IVM

Image, Video, and Multimedia Communications Laboratory

Digital Image Processing

Hongkai Xiong

熊红凯

http://ivm.sjtu.edu.cn

电子工程系

上海交通大学

Mar. 2016

Today

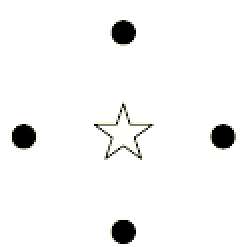
- Some Basic Relationships Between Pixels
- Intensity Transformation
- Spatial Filtering

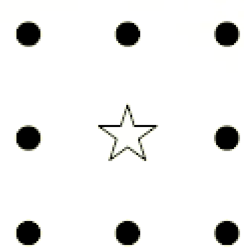
Today

- Some Basic Relationships Between Pixels
- Intensity Transformation
- Spatial Filtering

Some Basic Relationships Between Pixels

- Neighbors of a Pixel
  - A pixel p at coordinates (x, y) has four *horizontal* and *vertical* neighbors whose coordinates are given by
  $$(x+1,y),(x-1,y),(x,y+1),(x,y-1)$$
  This set of pixels, called the *4-neighbors* of p, is denoted by $N_4(p)$.
  - The four diagonal neighbors of p has coordinates
  $$(x+1,y+1),(x+1,y-1),(x-1,y+1),(x-1,y-1)$$
  and are denoted by $N_d(p)$. These points, together with the *4-neighbors*, are called the *8-neighbors* of p, denoted by $N_8(p)$.





Some Basic Relationships Between Pixels

- Adjacency, Connectivity, Regions, and Boundaries
  - Let V be the set of gray-level values used to define adjacency
    - In a binary image, V={1}
    - In a gray-scale image, V could be any subset of possible values 0 to 255.
  - (a) *4-adjacency*. Two pixels p and q with values from V are *4-adjacent* if q is in the set $N_4(p)$.
  - (b) *8-adjacency*. Two pixels p and q with values from V are *8-adjacent* if q is in the set $N_8(p)$.
  - (c) *m-adjacency*. Two pixels p and q with values from V are *m-adjacent* if
    - (i) q is in $N_4(p)$, or
    - (ii) q is in $N_D(p)$ and the set $N_4(p) \cap N_4(q)$ has no pixels whose values are from V

Some Basic Relationships Between Pixels

- Adjacency, Connectivity, Regions, and Boundaries
  - Mixed adjacency is a modification of *8-adjacency*. It is introduced to eliminate the ambiguities that often arise when *8-adjacency* is used.

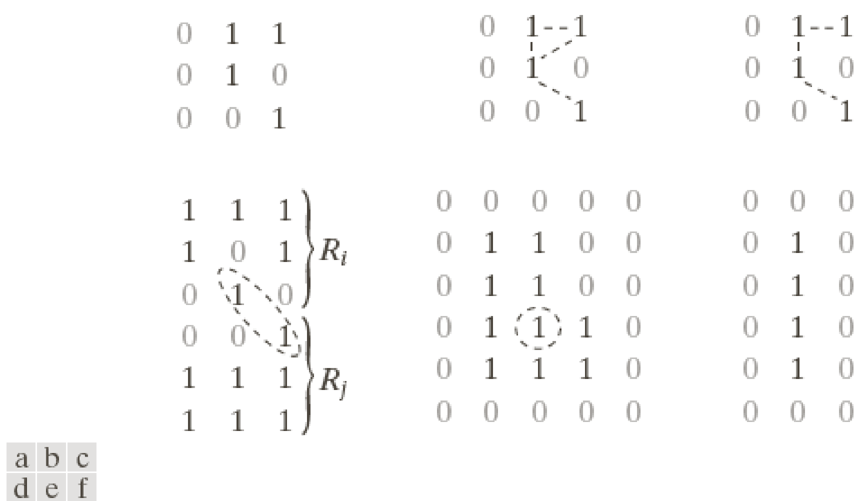


FIGURE 2.25 (a) An arrangement of pixels. (b) Pixels that are 8-adjacent (adjacency is shown by dashed lines; note the ambiguity). (c) *m*-adjacency. (d) Two regions that are adjacent if 8-adjecency is used. (e) The circled point is part of the boundary of the 1-valued pixels only if 8-adjacency between the region and background is used. (f) The inner boundary of the 1-valued region does not form a closed path, but its outer boundary does.

Some Basic Relationships Between Pixels

- Adjacency, Connectivity, Regions, and Boundaries
  - A (digital) path (or curve) from pixel p with coordinates (x, y) to pixel q with coordinates (s, t) is a sequence of distinct pixels with coordinates

    $$(x_0, y_0), (x_1, y_1), \ldots, (x_n, y_n)$$

    Where $(x_0, y_0) = (x, y), (x_n, y_n) = (s, t)$, and pixels (x_{i-1}, y_{i-1}) and (x_i, y_i) are adjacent for $1 \le i \le n$.
  - If $(x_0, y_0) = (x_n, y_n)$, the path is a *closed* path.
  - We can define 4-, 8-, or m-paths depending on the type of adjacency specified.
  - Note the absence of ambiguity in the m-path.

Some Basic Relationships Between Pixels

- Adjacency, Connectivity, Regions, and Boundaries
  - Let *S* represent a subset of pixels in an image. Two pixels *p* and *q* are said to be *connected* in *S* if there exists a path between them consisting entirely of pixels in *S*. For any pixel *p* in *S*, the *set* of pixels that are connected to it in *S* is called a *connected component* of *S*. If it only has one connected component, then set S is called a *connect set*.
  - Let *R* be a subset of pixels in an image. We call *R* a region of the image if *R* is a connected set. The *boundary* (also called *border* or *contour*) of a region *R* is the set of pixels in the region that have one or more neighbors that are not in *R*.
  - Edges are formed from pixels with derivative values that exceed a preset threshold.

Some Basic Relationships Between Pixels

- Distance Measures
  - For pixels p, q, and z, with coordinates (x, y), (s, t), and (v, w), respectively, D is a *distance function* or metric if
    - (a) $D(p,q) \geq 0 \quad (D(p,q) = 0 \quad iff \quad p = q)$
    - (b) $D(p,q) = D(q,p)$
    - (c) $D(p,z) \leq D(p,q) + D(q,z)$

Some Basic Relationships Between Pixels

- Distance Measures
  - *Euclidean distance*

$$D_e(p,q) = \left[(x-s)^2 + (y-t)^2\right]^{1/2}$$

  - *D_4 distance* (also called *city-block distance*)

$$D_4(p,q) = |x-s| + |y-t|$$

  - *D_8 distance* (also called *chessboard distance*)

$$D_8(p,q) = max(|x-s|,|y-t|)$$

Some Basic Relationships Between Pixels

- Distance Measures
  - *D_4 distance & D_8 distance*

$$\begin{array}{ccccc} & & 2 & & \\ & 2 & 1 & 2 & \\ 2 & 1 & 0 & 1 & 2 \\ & 2 & 1 & 2 & \\ & & 2 & & \end{array} \qquad \begin{array}{ccccc} 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{array}$$

  - Note that the *D_4 distance & D_8 distance* are independent of any paths that might exist between the points because these distances involve only the coordinates of the points.
  - The *D_m distance* between two points is defined as the shortest m-path between the points.

Today

- Some Basic Relationships Between Pixels
- Intensity Transformation
- Spatial Filtering

Basics

- The spatial domain processes can be denoted by the expression

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

- $f(x,y)$ is the input image, $g(x,y)$ is the output image, and T is an operator on f defined over a neighborhood of point (x, y)

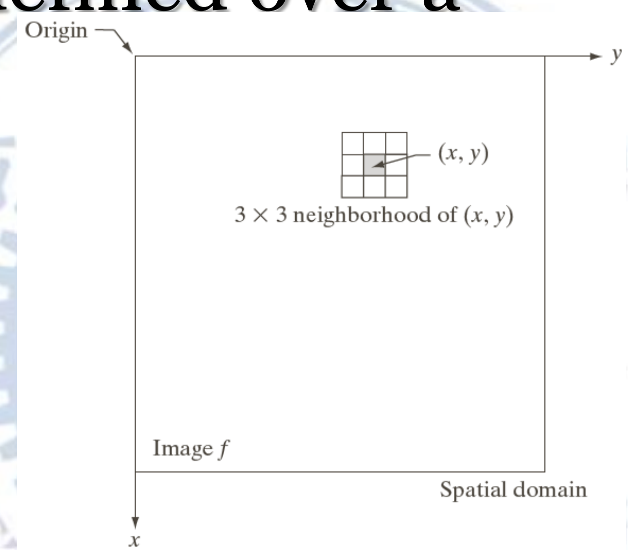


FIGURE 3.1 A 3×3 neighborhood about a point (x, y) in an image in the spatial domain. The neighborhood is moved from pixel to pixel in the image to generate an output image.

Basics

- The neighborhood along with a predefined operator is called *spatial filter* (also referred to as a *spatial mask*, *kernel*, *template*, or *window*), as we will discuss in detail in Section 3.4
- The smallest possible neighborhood is of size 1×1 .In this case, *g* depends only on the value of *f* at a single point (*x*, *y*) and *T* becomes *intensity transformation function*

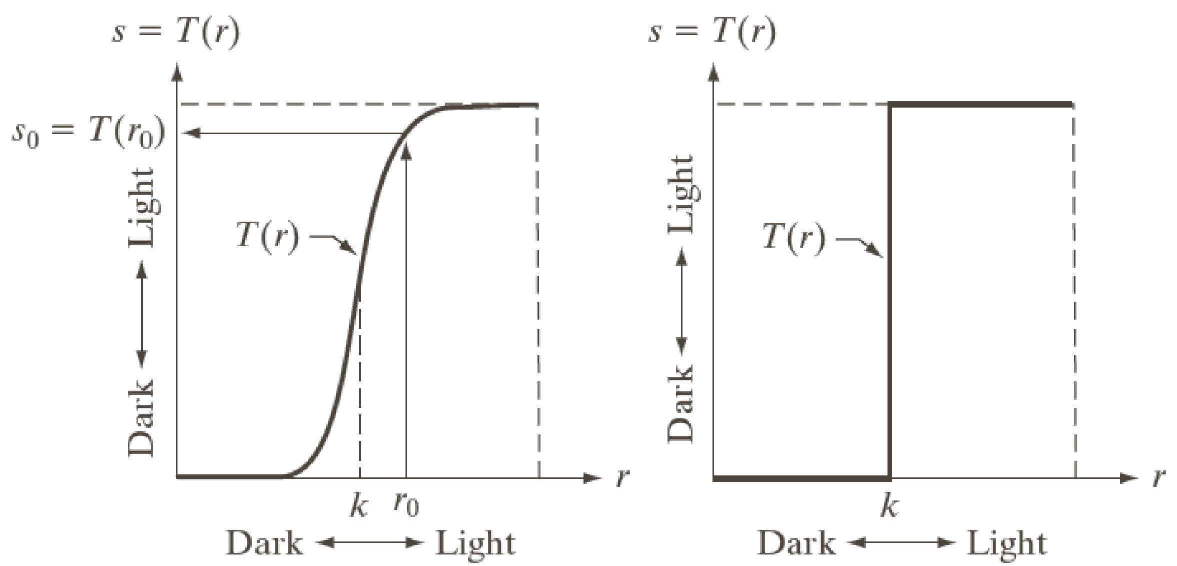
$$s = T(r)$$

Also called *gray-level transformation*, *mapping*, or *point processing*. *s* and *r* are variables denoting, respectively the intensity of *g* and *f* at any point (*x*, *y*)

Point Processing

- What can they do ?
- What's the form of T ?

- **Important:** every pixel for himself – spatial information completely lost!



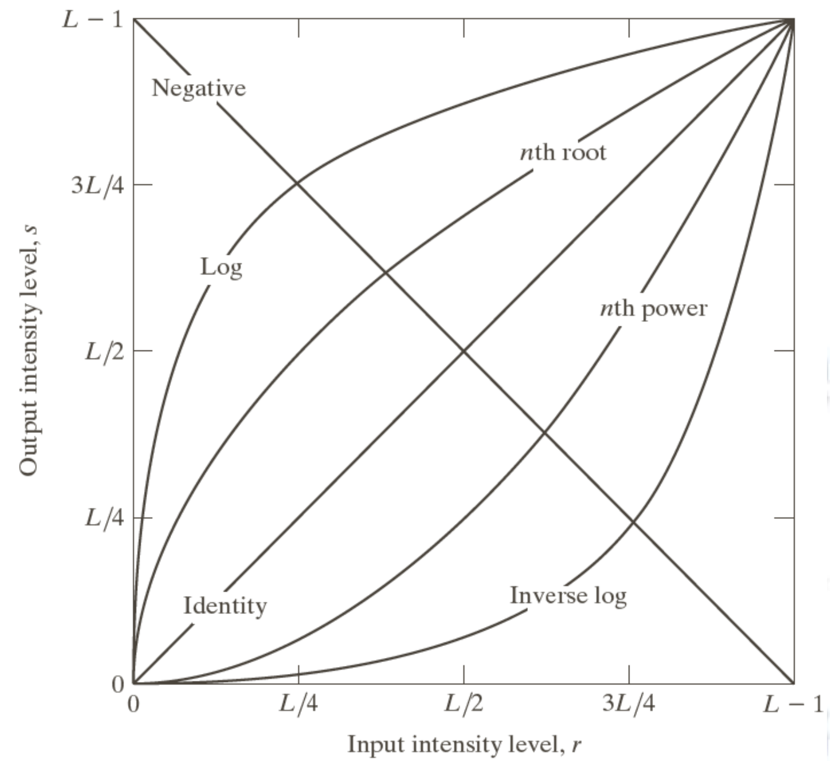
a b

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

Point Processing

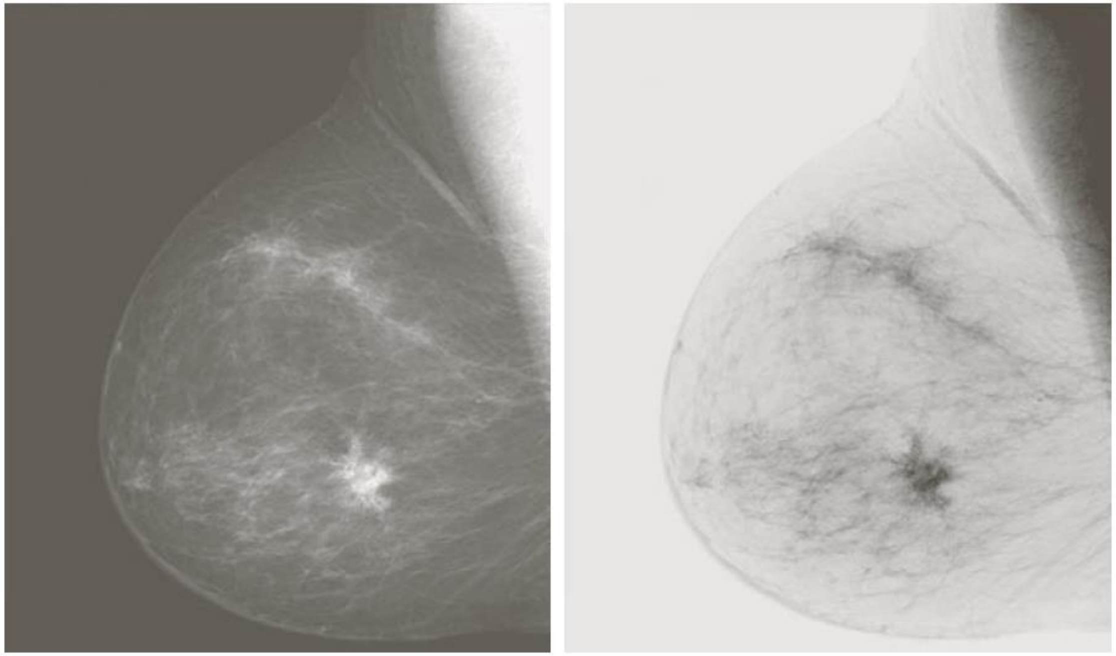
- Some basic intensity transformation functions
  - Image negatives

$$s = L - 1 - r$$
$$r \in [0,\ L-1]$$



Point Processing

- Some basic intensity transformation functions
  - Image negatives



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.)

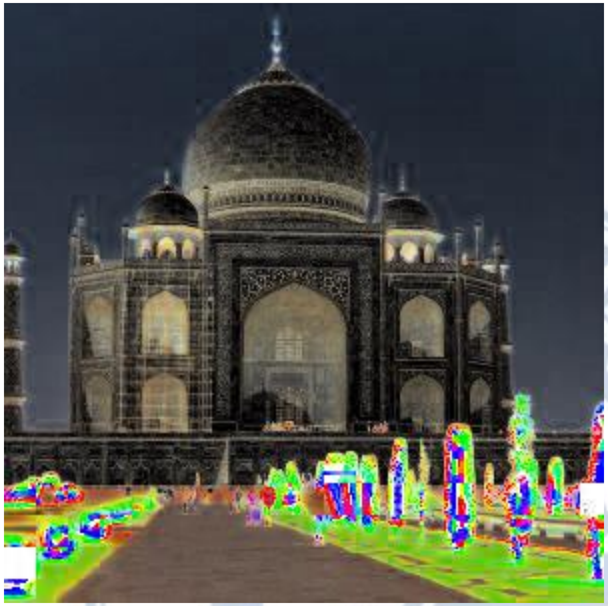
Point Processing

- Some basic intensity transformation functions
  - Image negatives

  - Inverse the colors

Point Processing

- Some basic intensity transformation functions
  - Image negatives

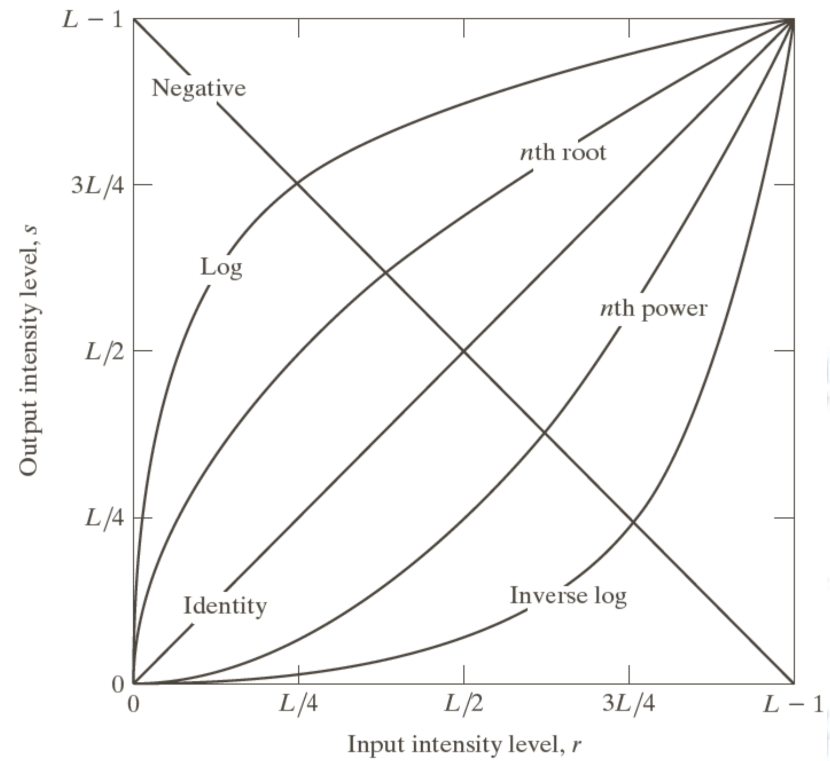


  - Inverse the brightness of each pixel

Point Processing

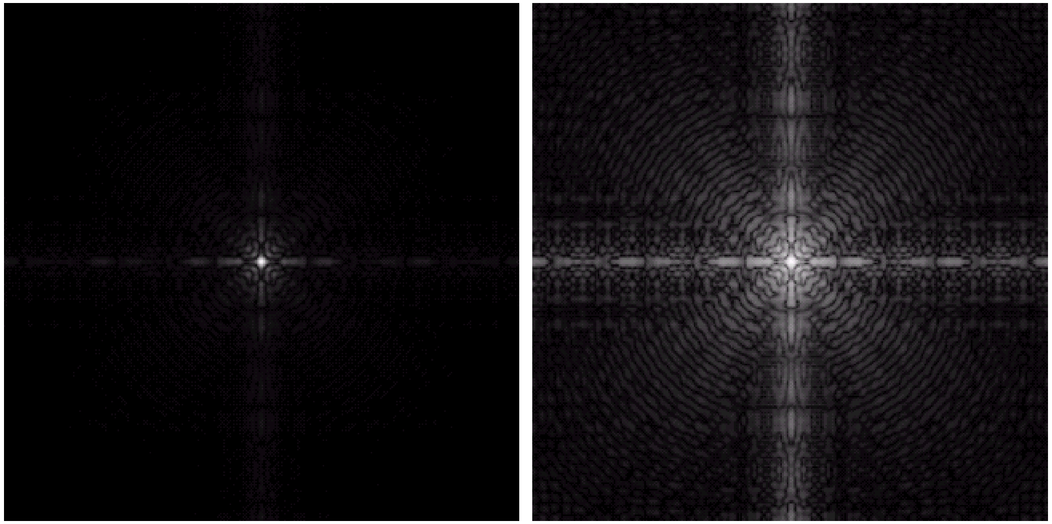
- Some basic intensity transformation functions
  - Log transformations

$$s = c\log(1+r)$$
$$c\ is\ a\ constant\ and\ r \geq 0$$



Point Processing

- Some basic intensity transformation functions
  - Log transformations

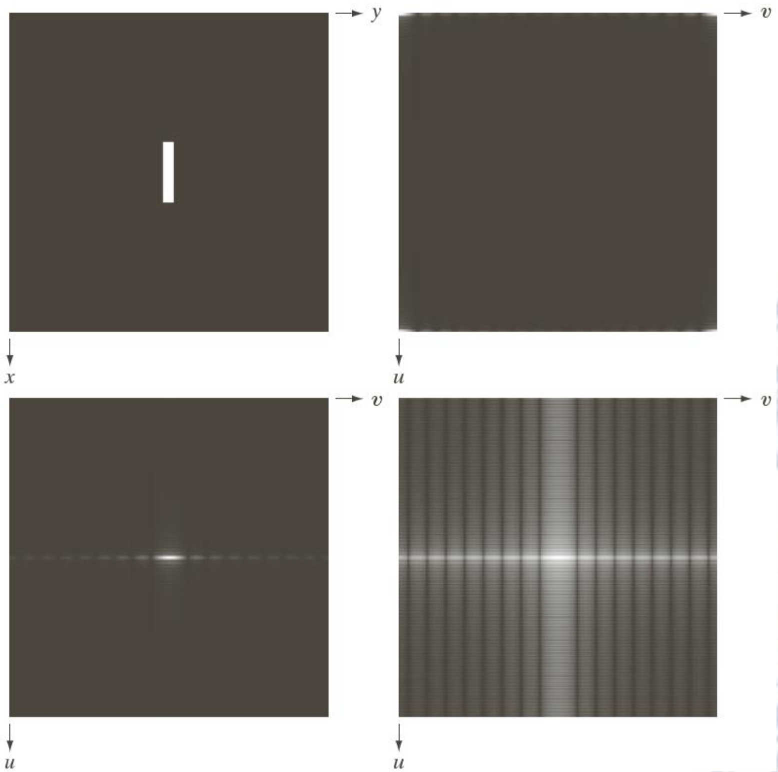


a b

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

Point Processing

- Some basic intensity transformation functions
  - Log transformations



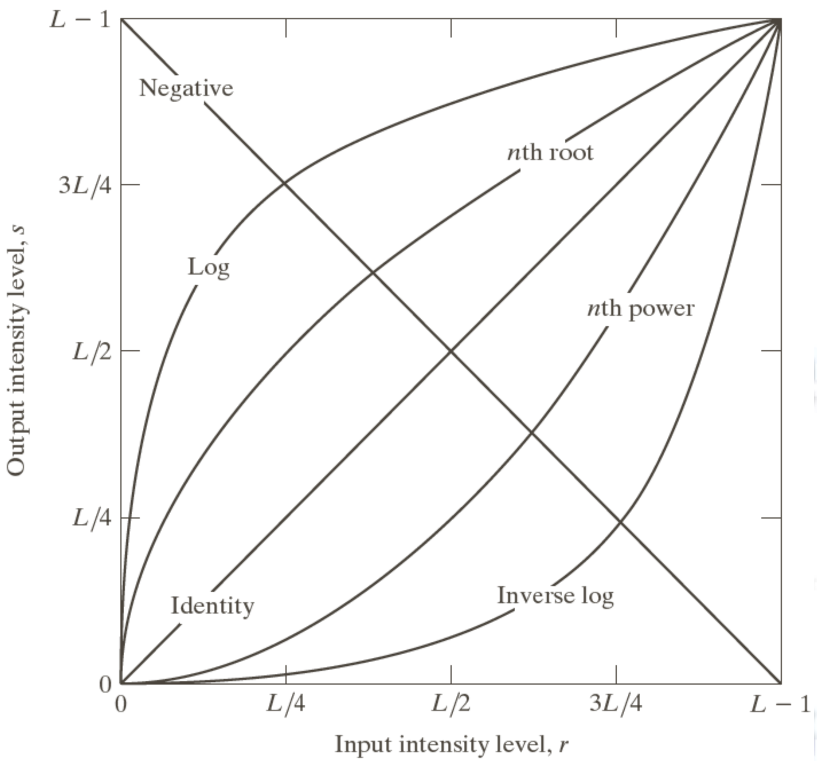
Point Processing

- Some basic intensity transformation functions
  - Power-Law (Gamma) transformations

$$s = cr^{\gamma}$$
$$c \text{ and } \gamma \text{ are positive constants}$$

  - Account for an offset

$$s = c(r+\varepsilon)^{\gamma}$$
$$c \text{ and } \gamma \text{ are positive constants}$$



Point Processing

- Some basic intensity transformation functions
  - Power-Law (Gamma) transformations

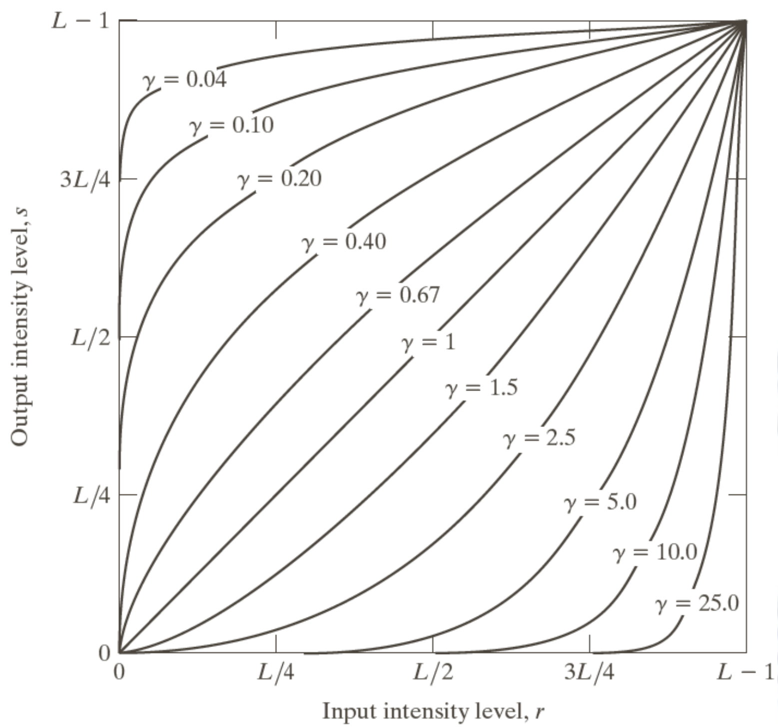
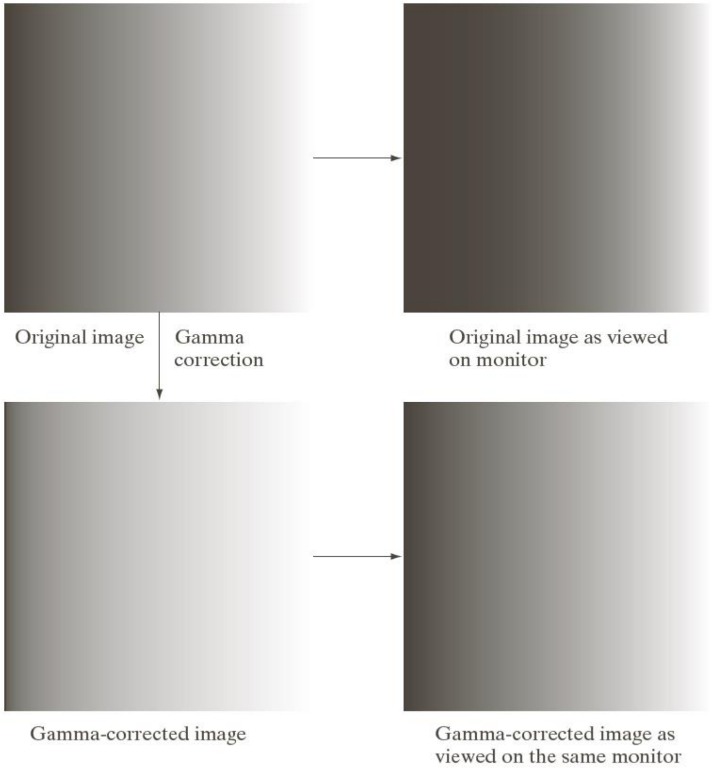


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

Point Processing

- Some basic intensity transformation functions
  - Power-Law (Gamma) transformations

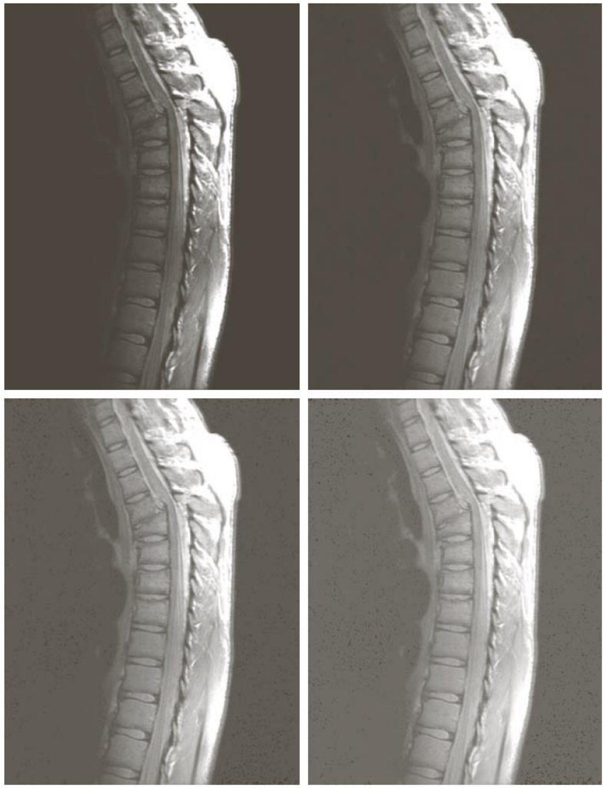


a	b
c	d

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

Point Processing

- Some basic intensity transformation functions
  - Power-Law (Gamma) transformations



a b
c d

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

Point Processing

- Some basic intensity transformation functions
  - Power-Law (Gamma) transformations

a	b
c	d

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

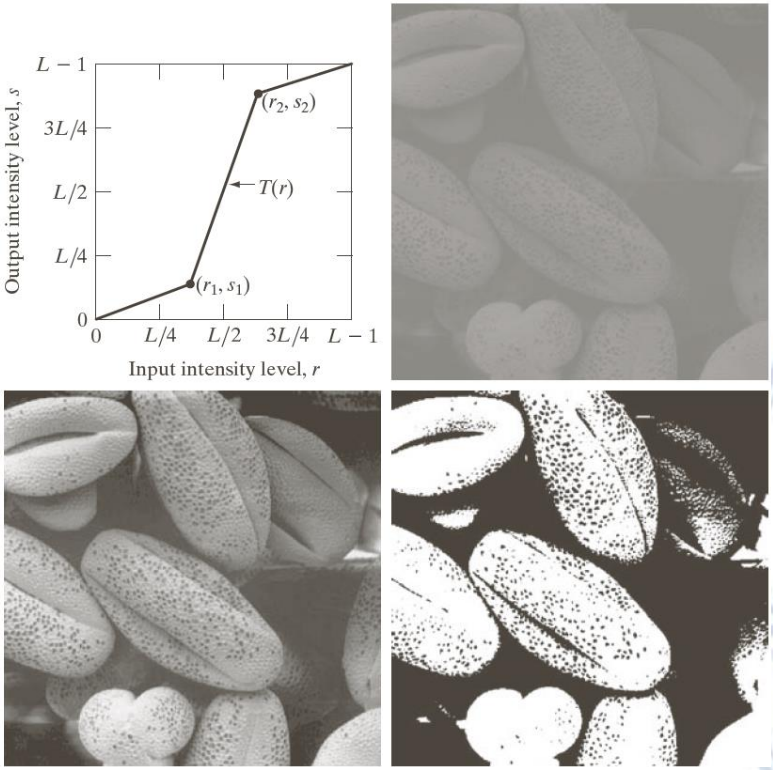
Point Processing

- Some basic intensity transformation functions
  - Piecewise-linear transformations
    - Contrast stretching

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

Point Processing

- Piecewise-linear transformations
  - Contrast stretching



a b
c d

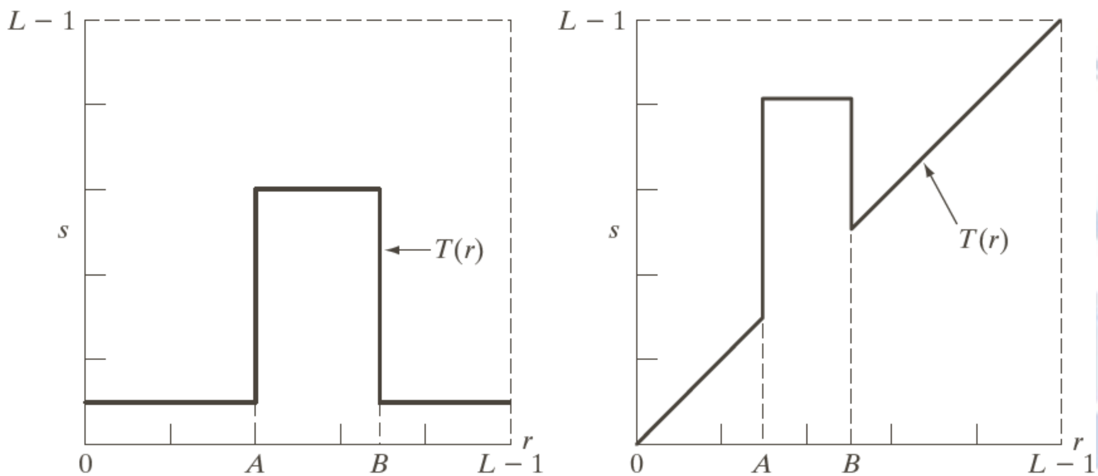
FIGURE 3.10 Contrast stretching. (a) Form of transformation function. (b) A low-contrast image. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

Point Processing

- Piecewise-linear transformations
  - Intensity-level slicing

a b

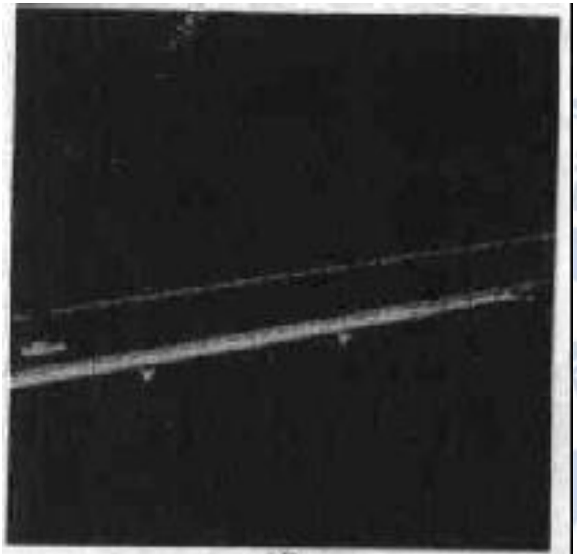
FIGURE 3.11 (a) This transformation highlights intensity range [A, B] and reduces all other intensities to a lower level. (b) This transformation highlights range [A, B] and preserves all other intensity levels.



Point Processing

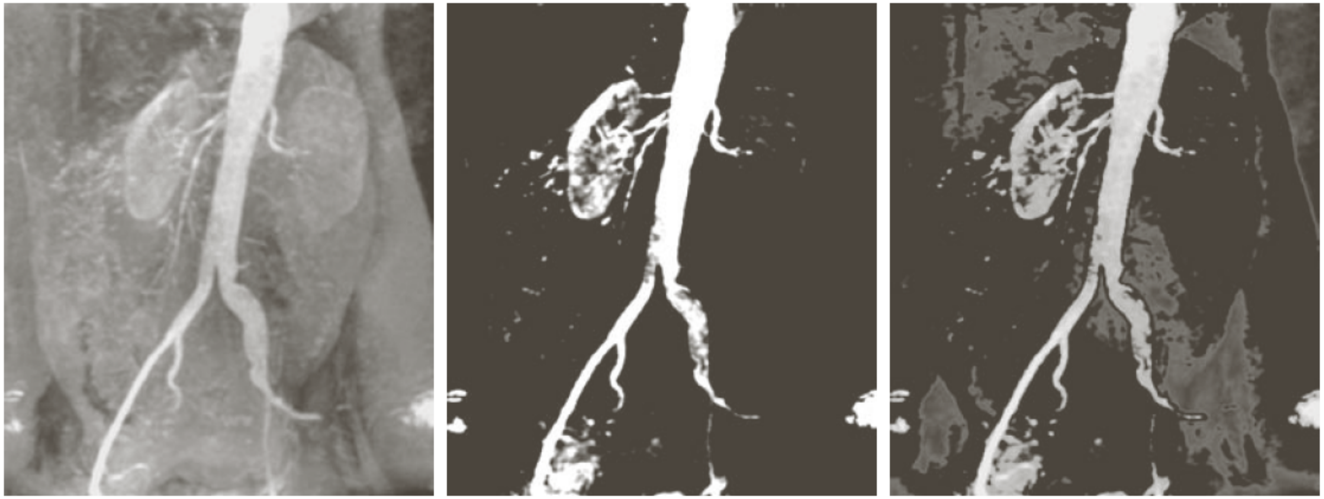
- Piecewise-linear transformations
  - Intensity-level slicing


(c)


(d)

Point Processing

- Piecewise-linear transformations
  - Intensity-level slicing



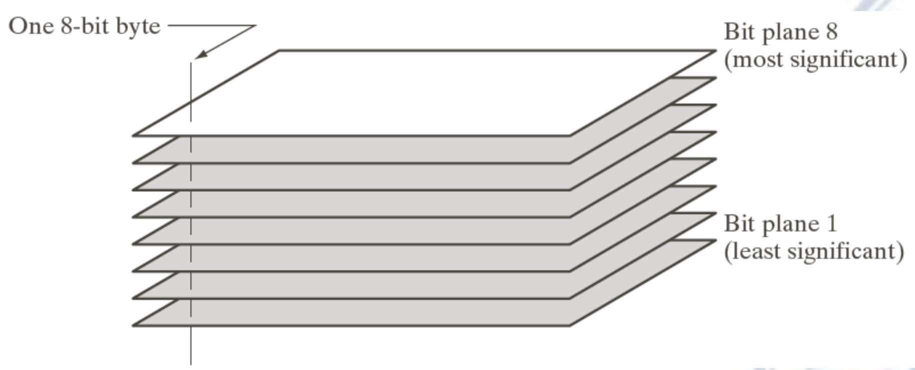
a b c

FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected area set to black, so that grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

Point Processing

- Piecewise-linear transformations
  - Bit-plane slicing

  Pixels are digital numbers composed of bits. The intensity of each pixel in a 256-level gray-scale image is composed of 8bits.



Point Processing

- Piecewise-linear transformations
  - Bit-plane slicing

Point Processing

- Piecewise-linear transformations
  - Bit-plane slicing

a b c

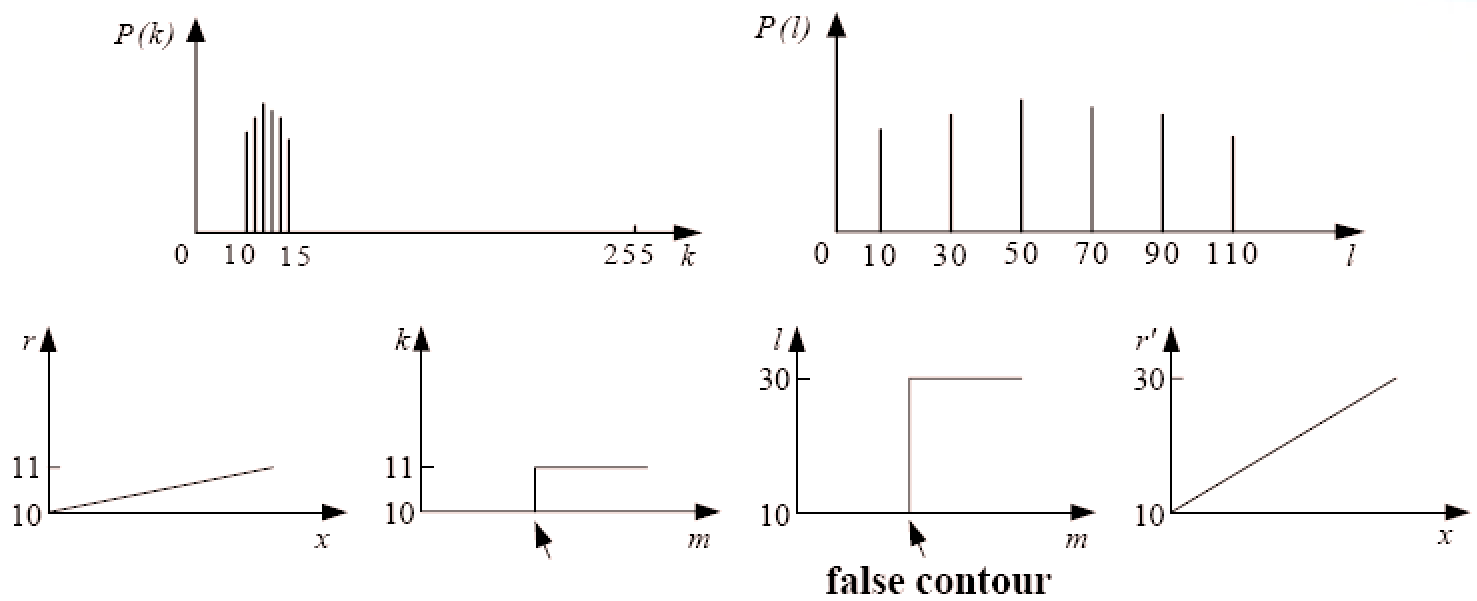
FIGURE 3.15 Images reconstructed using (a) bit planes 8 and 7; (b) bit planes 8, 7, and 6; and (c) bit planes 8, 7, 6, and 5. Compare (c) with Fig. 3.14(a).

Point Processing

- False contour
  - Discrete gray levels
  - Lack of effective gray levels

Point Processing

- False contour



Point Processing

- False contour

Point Processing

- Histogram processing
  - Histograms are the basis for numerous spatial domain processing techniques
  - Histograms are the statistical diagrams of gray level distribution
- For continuous gray level

$$p(r) = \lim_{\Delta r \to 0} \frac{A(r + \Delta r) - A(r)}{\Delta r \cdot A}$$

Point Processing

- Histogram processing
  - For discrete gray level

$$h(r_k) = n_k, \quad r_k \in [0, L-1]$$

where r_k is the kth intensity value and n_k is the number of pixels in the image with intensity r_k. Thus, a normalized histogram is given by

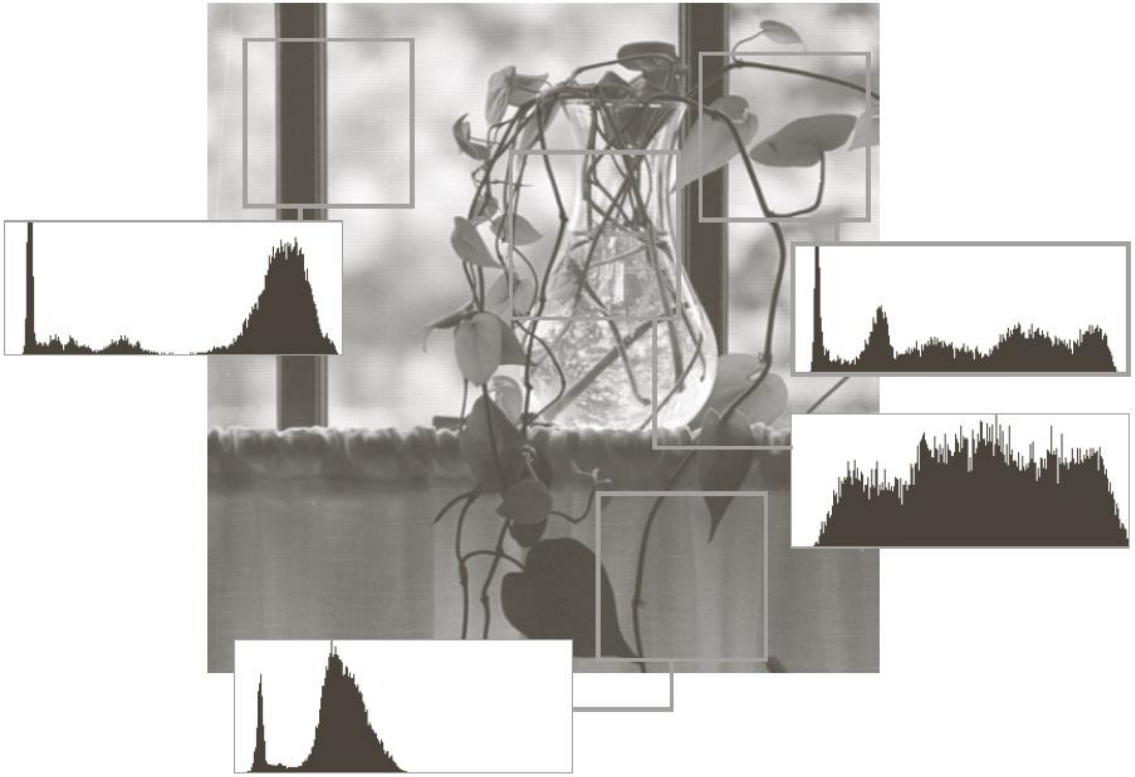
$$p(r_k) = n_k / MN, \text{ for } k = 0,1,2,\ldots,L-1$$

M and N are the row and column dimensions of the image. And we have

$$\sum_{k=0}^{L-1} p_k(r_k) = 1$$

Point Processing

- Histogram processing



Point Processing

- Histogram processing

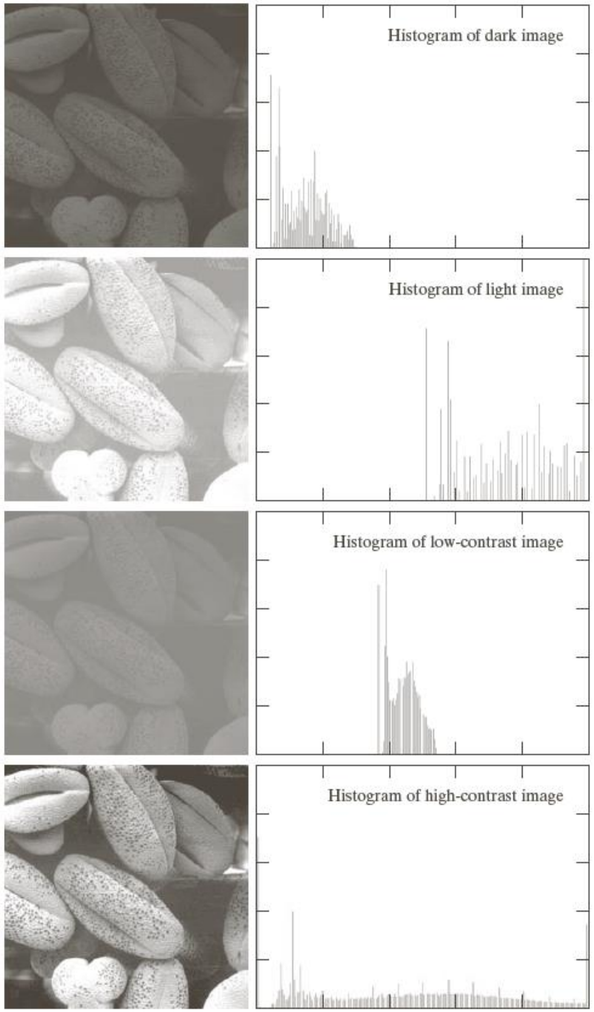
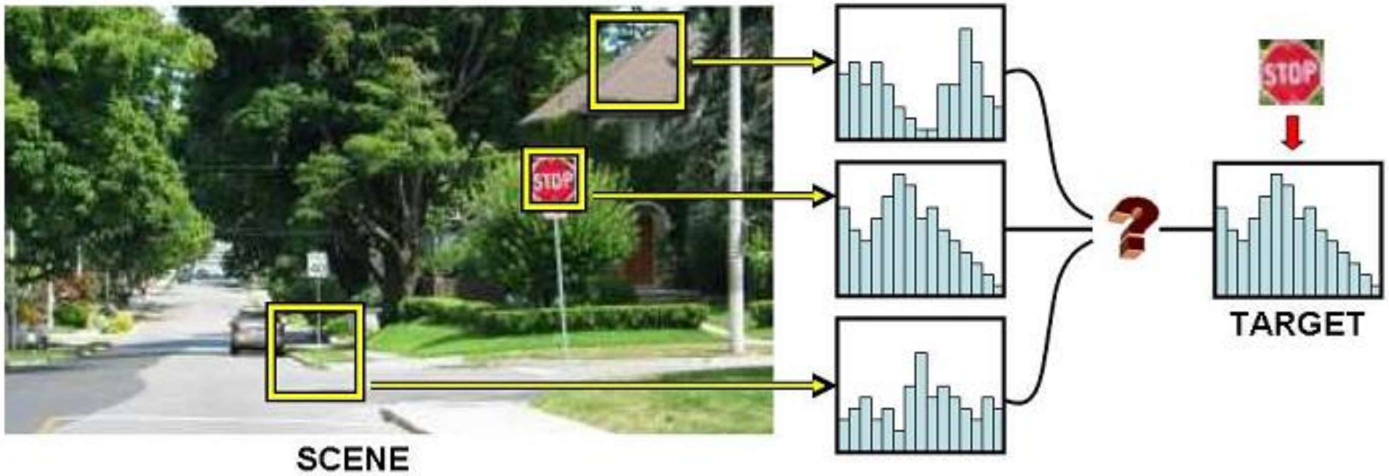


FIGURE 3.16 Four basic image types: dark, light, low contrast, high contrast, and their corresponding histograms.

Point Processing

- In many cases histograms are needed for local areas in an image
- Examples:
  - Pattern detection
  - adaptive enhancement
  - adaptive thresholding
  - tracking



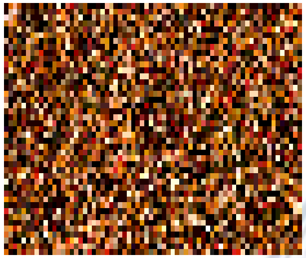
Histogram usage

- Digitizing parameters
- Measuring image properties:
  - Average
  - Variance
  - Entropy
  - Contrast
  - Area (for a given gray-level range)
- Threshold selection
- Image distance
- Image Enhancement
  - Histogram equalization
  - Histogram stretching
  - Histogram matching

Point Processing

- Histogram processing
  - Histograms haven't any position information

    Reshuffling all pixels within the image



Its histogram won't change. No point processing will be affected...

Spatial information is discarded

Point Processing

- Histogram Equalization & Histogram Specification
  - Histogram equalization
    - The histogram of is a uniform histogram
  - Histogram specification
    - The histogram of has a specified shape
  - Histogram equalization is a special example of histogram specification
  - Theoretical argument of histogram equalization
    - Principle of the biggest entropy

$$H_c = -\int_{r_{min}}^{r_{max}} p(r) \log p(r)\, dr$$

    - While the histogram is equalized, the entropy of the image is biggest, which means that the human visual system can obtain the maximum information

Histogram Equalization

Transformation function

$$s = T(r) \quad 0 \le r \le L-1$$

Assume that:

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

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

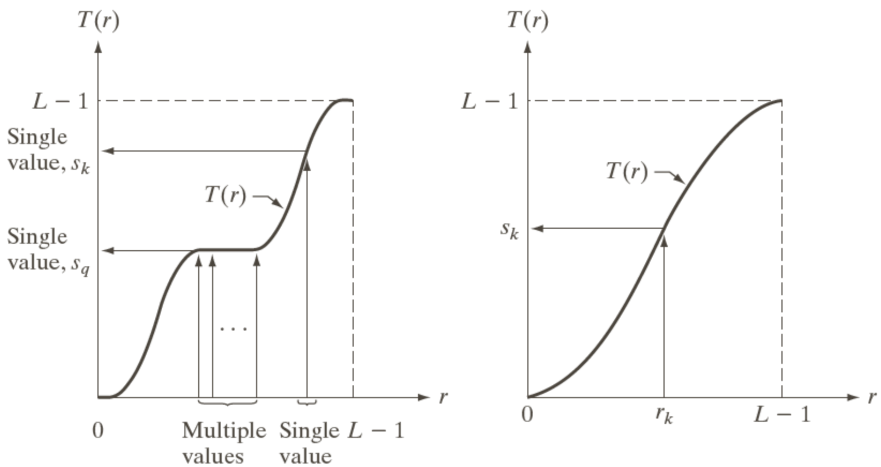
In some formulations to be discussed later, we use the inverse

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

change condition (a) to

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

Histogram Equalization



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.

Histogram Equalization

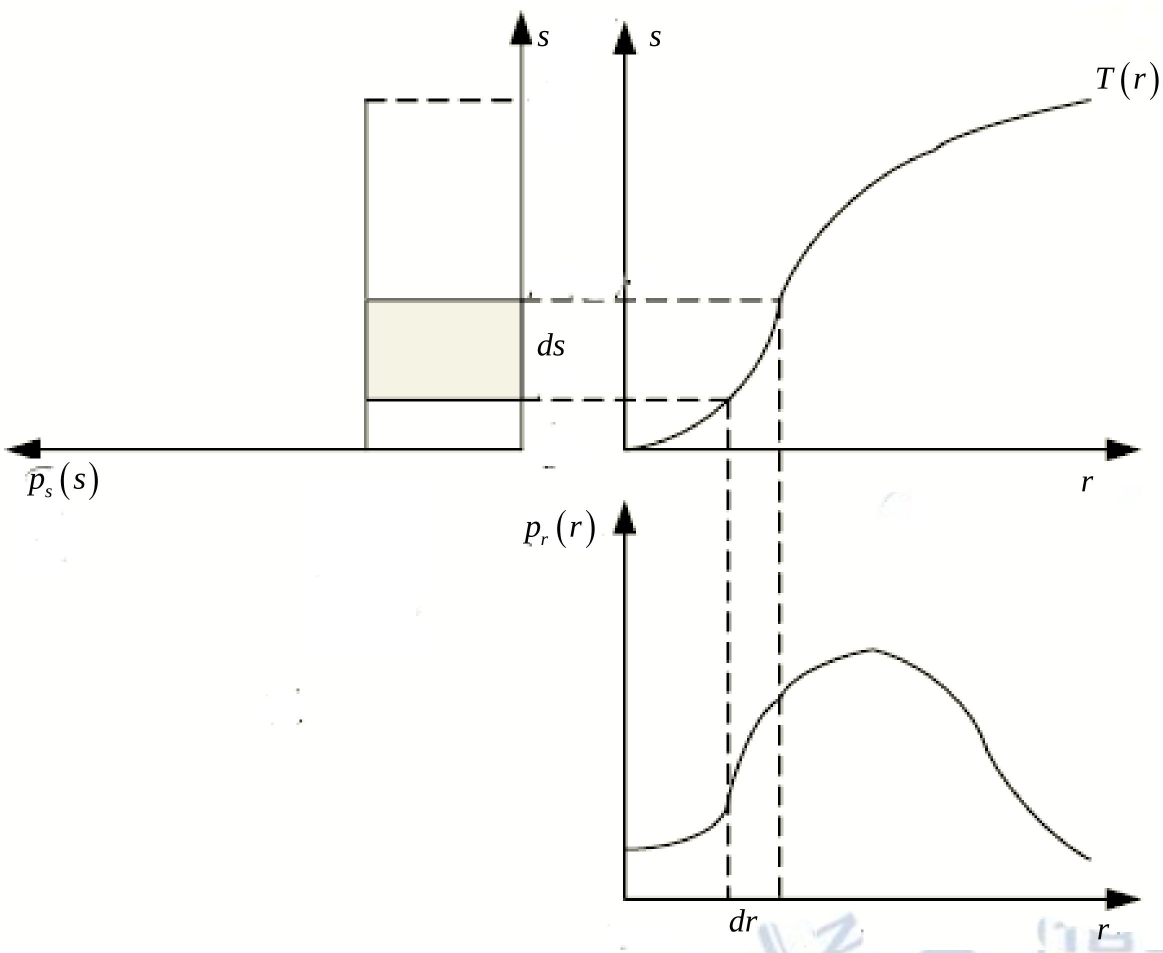
- $p_r(r)$ and $p_s(s)$ denote the PDFs of r and s, respectively. $p_r(r)$ and $T(r)$ are known and $T(r)$ is continuous and differentiable. Then the PDF of the transformed variable s can be obtained using the simple formula

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

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

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

Histogram Equalization



Histogram Equalization

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

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

Histogram Equalization

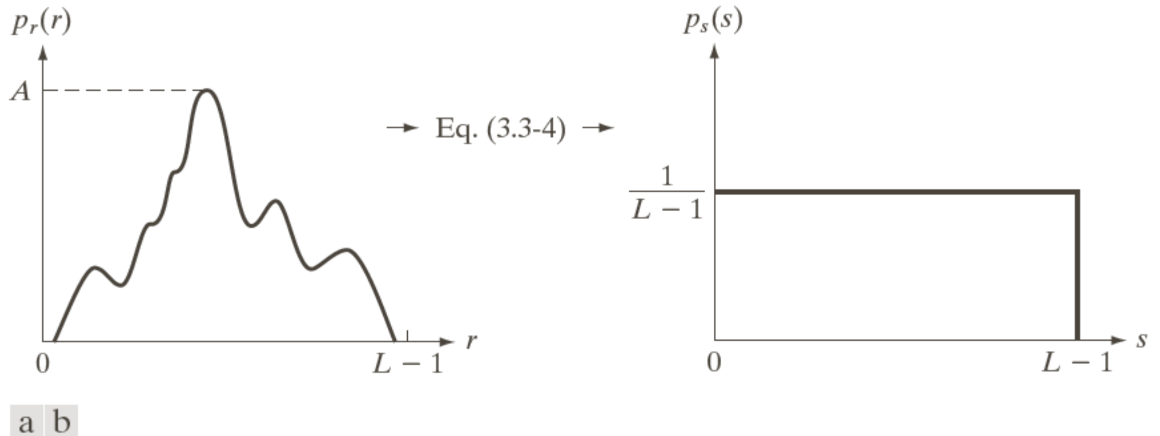


FIGURE 3.18 (a) An arbitrary PDF. (b) Result of applying the transformation in Eq. (3.3-4) to all intensity levels, r. The resulting intensities, s, have a uniform PDF, independently of the form of the PDF of the r's.

Histogram Equalization

- For discrete values, we deal with probabilities (histogram values) and summations instead of probability density functions and integrals. The condition s of monotonicity stated earlier apply also in the discrete case

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

- A plot of $p_r(r_k)$ versus r_k is commonly referred to as a *histogram*

Histogram Equalization

- The discrete form of the transformation is

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

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

Histogram Equalization

- Example 3.4 & 3.5

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.

Histogram Equalization

- Example 3.4 & 3.5

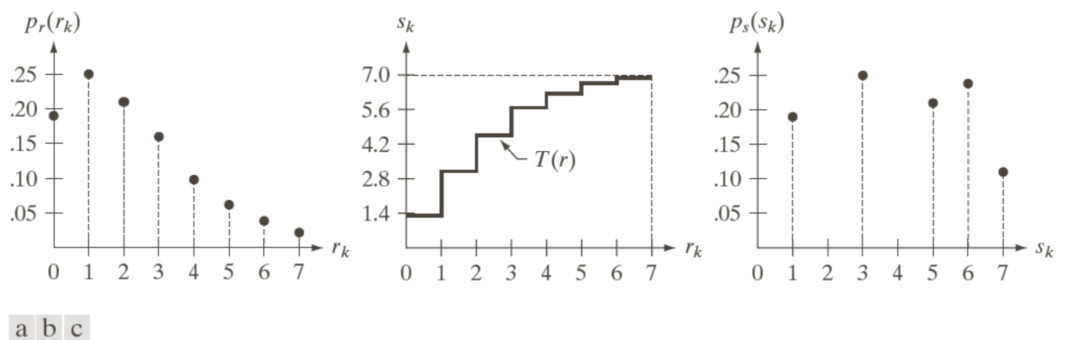


FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

Histogram Equalization

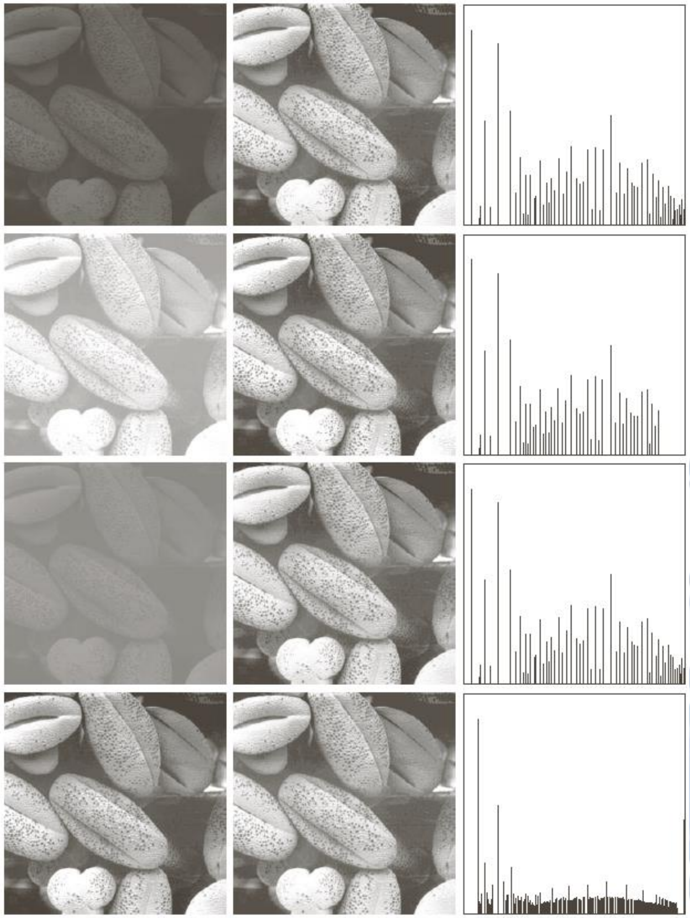


FIGURE 3.20 Left column: images from Fig. 3.16. Center column: corresponding histogram-equalized images. Right column: histograms of the images in the center column.

Histogram Equalization

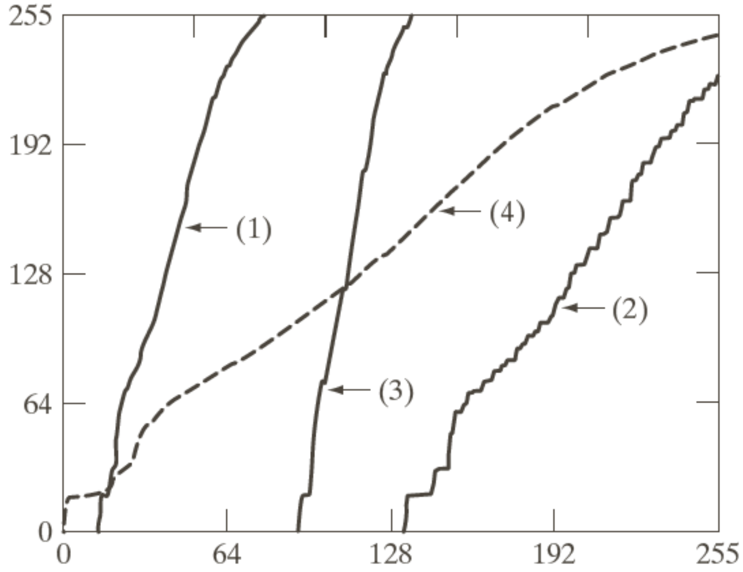


FIGURE 3.21 Transformation functions for histogram equalization. Transformations (1) through (4) were obtained from the histograms of the images (from top to bottom) in the left column of Fig. 3.20 using Eq. (3.3-8).

Histogram Equalization

- The *inverse transformation* from *s* back to *r* is defined by

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

- Problem 3.10(3rd edition)
- Problem 3.9(2nd edition)

Histogram Matching

- *Histogram equalization* automatically determines a transformation function that seeks to produce an output image that has a uniform histogram.
- *Histogram matching* (or *histogram specification*), generates a processed image that has a specified histogram.

Histogram Matching

- Start from continuous intensities
  - r and z denote the intensity levels of the input and output images, respectively
  - $p_r(r)$ and $p_z(z)$ denote their continuous PDFs, while $p_r(r)$ is estimated from given input image and $p_z(z)$ is the specified PDF that we wish to have
  - Let s be a random variable with the property

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

Histogram Matching

- Define another random variable z with the property

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

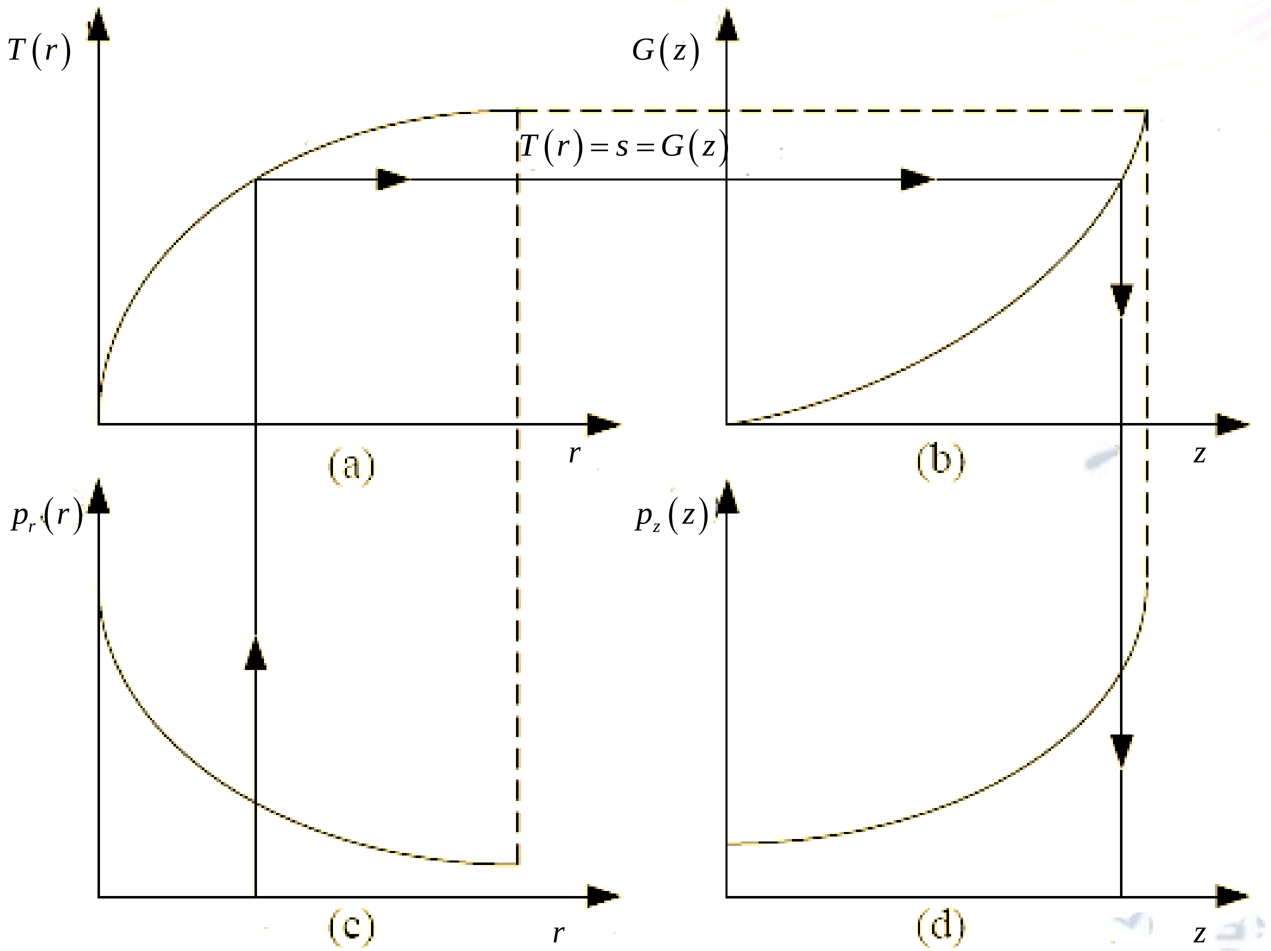
- From these equations that

$$G(z) = T(r)$$

- That z must satisfy the condition

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

Histogram Matching



Histogram Matching

- Discrete formulation

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

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

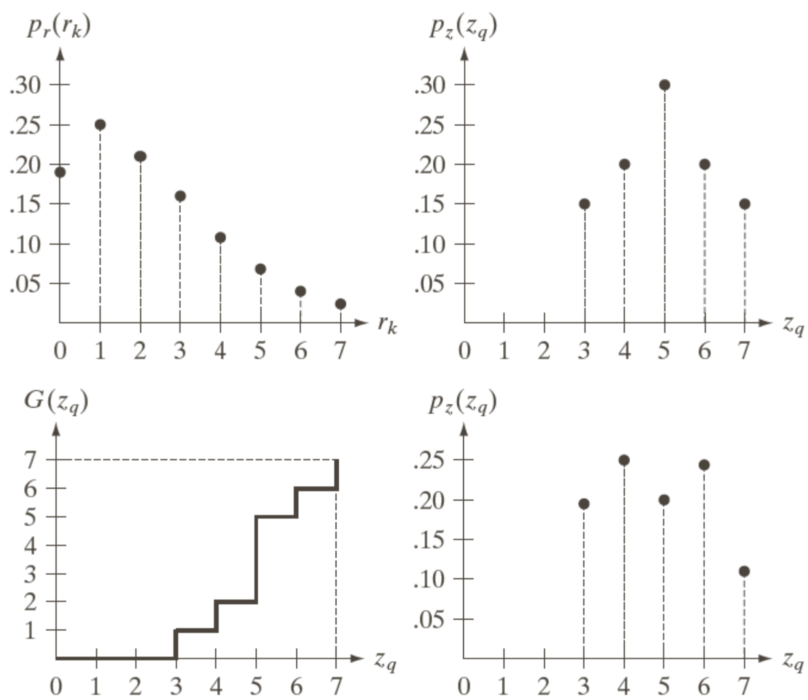
$$G(z_q) = (L-1)\sum_{i=0}^{q} p_z(z_i)$$

$$G(z_q) = s_k$$

$$z_q = G^{-1}(s_k)$$

Histogram Matching

- Example 3.7, 3.8



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 Matching

- Example 3.7, 3.8

z_q	Specified $p_z(z_q)$	Actual $p_z(z_k)$
$z_0 = 0$	0.00	0.00
$z_1 = 1$	0.00	0.00
$z_2 = 2$	0.00	0.00
$z_3 = 3$	0.15	0.19
$z_4 = 4$	0.20	0.25
$z_5 = 5$	0.30	0.21
$z_6 = 6$	0.20	0.24
$z_7 = 7$	0.15	0.11

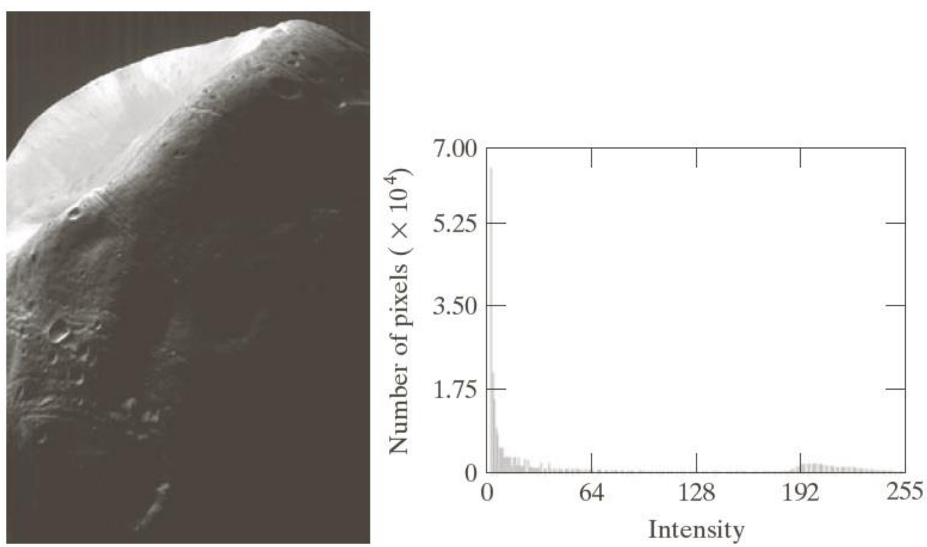
TABLE 3.2 Specified and actual histograms (the values in the third column are from the computations performed in the body of Example 3.8).

z_q	$G(z_q)$
$z_0 = 0$	0
$z_1 = 1$	0
$z_2 = 2$	0
$z_3 = 3$	1
$z_4 = 4$	2
$z_5 = 5$	5
$z_6 = 6$	6
$z_7 = 7$	7

TABLE 3.3 All possible values of the transformation function G scaled, rounded, and ordered with respect to z.

Histogram Matching

- Histogram equalization is not always good

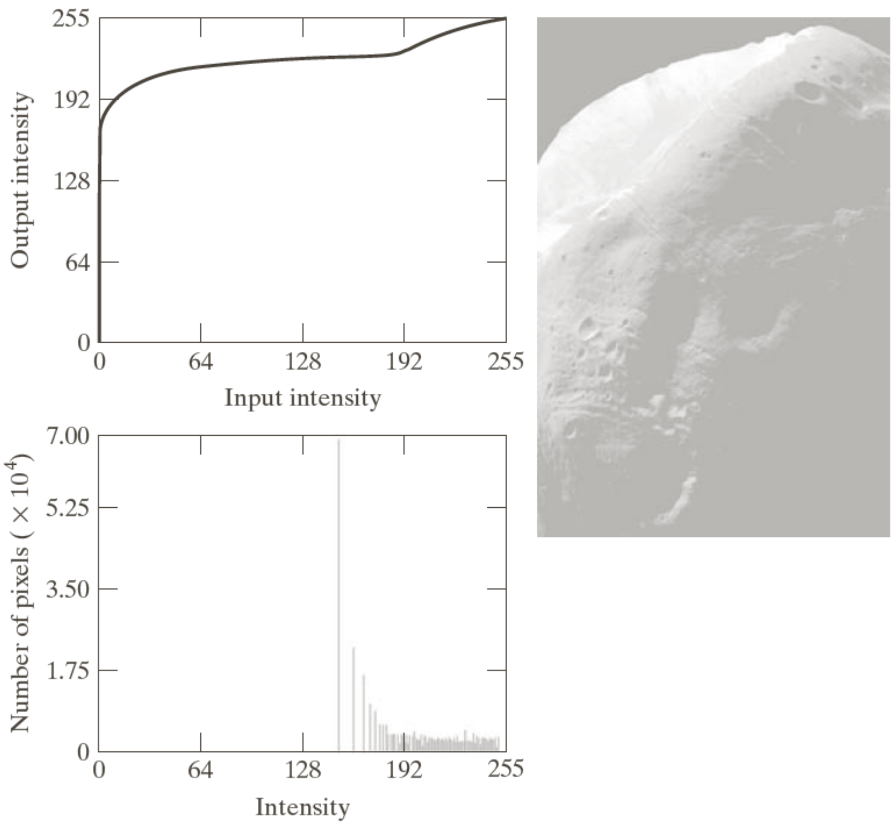


a b

FIGURE 3.23 (a) Image of the Mars moon Phobos taken by NASA's *Mars Global Surveyor.* (b) Histogram. (Original image courtesy of NASA.)

Histogram Matching

- Histogram equalization is not always good

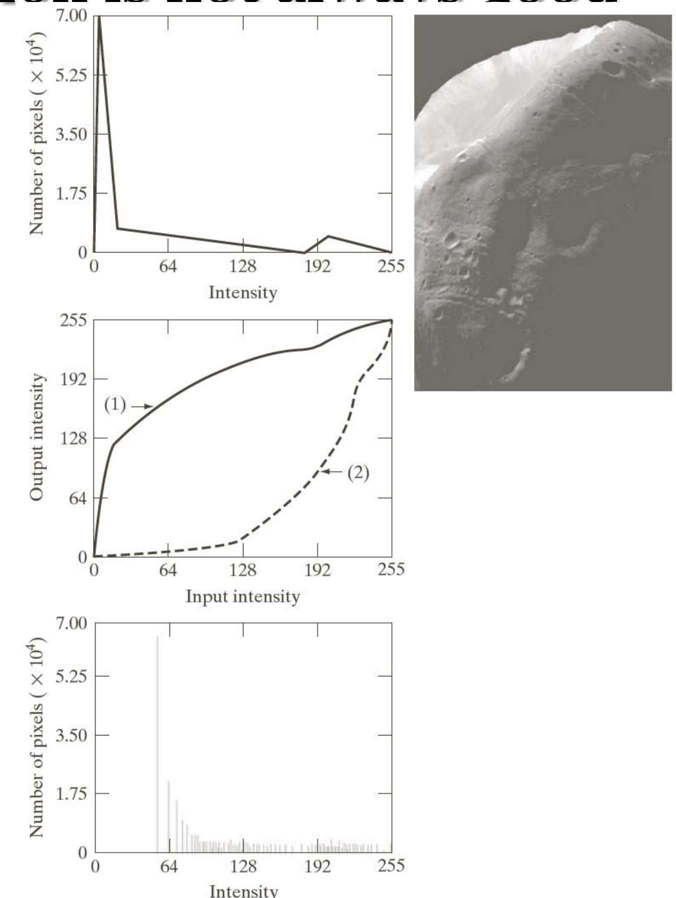


a b
c

FIGURE 3.24
(a) Transformation function for histogram equalization. (b) Histogram-equalized image (note the washed-out appearance). (c) Histogram of (b).

Histogram Matching

- Histogram equalization is not always good



a c
b
d

FIGURE 3.25
(a) Specified histogram.
(b) Transformations.
(c) Enhanced image using mappings from curve (2).
(d) Histogram of (c).

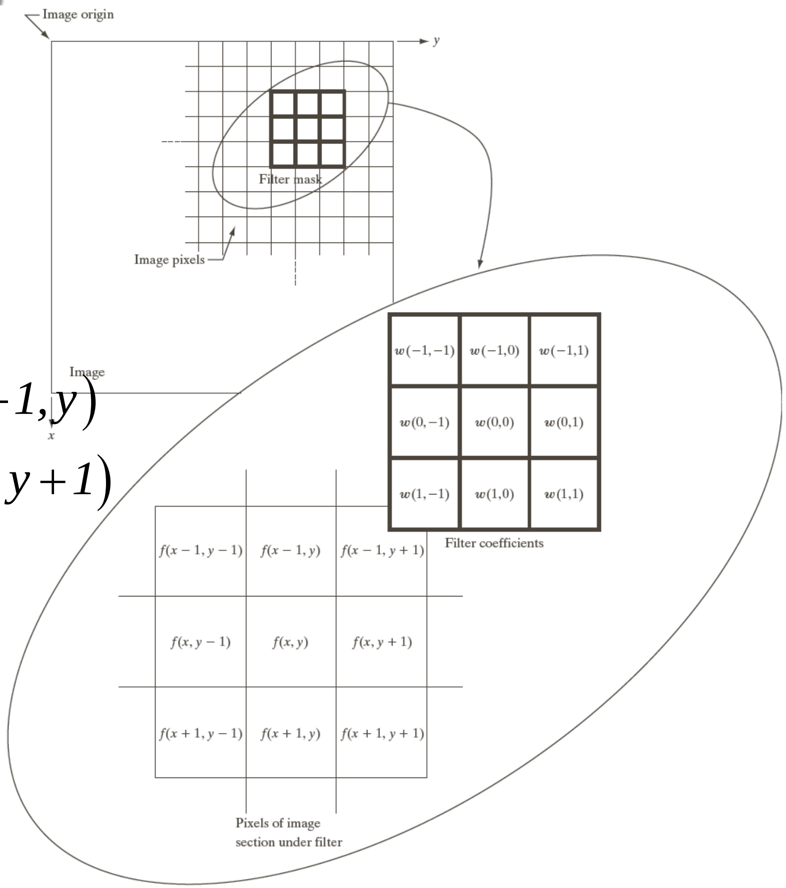
Today

- Some Basic Relationships Between Pixels
- Intensity Transformation
- Spatial Filtering

Spatial Filtering

- The mechanics of spatial filtering
  - For a mask of size 3 by 3

$$g(x,y) = w(-1,-1)\,f(x-1,y-1) + w(-1,0)\,f(x-1,y)$$
$$+\ldots + w(0,0)\,f(x,y) + \ldots + w(1,1)\,f(x+1,y+1)$$



Spatial Filtering

- The mechanics of spatial filtering
  - Generally , for a mask of size m by n, we assume that m=2a+1 and n=2b+1,where a and b are positive integers.
  - Linear spatial filtering of an image of size M by N with a filter of size m by n is given by the expression

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

Spatial Filtering

- Some yucky details
  - What about near the edge?
    - the filter window falls off the edge of the image
    - need to extrapolate
    - methods:
      - clip filter (black)
      - wrap around
      - copy edge
      - reflect across edge
      - vary filter near edge

Spatial Filtering

- Relationship between spatial correlation and convolution
  - Page 168, section 3.4.2, DIP(3rd Edition)

Spatial Filtering

- Vector representation of linear filtering

$$
\begin{aligned}
R &= w_1 z_1 + w_2 z_2 + \ldots + w_{mn} z_{mn} \\
&= \sum_{k=1}^{mn} w_k z_k \\
&= \mathbf{w}^{\mathbf{T}} \mathbf{z}
\end{aligned}
$$

  - where the *w*s are the coefficients of an m by n filter and the *z*s are the corresponding image intensities encompassed by the filter

Spatial Filtering

- Vector representation of linear filtering
  - As an example

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9 = \sum_{k=1}^{9} w_k z_k = \mathbf{w}^{\mathbf{T}} \mathbf{z}$$

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

  - where the ws are 9-dimensional vectors formed from the coefficients of the mask and the image intensities encompassed by the mask, respectively

Smoothing Spatial Filters

- Image smoothing is used for blurring and for noise reduction
  - Blurring is used in preprocessing steps, such as removal of small details from an image prior to object extraction, and bridging of small gaps in lines or curves
  - Noise reduction can be accomplished by blurring with a linear filter and also by nonlinear filtering
- Disadvantage of image smoothing
  - edges or details in the image will be blurred
- Principle of image smoothing
  - remain most of the details in the image

Smoothing Spatial Filters

$$\frac{1}{9} \times \begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array} \qquad \frac{1}{16} \times \begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

- Smoothing linear filters
  - Use of first filter yields the standard averages of the pixels under the mask. A spatial averaging filter in which all coefficients are equal sometimes called a *box filter*
  - The second mask yields a so-called *weighted average*, giving more weight to some pixels at the expense of the others. (Gaussian kernel)

Smoothing Spatial Filters

- Smoothing linear filters
  - The general implementation for filtering an M by N image with a weighted averaging filter of size m by n (m and n odd) is given by the expression

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

Smoothing Spatial Filters

- Smoothing linear filters

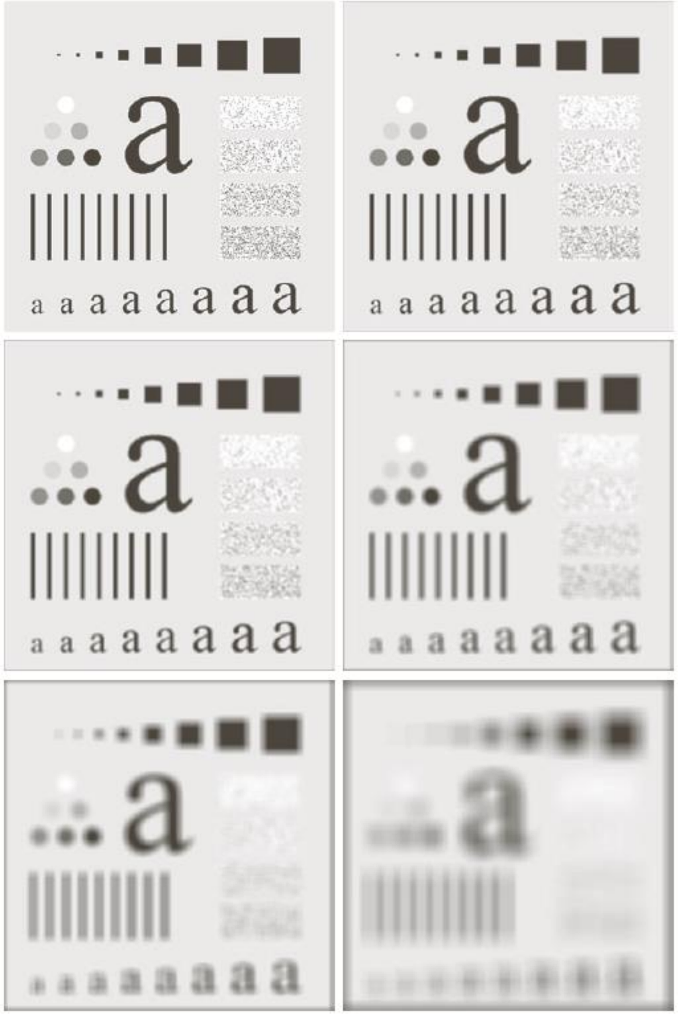
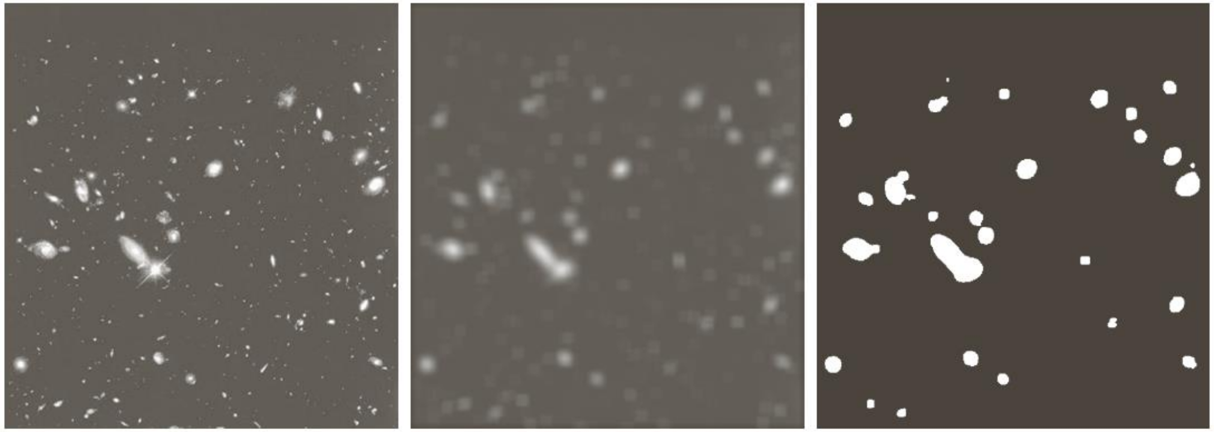


FIGURE 3.33 (a) Original image, of size 500 × 500 pixels. (b)–(f) Results of smoothing with square averaging filter masks of sizes $m = 3, 5, 9, 15$, and 35, respectively. The black squares at the top are of sizes 3, 5, 9, 15, 25, 35, 45, and 55 pixels, respectively; their borders are 25 pixels apart. The letters at the bottom range in size from 10 to 24 points, in increments of 2 points; the large letter at the top is 60 points. The vertical bars are 5 pixels wide and 100 pixels high; their separation is 20 pixels. The diameter of the circles is 25 pixels, and their borders are 15 pixels apart; their intensity levels range from 0% to 100% black in increments of 20%. The background of the image is 10% black. The noisy rectangles are of size 50 × 120 pixels.

a b
c d
e f

Smoothing Spatial Filters

- Smoothing linear filters



a b c

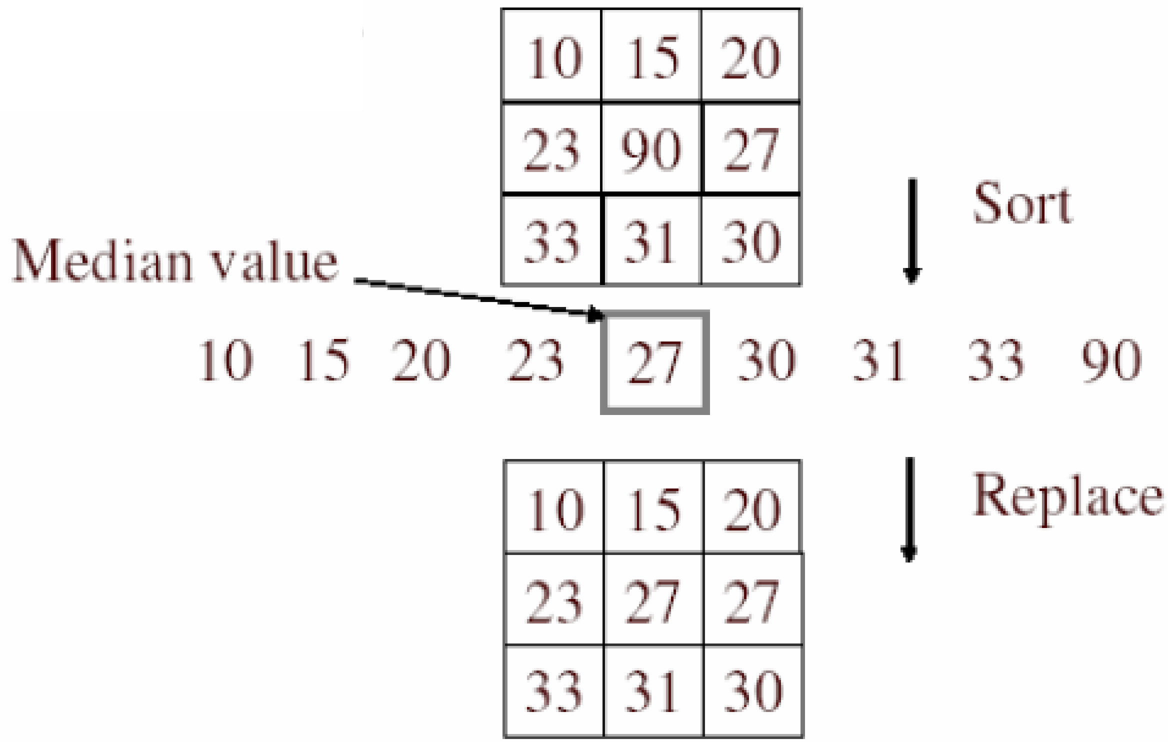
FIGURE 3.34 (a) Image of size 528 × 485 pixels from the Hubble Space Telescope. (b) Image filtered with a 15 × 15 averaging mask. (c) Result of thresholding (b). (Original image courtesy of NASA.)

Smoothing Spatial Filters

- Order-statistic filters
  - Order-statistic filters are nonlinear spatial filters whose response is based on ordering the pixels contained in the image area encompassed by the filter, and then replacing the value of the center pixel with the value determined by the ranking result.
  - Median, max, min filtering.
  - The median filters are the best known and particularly effective in the presence of impulse noise

Smoothing Spatial Filters

- A Median Filter operates over a window by selecting the median intensity in the window.

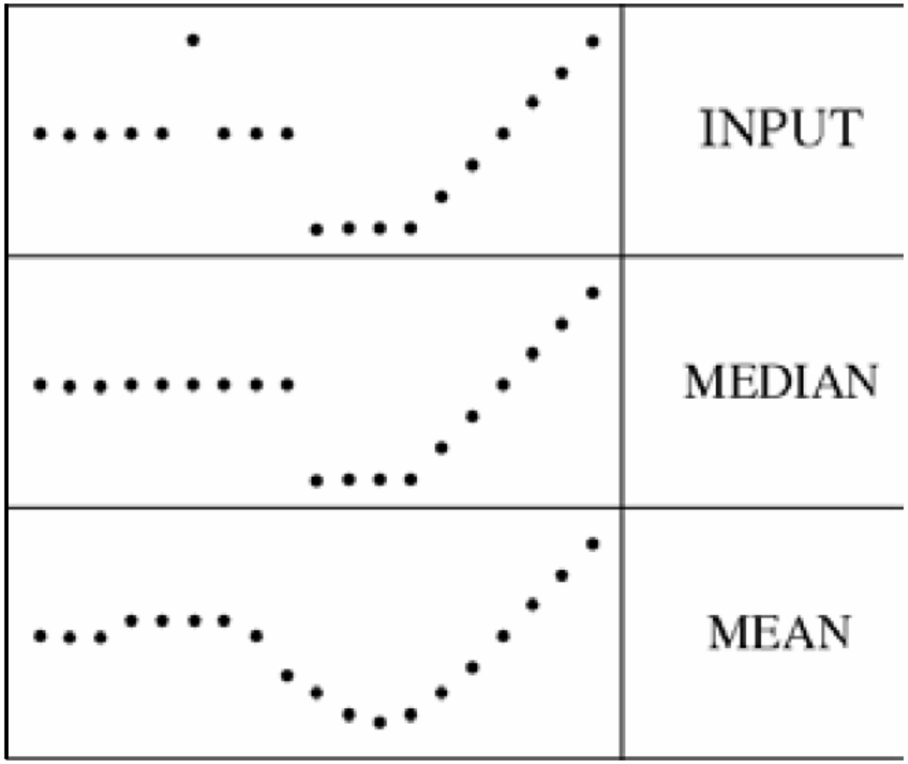


- Is median filtering linear?

Smoothing Spatial Filters

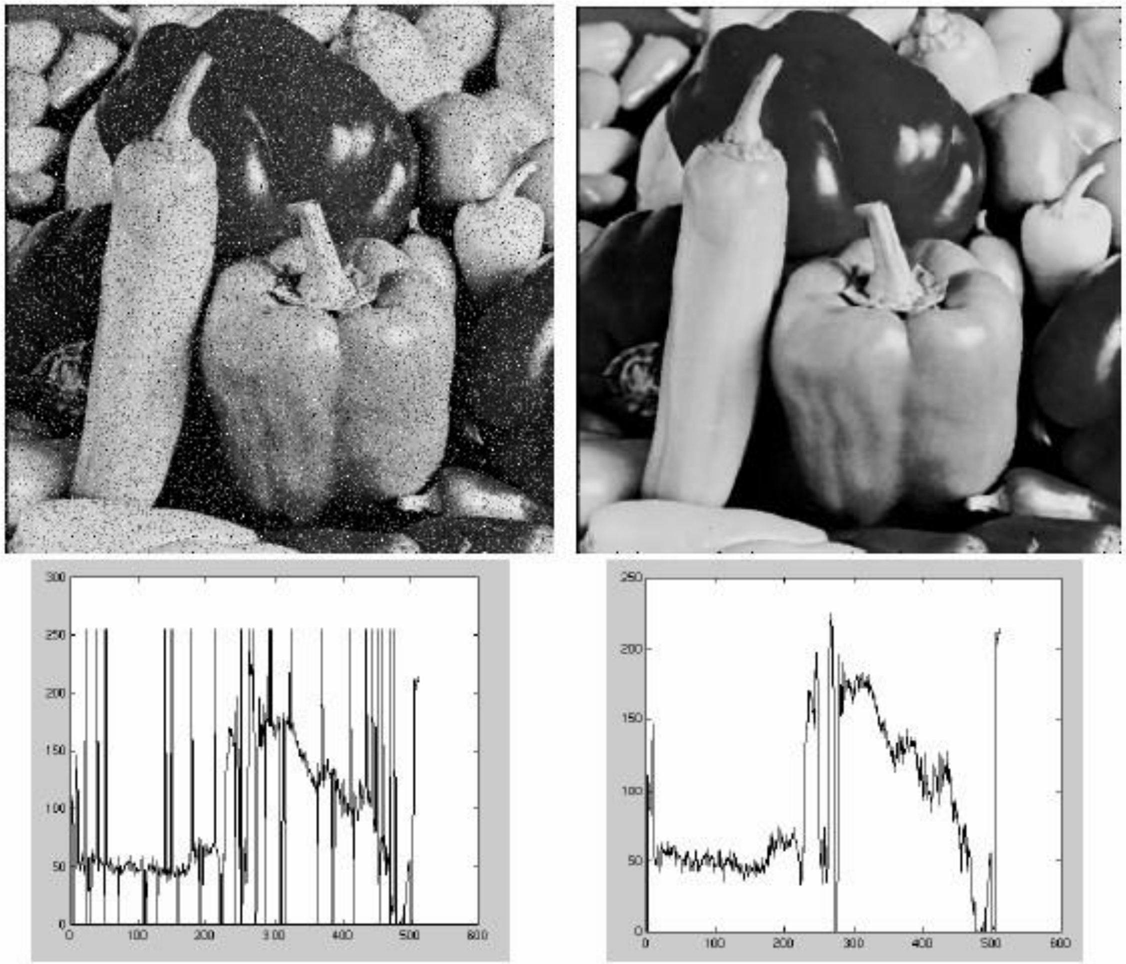
- What advantage does median filtering have over Gaussian filtering?
  - Robustness to outliers

filters have width 5 :

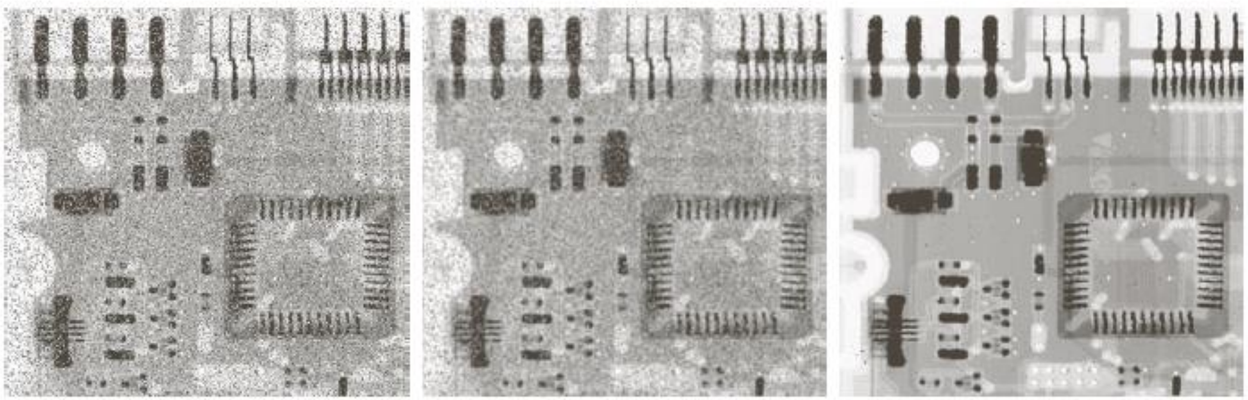


Smoothing Spatial Filters

Salt-and-pepper noise Median filtered



Smoothing Spatial Filters



a b c

FIGURE 3.35 (a) X-ray image of circuit board corrupted by salt-and-pepper noise. (b) Noise reduction with a 3 × 3 averaging mask. (c) Noise reduction with a 3 × 3 median filter. (Original image courtesy of Mr. Joseph E. Pascente, Lixi, Inc.)

Sharpening Spatial Filters

- Image sharpening is used to highlight fine detail in an image or to enhance detail that has been blurred
  - Blurred detail in the image is either in error or as a nature effect of a particular method of image acquisition
- Applications of image sharpening
  - Electronic printing, medical imaging, industrial inspection, autonomous guidance in military systems
- Principle of image sharpening
  - Image differentiation can enhance edges and other discontinuities (such as noise) and deemphasize areas with slowly varying gray-level values

Sharpening Spatial Filters

- Foundation
  - Any definition of a *first derivative*
    - Must be zero in areas of constant intensity
    - Must be nonzero at the onset of an intensity step or ramp
    - Must be nonzero along ramps
  - Any definition of a *second derivative*
    - Must be zero in constant areas
    - Must be nonzero at the onset and end of an intensity step or ramp
    - Must be zero along ramps of constant slope

Sharpening Spatial Filters

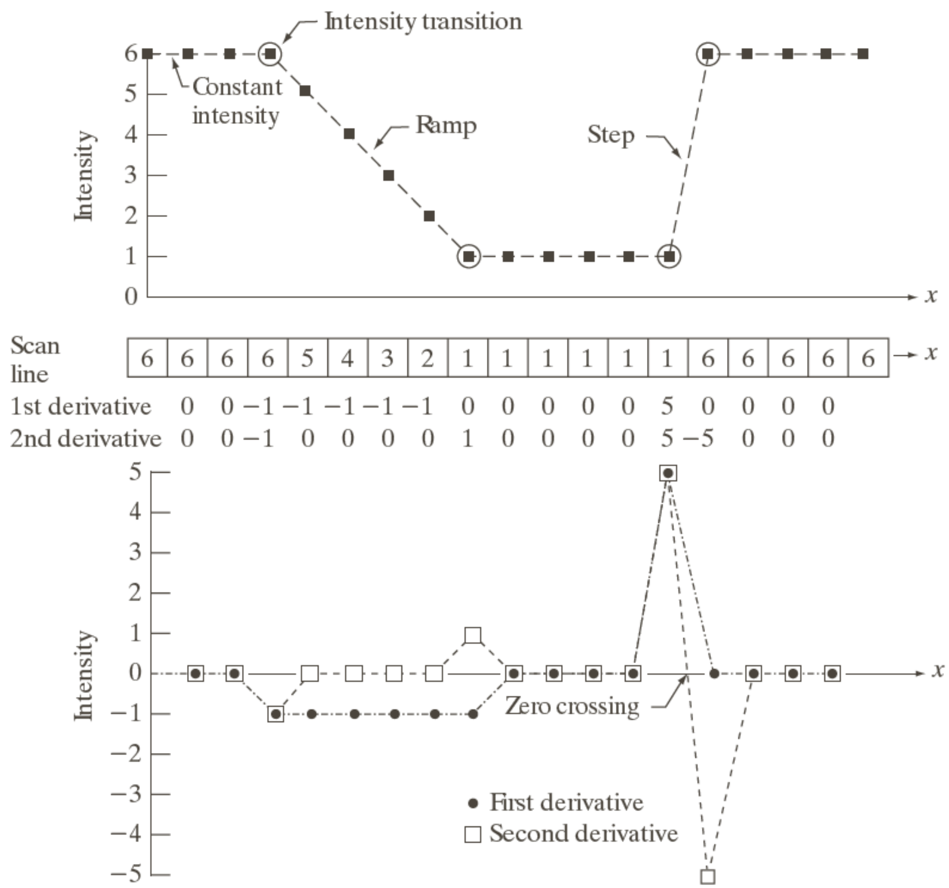
- First-order derivative of a one-dimensional function is the difference

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

- Second-order derivative of a one-dimensional function is the difference

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$

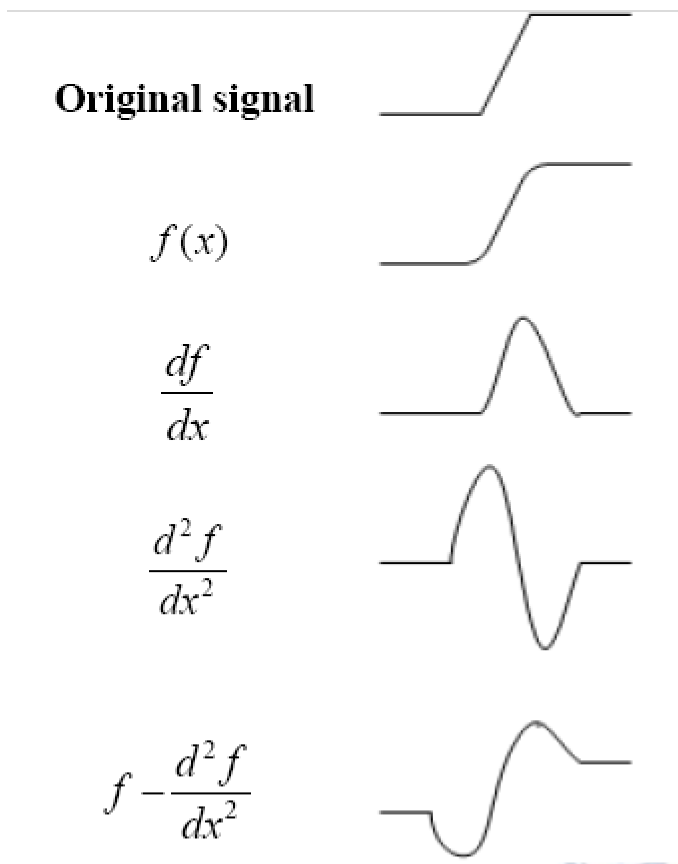
Sharpening Spatial Filters



a
b
c

FIGURE 3.36 Illustration of the first and second derivatives of a 1-D digital function representing a section of a horizontal intensity profile from an image. In (a) and (c) data points are joined by dashed lines as a visualization aid.

Sharpening Spatial Filters



Sharpening Spatial Filters

- The Laplacian
  - Laplacian operator

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

  - X-direction

$$\frac{\partial^2 f}{\partial x^2} = f\left(x+1,y\right) + f\left(x-1,y\right) - 2f\left(x,y\right)$$

  - Y-direction

$$\frac{\partial^2 f}{\partial y^2} = f\left(x,y+1\right) + f\left(x,y-1\right) - 2f\left(x,y\right)$$

Sharpening Spatial Filters

- The discrete Laplacian of two variables

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

(a)

0	1	0
1	−4	1
0	1	0

(b)

1	1	1
1	−8	1
1	1	1

(c)

0	−1	0
−1	4	−1
0	−1	0

(d)

−1	−1	−1
−1	8	−1
−1	−1	−1

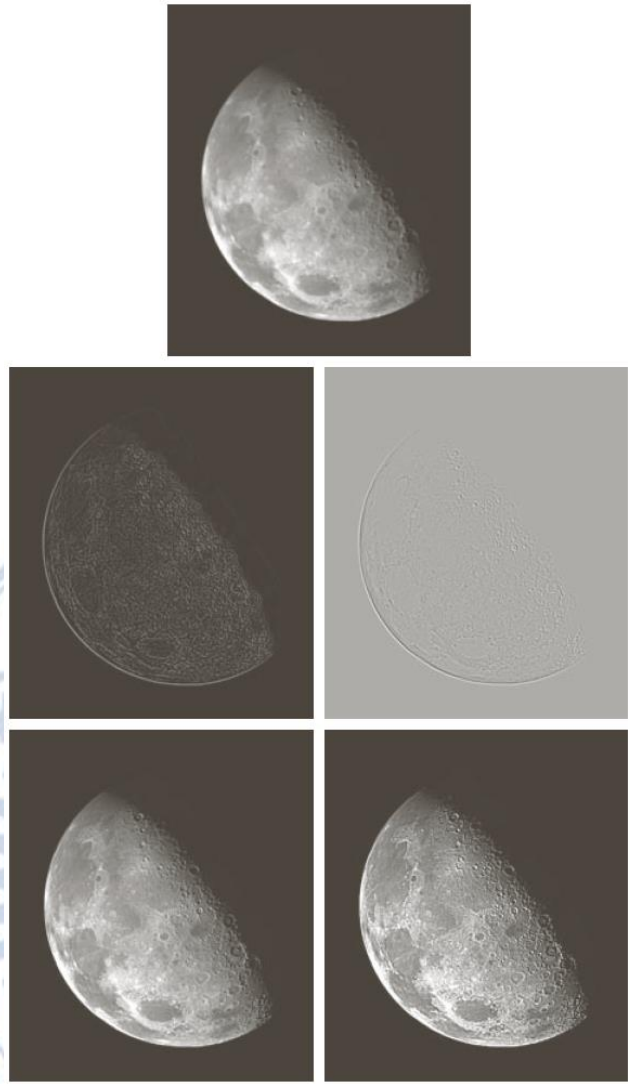
a b
c d

FIGURE 3.37 (a) Filter mask used to implement Eq. (3.6-6). (b) Mask used to implement an extension of this equation that includes the diagonal terms. (c) and (d) Two other implementations of the Laplacian found frequently in practice.

Sharpening Spatial Filters

- For image sharpening

$$g(x,y) = f(x,y) + c\left[\nabla^2 f(x,y)\right]$$



a
b c
d e

FIGURE 3.38 (a) Blurred image of the North Pole of the moon. (b) Laplacian without scaling. (c) Laplacian with scaling. (d) Image sharpened using the mask in Fig. 3.37(a). (e) Result of using the mask in Fig. 3.37(b). (Original image courtesy of NASA.)

Sharpening Spatial Filters

- Unsharp masking and highboost filtering
  - Unsharp masking
    - Blur the original image
    - Subtract the blurred image from the original (the resulting difference is called the mask)
    - Add the mask to the original

$$g_{mask}(x,y) = f(x,y) - \bar{f}(x,y)$$

$$g(x,y) = f(x,y) + k \bullet g_{mask}(x,y)$$

$-$

$=$

$+0.4$

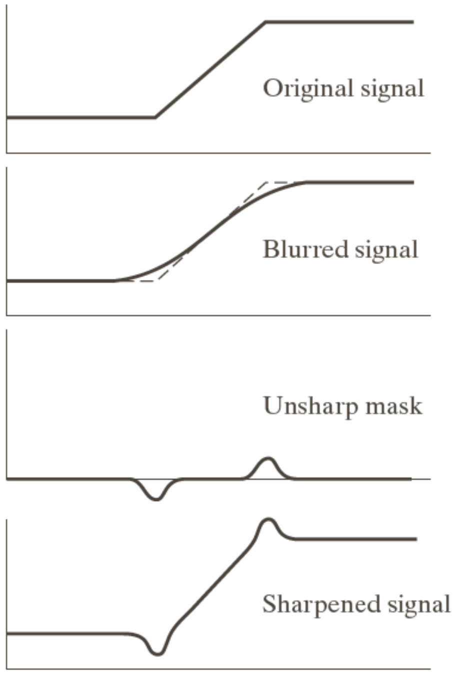
$=$

So, what does blurring take away?

Unsharp Masking (MATLAB)

```
image=imread('lena512.bmp');
im=mat2gray(image);
g= fspecial('gaussian', 25,4);
imblur = conv2(im,g,'same');
figure; imshow(imblur);
figure; imshow(im-imblur);
figure; imshow(im-imblur,[]);
figure; imshow(im+0.4*(im-imblur));
```

Sharpening Spatial Filters



a
b
c
d

FIGURE 3.39 1-D illustration of the mechanics of unsharp masking. (a) Original signal. (b) Blurred signal with original shown dashed for reference. (c) Unsharp mask. (d) Sharpened signal, obtained by adding (c) to (a).

a
b
c
d
e

FIGURE 3.40
(a) Original image.
(b) Result of blurring with a Gaussian filter.
(c) Unsharp mask. (d) Result of using unsharp masking.
(e) Result of using highboost filtering.

Sharpening Spatial Filters

- Using first-order derivatives for image sharpening – the gradient
  - The gradient of *f* at (*x*, *y*)

$$\nabla f \equiv grad(f) \equiv \begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

  - Magnitude

$$M(x,y) = mag(\nabla f) = \sqrt{g_x^2 + g_y^2}$$
$$\approx |g_x| + |g_y|$$

Sharpening Spatial Filters

(a)

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

(b)

−1	0
0	1

(c)

0	−1
1	0

(d)

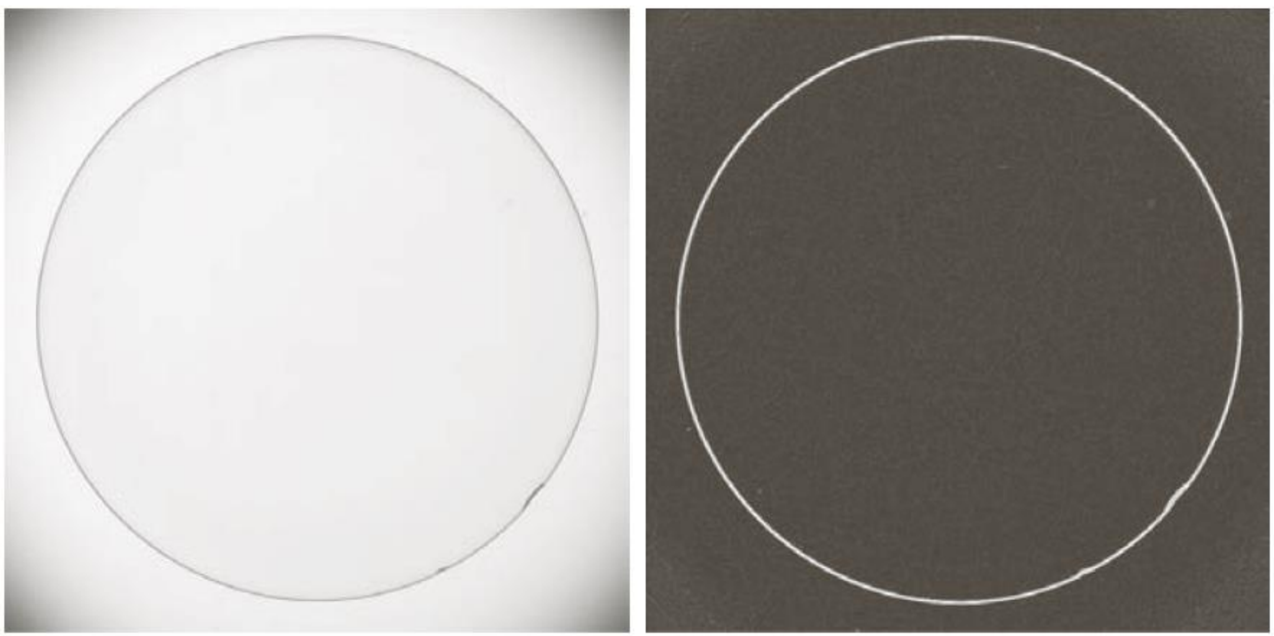
−1	−2	−1
0	0	0
1	2	1

(e)

−1	0	1
−2	0	2
−1	0	1

FIGURE 3.41 A 3×3 region of an image (the zs are intensity values). (b)–(c) Roberts cross gradient operators. (d)–(e) Sobel operators. All the mask coefficients sum to zero, as expected of a derivative operator.

Sharpening Spatial Filters



a b

FIGURE 3.42 (a) Optical image of contact lens (note defects on the boundary at 4 and 5 o'clock). (b) Sobel gradient. (Original image courtesy of Pete Sites, Perceptics Corporation.)

Homework

problem 3.7, 3.11, 3.23

Requirements of Project One now posted!

Thank You!