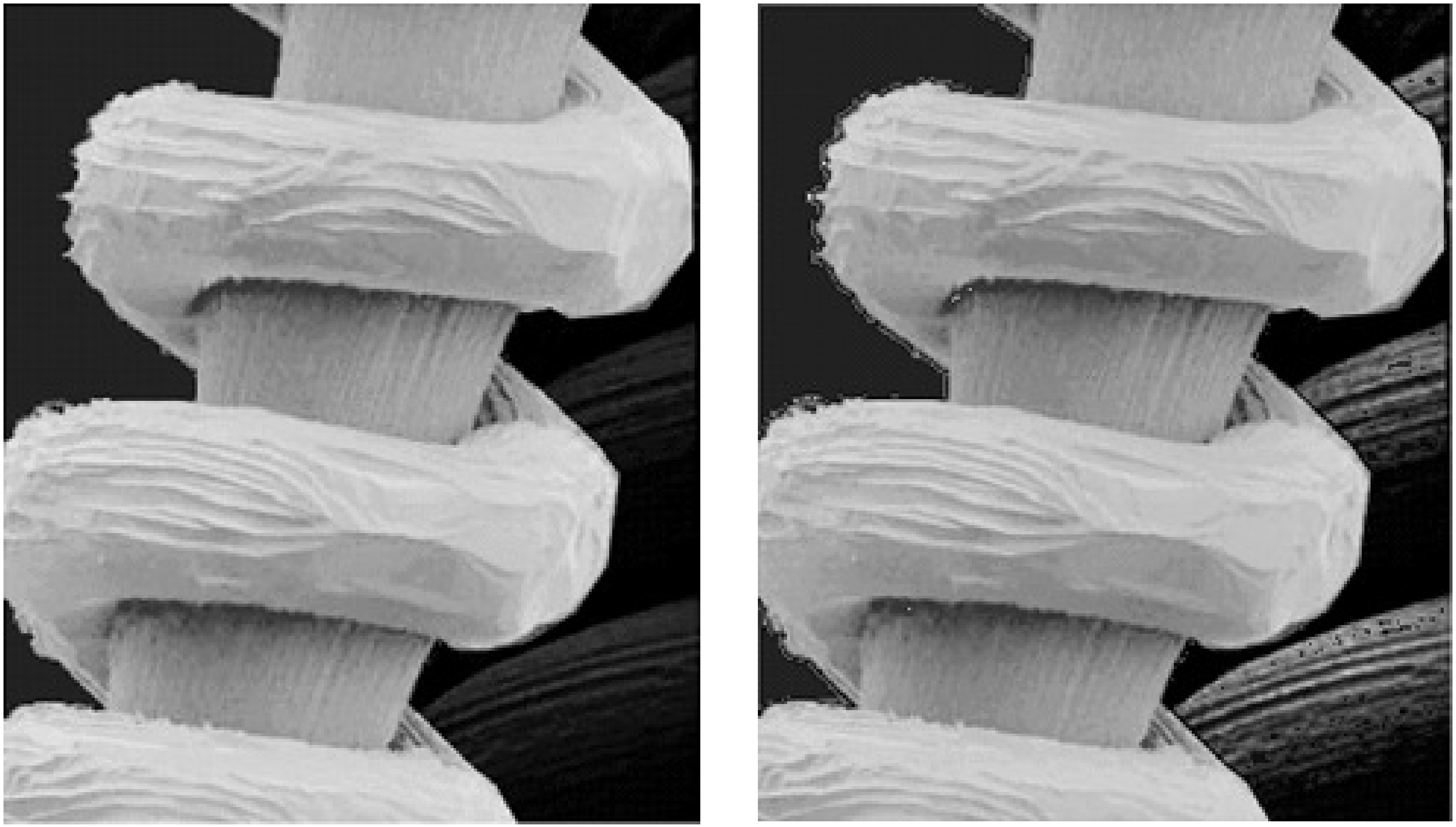


Image Substraction

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

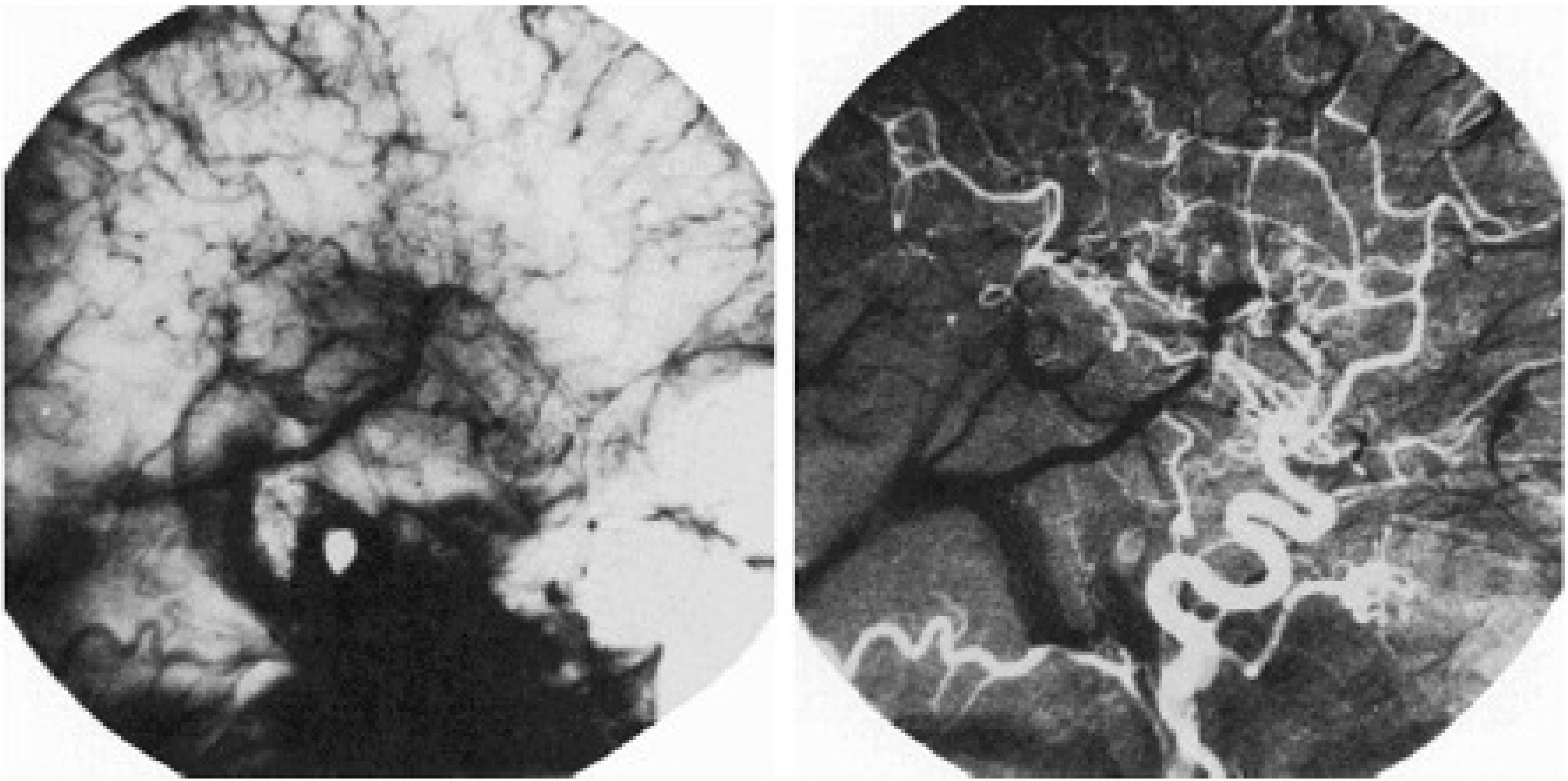


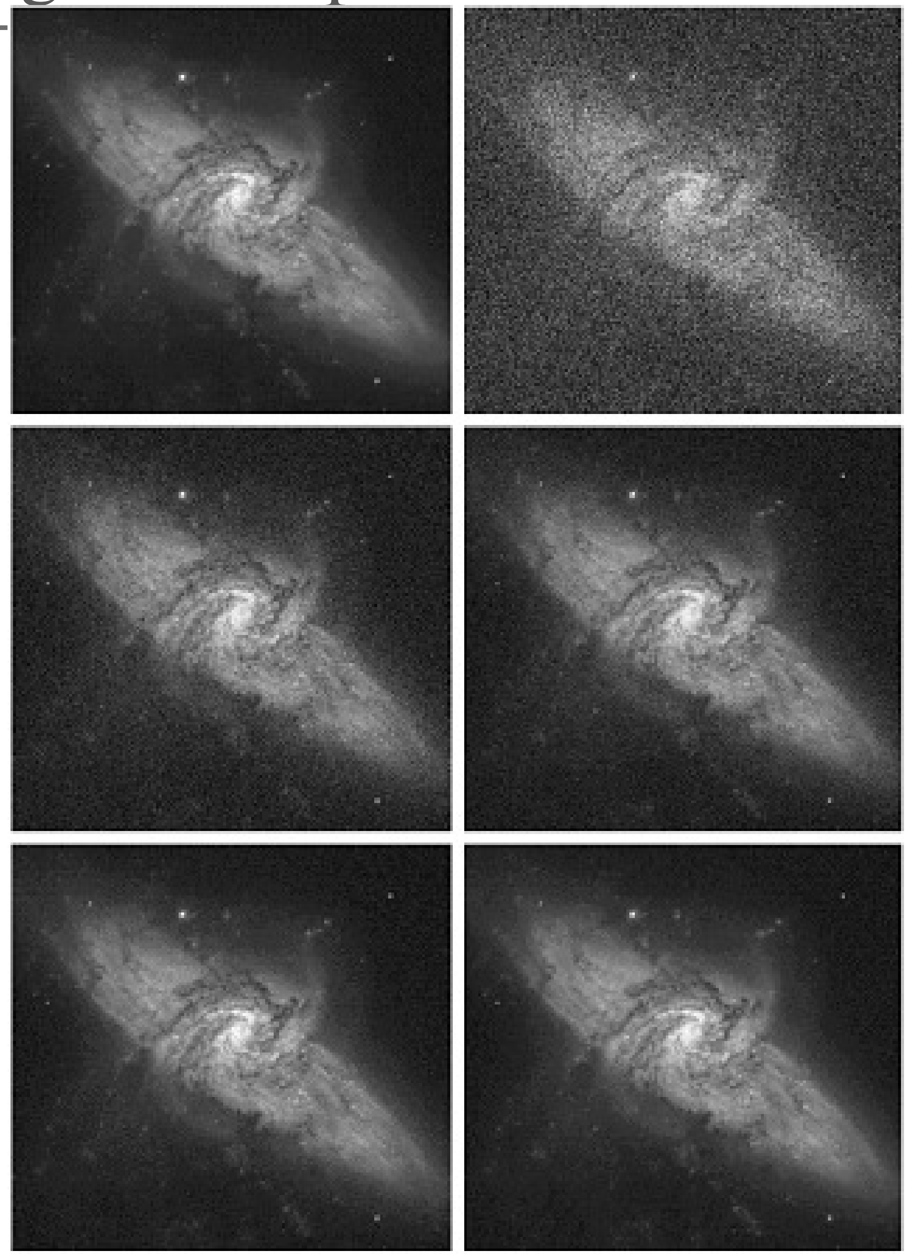
Image Averaging

- $g(x, y) = f(x, y) + \eta(x, y)$
  - $g(x, y)$: noisy image
  - $f(x, y)$: original image
  - $\eta(x, y)$: uncorrelated noise with zero average value
- We reduce the noise content by adding a set of noisy images $g_i x, y$
- $\overline{g}(x, y)$ is formed by:
  - $\overline{g}(x, y) = \frac{1}{K} \sum_{i=1}^{K} g_i(x, y)$

In theory:

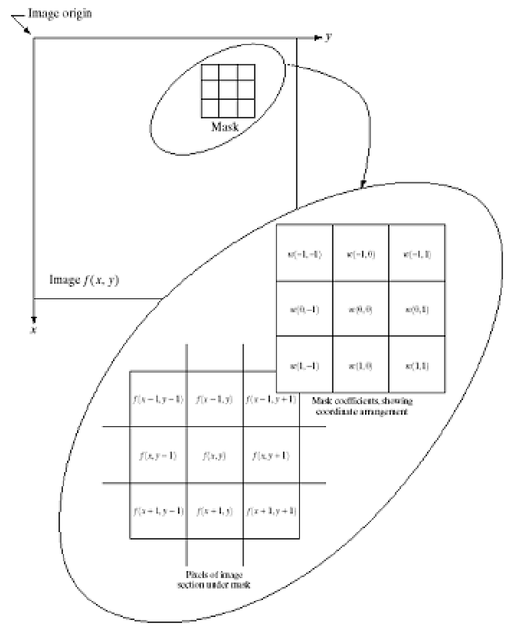
$$\hat{\overline{g}}(x, y) = f(x, y)$$

Image Averaging – Example



Spatial Filtering

- Filtering operations performed directly on the pixels of an image

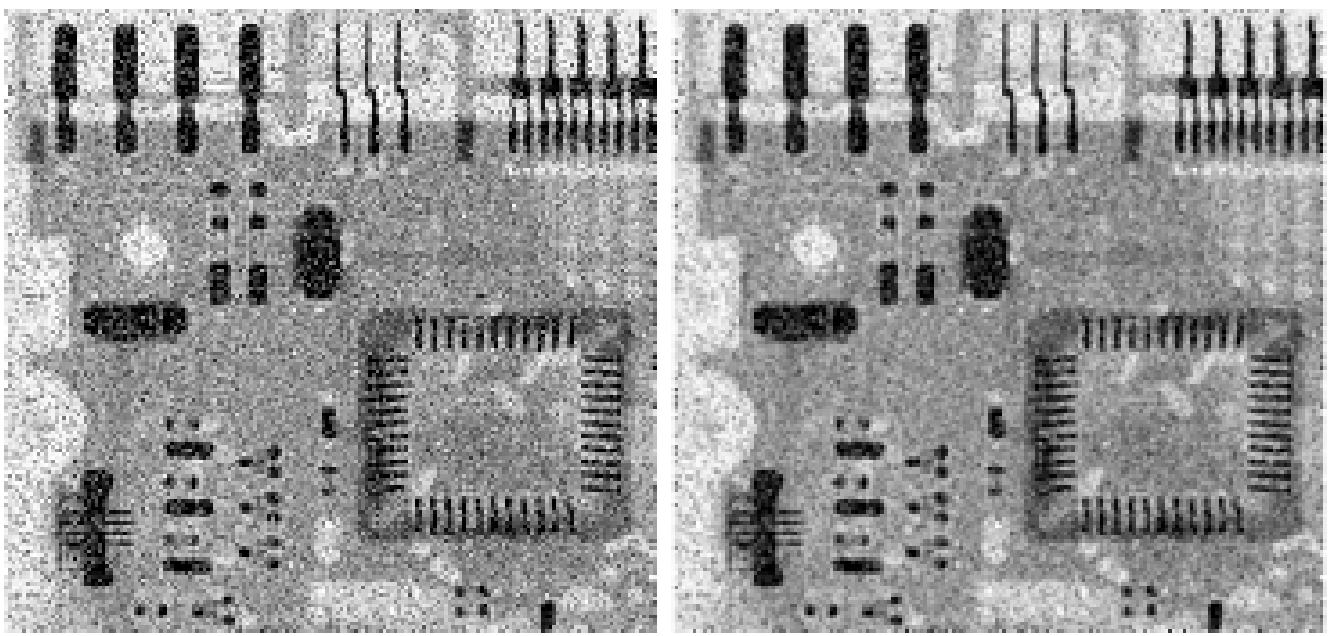


$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)\, f(x+s, y+t)$$

Weighted Averaging Filter

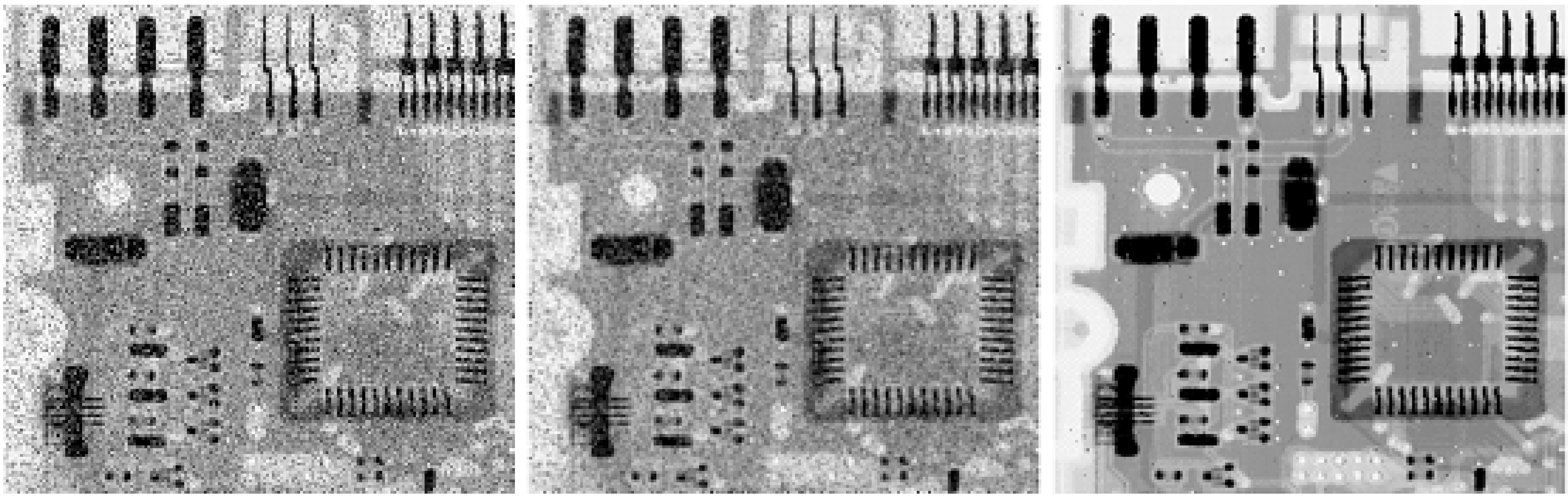
$$g(x,y) = \frac{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)\, f(x+s, y+t)}{\sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)}$$

- Example: image corrupted by a salt-and-pepper noise



Median Filter

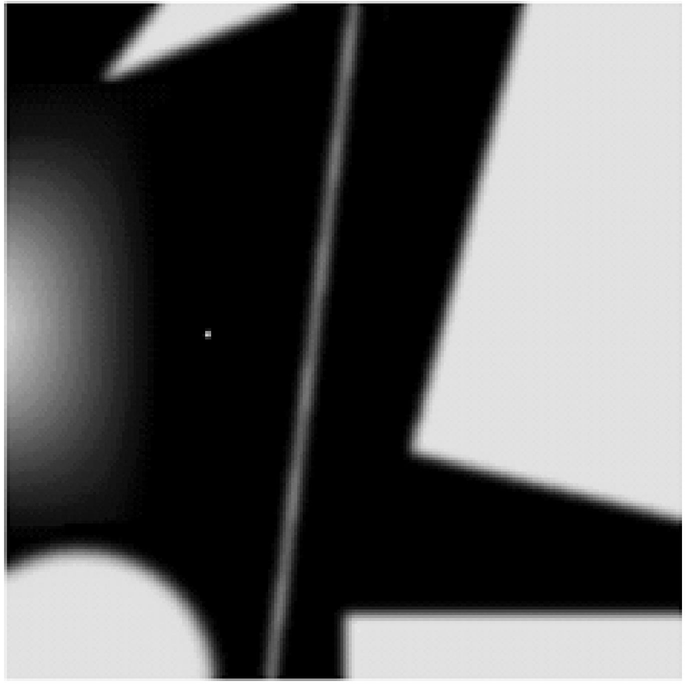
- $\hat{f}(x, y) = \text{median}_{(s,t) \in S_{xy}}(g(s, t))$

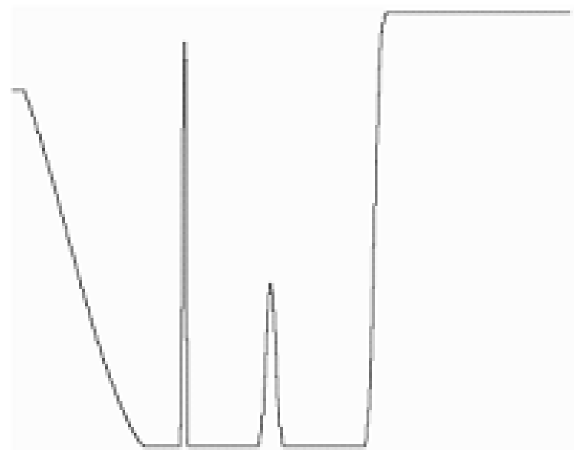


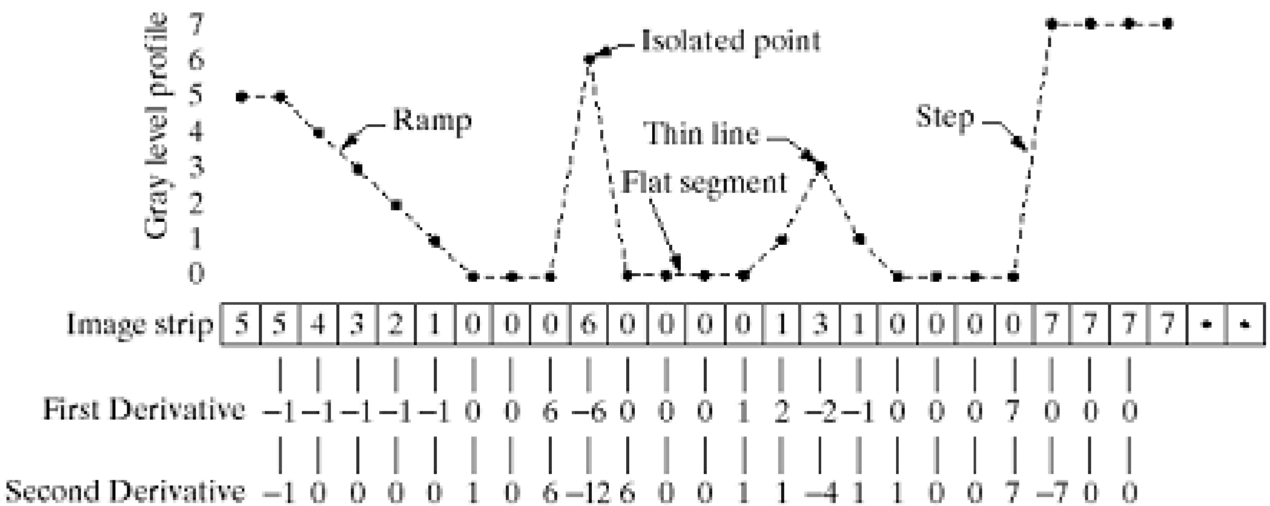
Sharpening Spatial Filters

- Sharpening can be achieved by spatial differention
- Derivative operators:
  - first-order derivative
  - second-order derivative
- emhance edges (and also noise...)
- deemphasize image areas with slow intensity variations
- first-order derivative:
  - $\frac{\partial f}{\partial x} = f(x+1) - f(x)$
- second-order derivative:
  - $\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$

First- and Second-order Derivatives (1)







First- and Second-order Derivatives (2)

- First-order derivatives produce thick edges
- Second-order derivatives have a stronger response to fine detail
- First-order derivatives have a stronger response to a gray-level step
- Second-order derivatives produce a double response at step changes in gray level
- In general, second-order derivatives better suit for enhancement

First derivatives for Enhancement (1)

- The gradient:

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

- The gradient magnitude:

$$\begin{aligned} \nabla \mathbf{f} \quad &= [G_x^2 + G_y^2]^{\frac{1}{2}} \\ &= [(\tfrac{\partial f}{\partial x})^2 + (\tfrac{\partial f}{\partial y})^2]^{\frac{1}{2}} \end{aligned}$$

- Approximation:

$$\nabla \mathbf{f} \approx |G_x + G_y|$$

First derivatives for Enhancement (2)

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

- simplest approximation:
  - $G_x = (z_8 - z_5)$
  - $G_y = (z_6 - z_5)$
- cross difference:
  - $G_x = (z_9 - z_5)$
  - $G_y = (z_8 - z_6)$

Then:

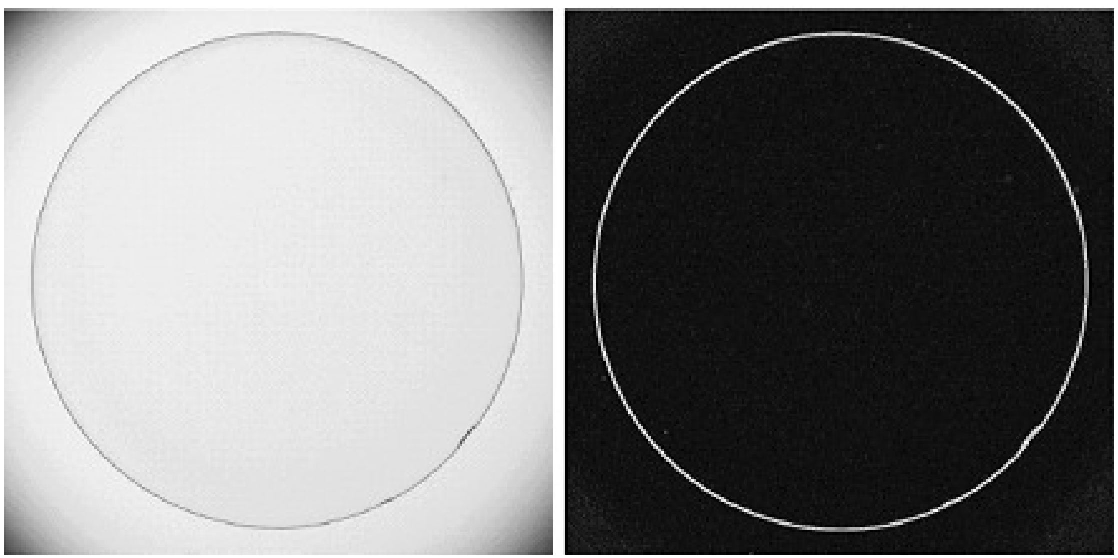
$$\nabla \mathbf{f} = |z_9 - z_5| + |z_8 - z_6|$$

Sobel Operator

$$\nabla \mathbf{f} \approx \begin{aligned} & |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| \\ & +|(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)| \end{aligned}$$

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1



Laplacian

- $\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$
  - $\frac{\partial^2 f}{\partial x^2} = f(x+1, y) + f(x-1, y) - 2f(x, y)$
  - $\frac{\partial^2 f}{\partial y^2} = f(x, y+1) + f(x, y-1) - 2f(x, y)$
- $\nabla^2 f \;=\; f(x+1, y) + f(x-1, y) + f(x, y+1)$
  $+ f(x, y-1) - 4f(x, y)$

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

Laplacian for Image Enhancement (1)

- highlights gray level discontinuities
- deeamphasizes regions with slowy varying gray level

Laplacian for Image Enhancement (2)

$$g(x,y) = \begin{cases} f(x,y) - \nabla^2 f(x,y) & \text{if the center coefficient is negative} \\ f(x,y) + \nabla^2 f(x,y) & \text{if the center coefficient is positive} \end{cases}$$

Mask Composition (1)

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

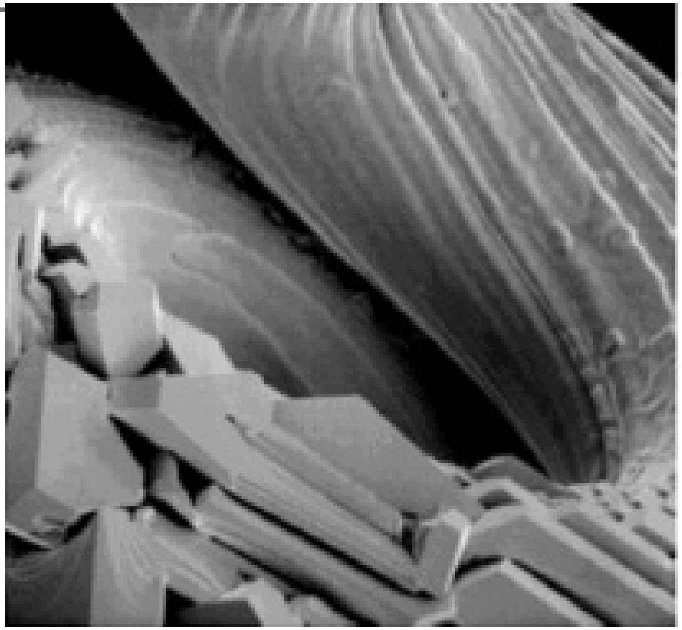
$$g(x,y)=f(x,y)-[f(x+1,y)+f(x-1,y)+f(x,y+1)+f(x,y-1)]+4f(x,y)$$

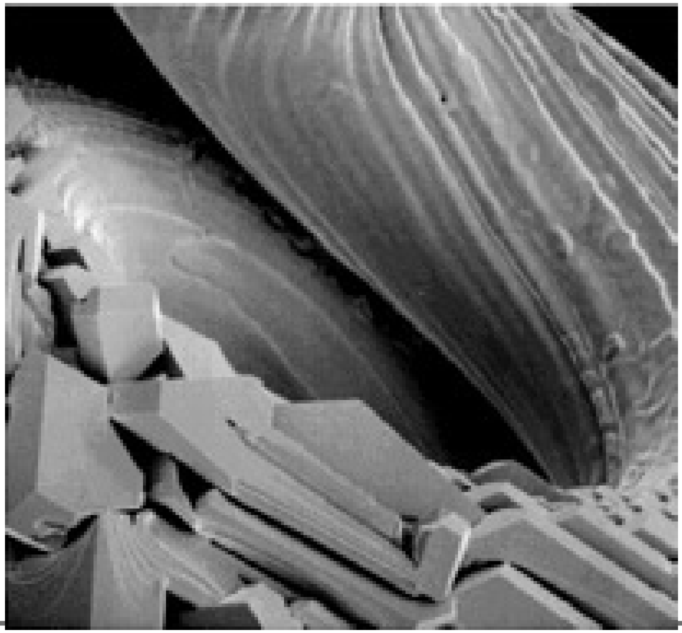
$$g(x,y)=5f(x,y)-[f(x+1,y)+f(x-1,y)+f(x,y+1)+f(x,y-1)]$$

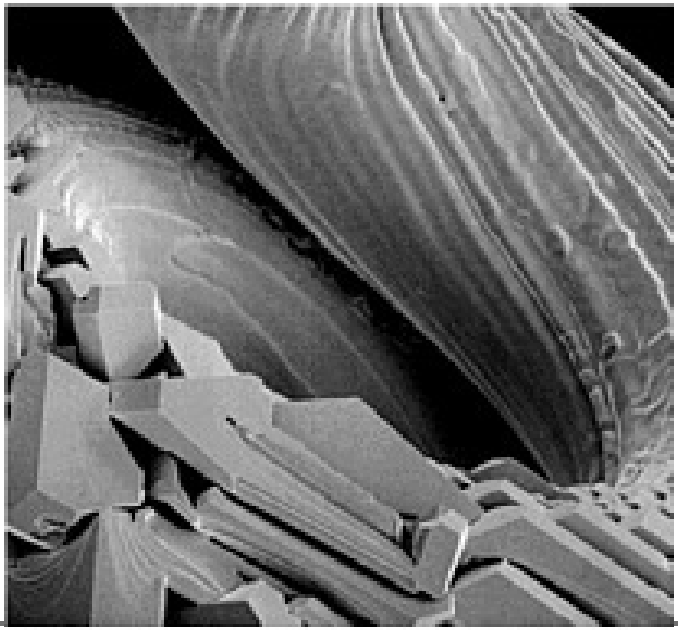
0	−1	0
−1	5	−1
0	−1	0

−1	−1	−1
−1	9	−1
−1	−1	−1

Mask Composition (2)







Unsharp Masking

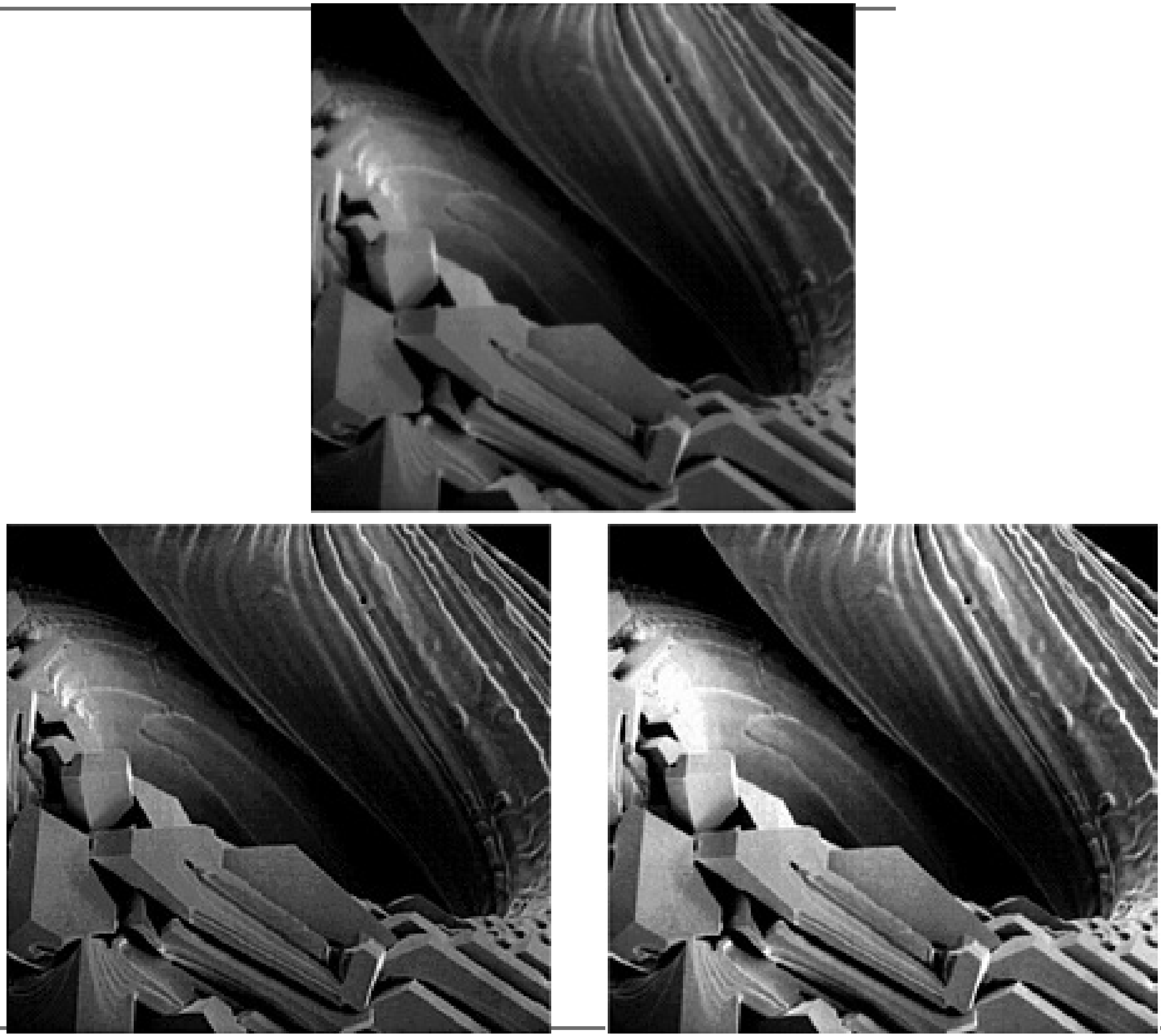
- Substracting a blurred version of an image from the image itself:
  - $f_s(x, y) = f(x, y) - \tilde{f}(x, y)$
  - $f_s(x, y)$: sharpened image
  - $\tilde{f}(x, y)$: blurred version of $f(x, y)$
- High-boost filtering:
  - $f_{hb}(x, y) = A\, f(x, y) - \tilde{f}(x, y)$
  - $f_h b(x, y)$: high-boosted image
  - $A \geq 1$

High-boost Filtering Using Laplacian (1)

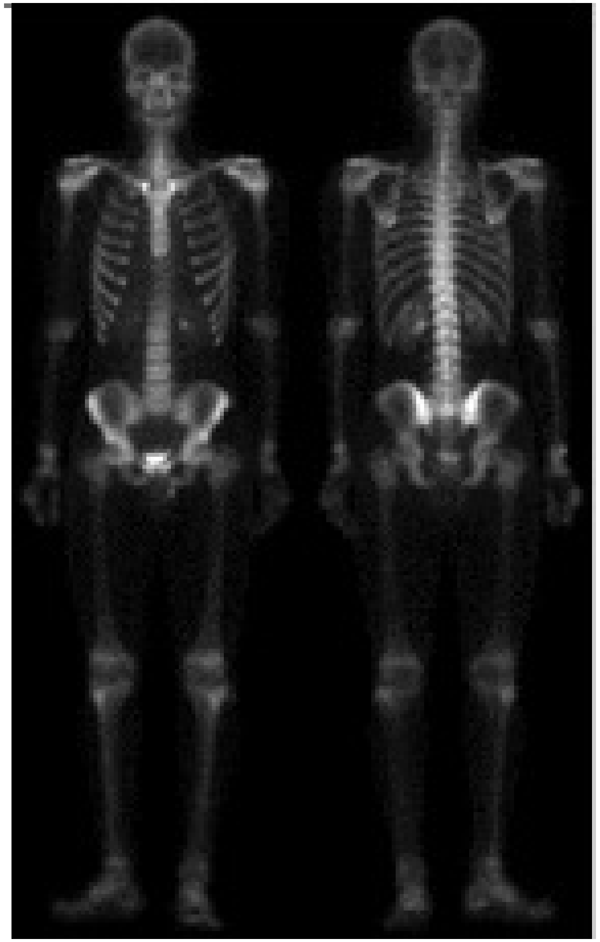
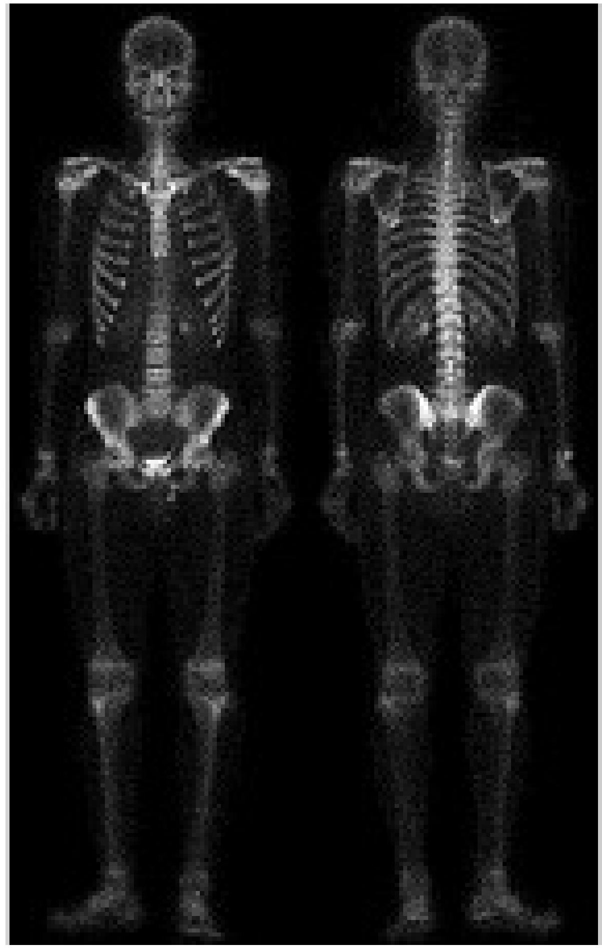
- Combining the two equations:
  - $f_{hb}(x, y) = (A\,f(x, y) - 1)f(x, y) + f(x, y) - \tilde{f}(x, y)$
  - $f_{hb}(x, y) = (A\,f(x, y) - 1)f(x, y) + f_s(x, y)$
  - Let say $f_s(x, y) = \nabla^2 f$ or $f_s(x, y) = -\nabla^2 f$
- Then:

$$f_{hb}(x, y) = \begin{cases} A\,f(x, y) - \nabla^2 f(x, y) \text{if center coefficient negative} \\ A\,f(x, y) + \nabla^2 f(x, y) \text{if center coefficient positive} \end{cases}$$

High-boost Filtering Using Laplacian (2)



Example

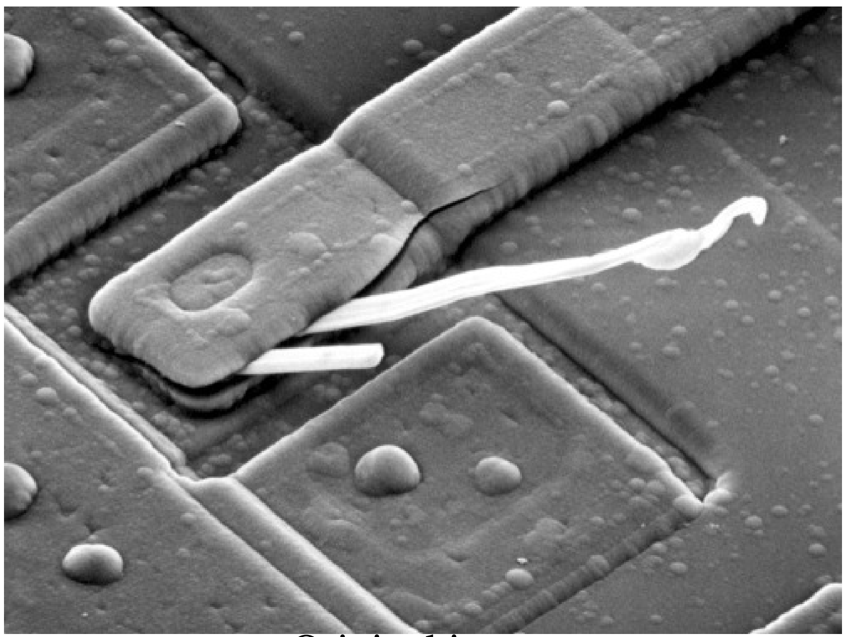




Spatial Domain Techniques: Summary

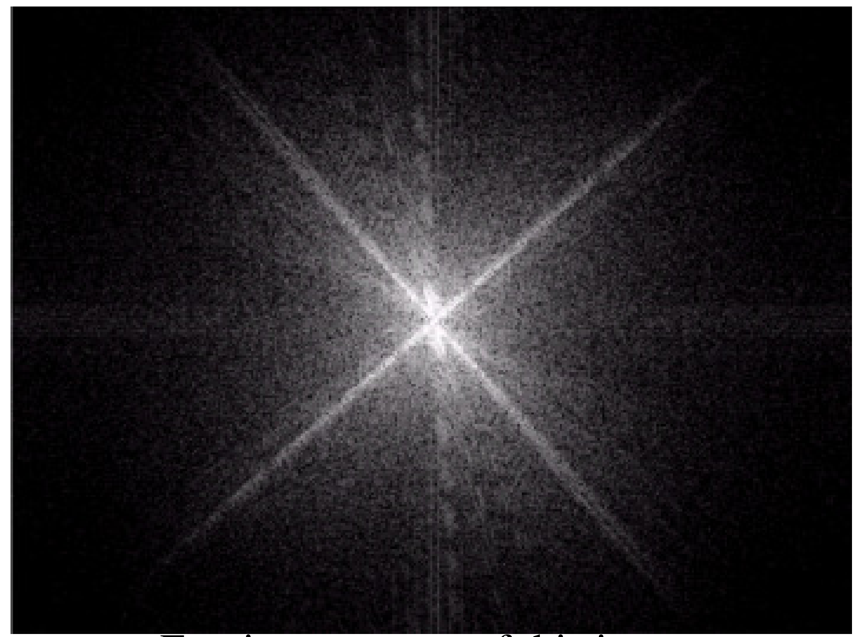
- Histogram equalization
- Histogram manipulation
- Basic statistics for image processing
- Filtering with spatial masks
- First-order and second-order derivatives
- Sobel filter

Frequency Domain – Background

- Jean Baptiste Joseph Fourier is the key!
- Frequency domain: space defined by values of the Fourier transform and its frequency variable (u, v)



Original image



Fourier spectrum of this image

Frequency Domain Filtering Operation

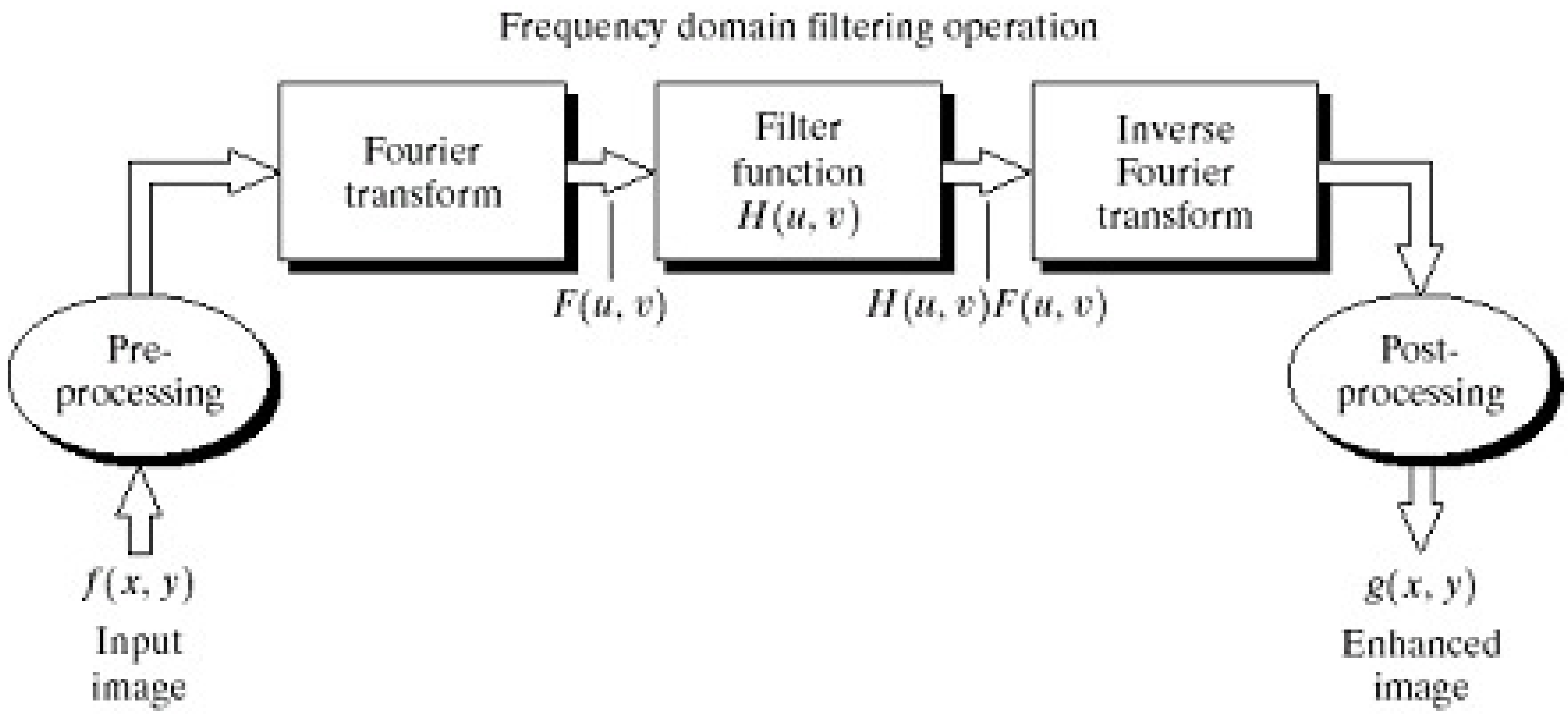


FIGURE 4.5 Basic steps for filtering in the frequency domain.