IMAGE PROCESSING(ACS511)

B. Tech V semester (Autonomous IARE R-16)

BY

Dr. R Obulakonda Reddy

Associate Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

DUNDIGAL, HYDERABAD - 500 043

UNIT 1
INTRODUCTION

Course learning outcomes

CLO's	Course Learning outcomes
CLO1	Understand the key concepts of Image Processing.
CLO2	Identify the origins of the Digital image processing
CLO3	Demonstrate the scope of the digital image processing in multiple fields
CLO4	Explore on overview of the components contained in the general purpose image processing system and its use in real time applications
CLO5	Describe the concept of elements of visual perception.

What is digital Imageprocessing?

Image –A two-dimensional signal that can be observed by human visual system

Digital image – Representation of images by sampling in time and space.

Digital image processing – Perform digital signal processing operations on digital images

What is digital image processing?(contd..)

- An image may be defined as a two- dimensional function, f(x,y) where x and y are spatial (plane) coordinates, and the amplitude off at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point

- When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image adigitalimage.

What is digital image processing?(contd..)

- A digital image is composed of a finite number of elements, each of which has a particular location and value

- These elements are referred to as picture elements, image elements and pixels.

- Pixel is the term most widely used to denote the elements of a digital image.

Origins of digital imageprocessing

- One of the first applications of digital images was in the newspaper industry, when pictures were first sent by submarine cable between London and New York.
- Specialized printing equipment coded pictures for cable transmission and then reconstructed them at the receiving end.
- Image was transmitted in this way and reproduced on a telegraph printer fitted with typefaces simulating a halftone pattern.

Origins of digital image processing(contd..)

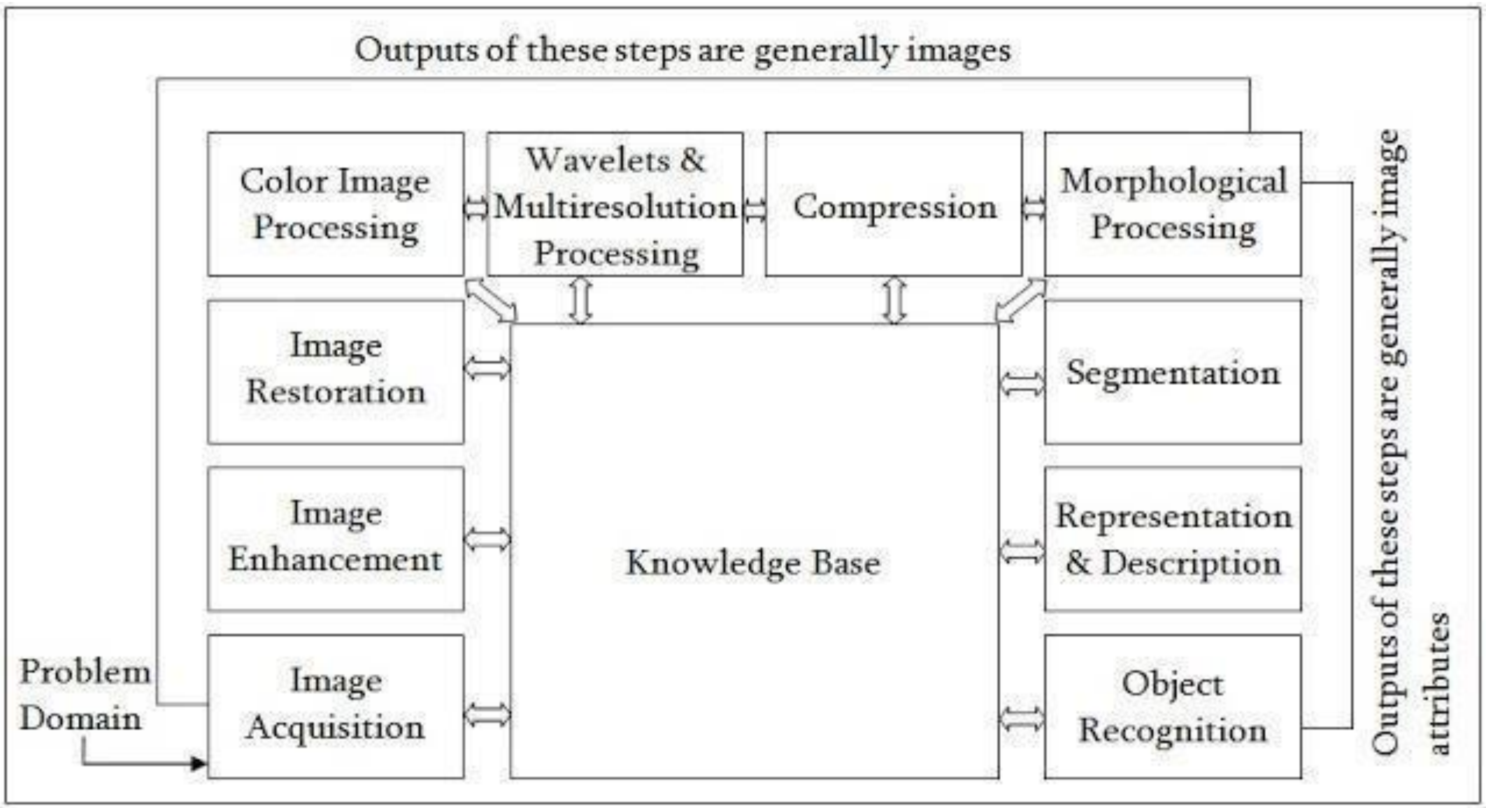
- The printing technique based on photographic reproduction made from tapes perforated at the telegraph receiving terminal from 1921.
- The improvements are tonal quality and in resolution.
- The early Bartlane systems were capable of coding images in five distinct levels of gray.
- This capability was increased to 15 levels in 1929.

Examples of fields that use DIP

- Gamma ray imaging
- X-ray Imaging (oldest source of EM radiation)
- Imaging in the visible and infrared bands
- Imaging in the microwave band
- Imaging in the radio band
- Other Imaging Modalities Acoustic images, electron microscopy and synthetic (computer – generated images)

Fundamental stepsin DIP

There are some fundamental steps but as they are fundamental, all these steps may have sub-steps. The fundamental steps are described below with aneatdiagram.



Fundamental steps in DIP(Contd..)

1. **Image Acquisition:**

This is the first step or process of the fundamental steps of digital image processing. Image acquisition could be as simple as being given an image that is already in digital form. Generally, the image acquisition stage involves pre- processing, such as scaling etc.

2. **Image Enhancement:**

Image enhancement is among the simplest and most appealing areas of digital image processing. Basically, the idea behind enhancement techniques is to bring out detail that is obscured, or simply to highlight certain features of interest in an image. Suchas, changing brightness & contrast etc.

3. **Image Restoration:**

Image restoration is an area that also deals with improving the appearance of an image. However, unlike enhancement, which is subjective, image restoration is objective, in the sense that restoration techniques tend to be based on mathematical or probabilistic models of image degradation.

Fundamental stepsin DIP(Contd..)

4. **Color ImageProcessing:**Color image processing is an area that has been gaining its importance because of the significant increase in the use of digital images over the Internet. This may include color modeling and processing in adigital domainetc.
5. **Wavelets and Multi-ResolutionProcessing:**Wavelets are the foundation for representing images in various degrees of resolution. Images subdivision successively into smaller regions for data compression and for pyramidalrepresentation.
6. **Compression:**Compression deals with techniques for reducing the storage required to save an image or the bandwidth to transmit it. Particularly in the uses of internet it is very much necessary to compressdata.
7. **Morphological Processing:** Morphological processing deals with tools forextracting image components that are useful in the representation and description of shape.

Fundamental stepsin DIP(Contd..)

8. **Segmentation:** Segmentation procedures partition an image into its constituent parts or objects. In general, autonomous segmentation is one of the most difficult tasks in digital image processing.
9. **Representation and Description:** Representation and description almost always follow the output of a segmentation stage, which usually is raw pixel data, constituting either the boundary of a region or all the points in the region itself. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computerprocessing.
10. **Object recognition:** Recognition is the process that assigns a label, such as, "vehicle" to an object based on its descriptors.
11. **Knowledge Base:** Knowledge may be as simple as detailing regions of an image where the information of interest is known to be located, thus limiting the search that has to be conducted in seeking that information.

Components of image processing system

Figure shows different components of image processing system.

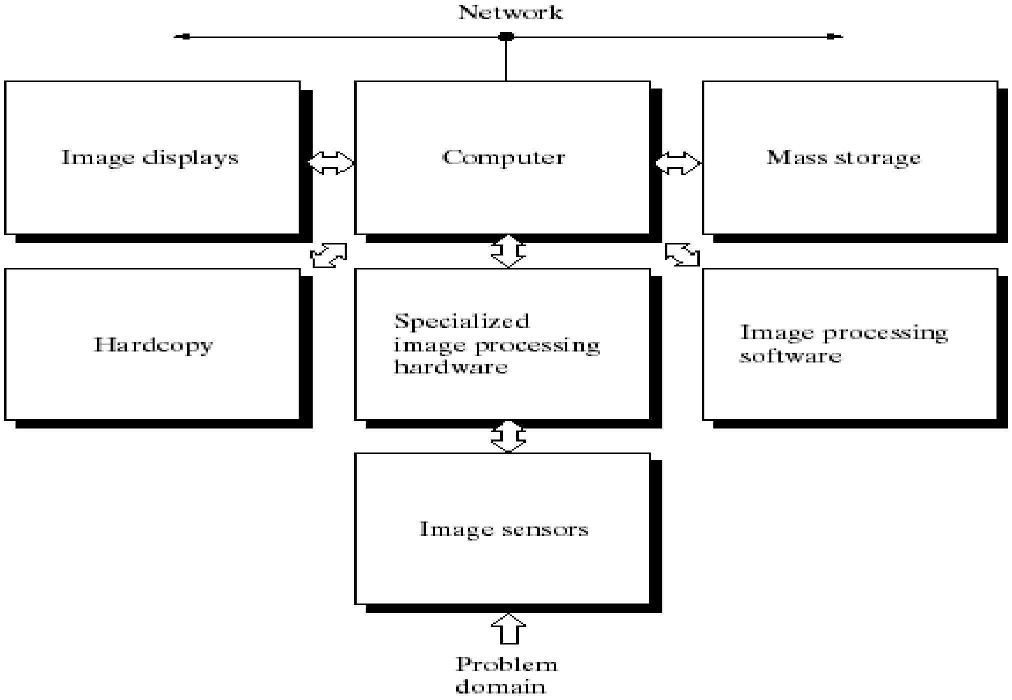


FIGURE 1.24 Components of a general-purpose image processing system.

Components of image processing systemimageprocessing

1. In sensing, two elements are required to acquire digital images. The first is physical device that is sensitive to the energy radiated by the object we wish to image. The second called adigitizer, is adevice for converting the output of the physical sensing device into digital form.
2. Specialized image processing hardware usually consists of the digitizer plus hardware that performs other primitive operations such as arithmetic and logical operations (ALU).
3. The computer is an image processing system is a general purpose to supercomputer
4. Software which include image processing specialized modules thatperform specific tasks.

Components of image processing

5. Mass storage capability is a must in image processing applications.
6. Image displays in use today are mainly color tv monitors.
7. hardcopy devices for recording images include laser printers, film cameras, inkjet units and cdrom
8. Networking for communication

Elements of visual perception

Structure of human eye:

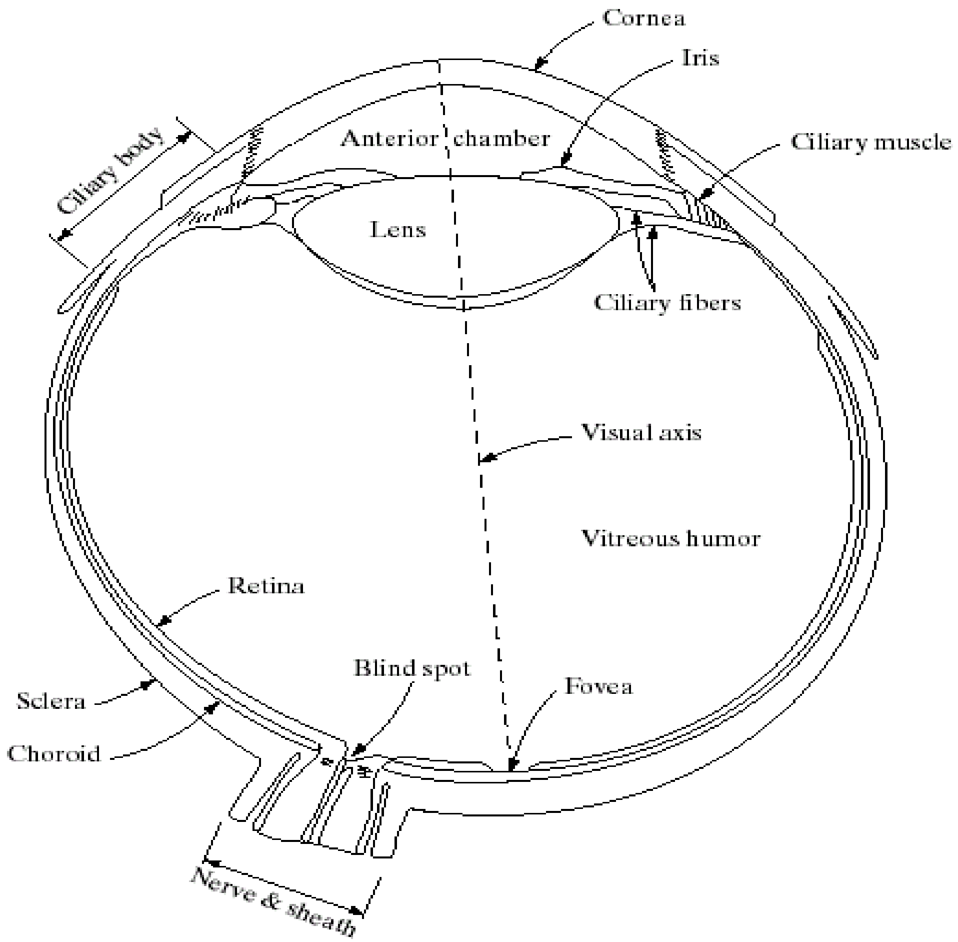


FIGURE 2.1 Simplified diagram of a cross section of the human eye.

Elements of visual perception(Contd..)

The eye is nearly a sphere, with an average diameter of approximately 20mm.

Three membranes enclose the eye:

1. **Cornea:** The cornea is a tough, transparent tissue that covers the anterior surface of the eye.
2. **Sclera:** sclera is an opaque membrane that encloses the remainder of the optic globe.
3. **Choroid:** The choroid lies directly below the sclera. This membrane contains a net- work of blood vessels that serve as the major source of nutrition to the eye. The choroid coat is heavily pigmented and hence helps to reduce the amount of extraneous light entering the eye and the backscatter within the optical globe.

Elements of visual perception(Contd..)

- The lens is made up of concentric layers of fibrous cells and is suspended by fibers that attach to the ciliary body. It contains 60 to 70% water, about 6% fat, and more protein than any other tissue in the eye.
- The innermost membrane of the eye is the retina, which lines the Inside of all entire posterior portion.
- When the eye is properly focused, light from an object outside the eye is imaged on the retina. Pattern vision is afforded by the distribution of discrete light receptors over the surface of the retina.

There are two classes of receptors: cones and rods.

- The cones in each eye number between 6 and 7 million
- They are located primarily in the central portion of the retina, called the fovea, and are highly sensitive to color.

Elements of visual perception(Contd..)

- Muscles controlling the eye rotate the eyeball until the image of an object of interest falls on the fovea.
- Cone vision is called photopic or bright-light vision.
- The number of rods is much larger: Some 75 to 150 million are distributed over the retinal surface.

Elements of visual perception(Contd..)

Image formation in theeye:

- Theprincipal difference between the lens of the eye and an ordinary optical lens is thatthe former is flexible.
- The shape of the lens is controlled by tension in the fibers of the ciliary body.
- To focus on distant objects, the controlling muscles cause the lens to be relatively flattened.
- Similarly, these muscles allow the lens to becomethicker in order to focus on objects near the eye.
- The distance between the center of the lens and the retina called the focal length varies from approximately 17 mm to about 14 mm, as the refractive power of the lens increases from its minimum to its maximum.

Elements of visual perception(Contd..)

- When the eye focuses on an object farther away the lens exhibits its lowest refractive power.
- When the eye focuses on a near by object, the lens is most strongly refractive.
- For example, the observer is looking at a tree 15 m high at a distance of 100m.
- If h is the height in mm of that object in the retinal image, the geometry of Fig. yields 15/100 = h/17 or h=2.55mm.

Simple image formation model

- Images are represented by two-dimensional functions of the form f(x,y).
- The value or amplitude of f at spatial coordinates (x,y) gives the intensity (brightness) of the image at that point.
- As light is a form of energy, f(x,y) must be nonzero and finite.

The function f(x,y) may be characterized by two components:

1.The amount of source illumination incident on the scene being viewed

2.The amount of illumination reflected by the objects in the scene.

These are called the *illumination and reflectance components and are denoted by i(x,y) and r(x,y), respectively.*

Simple image formation model(Contd..)

- The two functions combine asa product to formf(x,y):f(x,y)=i(x,y)r(x,y)

  r(x,y)=0 --- total absorption

  1 --- total reflection

- The intensity of a monochrome image f at any coordinates(x,y) the *gray level(l)of the imageat that point.*
- *That is,l=f(x0,y0)Llies in therange*

Lmin≤l≤ Lmax where Lmin= lmin rmin and Lmax=lmax rmax

Sampling and quantization

- To create an image which is digital, we need to convert continuous data into digital form. There are two steps in which it is done.

  1. Sampling

  2. Quantization

- Since an image is continuous not just in its co-ordinates (x axis), but also in its amplitude (y axis), so the part that deals with the digitizing of co-ordinates is known as sampling. and the part that deals with digitizing the amplitude is known as quantization.

Representing digital images

A digital image can be represented in matrix form:

	0	1	2	3	...	N-1
0	$f(0,0)$	$f(0,1)$	$f(0,2)$	$f(0,3)$		$f(0,N-1)$
1	$f(1,0)$	$f(1,1)$	$f(1,2)$	$f(1,3)$		$f(1,N-1)$
2	$f(2,0)$	$f(2,1)$	$f(2,2)$	$f(2,3)$		$f(2,N-1)$
3	$f(3,0)$	$f(3,1)$	$f(3,2)$	$f(3,3)$	...	$f(3,N-1)$
...	$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$
M-1	$f(M-1,0)$	$f(M-1,1)$	$f(M-1,2)$	$f(M-1,3)$		$f(M-1,N-1)$

Representing digital Images(Contd..)

The number of gray levels is chosen to be a power of 2 for practical reasons: L=2n, which generates gray values ranging from lmin=0 to l max=2n-1

We assume that the discrete levels are equally spaced and that they are integers in the interval [0, L-1] .

Sometimes the range of values spanned by the gray scale is called the dynamic range of an

- The number of sampling points N, M is set by the sensor array.
- The number, b, of bits required to store a digitized image is b=N*M*n

Spatial and Gray-level Resolution

Spatial resolution is the smallest level of detail discernable in an image
Number of line pairs per millimeter, say100 line pairs per millimeter.

Gray-level resolution is the smallest discernable change in gray level.
Very subjective.

Spatial and Gray-level Resolution(Contd..)

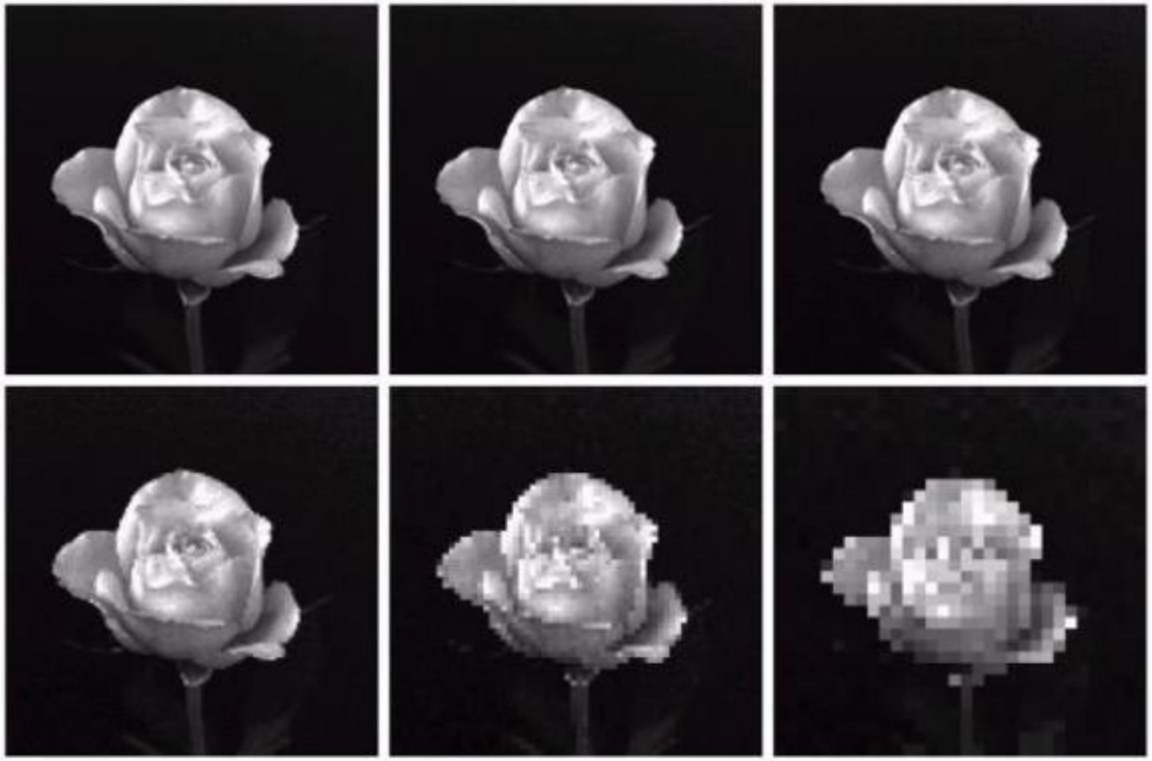


FIGURE 2.20 (a) 1024 × 1024, 8-bit image. (b) 512 × 512 image resampled into 1024 × 1024 pixels by row and column duplication. (c) through (f) 256 × 256, 128 × 128, 64 × 64, and 32 × 32 images resampled into 1024 × 1024 pixels.

Spatial and Gray-level Resolution(Contd..)

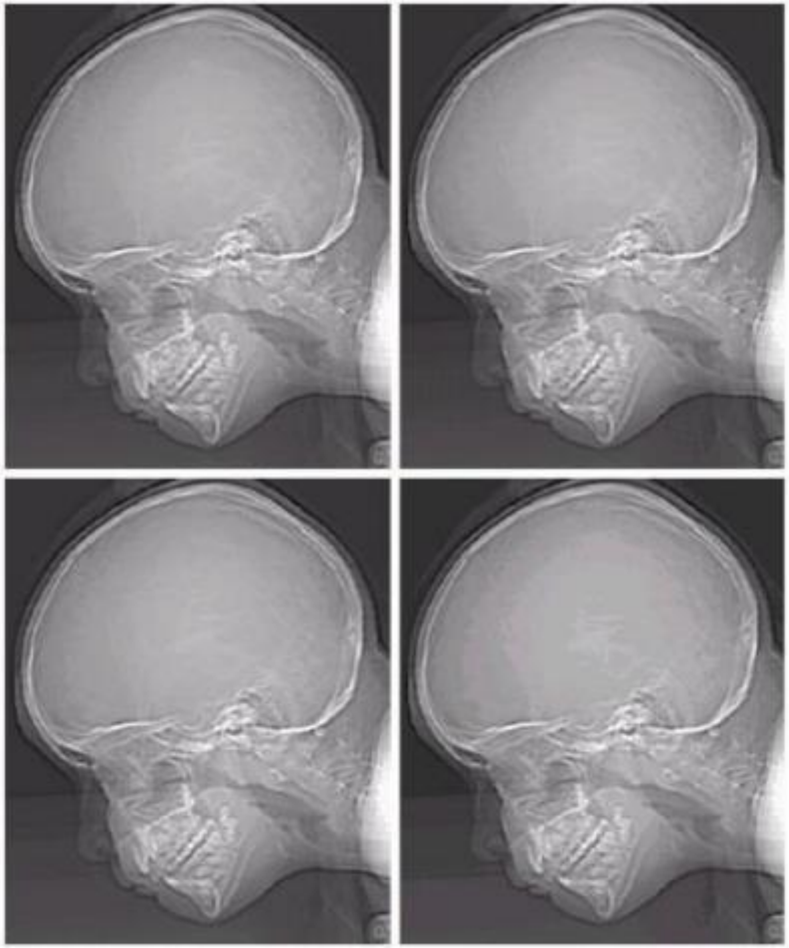


FIGURE 2.21 (a) 452 × 374, 256-level image. (b)–(d) Image displayed in 128, 64, and 32 gray levels, while keeping the spatial resolution constant.

Zooming and Shrinking Digital Images

Zooming:-It may be viewed as oversampling. Increasing no-of pixels in an image so that image appears larger.

It requires two steps:

- Creation of new pixels
- Assignment of gray level to those new locations.

Zooming and Shrinking Digital Images(Contd..)

Zooming Methods

- Nearest neighbor interpolation
- Bilinear interpolation
- K-times zooming

Shrinking:-It may be viewed as under sampling.

It is performed by row-column deletion

Basic relationshipsbetween pixels

- Neighborhood
- Adjacency
- Connectivity
- Paths
- Regions andboundaries

Basic relationshipsbetween pixels(Contd..)

Neighboursofpixel

- Any pixel p(x, y) has two vertical and two horizontal neighbors, given by (x+1, y), (x-1, y),(x, y+1), (x,y-1)
- This set of pixels are called the 4-neighbors of P, and is denoted by N4(P).
- Eachof them areat aunit distancefromP.

Basic relationships between pixels(Contd..)

- The four diagonal neighbors of p(x,y) are given by, (x+1, y+1), (x+1, y-1), (x-1, y+1), (x-1 ,y-1)
- This set is denoted by N D(P).
- The points ND(P) and N4(P) are together known as 8-neighbors of the point P, denoted by N8(P).
- Some of the points in the N4, ND and N8 may fall outside image when P lies on the border of image.

Basic relationshipsbetween pixels(Contd..)

- Two pixels are connected if they are neighbors and their gray levels satisfy some specified criterion ofsimilarity.
- For example, in a binary image two pixels are connected if they are 4-neighbors and have samevalue(0/1).

Let V be set of gray levels values used to defineadjacency.

<u>4-adjacency</u>: Two pixels p and q with values from V are 4- adjacent if q is in the set N4 (p).

<u>8-adjacency:</u> Two pixels p and q with values from V are 8- adjacent if q is in the set N8 (p).

<u>m-adjacency</u>: Two pixels p and q with values from V are m- adjacent if, – q is in N 4(P). – q is in N D(p) and the set [ ] is empty (has no pixels whose values are from V).

Basic relationshipsbetween pixels(Contd..)

Connectivity:

Let V be the set of gray-level values used to define connectivity; then Twopixels p, q that have values from the setVare:

a. 4-connected, ifq isin the set N4(p)

b. 8-connected, if q isin the set N8(p)

c. m-connected,iff

i. q isin N4(p)or

ii. q is in ND(p) and the set N4 (p)∩N4(q) is empty.

Basic relationshipsbetween pixels(Contd..)

Paths

- Apath 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), (x_2,y_2) …(x_n, y_n), where (x_0, y_0)=(x, y) and (x_n,y_n)=(s, t); (x_i, y_i) is adjacent to (x_{i-1}, y_{i-1}) ,$1 \le i \le n$.
- Heren isthe length of the path.
- Wecandefine 4-, 8-, and m-paths basedon type of adjacencyused.

Basic relationshipsbetween pixels(Contd..)

Regions and Boundaries

- A subset R of pixels in an image is called a Region of the image if Ris aconnectedset.
- The boundary of the region Ris the set of pixels in the region tha t have one or more neighbors that are not in R.

Distance measures

Given pixels p, q and z with coordinates (x, y), (s, t), (u, v) respectively, the distance function Dhasfollowing properties:

a. $D(p, q) \geq 0$ [$D(p, q) = 0$, iff $p = q$]

b. $D(p, q) = D(q, p)$

c. $D(p, z) \leq D(p, q) + D(q, z)$

Basic relationshipsbetween pixels(Contd..)

Thefollowing are the different Distance measures:

1. Euclidean Distance:

$$De(p,q) = [(x-s)(X-s)+(y-t)(y-t)]$$

2. City Block Distance:

$$D4(p,q) = |x-s| + |y-t|$$

3. Chess Board Distance:

$$D8(p,q) = \max(|x-s|, |y-t|)$$

Linear and Nonlinear operations

- General operator, H, that performs an output image, $g(x,y)$, for a given input image, $f(x,y)$

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

- H is said to be linear operator if

  $$H[a_i f_i(x,y)+a_j f_j(x,y)]=a_i H[f_i(x,y)]+a_j H[f_j(x,y)]=a_i g_i(x,y)+a_j g_j(x,y)$$

  where a_i, a_j are arbitrary constants and $f_i(x,y), f_j(x,y)$ are images of same size.

- For example sum is a linear operator and max is nonlinear operator

UNIT 2
IMAGE ENHANCEMENT IN SPATIAL DOMAIN

Course Learning Outcome

CLOs	Course Learning Outcome
CLO6	Use the concept of sampling and quantization in generating digital images
CLO7	Explore on the basic relationships existed between the pixels in the image
CLO8	Illustrate different mathematical tools used in image intensity transformations for quality enhancement
CLO9	Use histogram processing techniques in image enhancement and noise reduction

Image Enhancement

- The principal objective of enhancement is to process an image so that the result is more suitable for a special process
- Image Enhancement Fall Into two categories: Enhancement in spatial domain and Frequency domain.
- The term spatial domain refers to the Image Plane itself which is DIRECT manipulation of pixels.
- Frequency domain processing techniques are based on modifying the Fourier transform of an image.

Image Enhancement in Spatial domain

- Spatial Domain=Aggregate of pixels composing animage.
- Spatial Domain Methods=Procedures that operate directly on these pixels. Denoted by: g(x,y)=T[f(x,y)]

F(x,y) : Input Image,

T:Operator on Image

g(x,y): ProcessedImage.

Talso can operate on aset of Images.

<u>Definition of Neighborhood</u>:

Input for Process: A neighborhood about a point (x,y). The simplest form of input is a one pixel neighborhood. s=T(r) T:Transformation Function s,r : gray level of f(x,y) andg(x,y) respectively.

Basic gray-level transformations

Some basic gray-level transformations are

1. Image Negatives
2. Log transformation
3. Power-law transformation(Gamma transformation)
4. Piece-wise linear transformation

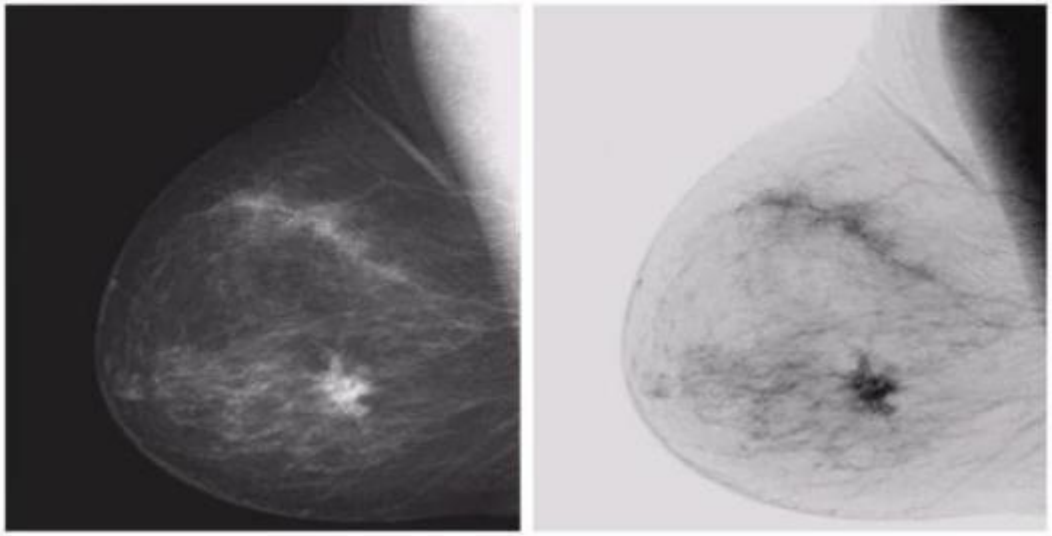
Image Negatives

- Image transformation is given by the followingequation

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

L-1 maximum gray level

- It Produces photographic negative. Some details are easierto spot if we go from black and white to white and black.



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.)

Logtransformation

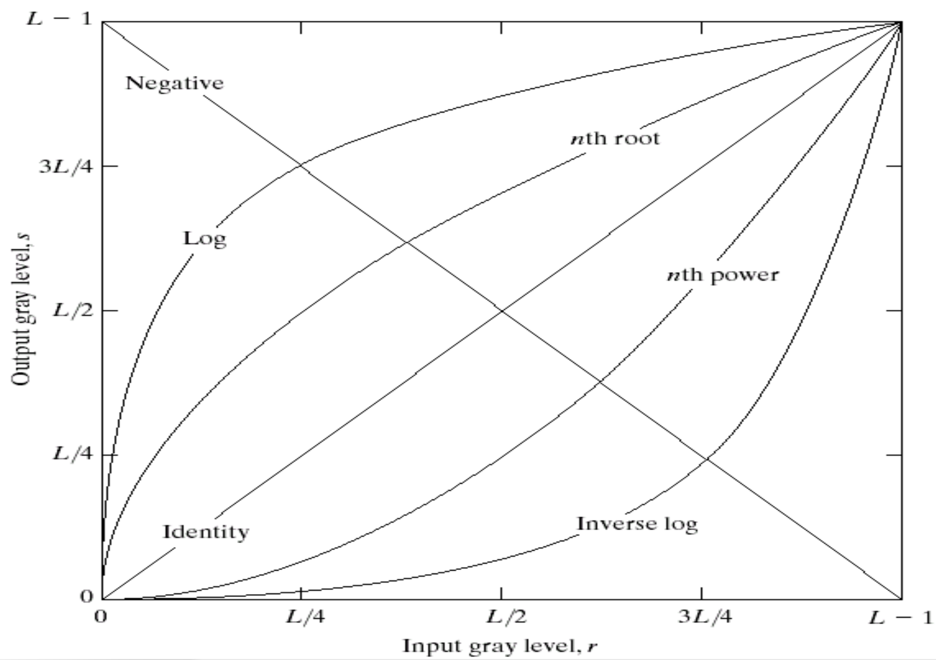
Logtransformation is given by

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

Inverse Logtransformation is given by

$$T(r) = \exp(r/c) - 1$$

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



Basic gray-level transformations (Contd..)

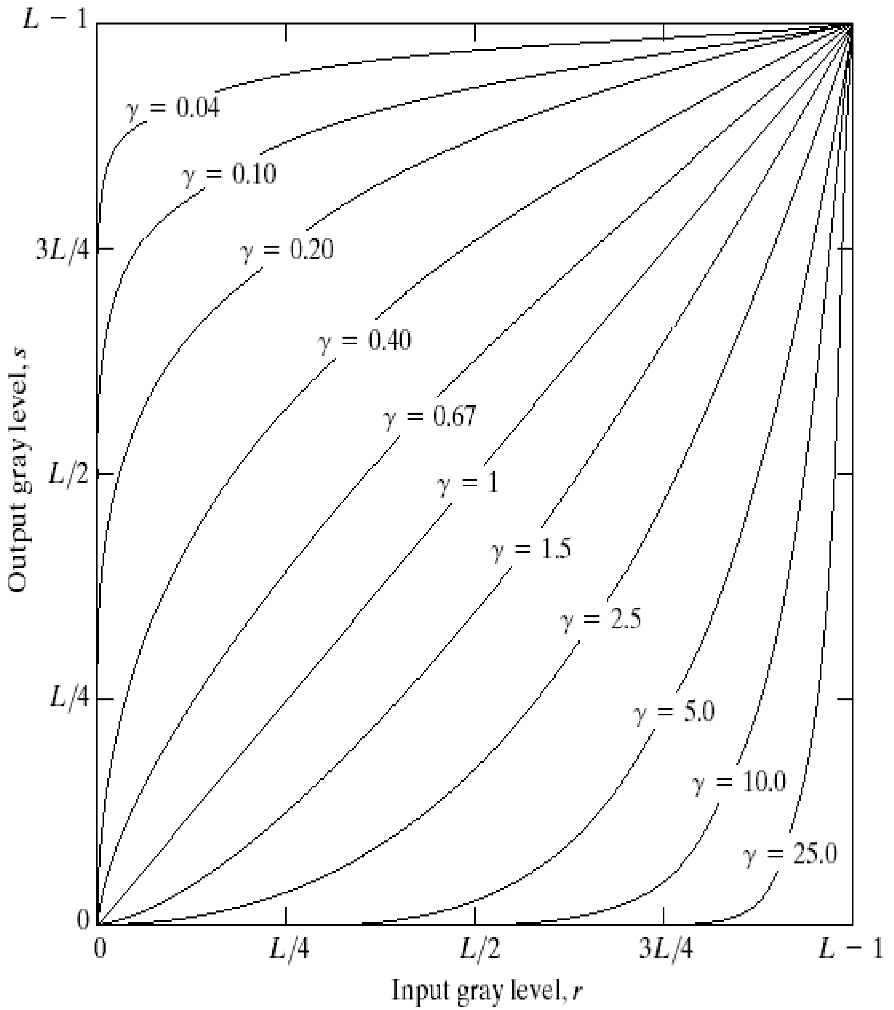
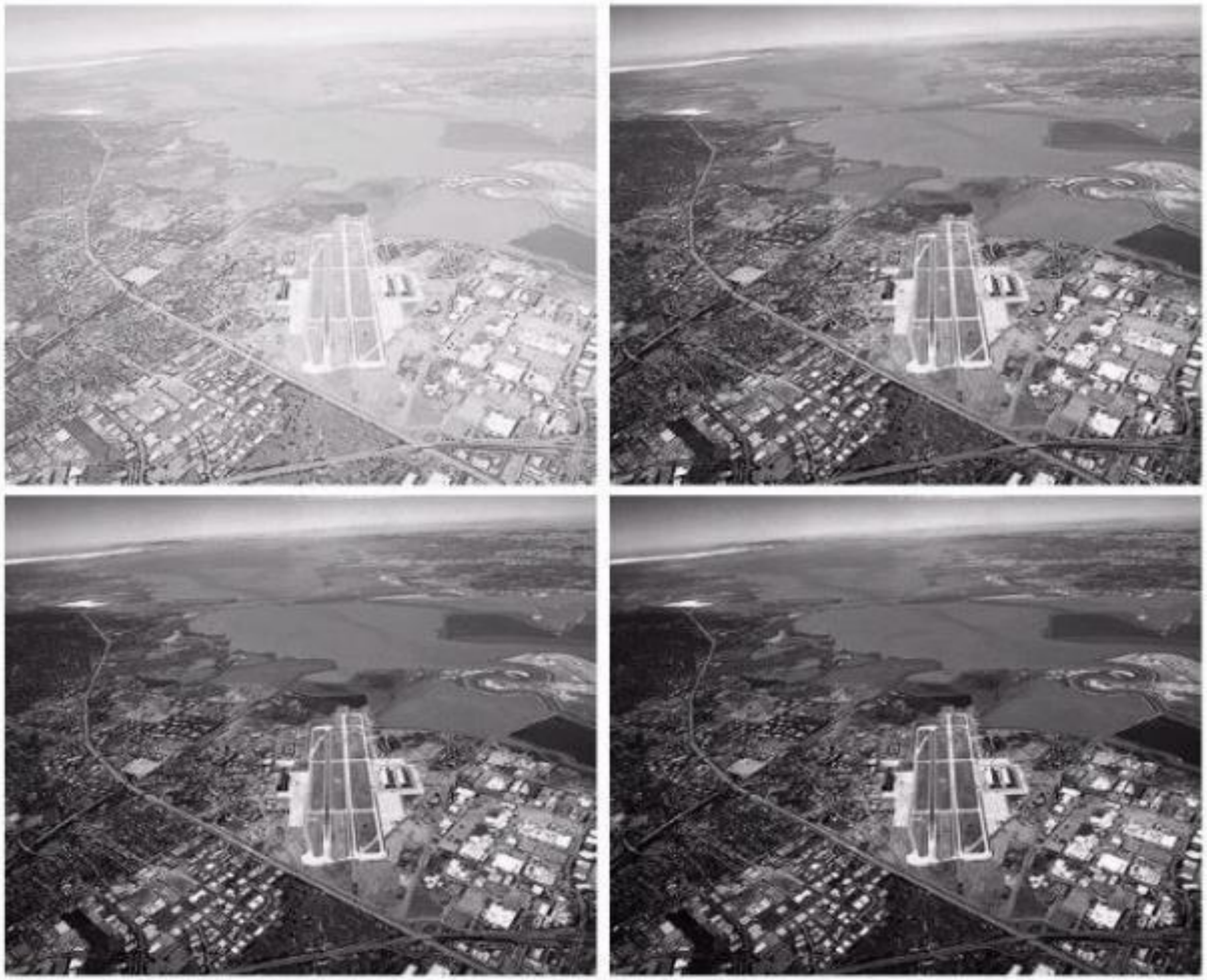


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases).

Basic gray-leveltransformations(Contd..)



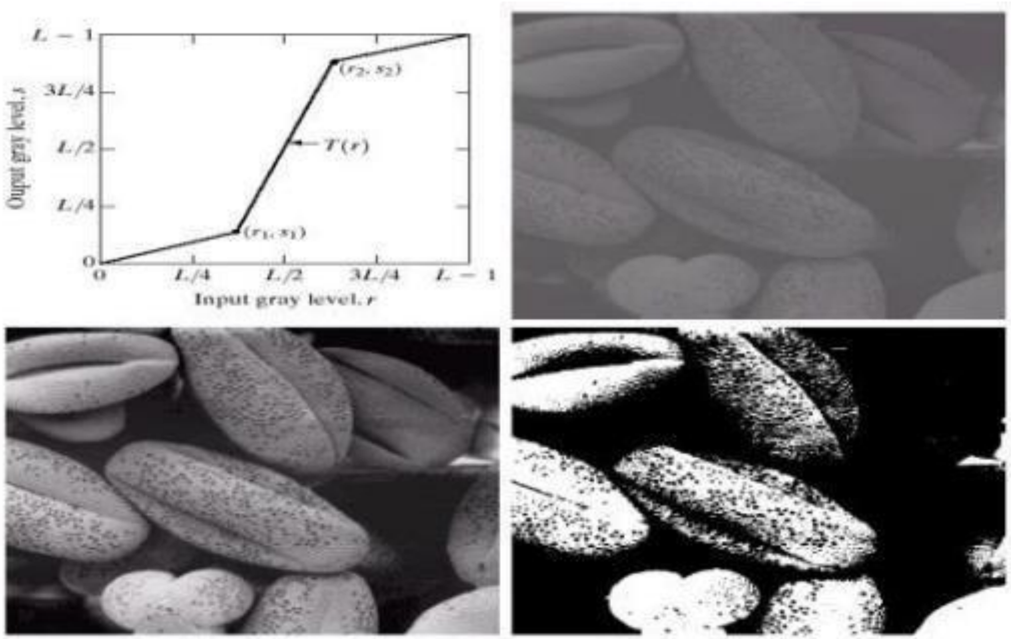
a b
c d

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.)

Basic gray-level transformations(Contd..)

- Piecewise-linear transformation functions
  - The form of piecewise functions can be arbitrarily complex

Contrast stretching

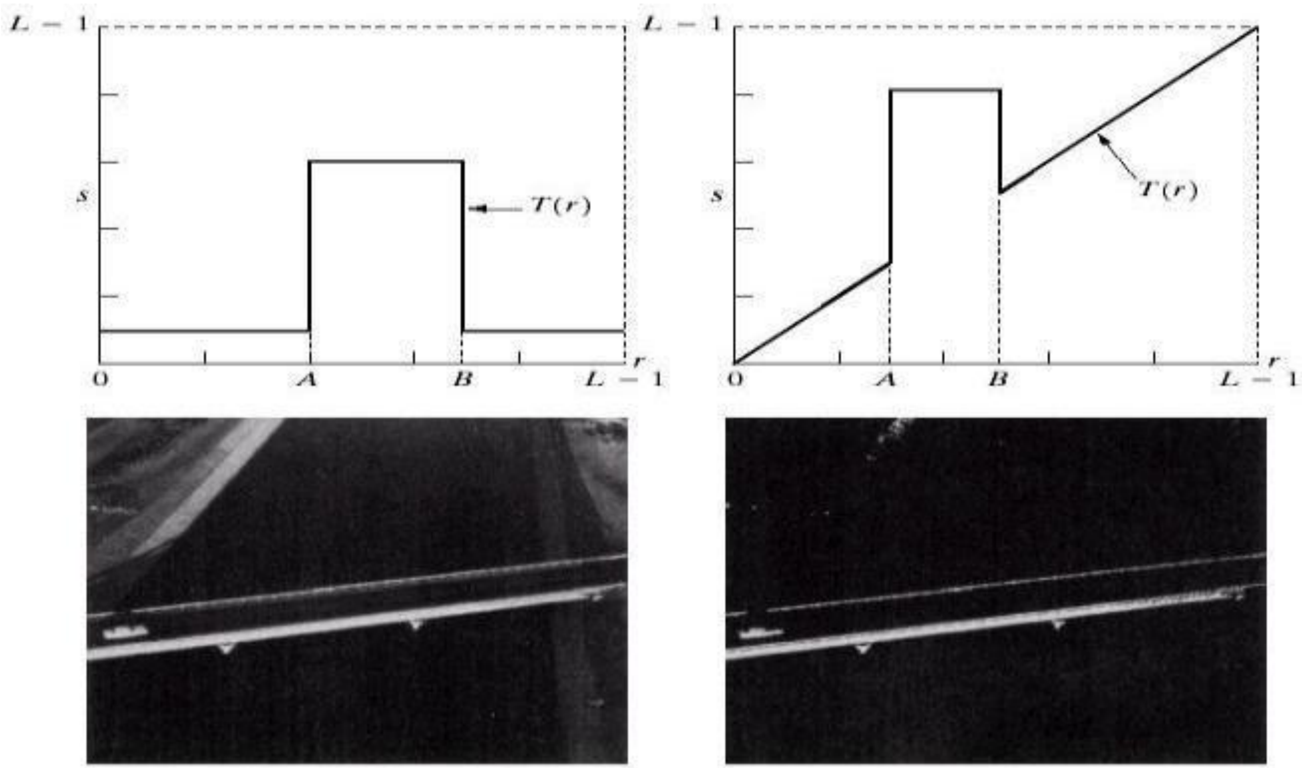


a b
c d

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.)

Basic gray-leveltransformations(Contd..)

- Gray-levelslicing



a b
c d

FIGURE 3.11 (a) This transformation highlights range [A, B] of gray levels and reduces all others to a constant level. (b) This transformation highlights range [A, B] but preserves all other levels. (c) An image. (d) Result of using the transformation in (a).

Basic gray-level transformations(Contd..)

- Bit-plane slicing

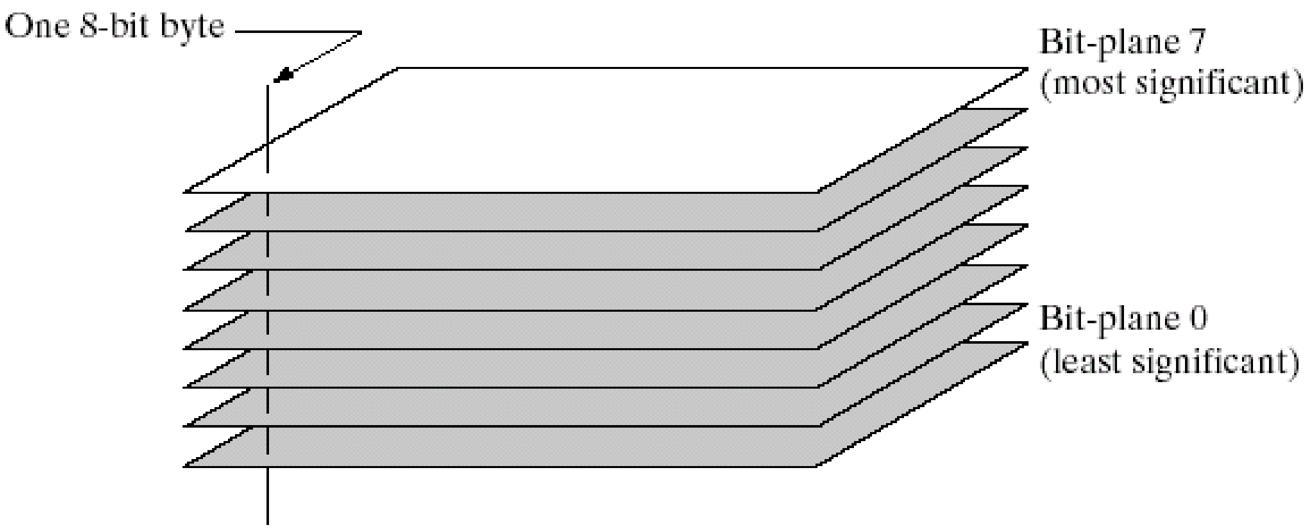
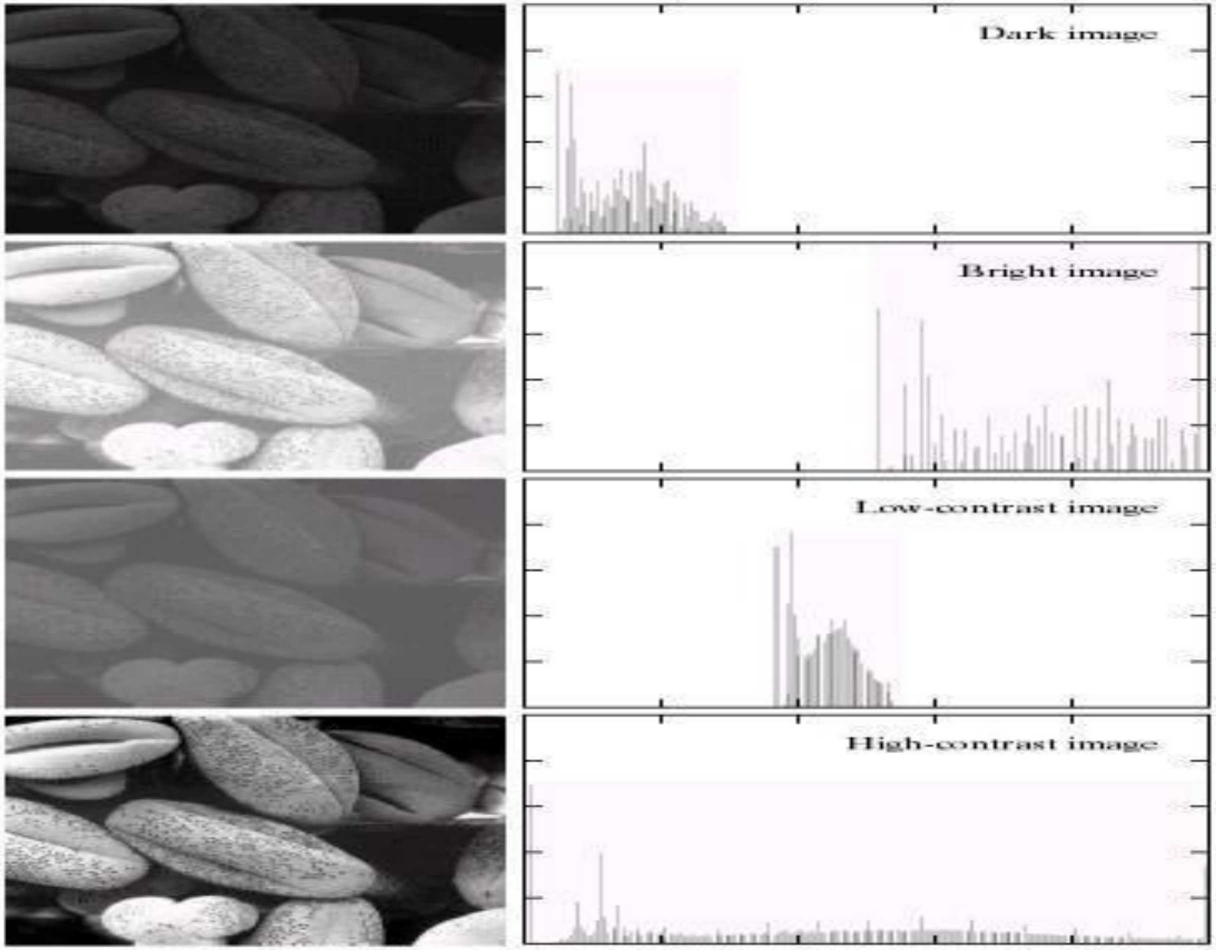


FIGURE 3.12 Bit-plane representation of an 8-bit image.

Histogram Processing

- Histogram
- h(rk)=nk
  - where rk is the kth gray level and is the number of pixels in the image having gray level rk
  - Normalized histogram
- P(rk)=nk/n

Histogram Processing



- Histogram Equilization

$S=T(r), 0 \leq r \leq 1$

$r=\text{inverse}T(s), 0 \leq s \leq 1$

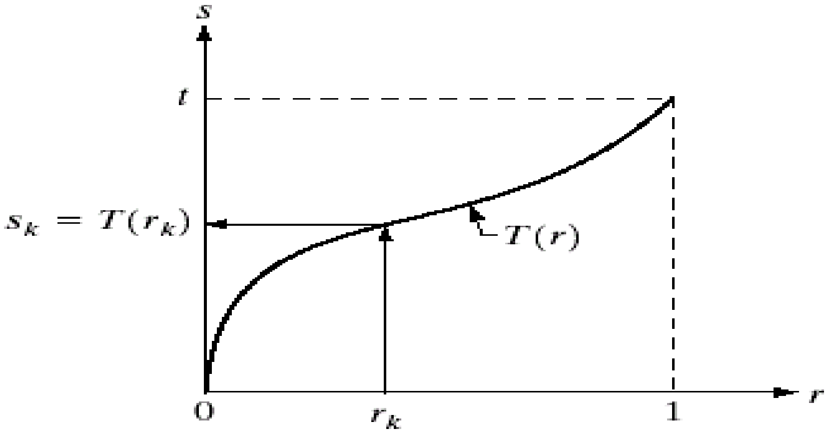


FIGURE 3.16 A gray-level transformation function that is both single valued and monotonically increasing.

Histogram Processing

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

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

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

$$p_s(s) = \frac{1}{L-1}$$

Histogram matching

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

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

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

PZ(Z) is the desiredPDF

Mechanics of spatial filtering

$$
\begin{aligned}
R = {} & w(-1,-1)\, f(x-1, y-1) + \\
& w(-1,0)\, f(x-1, y) + \cdots + \\
& w(0,0)\, f(x, y) + \cdots + \\
& w(1,0)\, f(x+1, y) + \\
& w(1,1)\, f(x+1, y+1)
\end{aligned}
$$

Fundamentals of spatial filtering

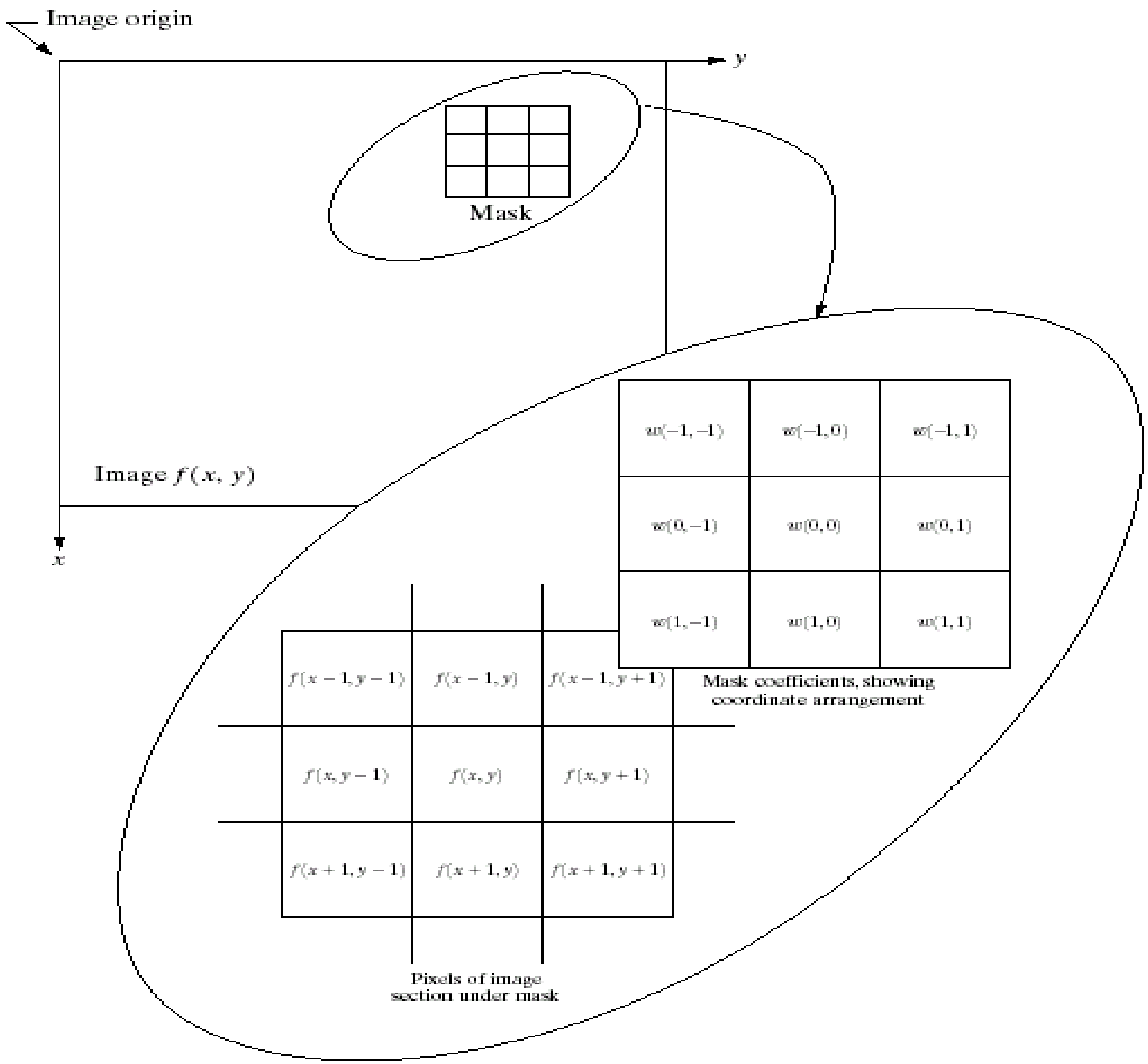


FIGURE 3.32 The mechanics of spatial filtering. The magnified drawing shows a 3×3 mask and the image section directly under it; the image section is shown displaced out from under the mask for ease of readability.

Fundamentals of spatial filtering

Correlation

Origin f w

(a) 0 0 0 1 0 0 0 0   1 2 3 2 8

(b) 0 0 0 1 0 0 0 0
1 2 3 2 8
Starting position alignment

Zero padding

(c) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 2 3 2 8

(d) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 2 3 2 8
Position after one shift

(e) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 2 3 2 8
Position after four shifts

(f) 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0
1 2 3 2 8
Final position

Full correlation result

(g) 0 0 0 8 2 3 2 1 0 0 0 0

Cropped correlation result

(h) 0 8 2 3 2 1 0 0

Convolution

Origin f w rotated 180°

0 0 0 1 0 0 0 0   8 2 3 2 1 (i)

0 0 0 1 0 0 0 0 (j)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (k)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (l)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (m)
8 2 3 2 1

0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 (n)
8 2 3 2 1

Full convolution result

0 0 0 1 2 3 2 8 0 0 0 0 (o)

Cropped convolution result

0 1 2 3 2 8 0 0 (p)

FIGURE 3.29 Illustration of 1-D correlation and convolution of a filter with a discrete unit impulse. Note that correlation and convolution are functions of *displacement*.

Fundamentals of spatial filtering

Origin $f(x, y)$

$$\begin{matrix}0&0&0&0&0\\0&0&0&0&0\\0&0&1&0&0\\0&0&0&0&0\\0&0&0&0&0\end{matrix}$$

$w(x, y)$

$$\begin{matrix}1&2&3\\4&5&6\\7&8&9\end{matrix}$$

(a)

Padded f

$$\begin{matrix}0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\end{matrix}$$

(b)

Initial position for w

$$\begin{matrix}1&2&3&0&0&0&0&0&0\\4&5&6&0&0&0&0&0&0\\7&8&9&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\end{matrix}$$

(c)

Full correlation result

$$\begin{matrix}0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&9&8&7&0&0&0\\0&0&0&6&5&4&0&0&0\\0&0&0&3&2&1&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\end{matrix}$$

(d)

Cropped correlation result

$$\begin{matrix}0&0&0&0&0\\0&9&8&7&0\\0&6&5&4&0\\0&3&2&1&0\\0&0&0&0&0\end{matrix}$$

(e)

Rotated w

$$\begin{matrix}9&8&7&0&0&0&0&0&0\\6&5&4&0&0&0&0&0&0\\3&2&1&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&1&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\end{matrix}$$

(f)

Full convolution result

$$\begin{matrix}0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&1&2&3&0&0&0\\0&0&0&4&5&6&0&0&0\\0&0&0&7&8&9&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0&0\end{matrix}$$

(g)

Cropped convolution result

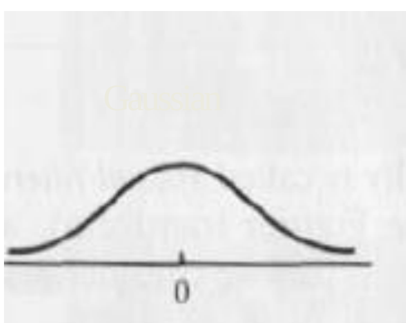
$$\begin{matrix}0&0&0&0&0\\0&1&2&3&0\\0&4&5&6&0\\0&7&8&9&0\\0&0&0&0&0\end{matrix}$$

(h)

FIGURE 3.30 Correlation (middle row) and convolution (last row) of a 2-D filter with a 2-D discrete, unit impulse. The 0s are shown in gray to simplify visual analysis.

Smoothing spatialfilters

1. Useful for reducing noise and eliminating small details.
2. The elements of the mask must bepositive.
3. Sumof mask elements is 1 (after normalization).



7 × 7 Gaussian mask

1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1

Smoothing spatialfilters(example)

- Useful for reducing noise and eliminating small details.

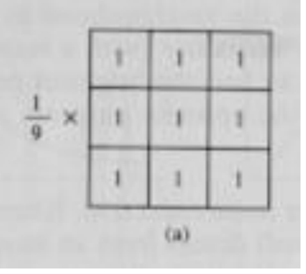
input image

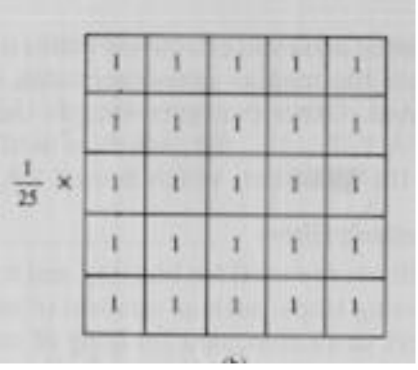
output image

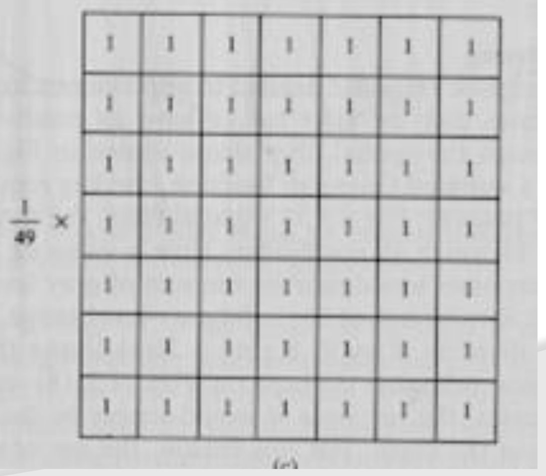
Smoothing spatial filters(averaging)

- Averaging
- Gaussian
- Median filtering(non-linear)

1.Averaging







Smoothing spatial filters(median)

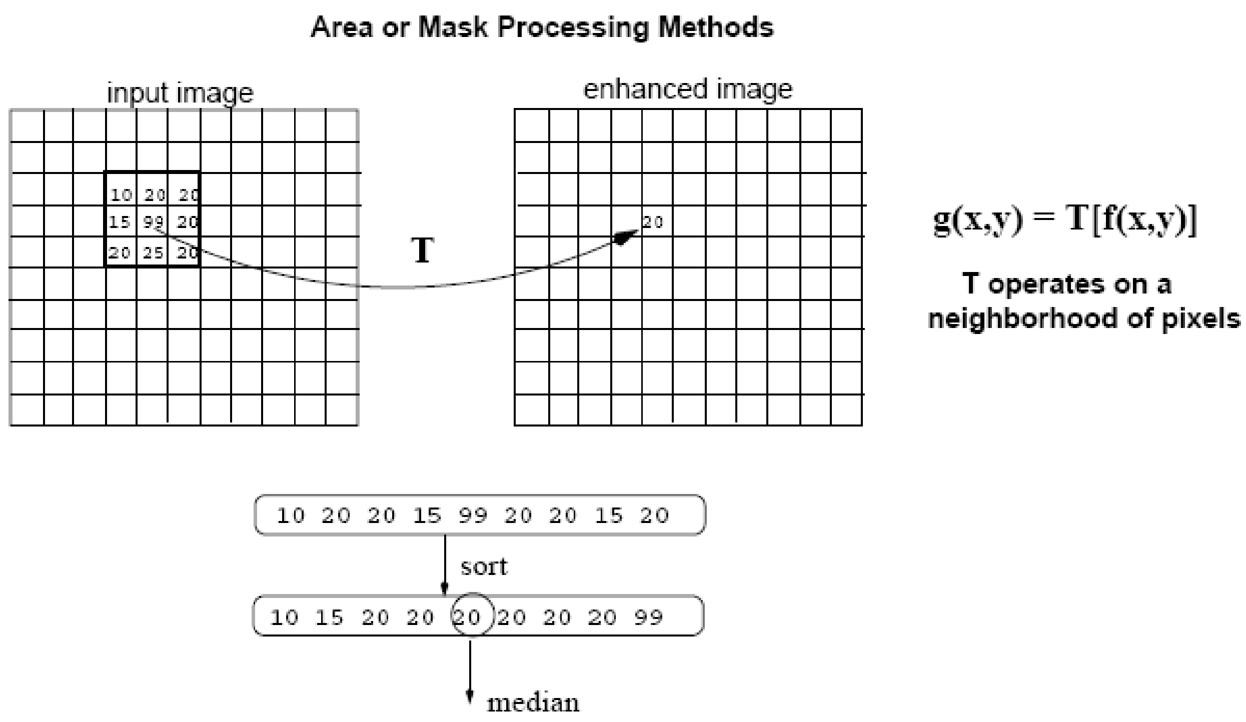
- Very effective for removing “salt and pepper”noise (i.e., random occurrences of black and white pixels).

averaging

median filtering

Smoothing spatialfilters(Median filtering)

- Replace each pixel by the median in a neighborhood around the pixel.



SharpeningFilters

- Unsharpmasking
- High Boostfilter
- Gradient (1st derivative)
- Laplacian (2nd derivative)

Sharpening Filters:Unsharp masking

- Obtain a sharp image by subtracting a lowpass filtered (i.e., smoothed) image from the original image.

$$Highpass = Original - Lowpass$$

(after contrast enhancement)GRAY COLOR

Sharpening Filters:high boost

- If **A=1**, we get unsharpmasking.
- If **A>1**, part of the original image is added back to the high pass filteredimage.
- One way to implement high boost filtering isusing the masksbelow:

A>=1
w = 9A-1

-1	-1	-1
-1	w	-1
-1	-1	-1

A=2
w = 17

-1	-1	-1
-1	17	-1
-1	-1	-1

Sharpening Filters:First order derivatives

- Taking the derivative of an image results in sharpening the image.
- The derivative of an image (i.e., 2D signal) can be computed using the gradient.

$$grad(f) = \begin{pmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{pmatrix}$$

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \approx f(x+1) - f(x) \quad (h=1)$$

$$\frac{\partial f}{\partial x} = \frac{f(x+\Delta x, y) - f(x,y)}{\Delta x} = f(x+1, y) - f(x,y), \quad (\Delta x = 1)$$

$$\frac{\partial f}{\partial y} = \frac{f(x, y+\Delta y) - f(x,y)}{-\Delta y} = f(x,y) - f(x, y+1), \quad (\Delta y = 1)$$

Sharpening Filters: Second order derivatives

- The Laplacian (2nd derivative) is defined as:

$$\nabla^2 = \nabla \cdot \nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

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

Approximate 2nd derivatives:

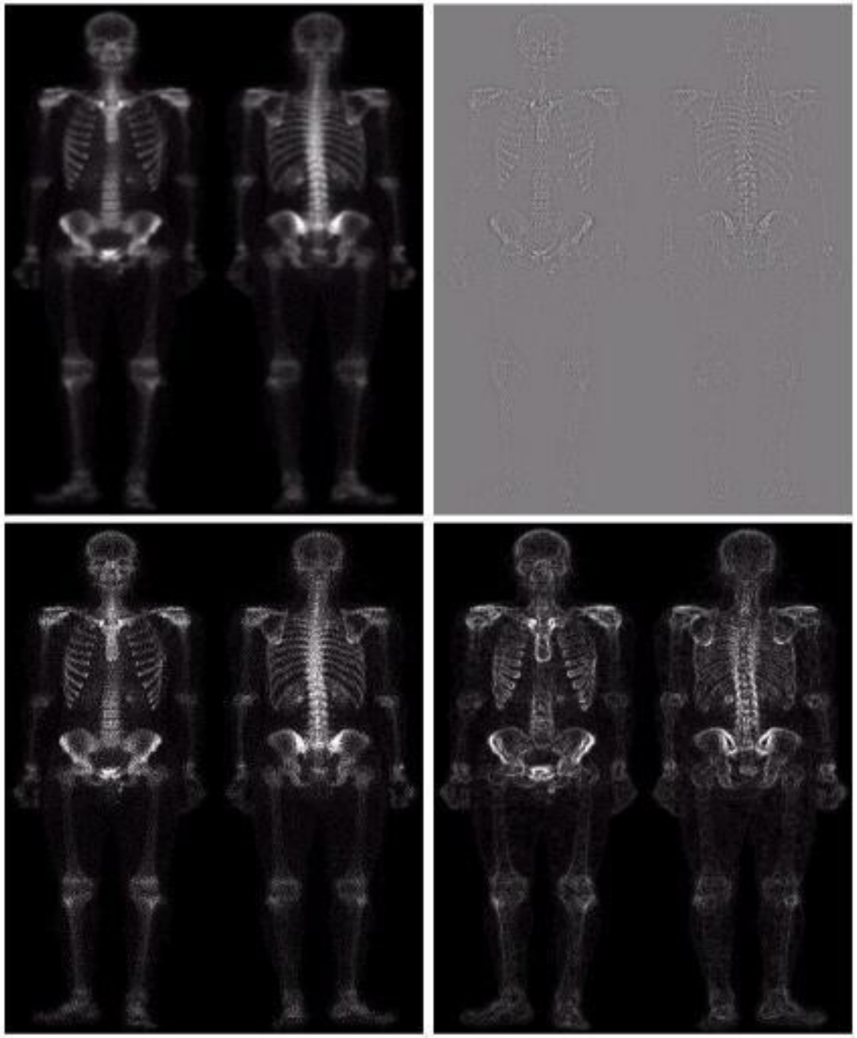
$$\frac{\partial^2 f}{\partial y^2} = f(i+1, j) - 2f(i, j) + f(i-1, j)$$

$$\nabla^2 f = -4f(i, j) + f(i, j+1) + f(i, j-1) + f(i+1, j) + f(i-1, j)$$

Combining spatial enhancement methods

a	b
c	d

FIGURE 3.46 (a) Image of whole body bone scan. (b) Laplacian of (a). (c) Sharpened image obtained by adding (a) and (b). (d) Sobel of (a).



Combining spatial enhancement methods

e	f
g	h

FIGURE 3.46 *(Continued)* (e) Sobel image smoothed with a 5×5 averaging filter. (f) Mask image formed by the product of (c) and (e). (g) Sharpened image obtained by the sum of (a) and (f). (h) Final result obtained by applying a power-law transformation to (g). Compare (g) and (h) with (a). (Original image courtesy of G.E. Medical Systems.)

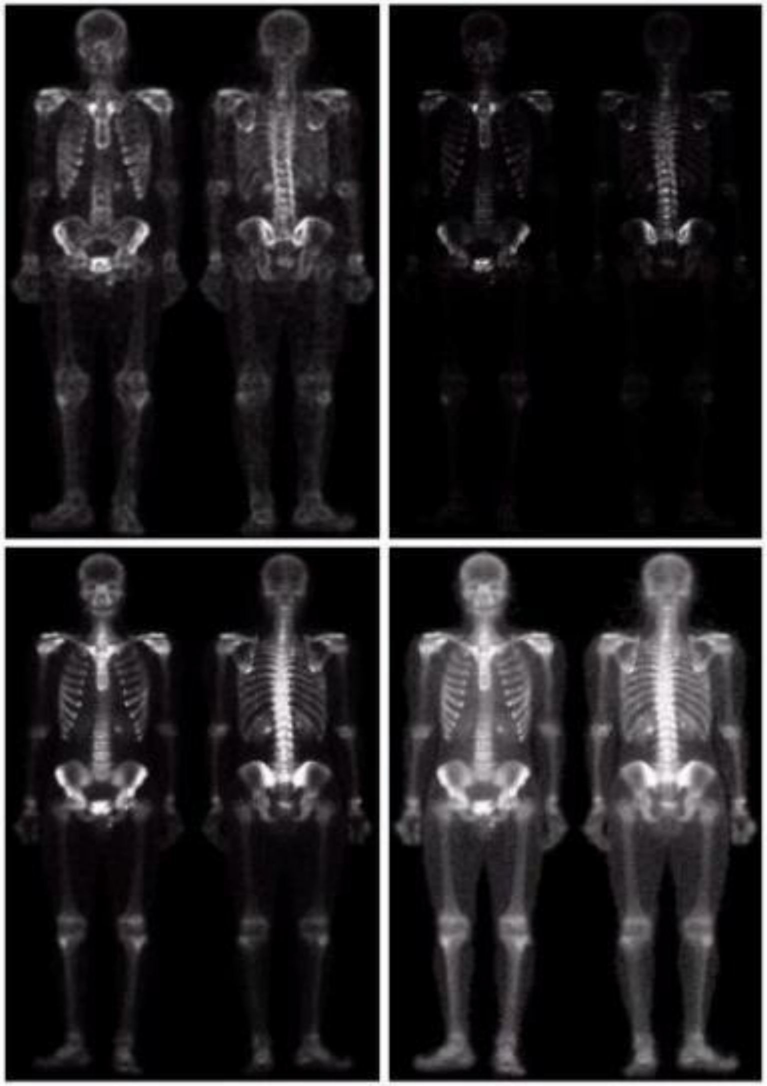


Image enhancement in frequency domain

- Any function that periodically repeats itself can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient (Fourier series).
- Even functions that are not periodic (but whose area under the curve is finite) can be expressed as the integral of sines and/or cosines multiplied by a weighting function (Fourier transform).
- The frequency domain refers to the plane of the two dimensional discrete Fourier transform of animage.
- The purpose of the Fourier transform is to represent a signal as a linear combination of sinusoidal signals of various frequencies.

Image enhancement frequency domain(Contd..)

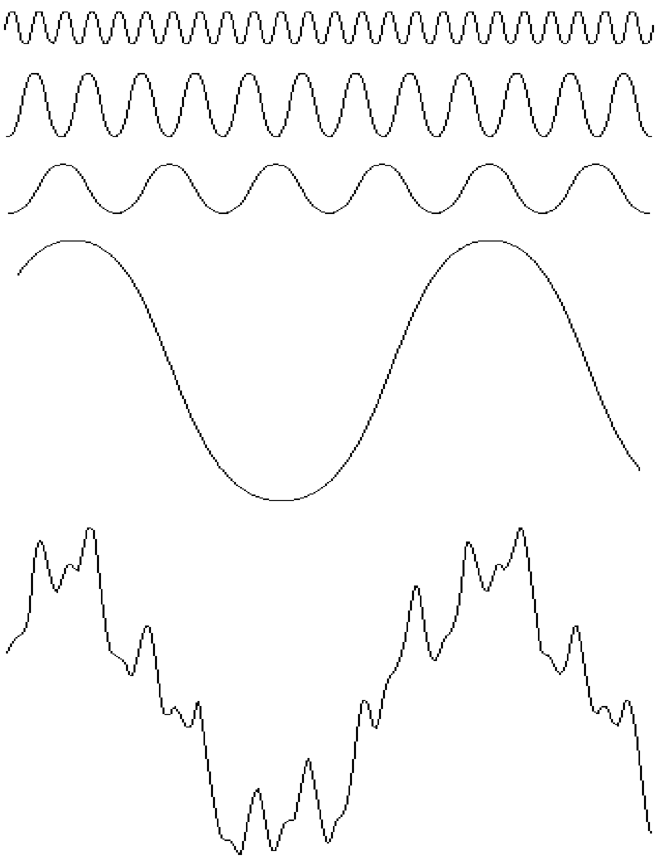


FIGURE 4.1 The function at the bottom is the sum of the four functions above it. Fourier's idea in 1807 that periodic functions could be represented as a weighted sum of sines and cosines was met with skepticism.

The one-dimensional Fourier transform and its inverse

Fourier transform (continuous case)

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-j2\pi ux} dx \quad \text{where } j = \sqrt{-1}$$

Inverse Fourier transform:

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{j2\pi ux} du$$

The two-dimensional Fourier transform and its inverse

Fourier transform (discrete case)DTC

$$F(u,v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) e^{-j2\pi(ux/M + vy/N)}$$

$$\text{for } u = 0,1,2,\ldots, M-1, v = 0,1,2,\ldots, N-1$$

Introduction to the Fourier Transform and the Frequency Domain(Contd..)

Inverse Fouriertransform:

$$f(x,y)=\sum_{u=0}^{M-1}\sum_{v=0}^{N-1}F(u,v)e^{j2\pi(ux/M+vy/N)}$$

$$\text{for } x=0,1,2,\ldots,M-1,\ y=0,1,2,\ldots,N-1$$

u,v : the transform or frequency variables

x,y : the spatial orimage variables

Filtering in the frequency domain

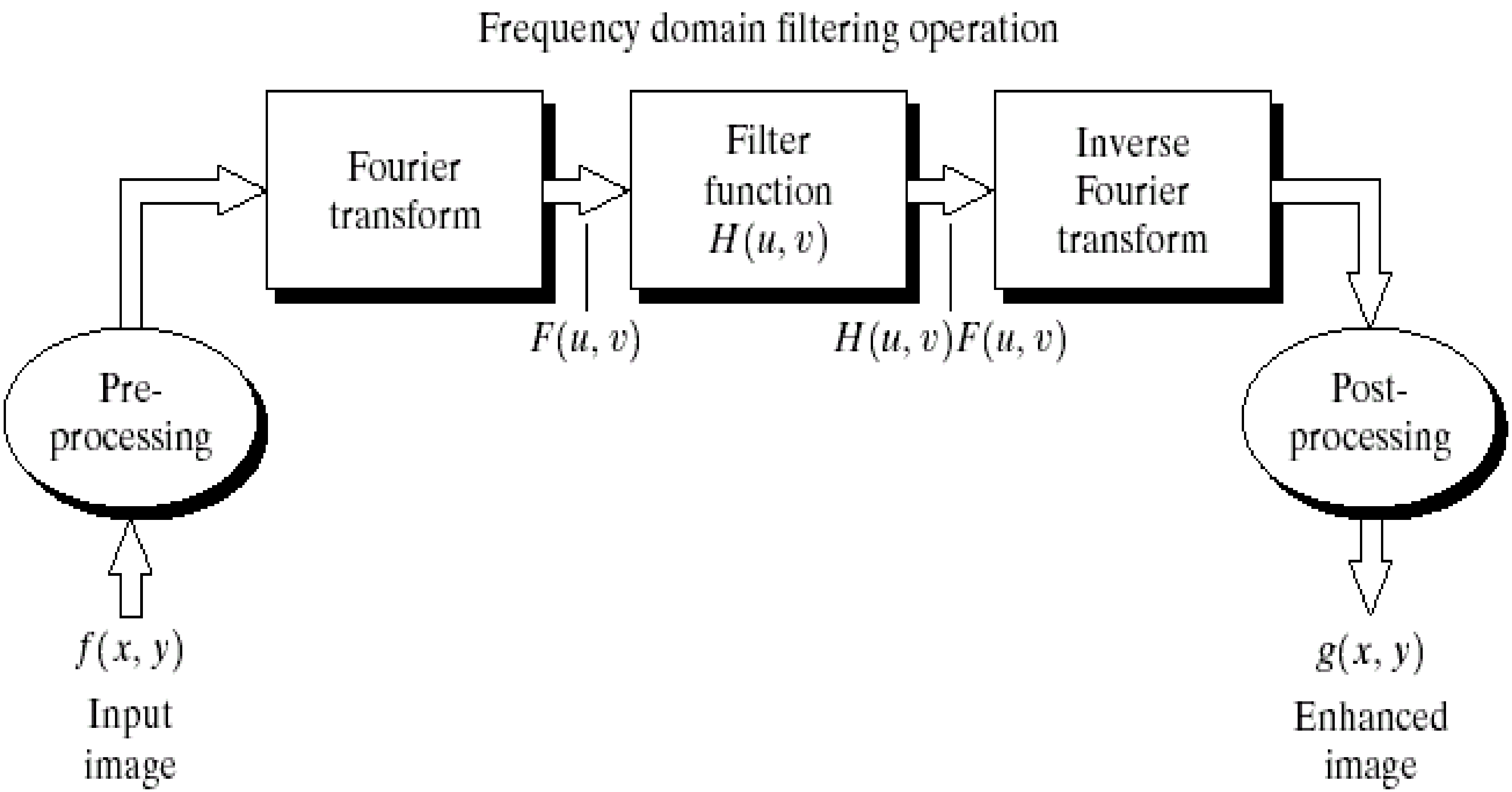


FIGURE 4.5 Basic steps for filtering in the frequency domain.

Smoothing frequency domain filters

- The basic model for filtering in the frequency domain
  $G(u,v)=H(u,v)F(u,v)$
  where $F(u,v)$: the Fourier transform of the image to be smoothed
  $H(u,v)$: a filter transfer function

- Smoothing is fundamentally a lowpass operation in the frequency domain.
- There are several standard forms of lowpass filters (LPF).
  - Ideal lowpass filter
  - Butterworth lowpass filter
  - Gaussian lowpass filter

Ideal low-pass filters

- The simplest lowpass filter is a filter that "cuts off" all high-frequency components of the Fourier transform that are at a distance greater than a specified distance D_0 from the origin of the transform.
- The transfer function of an ideal lowpass filter

$$H(u,v) = \begin{cases} 1 & \text{if } D(u,v) \le D_0 \\ 0 & \text{if } D(u,v) > D_0 \end{cases}$$

where $D(u,v)$: the distance from point (u,v) to the center of ther frequency rectangle

$$D(u,v) = \left[(u - M/2)^2 + (v - N/2)^2\right]^{\frac{1}{2}}$$

Butterworth Lowpass Filters (BLPFs) With order *n*

$$H(u,v) = \frac{1}{1+\left[D(u,v)/D_0\right]^{2n}}$$

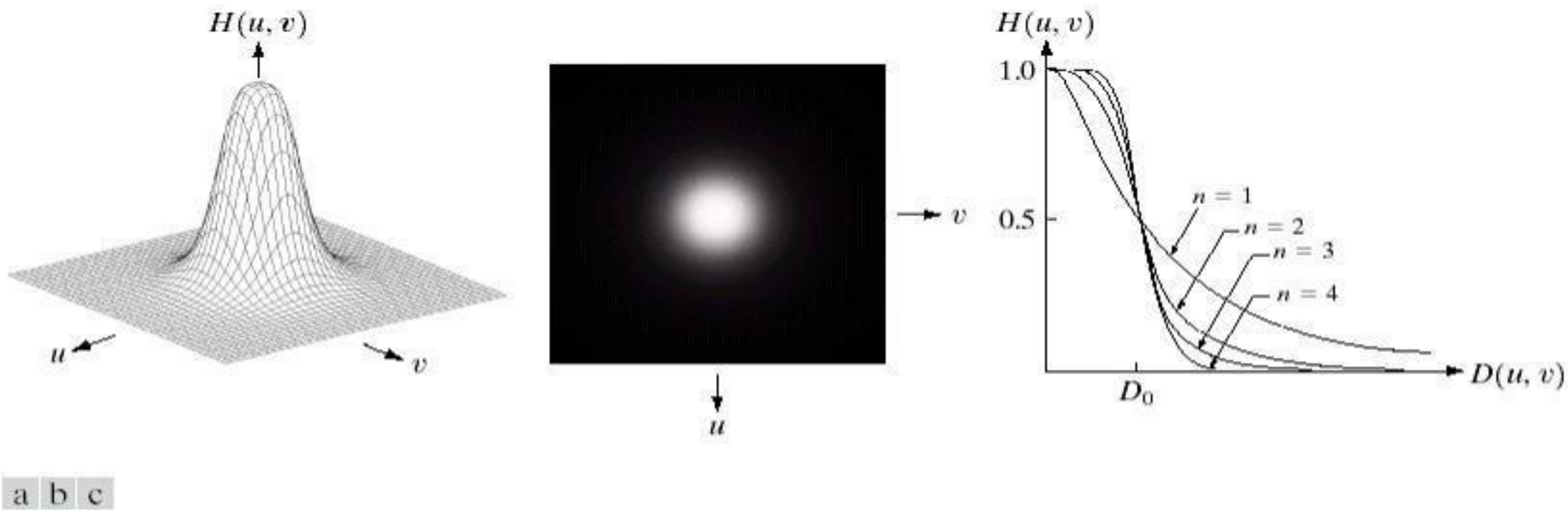


FIGURE 4.14 (a) Perspective plot of a Butterworth lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections of orders 1 through 4.

Gaussian low passfilters

$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

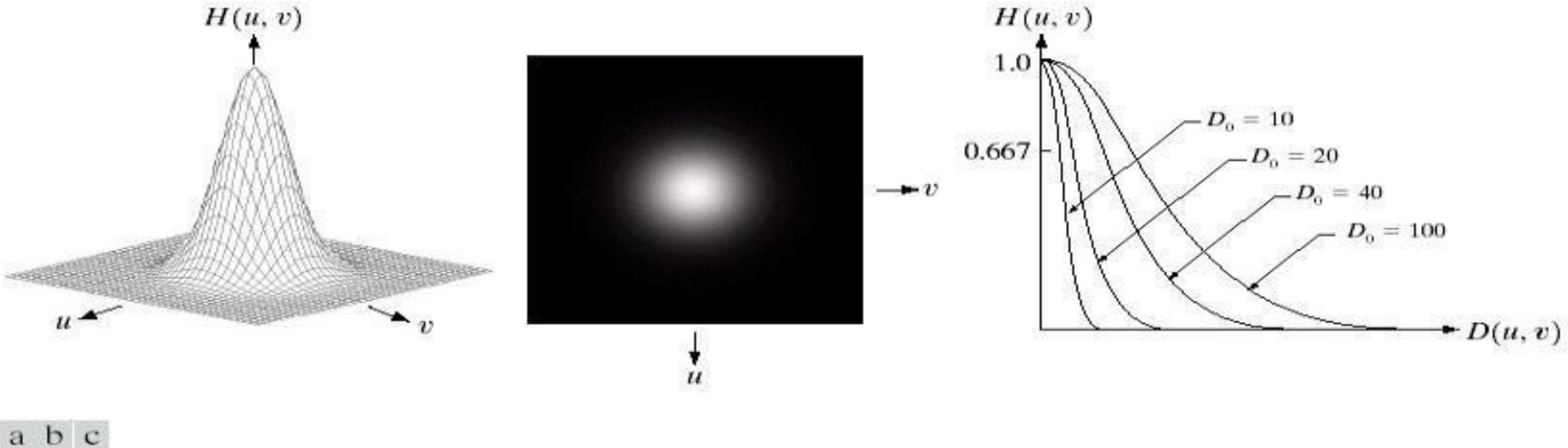


FIGURE 4.17 (a) Perspective plot of a GLPF transfer function. (b) Filter displayed as an image. (c) Filter radial cross sections for various values of D_0.

Sharpening frequency domain filters

$$H_{hp}(u,v) = H_{lp}(u,v)$$

Ideal highpass filter

$$H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \le D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}$$

Butterworth highpass filter

$$H(u,v) = \frac{1}{1 + [D_0 / D(u,v)]^{2n}}$$

Gaussian highpass filter

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

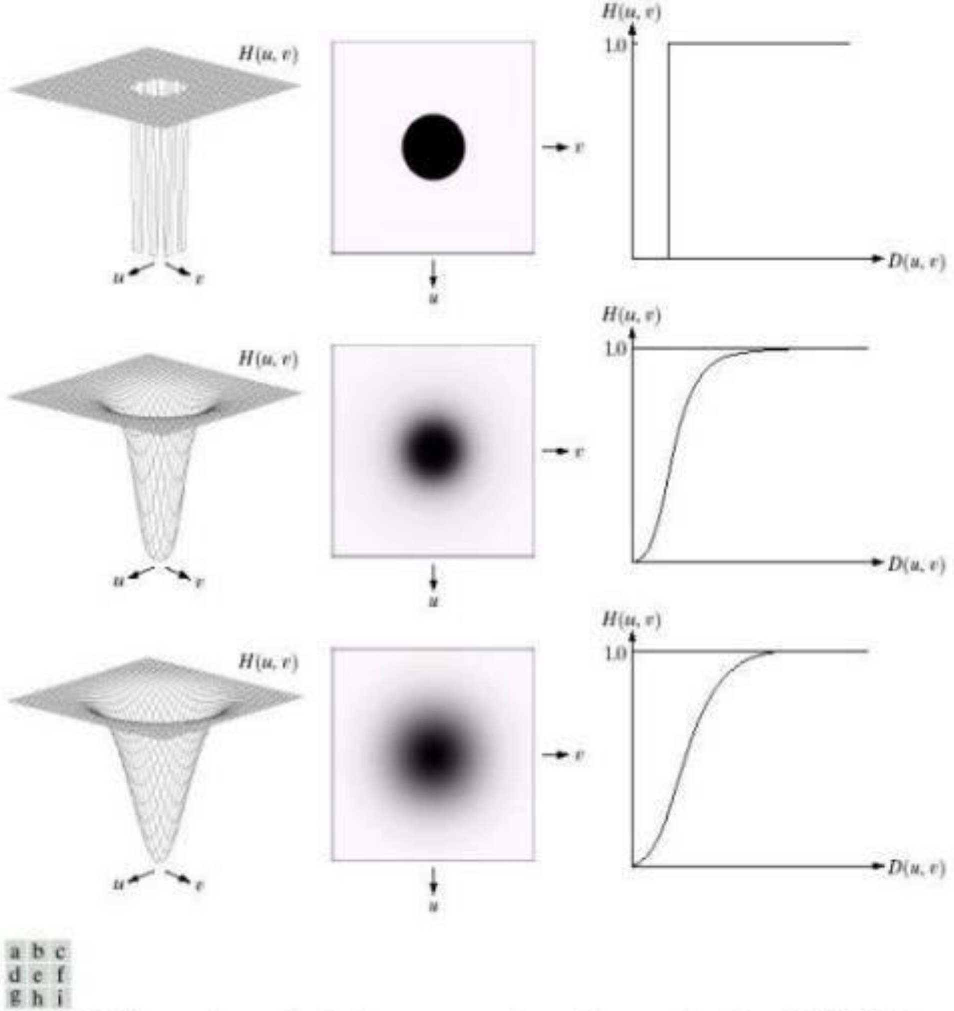
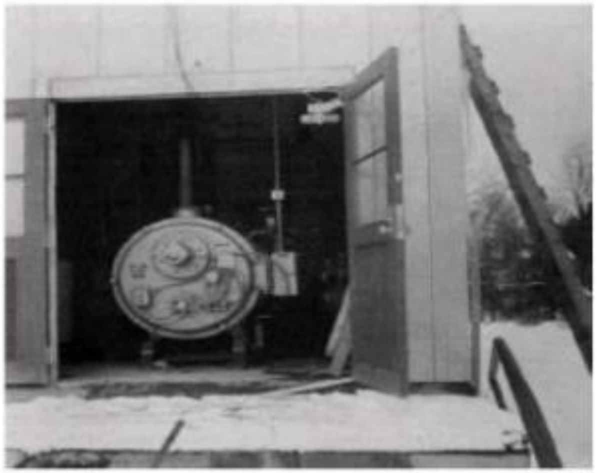


FIGURE 4.22 Top row: Perspective plot, image representation, and cross section of a typical ideal highpass filter. Middle and bottom rows: The same sequence for typical Butterworth and Gaussian highpass filters.

HomomorphicFiltering

- Many times, we want to remove shading effects from an image (i.e., due to uneven illumination)
  - Enhance high frequencies
  - Attenuate low frequencies but preserve fine detail



Homomorphic Filtering(Contd..)

- Consider the following model of imageformation:

$$f(x,y) = i(x,y)\, r(x,y)$$

i(x,y): illumination **r(x,y):** reflection

- In general, the illuminationcomponent i(x,y) varies **slowly** and affects **low** frequenciesmostly.
- In general, the reflection component r(x,y) varies **faster**and affects **high** frequencies mostly.

IDEA:separate low frequencies due to i(x,y) from high frequencies due tor(x,y)

Steps of Homomorphic Filtering

(1) Take $ln(f(x,y)) = ln(i(x,y)) + ln(r(x,y))$

(2) Apply FT: $F(ln(f(x,y))) = F(ln(i(x,y))) + F(ln(r(x,y)))$

or

$$Z(u,v) = Illum(u,v) + Refl(u,v)$$

(3) Apply H(u,v)

$$Z(u,v)H(u,v) = Illum(u,v)H(u,v) + Refl(u,v)H(u,v)$$

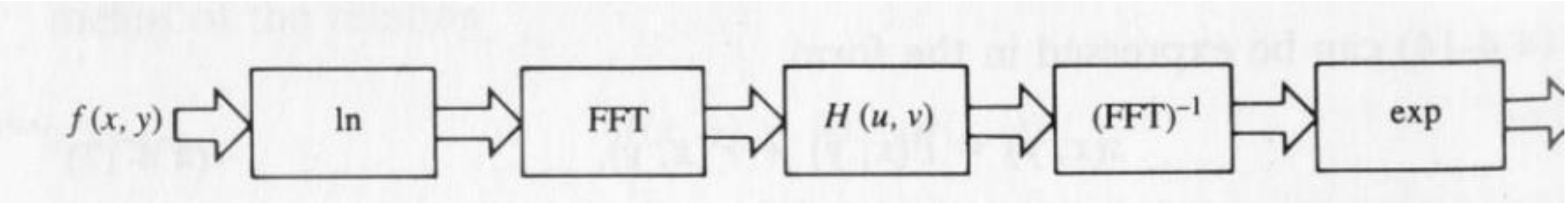
Steps of Homomorphic Filtering

(4) Take Inverse FT:

$$F^{-1}(Z(u,v)H(u,v)) = F^{-1}(Illum(u,v)H(u,v)) + F^{-1}(Refl(u,v)H(u,v))$$

or $$s(x,y) = i'(x,y) + r'(x,y)$$

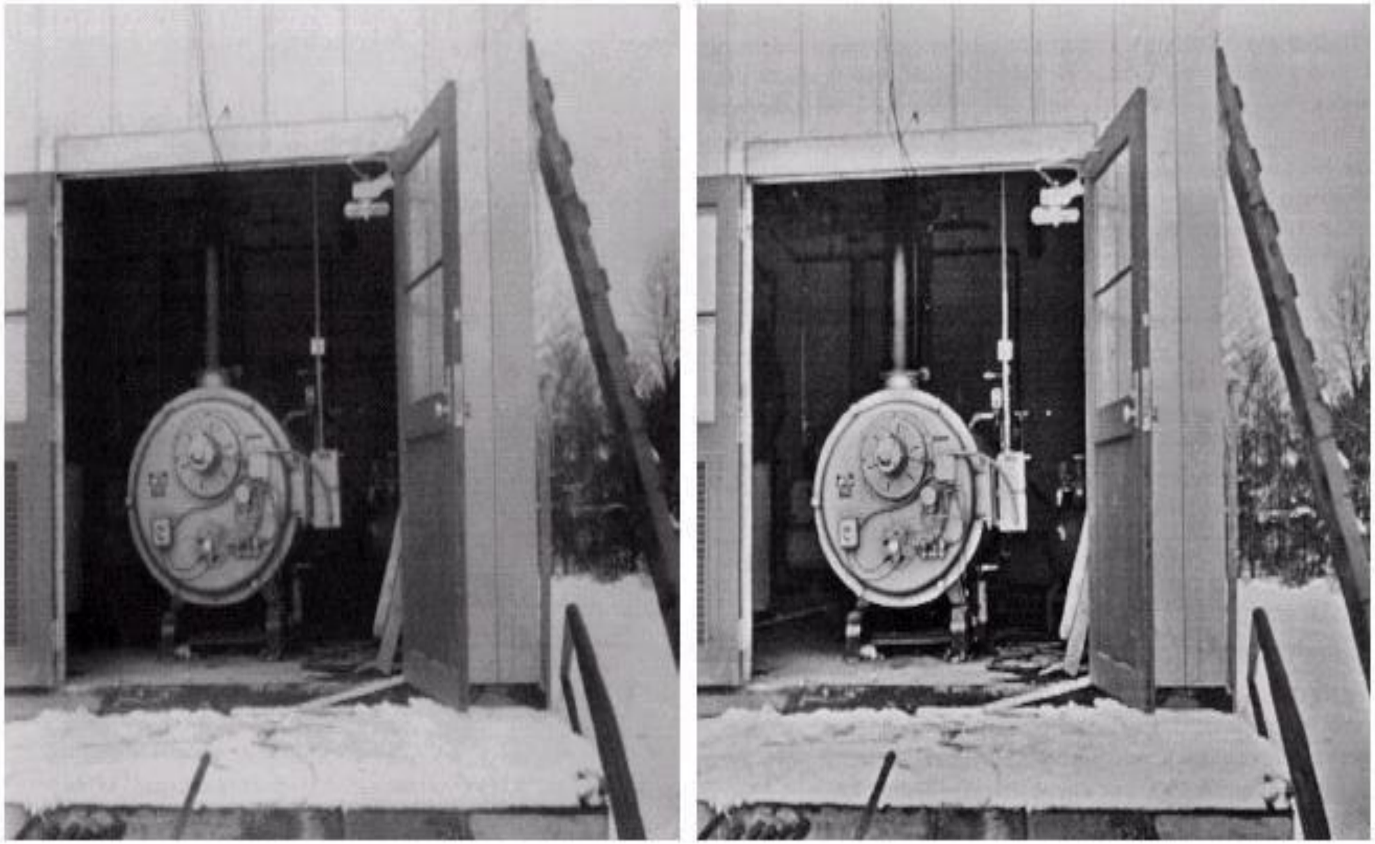
(5) Take exp( ) $$e^{s(x,y)} = e^{i'(x,y)}e^{r'(x,y)}$$



Homomorphic Filtering:Example

a b

FIGURE 4.33
(a) Original image. (b) Image processed by homomorphic filtering (note details inside shelter). (Stockham.)



UNIT 3
IMAGE RESTORATION AND FILTERING

Course Learning Outcome

CLOs	Course Learning Outcome
CLO10	Understand the impact of smoothing and sharpening filters in spatial domain.
CLO11	Apply the Fourier transform concepts on image function in frequency domain filters(low pass/high pass).
CLO12	Describe the concept of image degradation or restoration of images.

Introduction

Objective of image restoration

to recover a **distorted image** to the **original form** based on **idealized models**.

The distortion is due to

Image degradation in sensing environment e.g. random atmospheric turbulence

Noisy degradation from sensor noise.

Blurring degradation due to sensors

e.g. camera motion or out-of-focus

Geometric distortion

e.g. earth photos taken by a camera in a satellite

Image Enhancement

- Enhancement
  - Concerning the extraction of imagefeatures
  - Difficult to quantifyperformance
  - Subjective; making an image “lookbetter
- Restoration
  - Concerning the restoration of degradation
  - Performance can be quantified
  - Objective; recovering the original image

Noise models

- **Assuming degradation only due to additive noise (H=1)**
- **Noise from sensors**
  - Electronic circuits
  - Light level
  - Sensor temperature
- **Noise from environment**
  - Lightening
  - Atmospheric disturbance
  - Other strong electric/magnetic signals

Noise models

- **Assuming that noise is**
  - independent of spatial coordinates, and
  - uncorrelated with respect to the image content

Gaussian noise

- Probability density function (PDF)

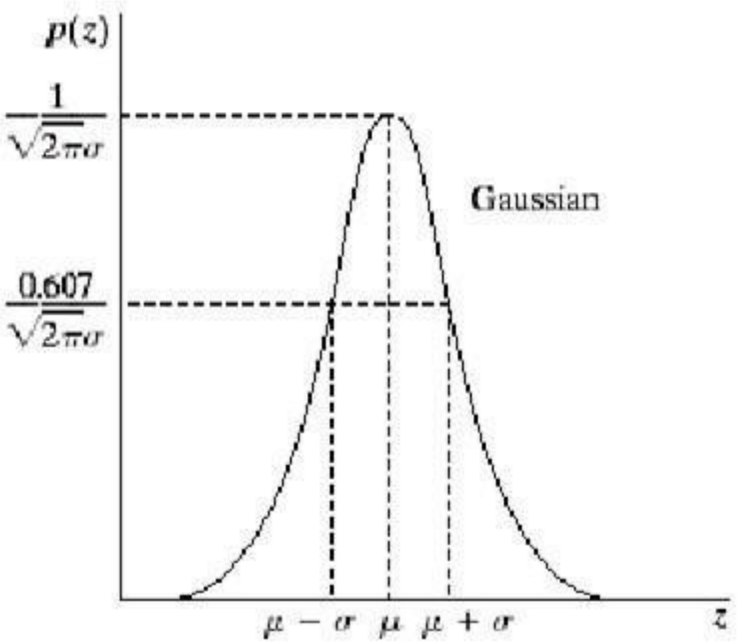
$$p(z) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(z-\mu)^2/2\sigma^2}$$

- z: gray level (Gaussian random variable)
- μ: mean of average value of z
- σ: standard deviation of z
- σ^2: variance of z

Noise models

PDF of Gaussian noise

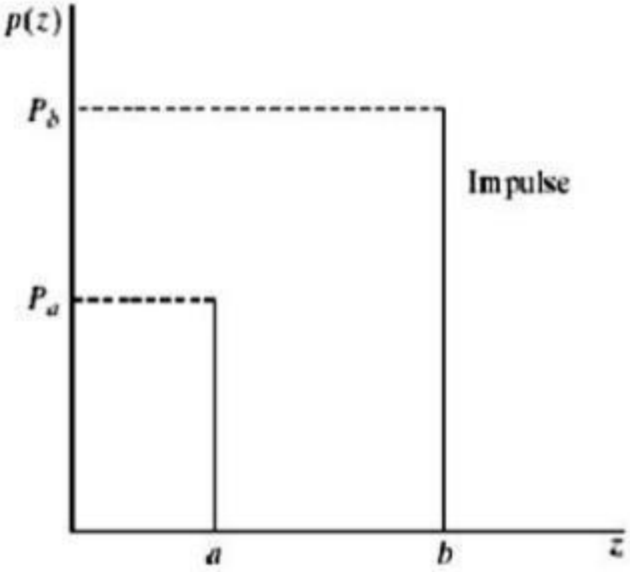
- 70% of z in $[\mu-\sigma, \mu+\sigma]$
- 90% of z in $[\mu-2\sigma, \mu+2\sigma]$



Noise models

Impulse (salt-and-pepper) noise

$$p(z) = \begin{cases} P_a & \text{for } z = a \\ P_b & \text{for } z = b \\ 0 & \text{otherwise} \end{cases}$$



- bipolar if $P_a \neq 0$, $P_b \neq 0$
- unipolar if one of P_a and P_b is 0
- noise looks like salt-and-pepper granules if $P_a \approx P_b$
- negative or positive; scaling is often necessary to form digital images
- extreme values occur (e.g. $a = 0$, $b = 255$)

Noise models

- **Other common noise models**
  - Rayleigh noise
  - Gamma noise
  - Exponential noise
  - Uniform noise

Noise models

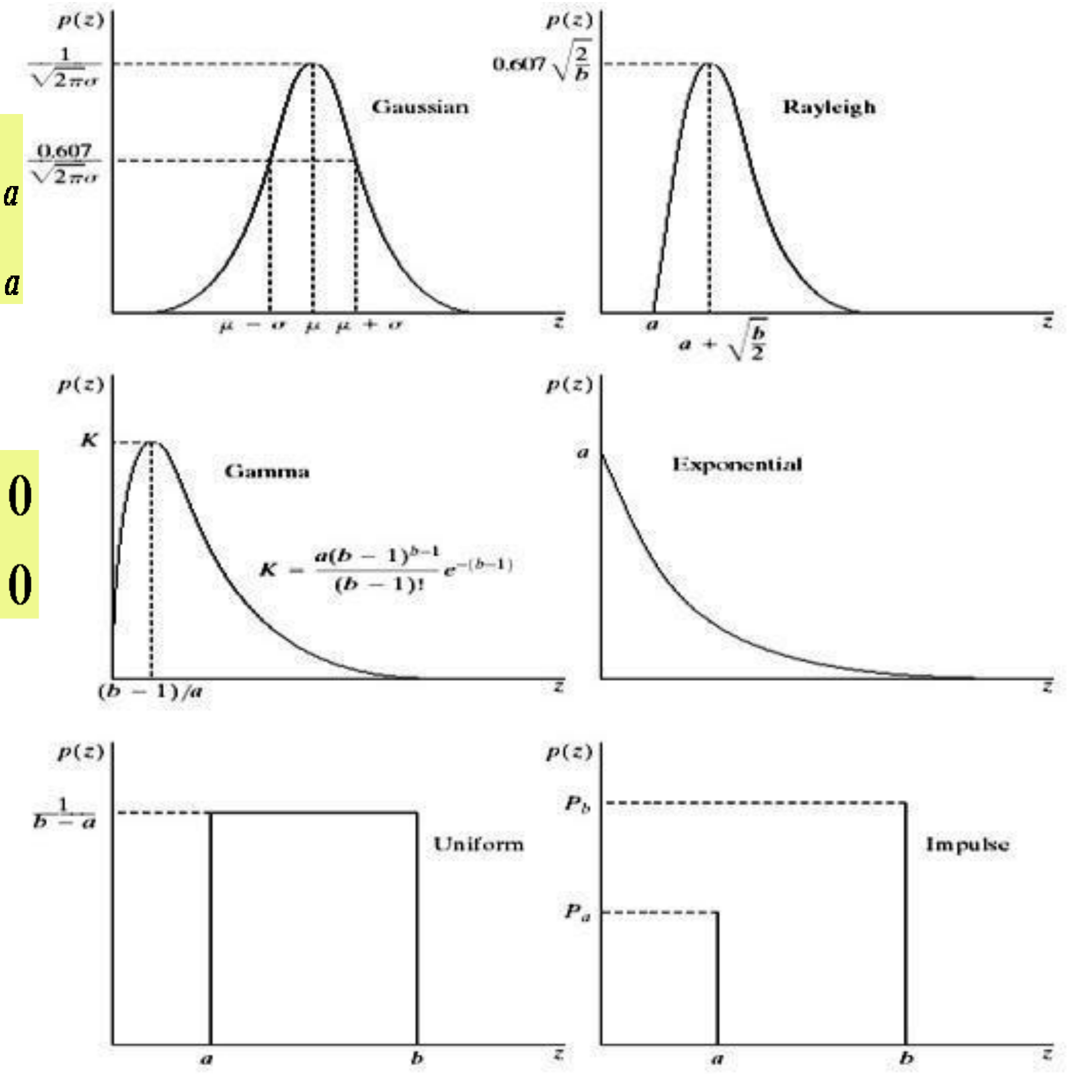
RayleighNoise

$$p(z) = \frac{2}{b}(z-a)e^{-(z-a)^2/b} \quad \text{for } z \ge a$$
$$= 0 \quad \text{for } z < a$$

Gamma(Erlang) Noise

Exponential Noise

$$p(z) = ae^{-az} \quad \text{for } z \ge 0$$
$$= 0 \quad \text{for } z < 0$$



Restoration by spatial filtering

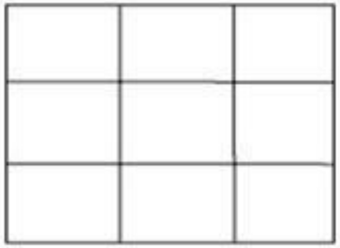
Assume noise is the only degradation source

$$g(x,y) = f(x,y) + \eta(x,y)$$

$$G(u,v) = F(u,v) + N(u,v)$$

Spatial filtering

- a means when only additive noise is present
- similar to enhancement in spatial domain



3×3 filter

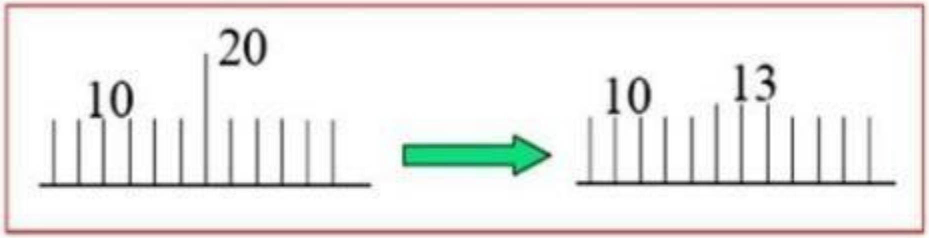
Restoration by spatial filtering

Mean filters (noise reduced by blurring)

- **Arithmetic mean filter**

$$\hat{f}(x,y) = \frac{1}{mn}\sum_{(s,t)\in S_{xy}} g(s,t)$$

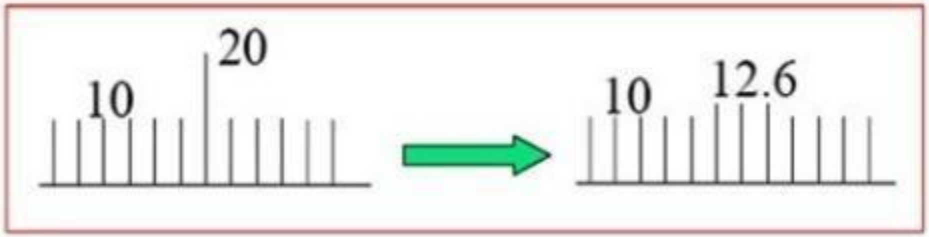
1 x 3 mask



- **Geometric mean filter**

$$\hat{f}(x,y) = \left[\prod_{(s,t)\in S_{xy}} g(s,t)\right]^{\frac{1}{mn}}$$

1D illustration



- Smoothing comparable to arithmetic mean filter
- Losing less image details

Restoration byspatial filtering

Order-statistics filters

- **Median filter**
  - handling both bipolar or unipolar impulse noise

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

- **Max filter**
  - finding the brightest points in an image
  - reducing pepper noise

$$\hat{f}(x,y) = \max_{(s,t)\in S_{xy}}\{g(s,t)\}$$

- **Min filter**
  - finding the darkest points in an image
  - reducing salt noise

$$\hat{f}(x,y) = \min_{(s,t)\in S_{xy}}\{g(s,t)\}$$

Restoration byspatial filtering

Order-statistics filters

- **Midpoint filter**
  - order statistics
  - averaging
  - work well for Gaussian noise

$$\hat{f}(x,y)=\frac{1}{2}\left[\max_{(s,t)\in S_{xy}}\{g(s,t)\}+\min_{(s,t)\in S_{xy}}\{g(s,t)\}\right]$$

- **Alpha-trimmed mean filter**
  - $d=0$: arithmetic mean filter
  - $d=mn-1$: median filter
  - suitable for the situation involving multiple types of noise

$$\hat{f}(x,y)=\frac{1}{mn-d}\sum_{(s,t)\in S_{xy}} g_r(s,t)$$

delete d/2 lowest and d/2 highest values first, then average the remaining

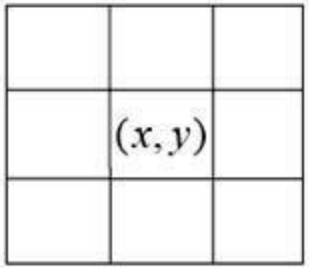
Restoration by spatial filtering

- **Filters discussed so far**
  - Do not consider image characteristics
- **Adaptive filters to be discussed**
  - Behaviors based on statistical characteristics of the subimage under a filter window
  - Better performance
  - More complicated
  - **Adaptive, local noise reduction filter**
  - **Adaptive median filter**

Restoration by spatial filtering

Adaptive, local noise reduction filter

- **Mean** of a random variable: a measure of average gray level in some region
- **Variance** of a random variable: a measure of average contrast in the region
- Response based on four quantities
  - $g(x,y)$: value of noisy image at (x,y)
  - σ_{η}^{2} : variance of the noise
  - m_L : local mean in S_{xy}
  - σ_L^2 : local variance in S_{xy}



S_{xy}

Restoration byspatial filtering

Algorithm of adaptive median filtering

- Level 0: Set initial window size at a new (x,y)
- Level A: If $z_{\min} < z_{\text{med}} < z_{\max}$, Go to Level B

  Else increase the window size

  If window size $\leq S_{\max}$, repeat Level A

  Else output z_{xy}.
- Level B: If $z_{\min} < z_{xy} < z_{\max}$, output z_{xy}

  Else output z_{med}.

Frequency domain filtering

- Pure sine wave

$$f(x,y) = A\sin(u_0 x + v_0 y)$$

-Appear as a pair of impulse (conjugate) in the frequency domain

$$F(u,v) = -j\frac{A}{2}\left[\delta\left(u-\frac{u_0}{2\pi}, v-\frac{v_0}{2\pi}\right) - \delta\left(u+\frac{u_0}{2\pi}, v+\frac{v_0}{2\pi}\right)\right]$$

Periodic noise reduction(Contd..)

- Bandreject filters
- Bandpass filters
- Notch filters
- Optimum notch filtering

Periodic noise reduction(Contd..)

Bandreject filters

- Reject an isotropic frequency

ideal Butterworth Gaussian

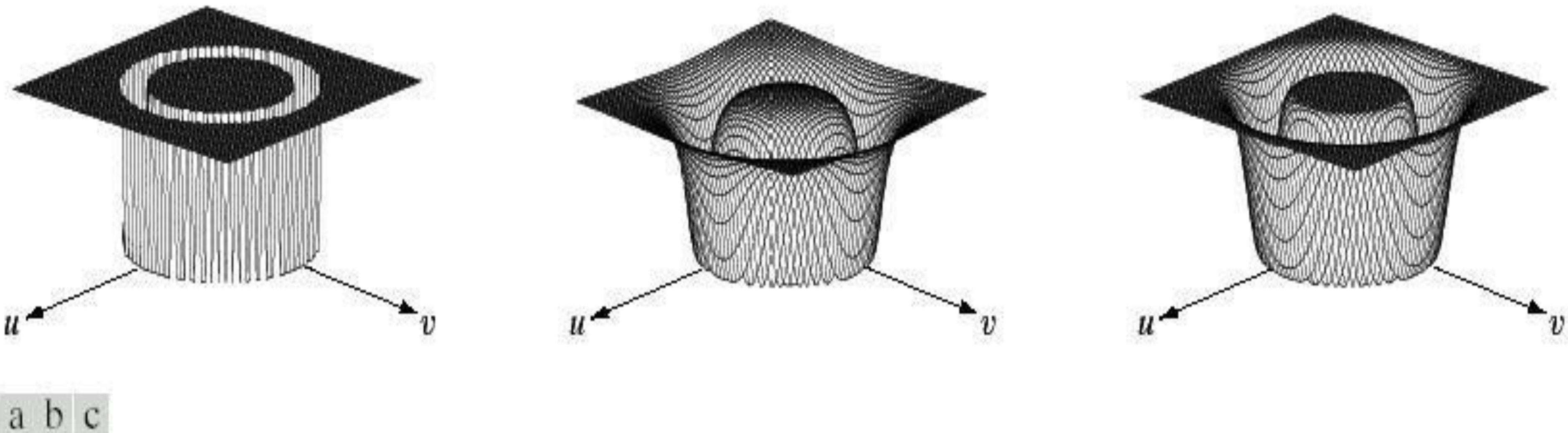
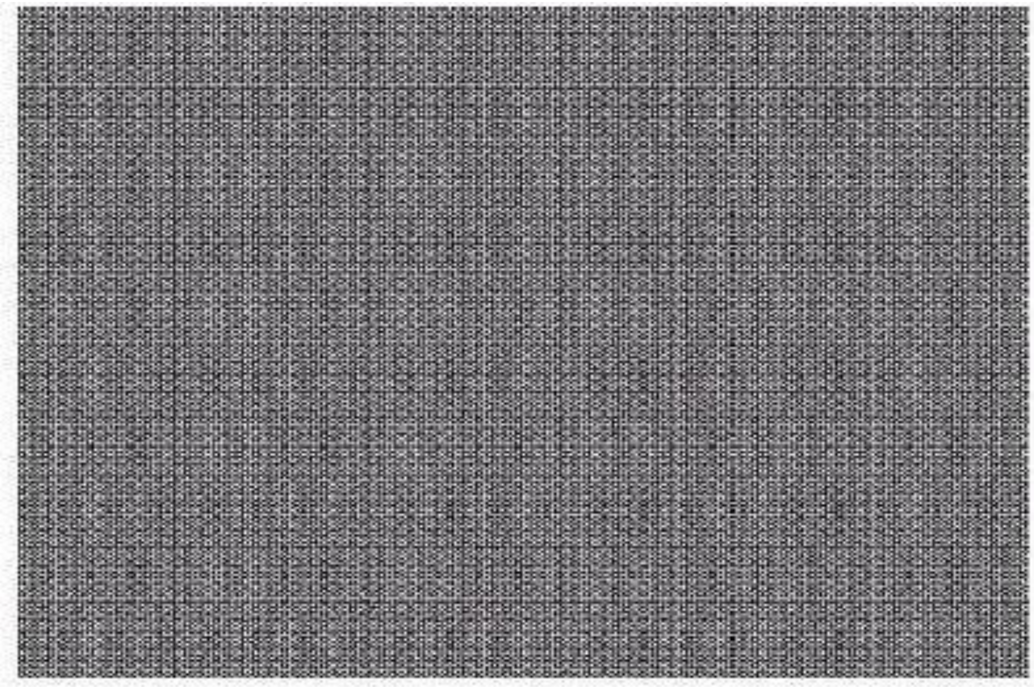


FIGURE 5.15 From left to right, perspective plots of ideal, Butterworth (of order 1), and Gaussian bandreject filters.

Periodic noise reduction(Contd..)

Bandpass filters

- $H_{bp}(u,v)=1- H_{br}(u,v)$

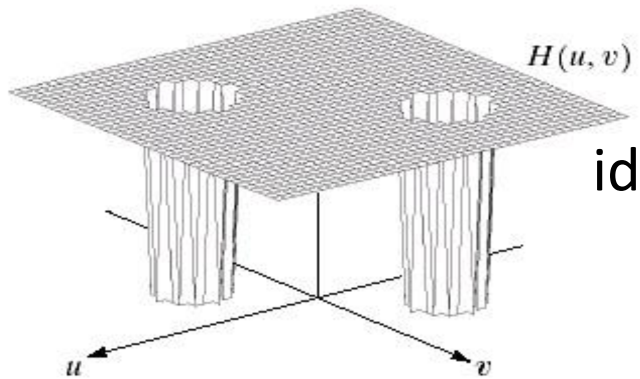


$$\Im^{-1}\left\{G(u,v)H_{bp}(u,v)\right\}$$

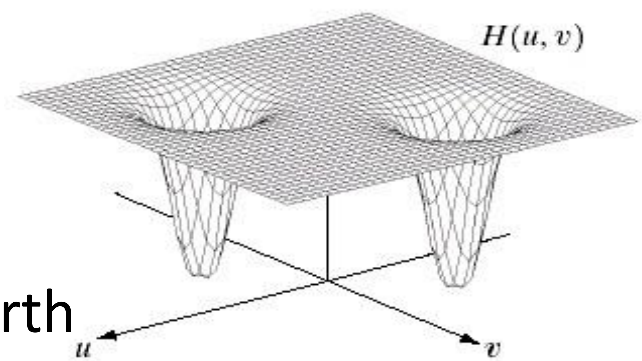
Periodic noise reduction(Contd..)

Notch filters

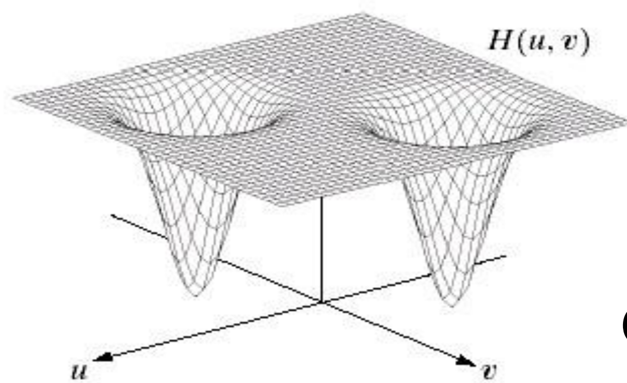
- Reject(or pass) frequencies in predefined neighborhoods about a center frequency



ideal



Butterworth



Gaussian

Linear, position-invariant degradation

Properties of the degradation function H

- Linear system
  $H[af_1(x,y)+bf_2(x,y)]=aH[f_1(x,y)]+bH[f_2(x,y)]$

- Position(space)-invariant system
  $H[f(x,y)]=g(x,y) \Leftrightarrow H[f(x-a, y-b)]=g(x-a, y-b)$

- c.f. 1-D signal
  LTI (linear time-invariant system)

Linear, position-invariant degradation(contd..)

$$f(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\delta\boxed{(x-\alpha,y-\beta)}d\alpha d\beta$$

impulse

$$g(x,y)=H[f(x,y)]=H\left[\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\delta(x-\alpha,y-\beta)d\alpha d\beta\right]$$

linear

$$g(x,y)=H[f(x,y)]=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} H\left[f(\alpha,\beta)\delta(x-\alpha,y-\beta)\right]d\alpha d\beta$$

$$g(x,y)=H[f(x,y)]=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)\boxed{H\left[\delta(x-\alpha,y-\beta)\right]}d\alpha d\beta$$

$h(x,y)=H\left[\delta(x,y)\right]$

$$h(x,\alpha,y,\beta)=H\left[\delta(x-\alpha,y-\beta)\right]$$

If position-invariant

$$H\left[\delta(x-\alpha,y-\beta)\right]=h(x-\alpha,y-\beta)$$

$$g(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)h(x-\alpha,y-\beta)d\alpha d\beta$$

$\eta(x,y)\neq 0$

$$g(x,y)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(\alpha,\beta)h(x-\alpha,y-\beta)d\alpha d\beta+\eta(x,y)$$

Linear, position-invariant degradation(contd..)

- Linear system theory is ready

- Non-linear, position-dependent system
  -May be general and more accurate
  -Difficult to solve computationally

- Image restoration: find H(u,v) and apply inverse process
  -Image deconvolution

Estimating the degradation function

- Estimation by Image observation
- Estimation by experimentation
- Estimation by modeling

<u>Estimation by image observation</u>

- Take a window in the image
  -Simple structure
  -Strong signal content
- Estimate the original image in the window

$$H_s(u,v) = \frac{G_s(u,v)}{\hat{F}_s(u,v)}$$

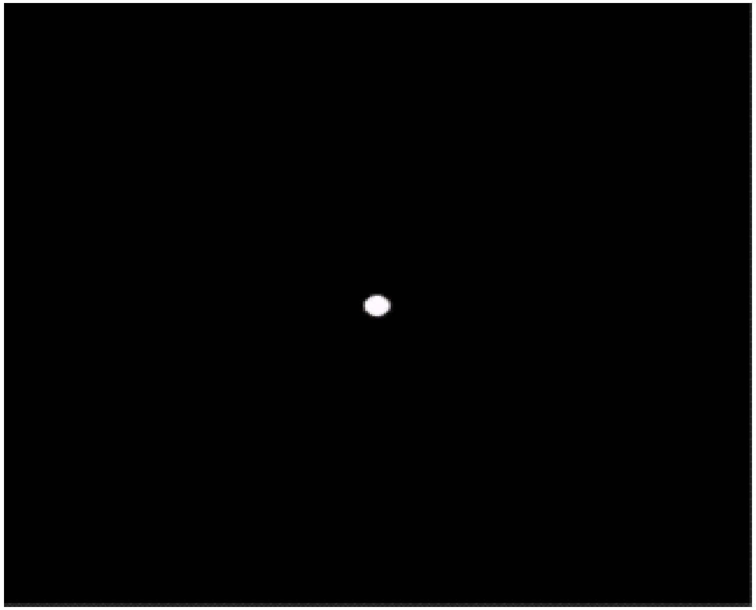
known

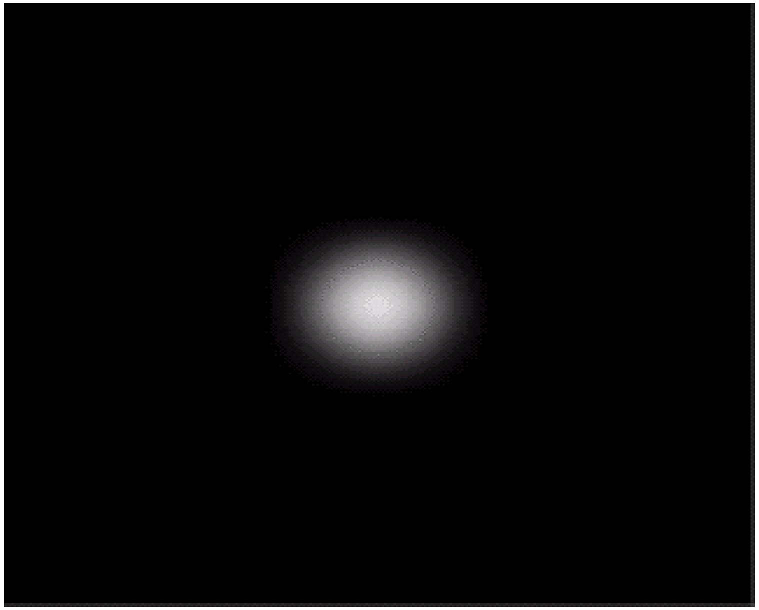
estimate

function(Contd..)

<u>Estimation by experimentation</u>

- If the image acquisition system is ready
- Obtain the impulse response





impulse

Impulse response

function(Contd..)

<u>Estimation by modeling</u>

Ex. Atmospheric model

$$H(u,v) = e^{-k(u^2+v^2)^{5/6}}$$

original

k=0.0025

k=0.001

k=0.00025

Estimating the degradation function(Contd..)

<u>Estimation by modeling</u>

- Derive a mathematical model

  Ex. Motion of image

$$g(x, y) = \int_0^T f(x - x_0(t), y - y_0(t))dt$$

Planar motion

Fourier transform

$$G(u,v) = F(u,v)\int_0^T e^{-j2\pi[ux_0(t)+vy_0(t)]}dt$$

Inverse filtering

- With the estimated degradation function H(u,v)

$$G(u,v)=F(u,v)H(u,v)+N(u,v)$$

Unknown noise

$$\Rightarrow \quad \hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} = F(u,v) + \frac{N(u,v)}{H(u,v)}$$

Estimate of original image

Problem: 0 or small values

Sol: limit the frequency around the origin

Minimum Mean Square Error Filtering

- Wiener filters, on the other hand, are based on a statistical approach
- If the spectral properties of the signals involved are known, a linear time-invariant filter can be designed whose output would be as close as possible to the original signal

- minimum mean square error: $e^2 = E\{ (f-f^c)^2\}$

$$F^c(u,v) = [1/H(u,v)] [|H(u,v|^2 / (|H(u,v|^2 + S_\eta(u,v)/S_f(u,v))] G(u,v)$$

$S_\eta(u,v) = |N(u,v)|^2$ power spectrum of noise

- Approximations of $S_\eta(u,v)/S_f(u,v)$:
  - K (constant)
  - $\gamma\ |P(u,v)|^2$ (power spectrum of Laplacian)
  - γ found by iterative method to minimize e^2 (constrained least squares filtering)

Minimum Mean Square Error Filtering(Contd..)

<u>Wiener Filtering</u>

$$\hat{F}(u,v) = \left[\frac{1}{|H(u,v)|} \frac{|H(u,v)|^2}{|H(u,v)|^2 + K} \right] G(u,v)$$

- $K = S_n(u,v)/S_f(u,v)$,
- $Sn(u,v) = |N(u,v)|^2$
- $Sf(u,v) = |F(u,v)|^2$

- $S_n(u,v)$ & $S_f(u,v)$ must be known
  - $S_n(u,v)$ the power spectrum of the noise,
  - $S_f(u,v)$ the power spectrum of the original image

Minimum Mean Square Error Filtering(Contd..)

Example Wiener filter

Original

Noise added

Pseudo-inverse

Wiener filter

Minimum Mean Square Error Filtering(Contd..)

Linear motion Wiener filter

a b c
d e f
g h i

FIGURE 5.29 (a) Image corrupted by motion blur and additive noise. (b) Result of inverse filtering. (c) Result of Wiener filtering. (d)–(f) Same sequence, but with noise variance one order of magnitude less. (g)–(i) Same sequence, but noise variance reduced by five orders of magnitude from (a). Note in (h) how the deblurred image is quite visible through a "curtain" of noise.

Constrained Least Squares Filtering

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \gamma |P(u,v)|^2} \right] G(u,v)$$

- P(u,v) is the fourier transform of the Laplacian operator

Constraint:

- $|g - H\ \ |^2 = |\eta|^2$
- R(u,v) = G(u,v) – H(u,v)
- Adjust γ from the constraint – by Newton-Raphson root-finding

Constrained Least Squares Filtering(Contd..)

- In the Fourier domain, the constrained least squares filter becomes:

$$F(k,l) = \frac{H^*(k,l)}{\left|H(k,l)\right|^2 + \lambda\left|Q(k,l)\right|^2} G(k,l)$$

- Keep always in mind to zero-pad the images properly.

Constrained Least Squares Filtering(Contd..)

a b c

FIGURE 5.30 Results of constrained least squares filtering. Compare (a), (b), and (c) with the Wiener filtering results in Figs. 5.29(c), (f), and (i), respectively.

Low noise: Wiener and CLS generate equal results.
High noise: CLS outperforms Wiener if λ is properly selected.
It is easier to select the scalar value for λ than to approximate the SNR which is seldom constant.

Geometric mean filter

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2}\right]^{\alpha}\left[\frac{H^*(u,v)}{|H(u,v)|^2 + \beta\left[\frac{S_\eta(u,v)}{S_f(u,v)}\right]}\right]^{1-\alpha} G(u,v)$$

•Geometric mean filter is quite useful when implementing restoration filters because it represents a family of filters combined into a single expression

UNIT 4
IMAGE PROCESSING

Course Learning Outcome

CLOs	Course Learning Outcome
CLO13	Understand the various kind of noise present in the image and how to restore the noisy image.
CLO14	Understand the differences of inverse, least square and Wiener filtering in restoration process of images
CLO15	Understand the color fundamentals and models in image processing
CLO16	Memorize the transformation techniques in pseudo color image processing.
CLO17	Use wavelet concepts in multi-resolution processing.

Color Image Fundamentals

The use of color is important in image processing because

•Color is a powerful descriptor that simplifies object identification and extraction.

•Humans can discern thousands of color shades and intensities, compared to about only two dozen shades of gray.

Color image processing is divided into two major areas:

•Full color processing :images are acquired with a full color sensor, such as a color TV camera or color scanner.

•Pseudo color processing: The problem is one of assigning a color to a particular monochrome intensity or range of intensities.

Color Image Fundamentals(Contd..)

•Physical phenomenon

Physical nature of color is known

•Psysio-psychological phenomenon

How human brain perceive and interpret color?

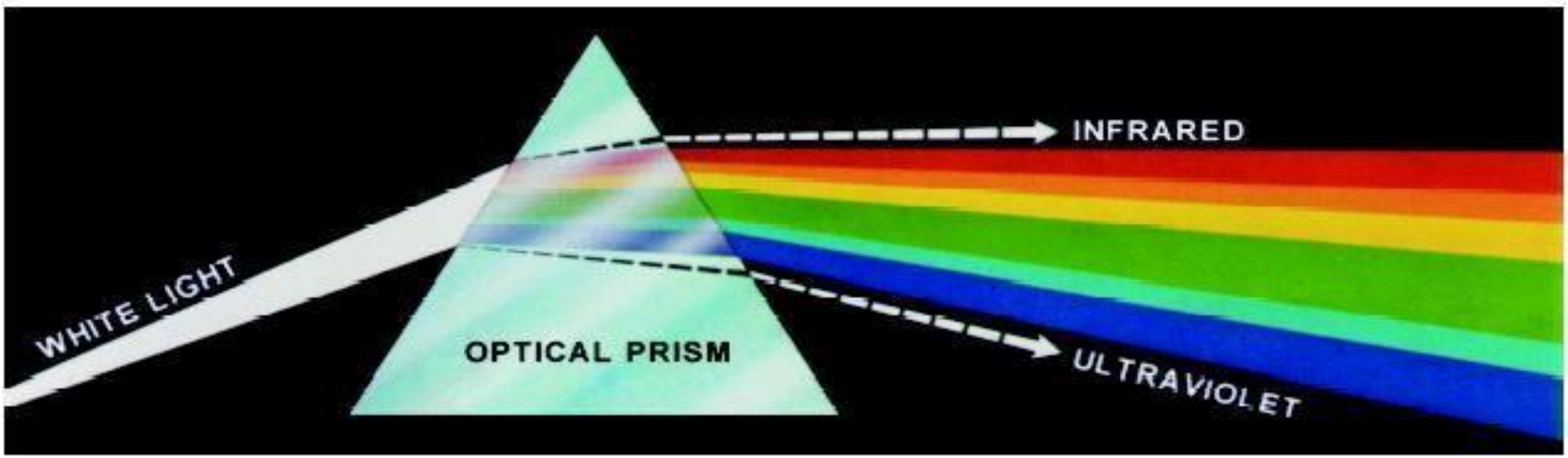


FIGURE 6.1 Color spectrum seen by passing white light through a prism. (Courtesy of the General Electric Co., Lamp Business Division.)

Color Image Fundamentals (Contd..)

Visible light:

Chromatic light span the electromagnetic spectrum (EM) from 400 to 700 nm

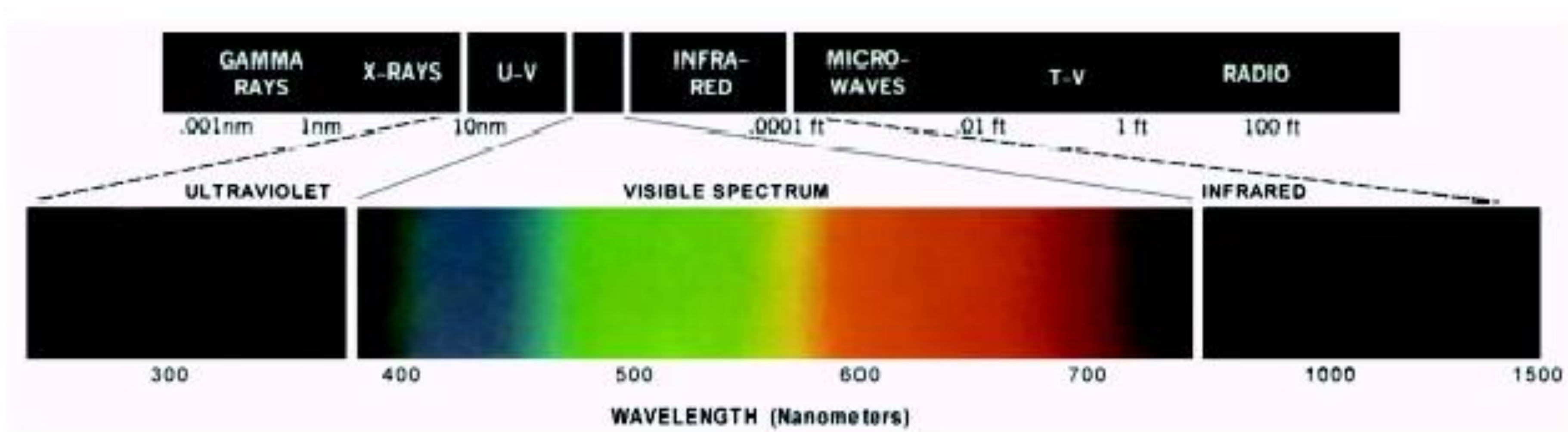
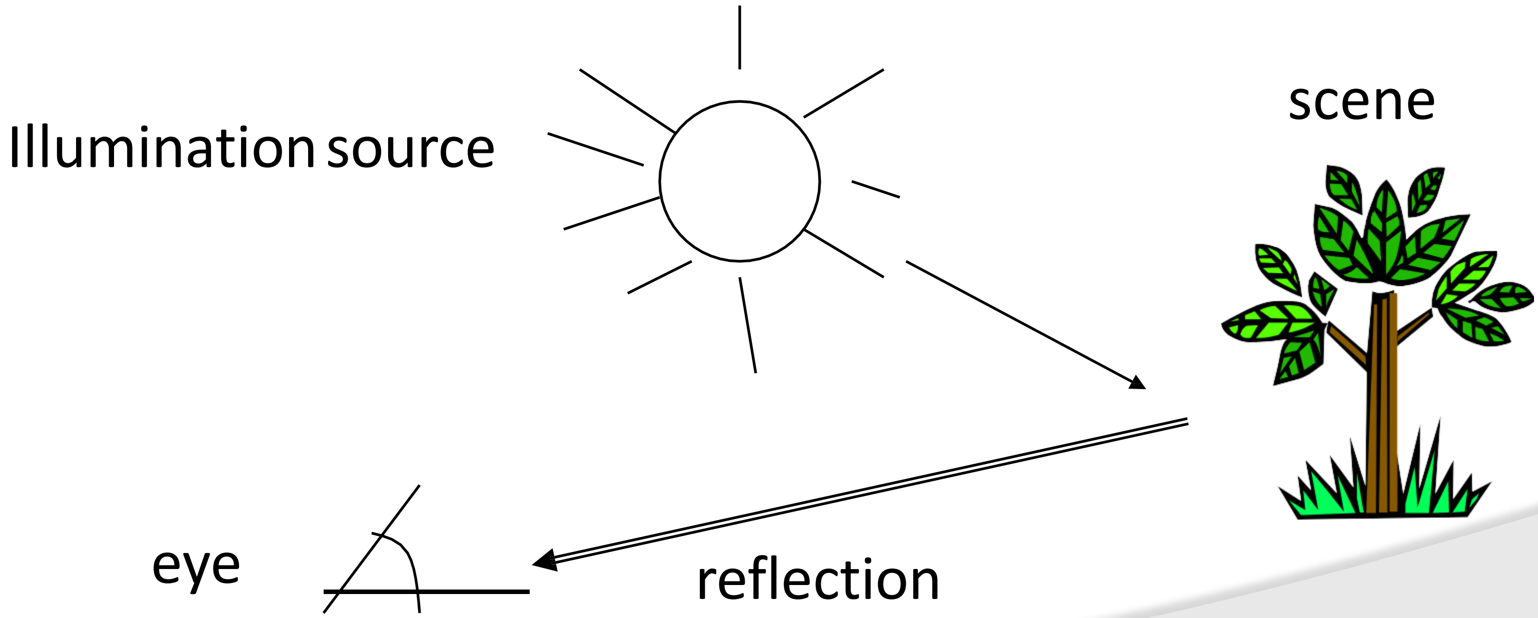


FIGURE 6.2 Wavelengths comprising the visible range of the electromagnetic spectrum. (Courtesy of the General Electric Co., Lamp Business Division.)

Color Image Fundamentals(Contd..)

•The color that human perceive in an object = the light reflected from the object



Color Image Fundamentals (Contd..)

Physical quantities to describe a chromatic light source

Radiance: total amount of energy that flow from the light source, measured in watts (W)

Luminance: amount of energy an observer *perceives* from a light source, measured in lumens (lm)

•Far infrared light: high radiance, but 0 luminance

Brightness: subjective descriptor that is hard to measure, similar to the achromatic notion of intensity.

How human eyes sense light?

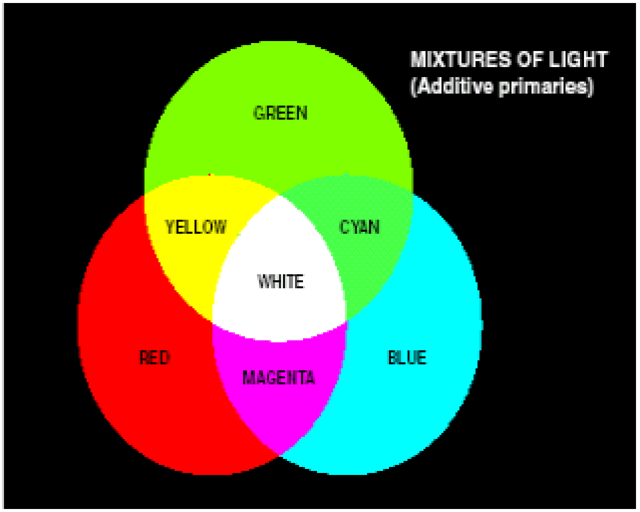
•6~7M Cones are the sensors in the eye

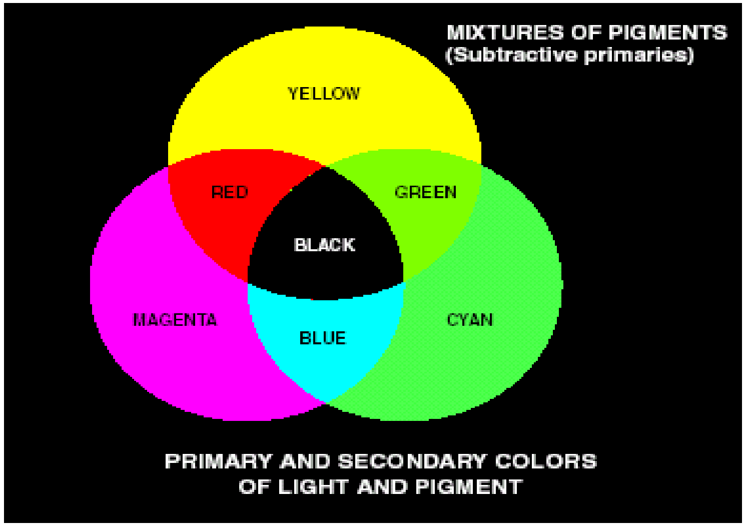
•3 principal sensing categories in eyes

•Red light 65%, green light 33%, and blue light 2%

Primary and secondary colors

- In 1931, CIE(International Commission on Illumination) defines specific wavelength values to the primary colors
  - B = 435.8 nm, G = 546.1 nm, R = 700 nm
  - However, we know that no single color may be called red, green, or blue
- Secondary colors: G+B=Cyan, R+G=Yellow, R+B=Magenta





Color Models

•Color model, color space, color system

- •Specify colors in a standard way
- •A coordinate system that each color is represented by a single point

•RGB model

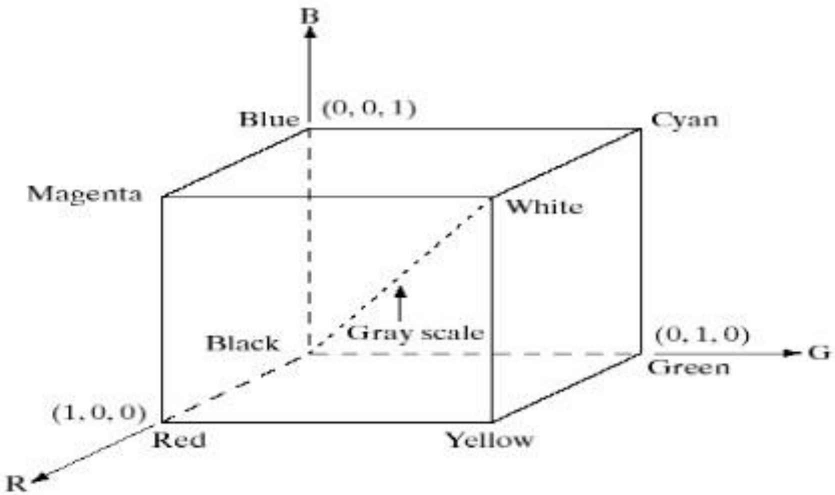
•CYM model

•CYMK model

•HSI model

•Suitable for hardware or applications

<u>RGB color model</u>

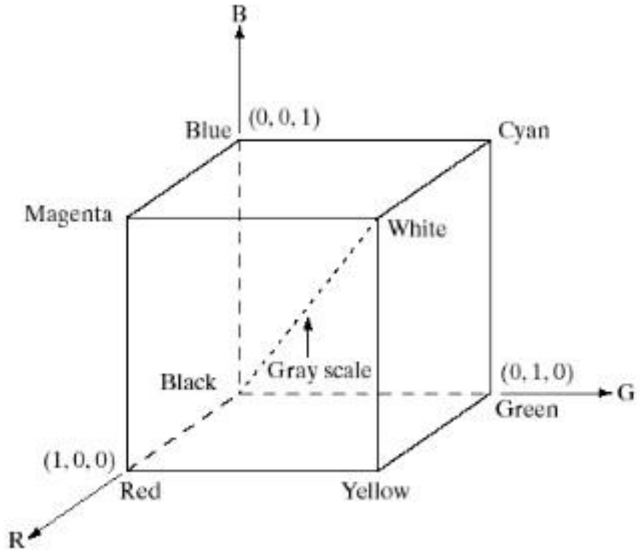


Color Models(Contd..)

•Pixel depth: the number of bits used to represent each pixel in RGB space

•Full-color image: 24-bit RGB color image

- •(R, G, B) = (8 bits, 8 bits, 8 bits)

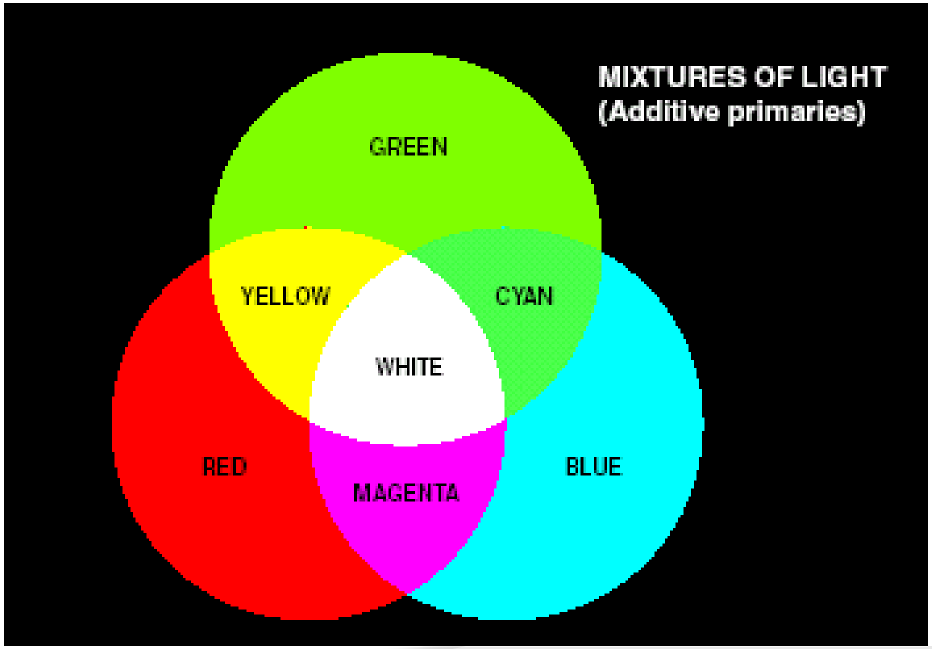


Color Models(Contd..)

CMY model (+Black = CMYK)

- CMY: secondary colors of light, or primary colors of pigments
- Used to generate hardcopy output

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

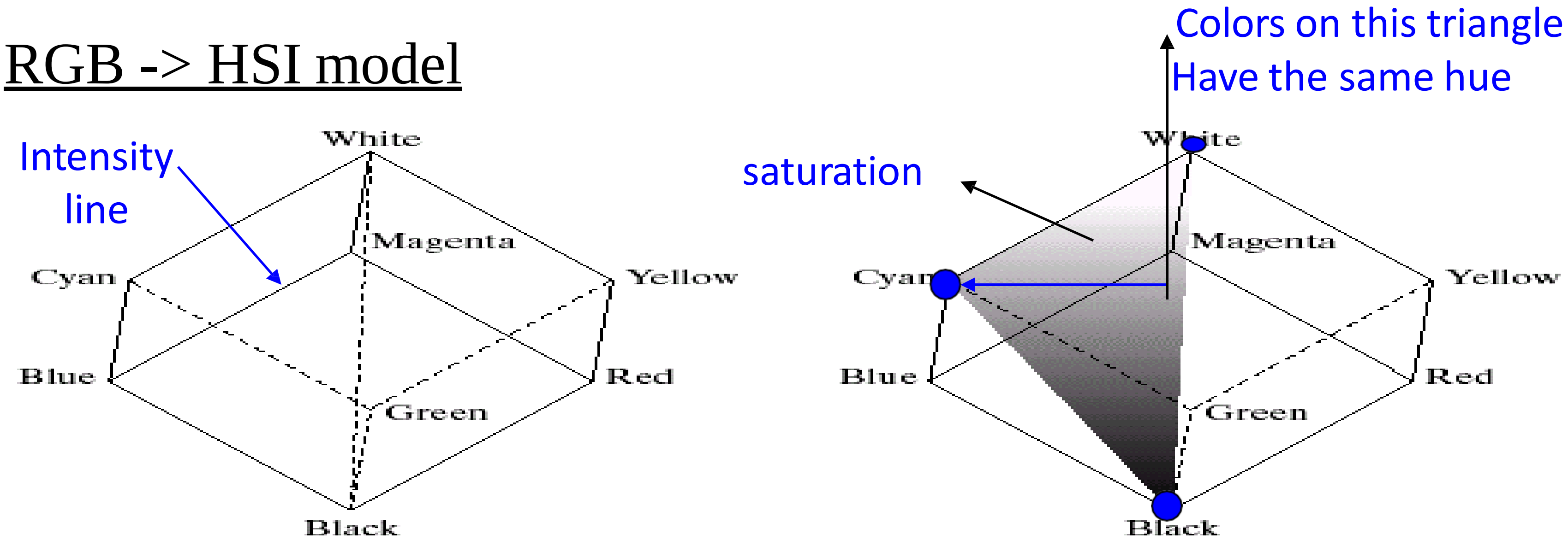


Color Models(Contd..)

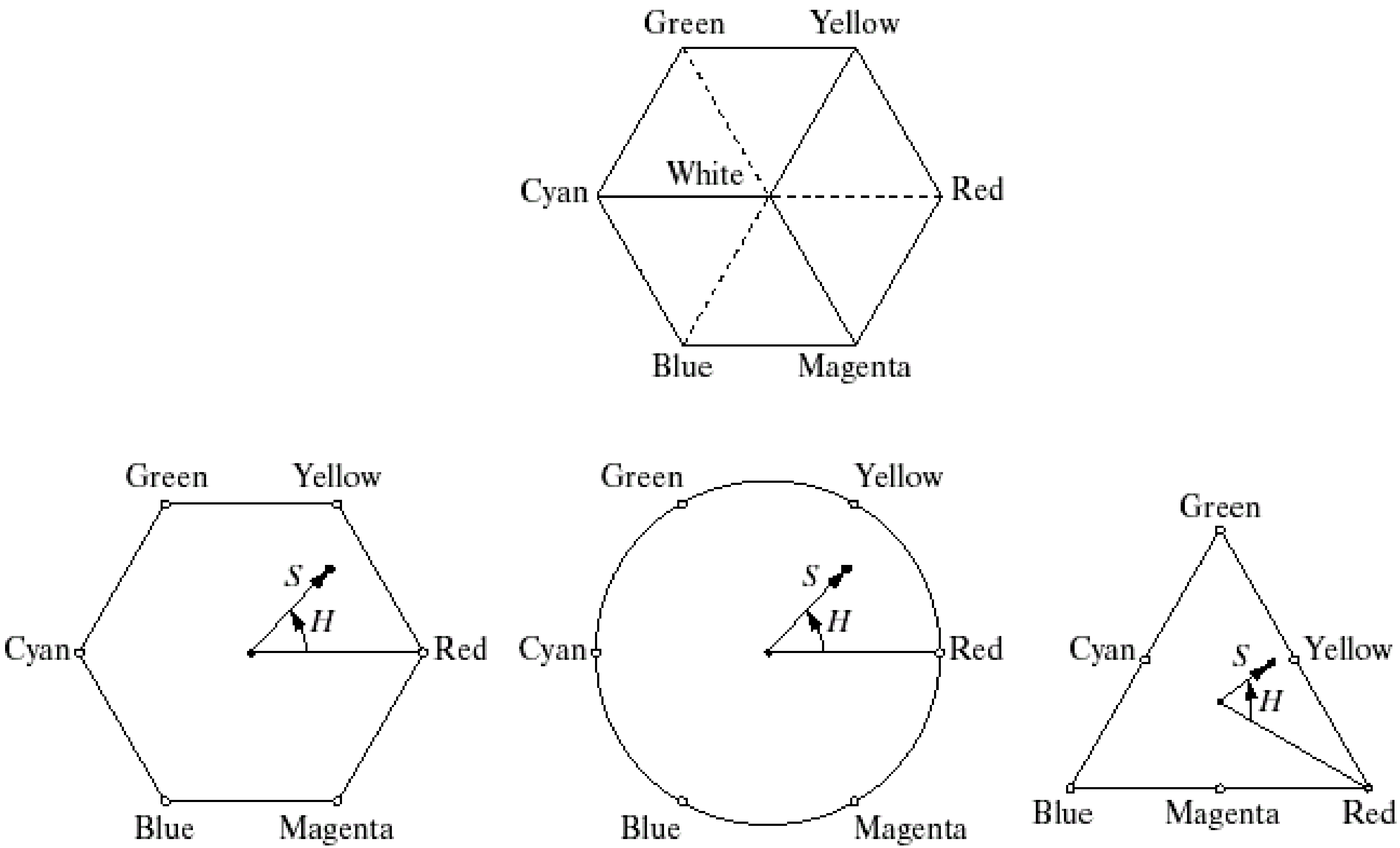
HSI color model

- Will you describe a color using its R, G, B components?
- Human describe a color by its hue, saturation, and brightness
    - Hue : color attribute
    - Saturation: purity of color (white->0, primary color->1)
    - Brightness: achromatic notion of intensity

RGB -> HSI model

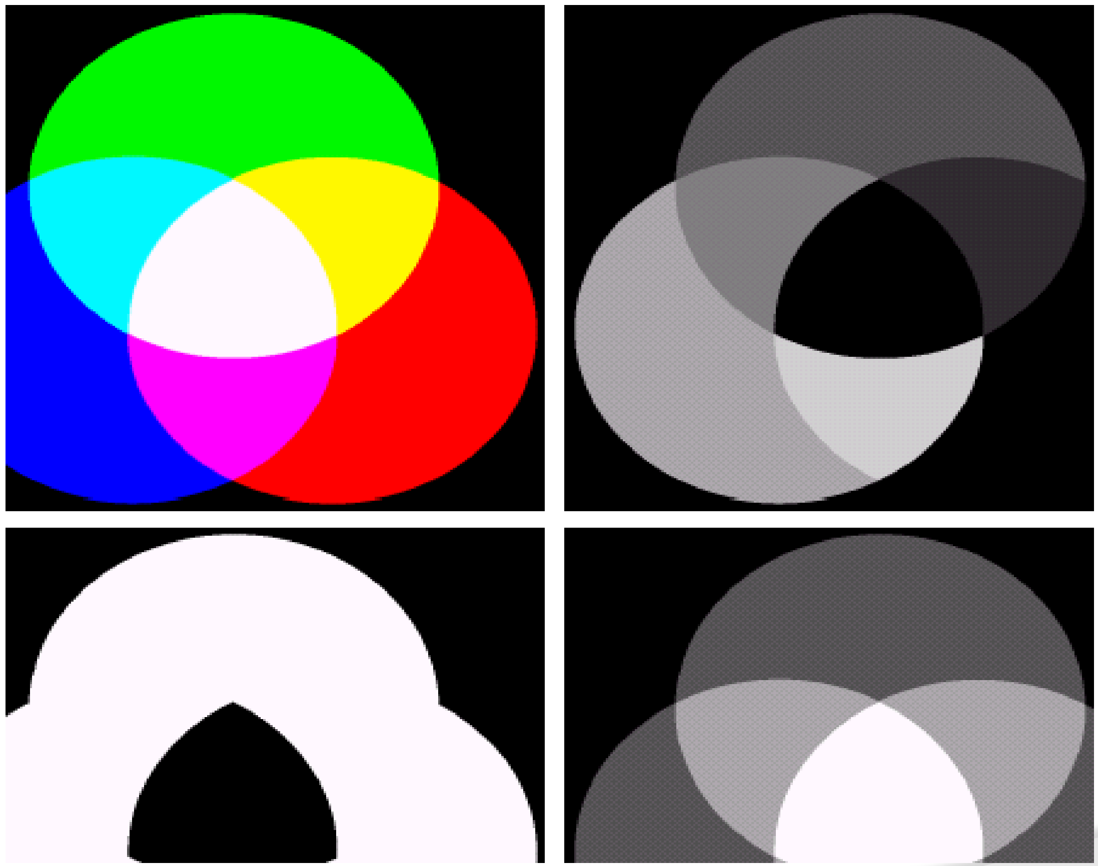


Color Models(Contd..)



Color Models(Contd..)

HSI component images



R,G,B

Hue

saturation

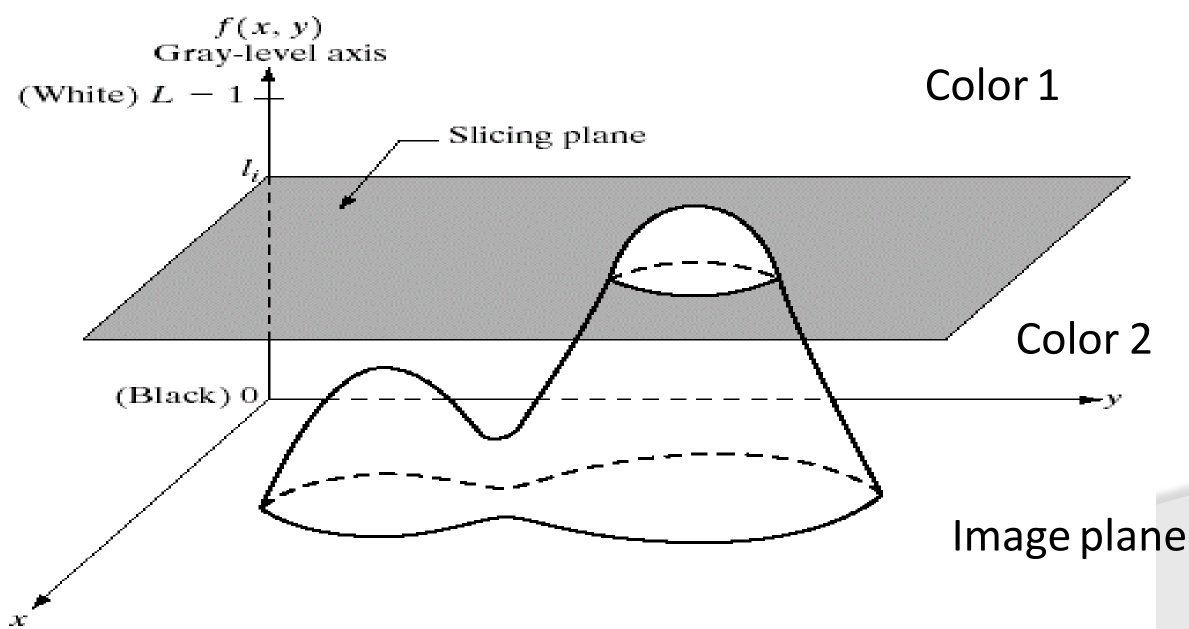
intensity

Pseudo-color image processing

- Assign colors to gray values based on a specified criterion
- For human visualization and interpretation of gray-scale events
- Intensity slicing
- Gray level to color transformations

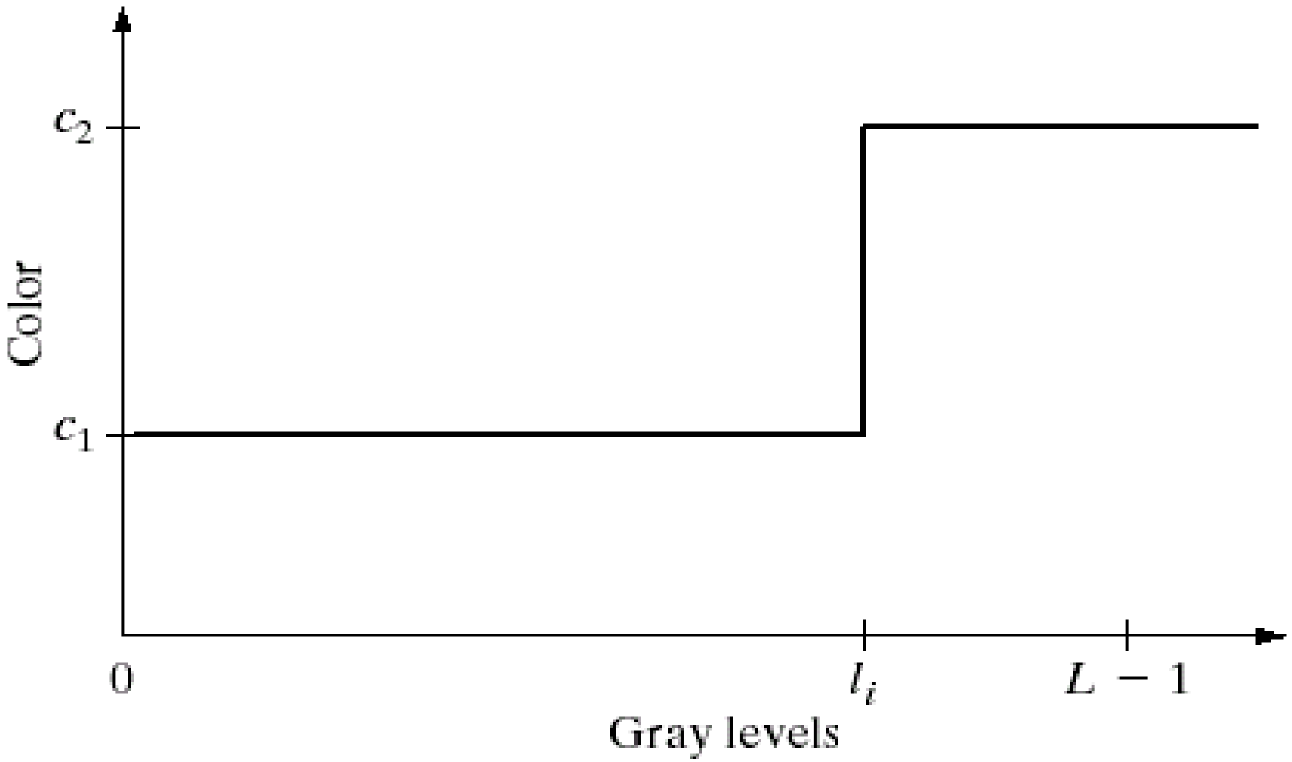
Intensity slicing

3-D view of intensity image



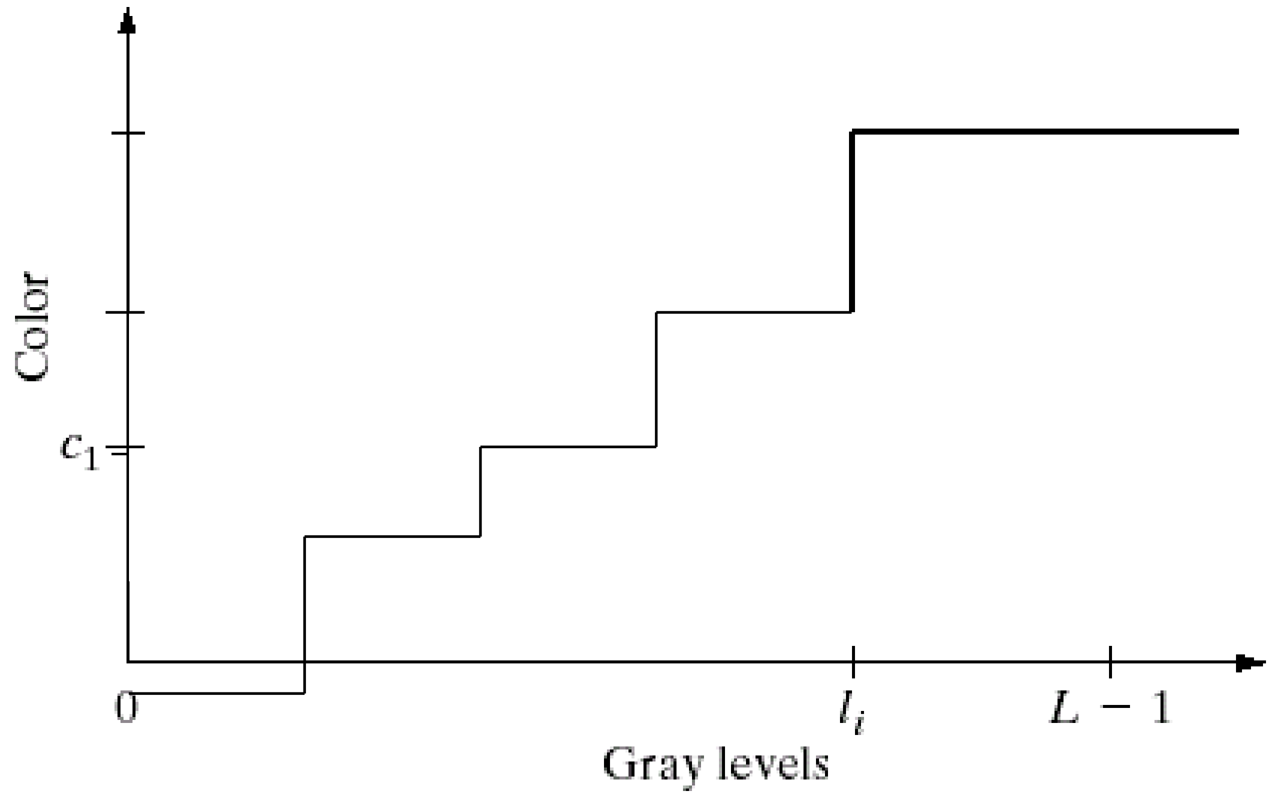
Pseudo-color image processing(Contd.)

Alternative representation of intensity slicing



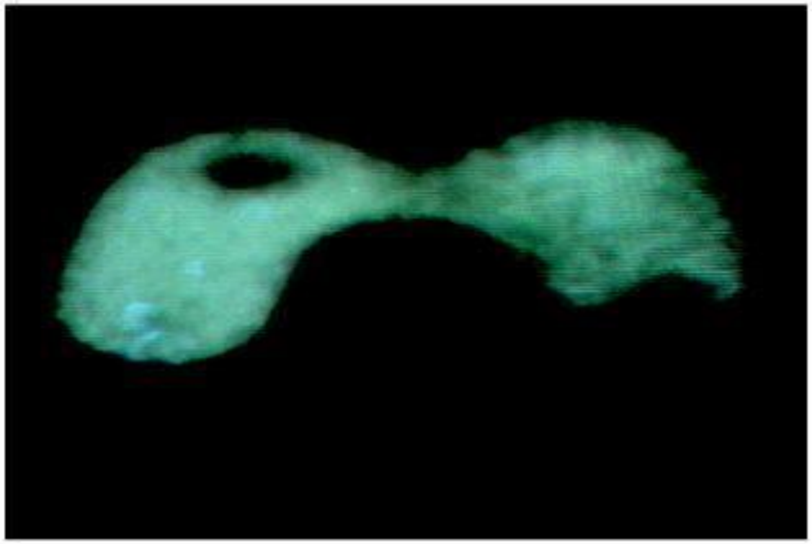
Pseudo-color image processing(Contd..)

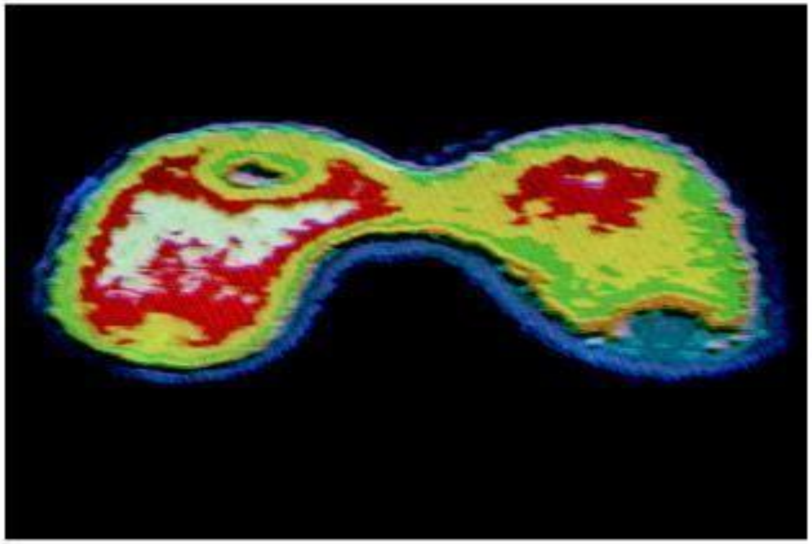
More slicing plane, more colors



Pseudo-color image processing(Contd..)

Application of Intensity slicing





Radiation test pattern → 8 color regions

Pseudo-color image processing(Contd..)

Gray level to color transformation

- General Gray level to color transformation

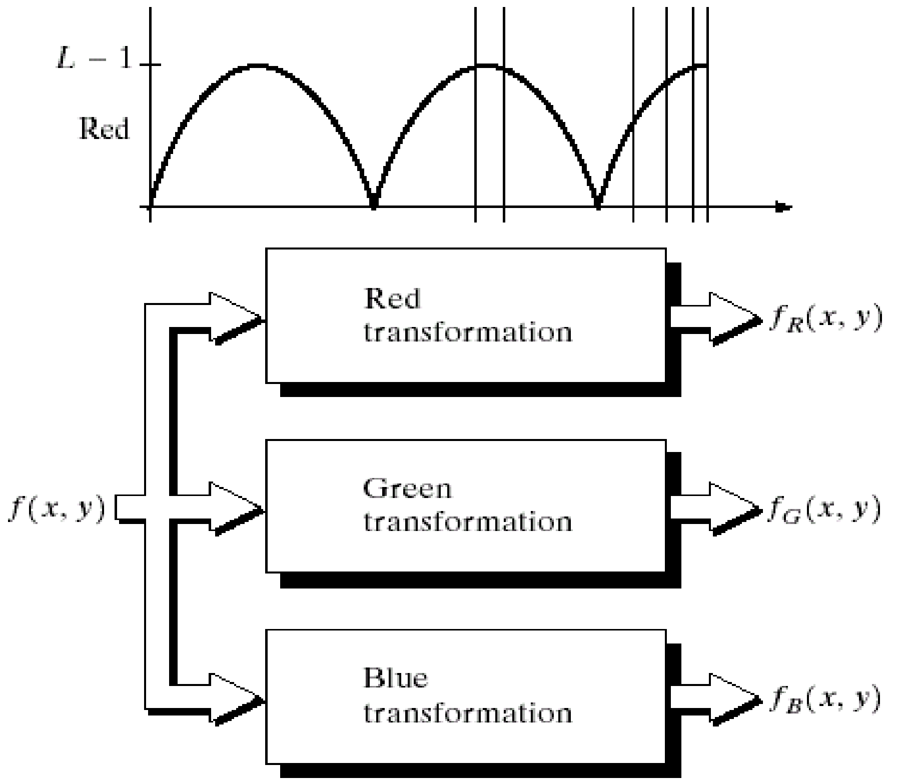
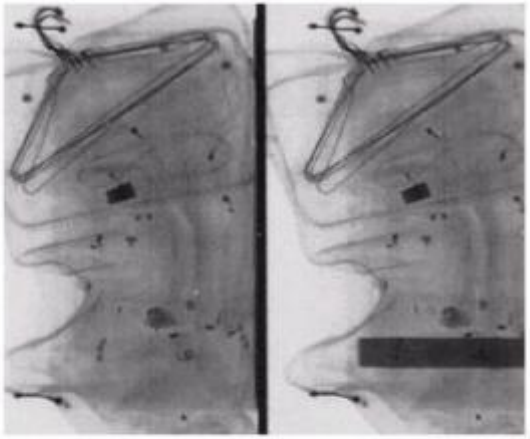
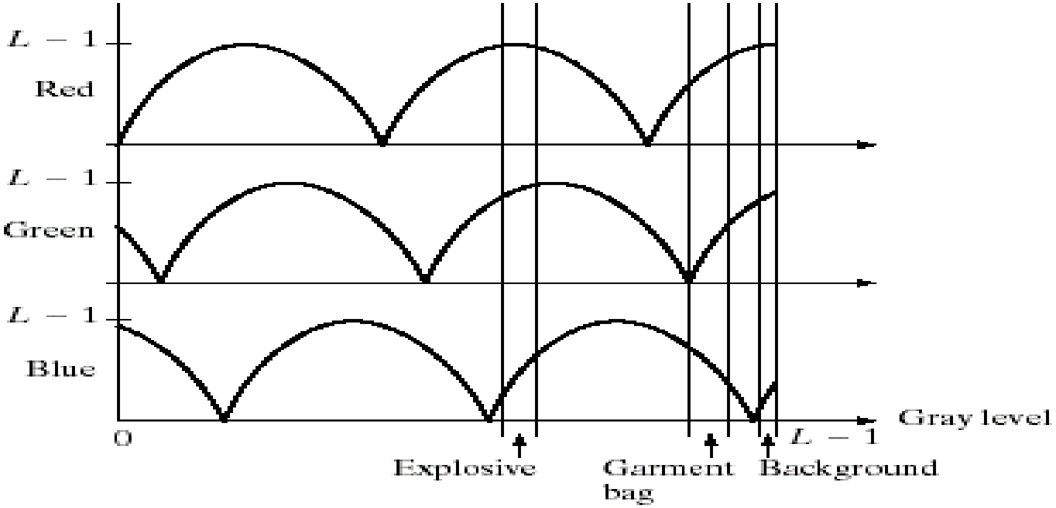


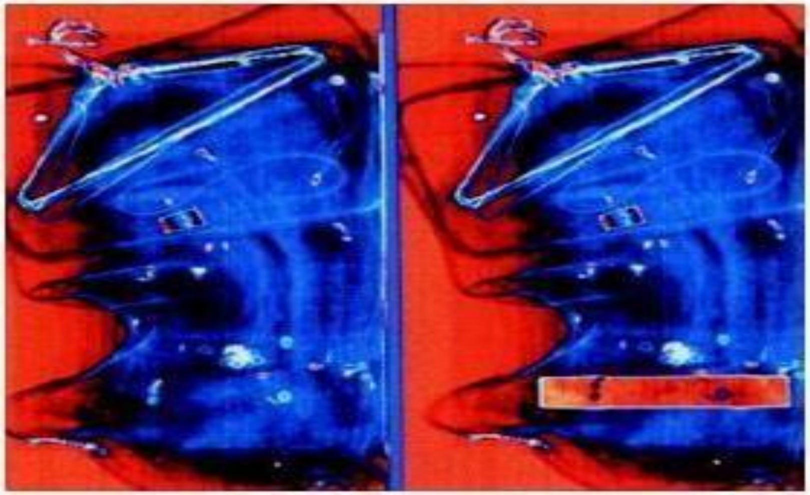
FIGURE 6.23 Functional block diagram for pseudocolor image processing. f_R, f_G, and f_B are fed into the corresponding red, green, and blue inputs of an RGB color monitor.

Pseudo-color image processing(Contd..)

Application of gray level to color transformation







Basics of Full-Color Image Processing

Color pixel

- A pixel at (x,y) is a vector in the color space
  - RGB color space

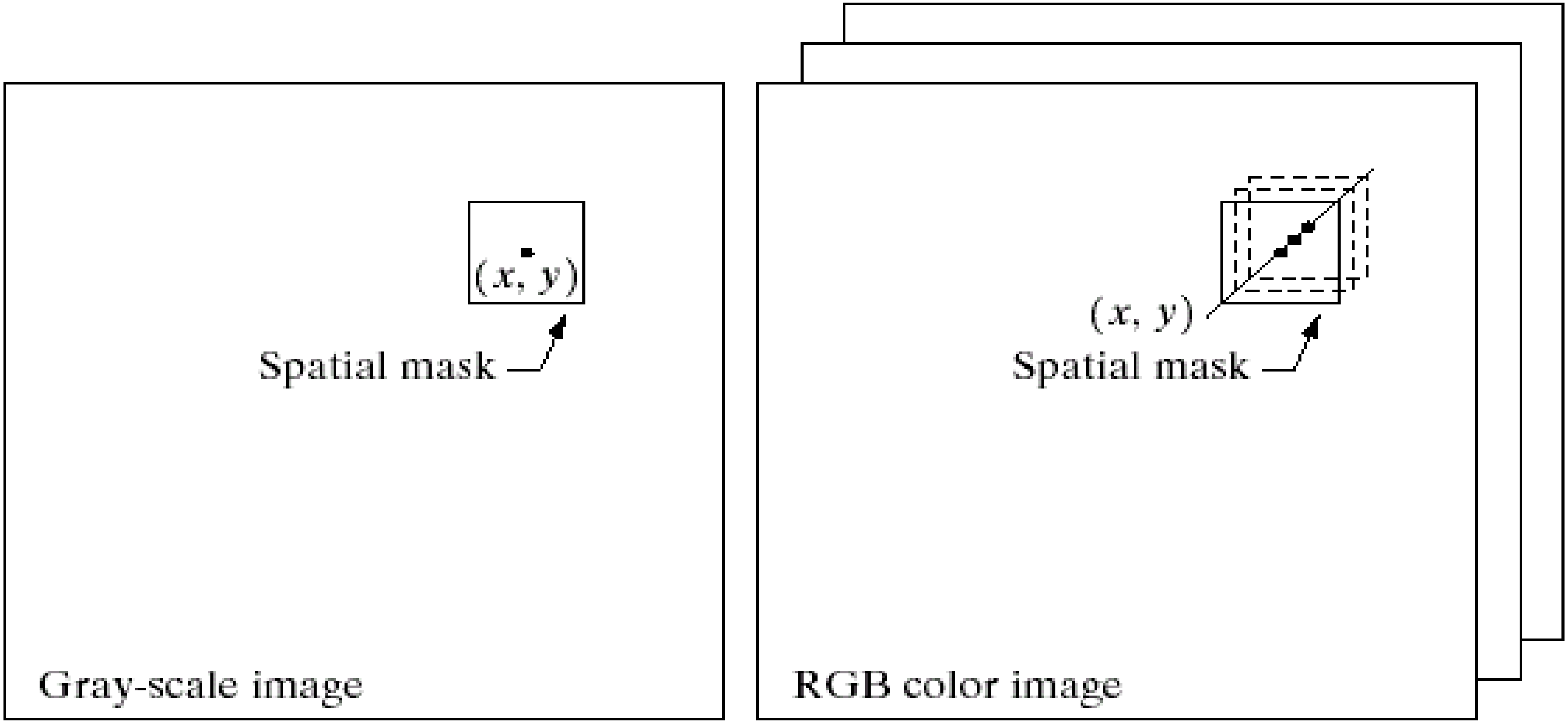
$$\mathbf{c}(x, y) = \begin{bmatrix} R(x, y) \\ G(x,y) \\ B(x, y) \end{bmatrix}$$

c.f. gray-scale image

f(x,y) = I(x,y)

Basics of Full-Color Image Processing(contd..)

Example: spatial mask



How to deal with color vector?

- Per-color-component processing
  - Process each color component
- Vector-based processing
  - Process the color vector of each pixel
- When can the above methods be equivalent?
  - Process can be applied to both scalars and vectors
  - Operation on each component of a vector must be independent of the other component

Basics of Full-Color Image Processing(contd..)

Two spatial processing categories

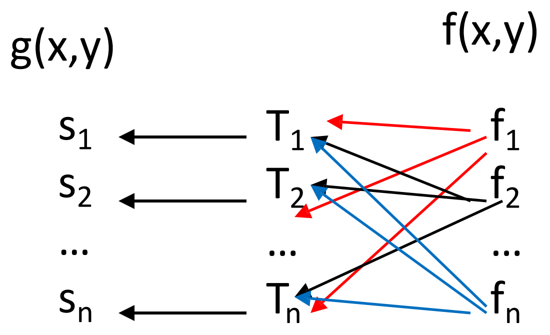
•Similar to gray scale processing studied before, we have to major categories

- •Pixel-wise processing
- •Neighborhood processing

Color Transformations

- Similar to gray scale transformation
  g(x,y)=T[f(x,y)]
- Color transformation

$$s_i = T_i(r_1, r_2, \dots, r_n)\,, \quad i = 1,2,\dots,n$$



Color Transformations(Contd..)

Use which color model in color transformation?

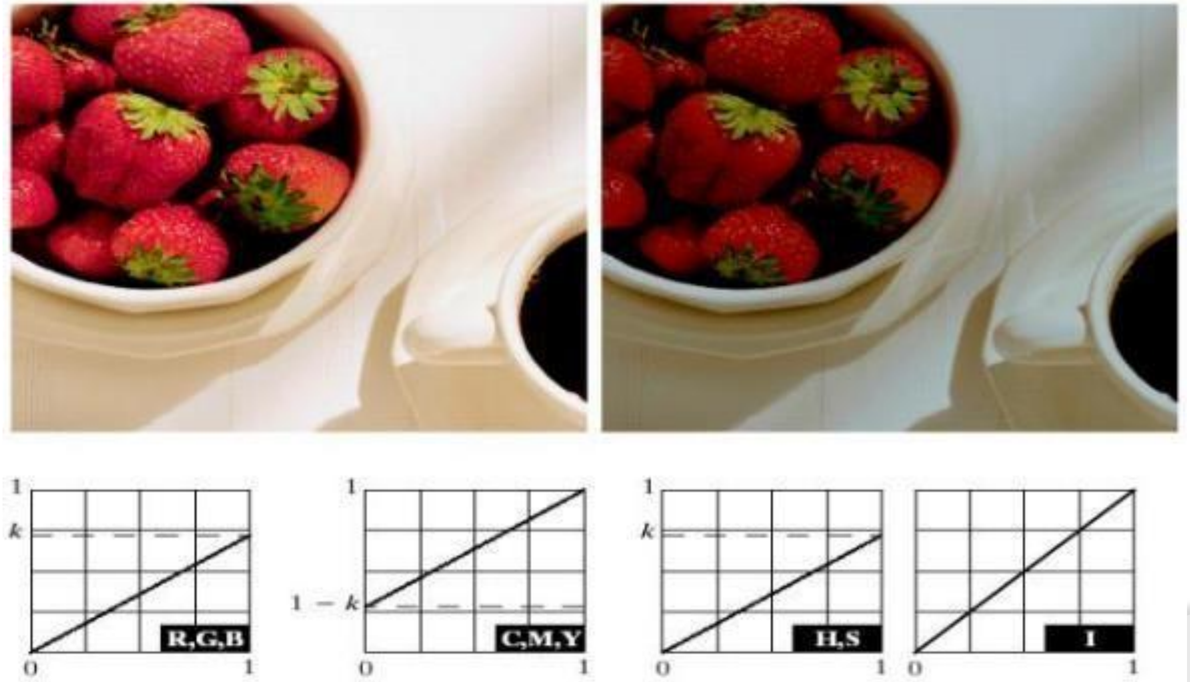
- RGB ⇔CMY(K) ⇔ HSI
- Theoretically, any transformation can be performed in any color model
- Practically, some operations are better suited to specific color model

Example: modify intensity of a color image

- Example: $g(x,y)=k\,f(x,y)$, $0<k<1$
- HSI color space
  - Intensity: $s_3 = k\,r_3$
  - Note: transform to HSI requires complex operations

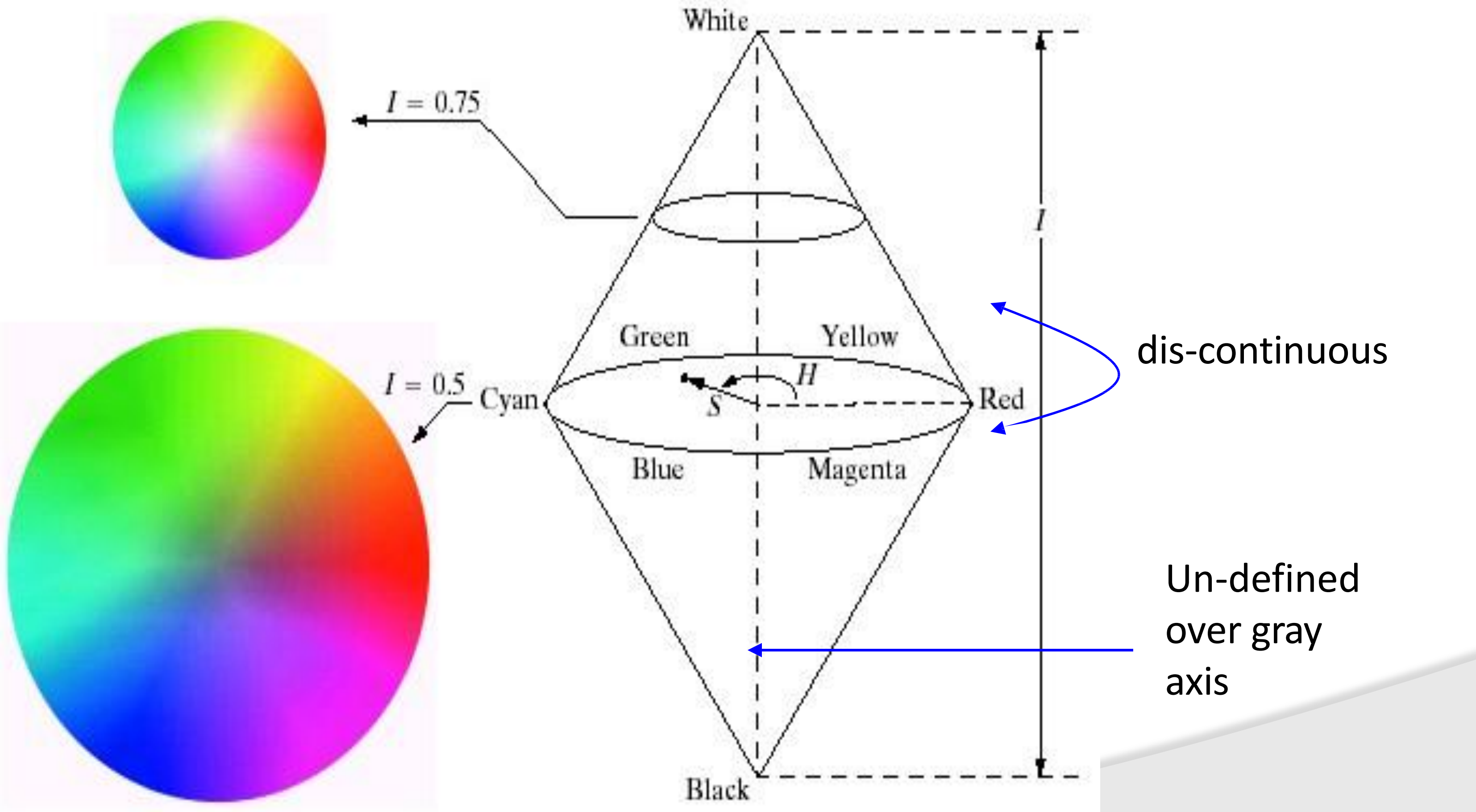
Color Transformations(Contd..)

- RGB color space
  - For each R,G,B component: $s_i = k\, r_i$
- CMY color space
  - For each C,M,Y component:
  - $s_i = k\, r_i + (1-k)$



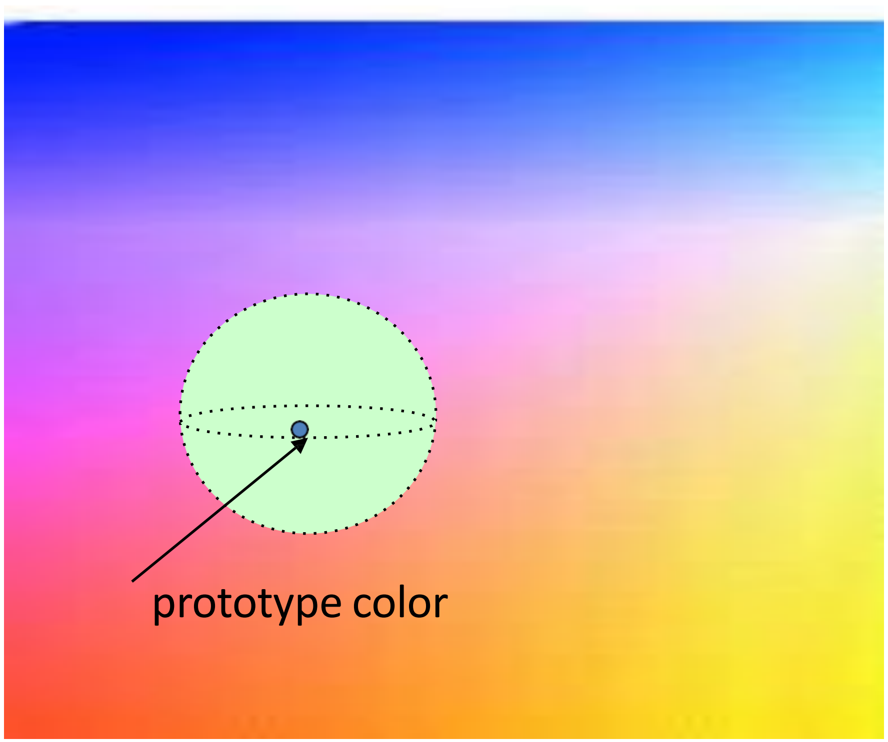
Color Transformations(Contd..)

Problem of using Hue component

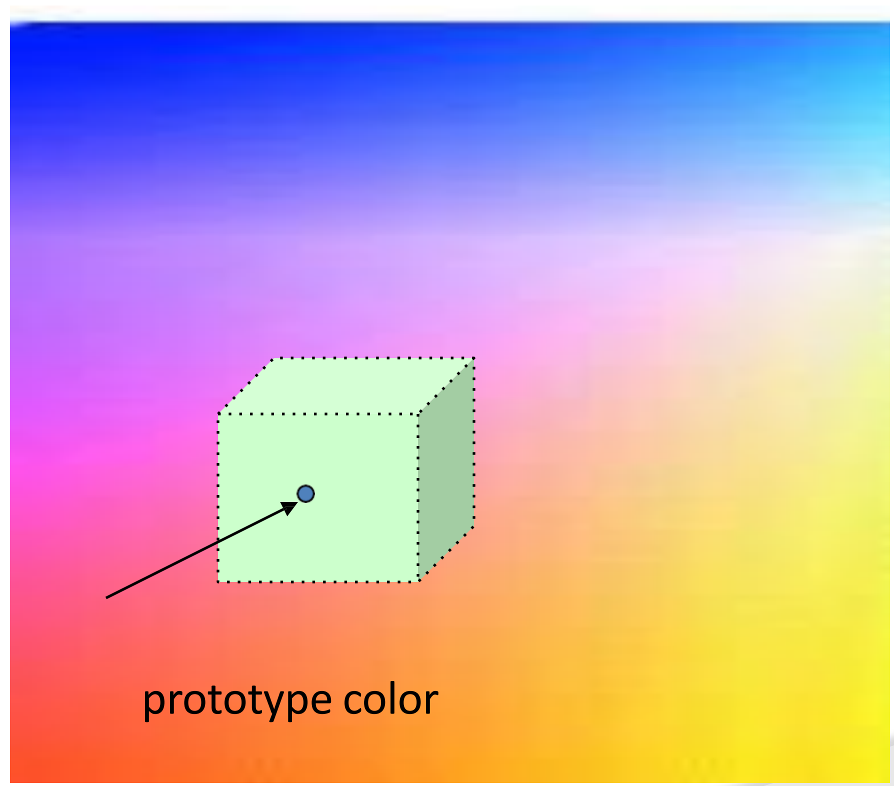


Color Transformations(Contd..)

Implementation of color slicing



Sphere region



Cube region

Color Transformations(Contd..)

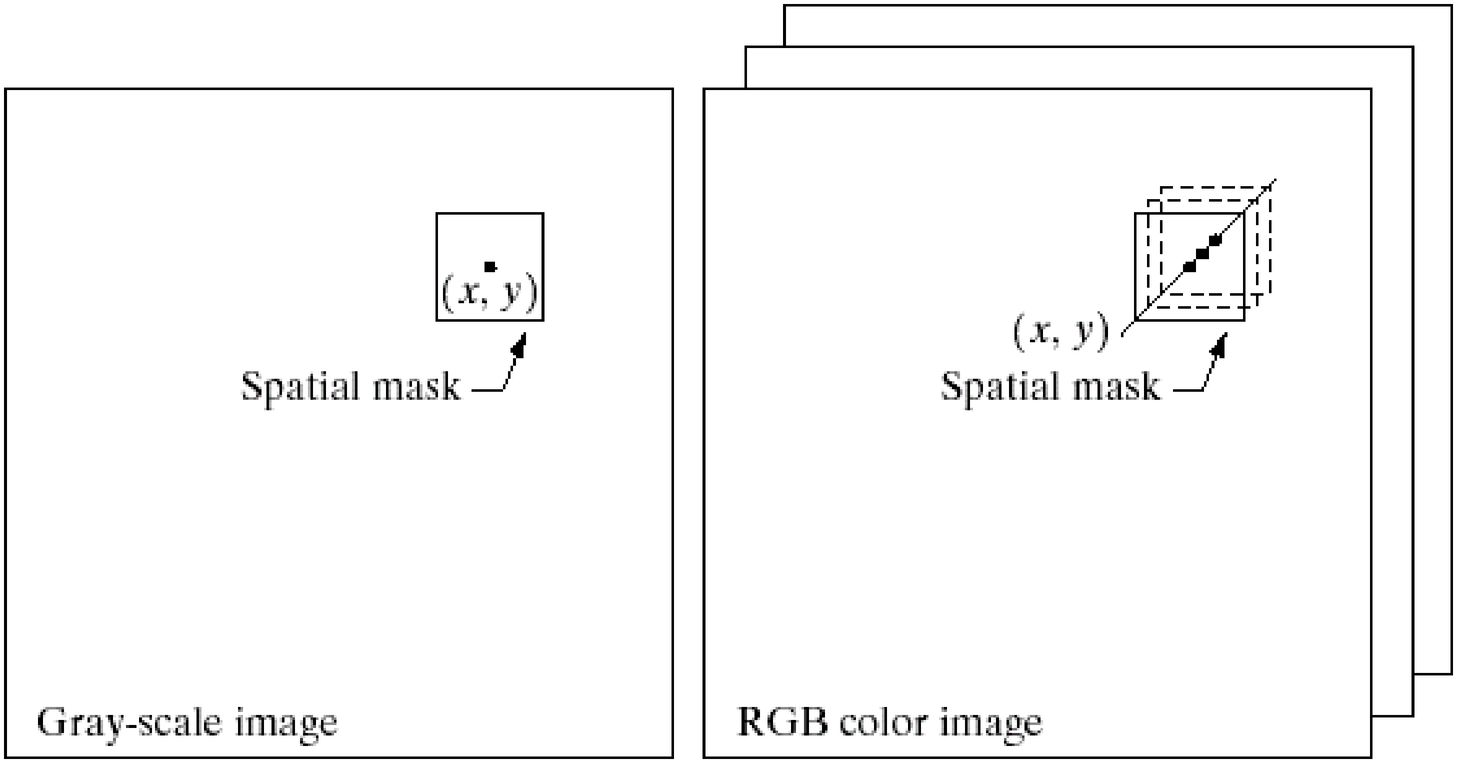
Application

cube

sphere

Color image smoothing

- Neighborhood processing



Color image smoothing: averaging mask

$$\bar{\mathbf{c}}(x, y) = \frac{1}{K} \sum_{(x,y)\in S_{xy}} \mathbf{c}(x, y)$$

vector processing

Neighborhood
Centered at (x,y)

$$\bar{\mathbf{c}}(x, y) = \begin{bmatrix} \frac{1}{K} \sum_{(x,y)\in S_{xy}} R(x, y) \\ \frac{1}{K} \sum_{(x,y)\in S_{xy}} G(x, y) \\ \frac{1}{K} \sum_{(x,y)\in S_{xy}} B(x, y) \end{bmatrix}$$

per-component processing

Smoothing and Sharpening(Contd..)

Smoothing and Sharpening(Contd..)

Example: 5x5 smoothing mask

RGB model

Smooth I
in HSI model

difference

a b c

FIGURE 6.40 Image smoothing with a 5 × 5 averaging mask. (a) Result of processing each RGB component image. (b) Result of processing the intensity component of the HSI image and converting to RGB. (c) Difference between the two results.

Color Segmentation

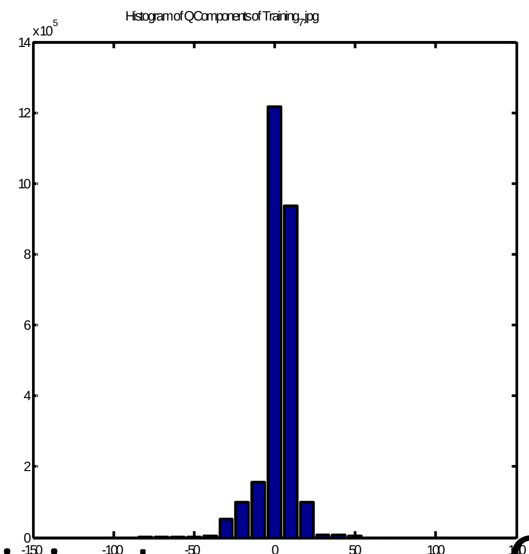
•View the YIQ color space:

-Y=luminance, I=hue, Q=saturation

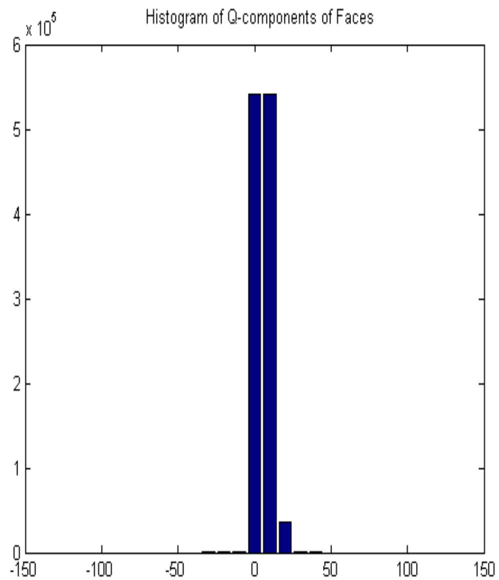
•Human skin occupy a small portion of the I and Q spaces.

•From training images, compare and contrast hue and saturation of faces only vs. entire image

Hue and Saturation



Training Image

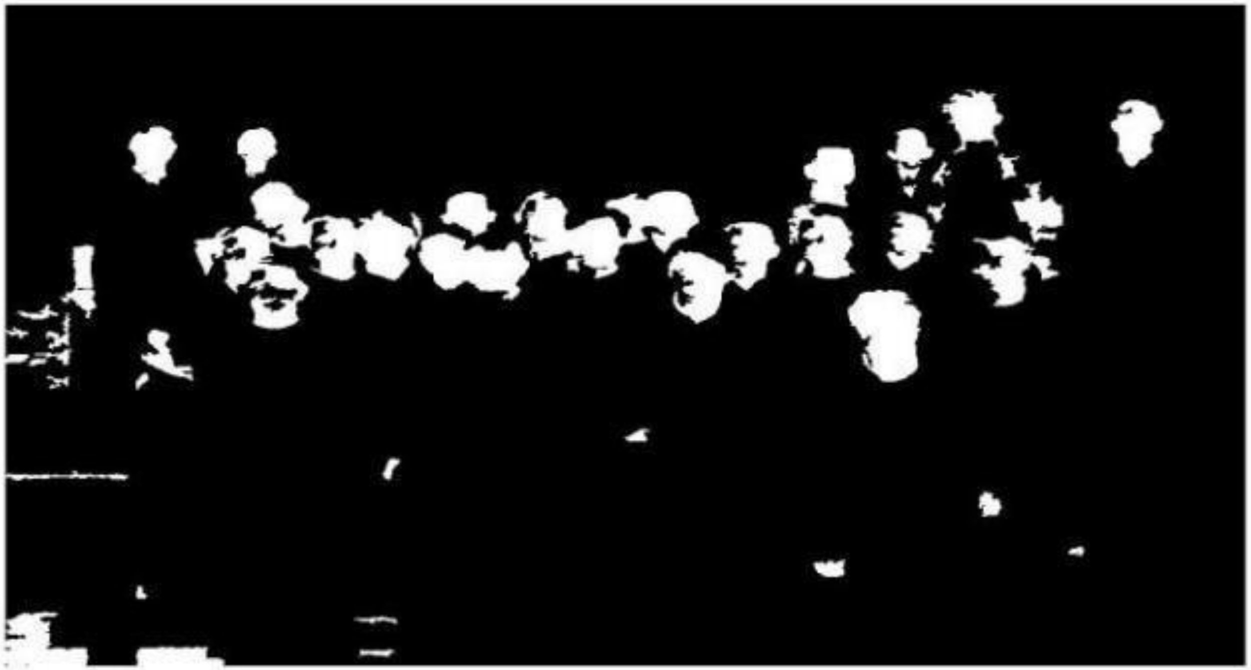


Faces

Q Distribution

Color Segmentation(Contd..)

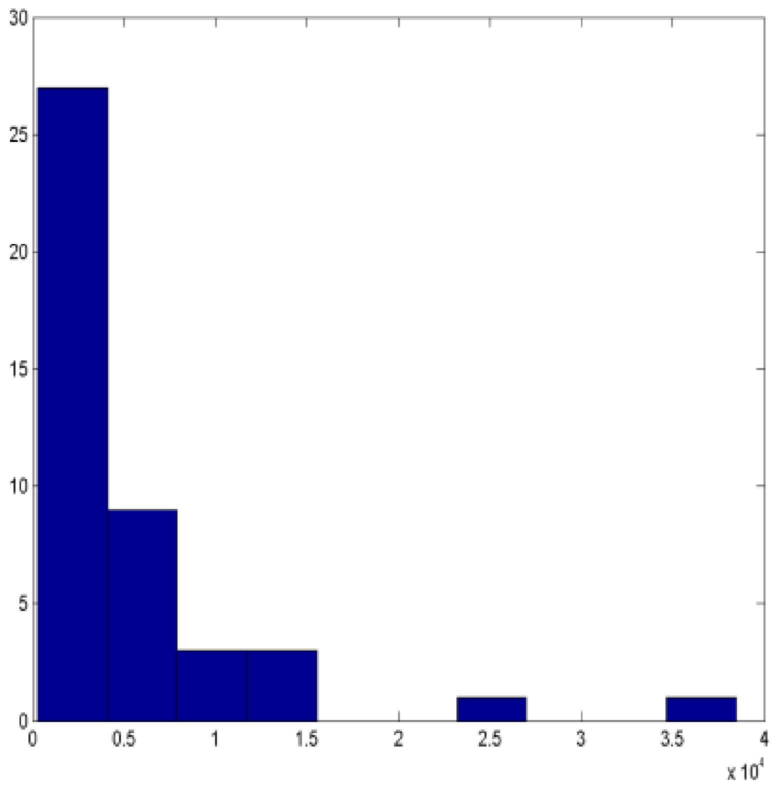
Mask After Color Segmentation

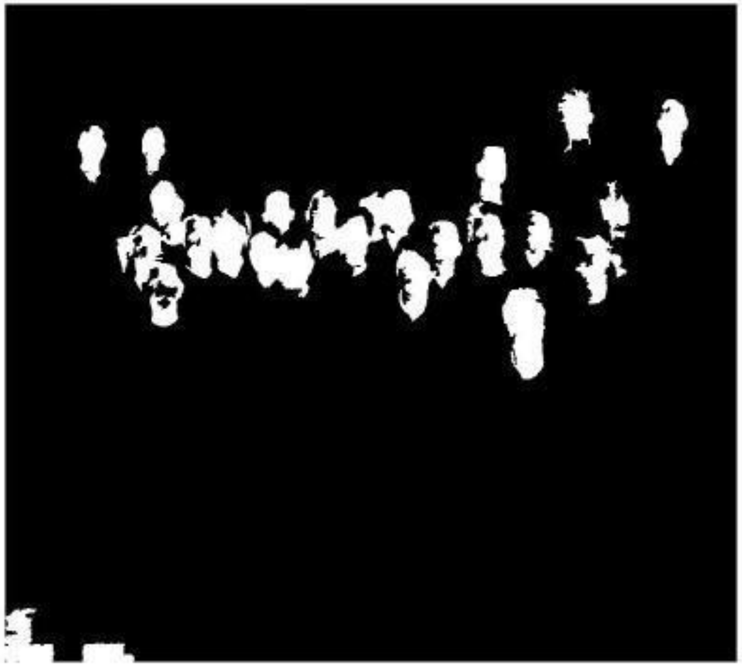


- Skin elements remain.
- Holes in faces later eliminated with hole-filling

Color Segmentation(Contd..)

Mask After Object Removal





Based on size distribution of remaining objects, remove small ones

Color Segmentation(Contd..)

<u>Region counting - Supplementary method</u>

- The edge outlines have clearly identifiable connected regions
- Can be counted, and statistics used to help reject clutter

Number of regions: 14

Number of regions: 43

Color Segmentation(Contd..)

Detection Algorithm

- Correlation – Degree of matching
- Dimensions – height, width
- Region counting – complexity of image

Single face

Correlation → Dimensions → Region counting

Multiple faces

Correlation → Dimensions → Region counting

Multi-face detection

Noise in Color Images

Noise can corrupt each color component independently

a b
c d

FIGURE 6.48 (a)–(c) Red, green, and blue component images corrupted by additive Gaussian noise of mean 0 and variance 800. (d) Resulting RGB image. [Compare (d) with Fig. 6.46(a).]

Noise is less noticeable in a color image

Noise in Color Images(Contd..)

FIGURE 6.49 HSI components of the noisy color image in Fig. 6.48(d). (a) Hue. (b) Saturation. (c) Intensity.

Noise in Color Images(Contd..)

a b
c d

FIGURE 6.50 (a) RGB image with green plane corrupted by salt-and-pepper noise. (b) Hue component of HSI image. (c) Saturation component. (d) Intensity component.

Color Image Compression

Original image

JPEG2000 FILE

After lossy compression with ratio 230:1

Wavelets and Multi-resolution Processing

- Fourier transform has its basis functions in sinusoids
- Wavelets based on small waves of varying frequency and limited duration
- In addition to frequency, wavelets capture temporal information
    - ✓Bound in both frequency and time domains
    - ✓Localized wave and decays to zero instead of oscillating forever
- Form the basis of an approach to signal processing and analysis known as *multiresolution theory*
    - ✓Concerned with the representation and analysis of images at different resolutions
    - ✓Features that may not be prominent at one level can be easily detected at another level

Wavelets and Multi-resolution Processing(Contd..)

Comparison with Fourier transform

- Fourier transform used to analyze signals by converting signals into a continuous series of sine and cosine functions, each with a constant frequency and amplitude, and of infinite duration
- Real world signals (images) have a finite duration and exhibit abrupt changes in frequency
- Wavelet transform converts a signal into a series of wavelets
- In theory, signals processed by wavelets can be stored more efficiently compared to Fourier transform
- Wavelets can be constructed with rough edges, to better approximate real-world signals
- Wavelets do not remove information but move it around, separating out the noise and averaging the signal
- Noise (or detail) and average are expressed as sum and difference of signal, sampled at different points

Wavelets and Multi-resolution Processing(Contd..)

- Objects in images are connected regions of similar texture and intensity levels
- Use high resolution to look at small objects; coarse resolution to look at large objects
    - If you have both large and small objects, use different resolutions to look at them
    - Images are 2D arrays of intensity values with locally varying statistics

Image Pyramids

- Originally devised for machine vision and image compression.
- It is a collection of images at decreasing resolution levels.
- Base level is of size 2^Jx2^J or NxN.
- Level j is of size 2^jx2^j.

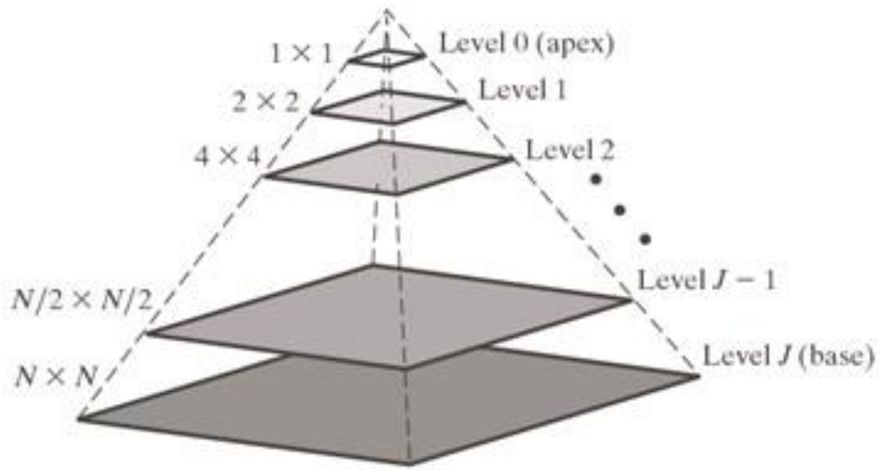


Image Pyramids(Contd..)

What is an Image Pyramid?

Low resolution

High resolution

Image Pyramids(Contd..)

<u>Approximation pyramid:</u>

At each reduced resolution level we have a filtered and downsampled image.

$$f_{\downarrow 2}(n) = f(2n)$$

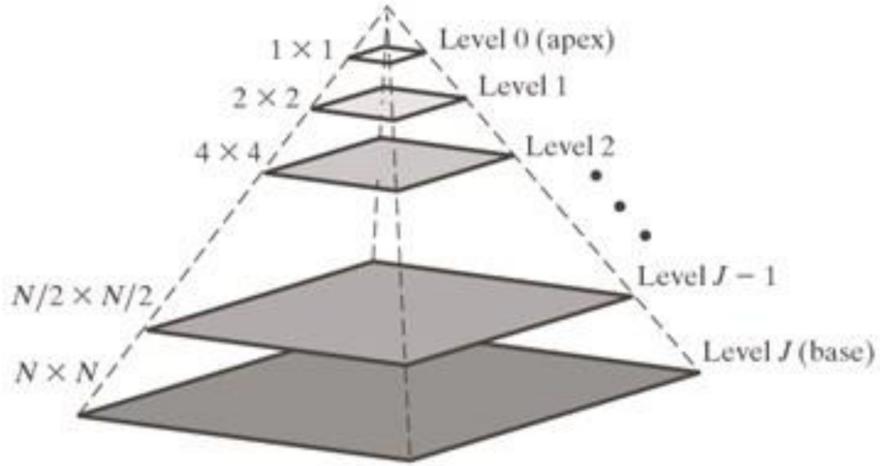
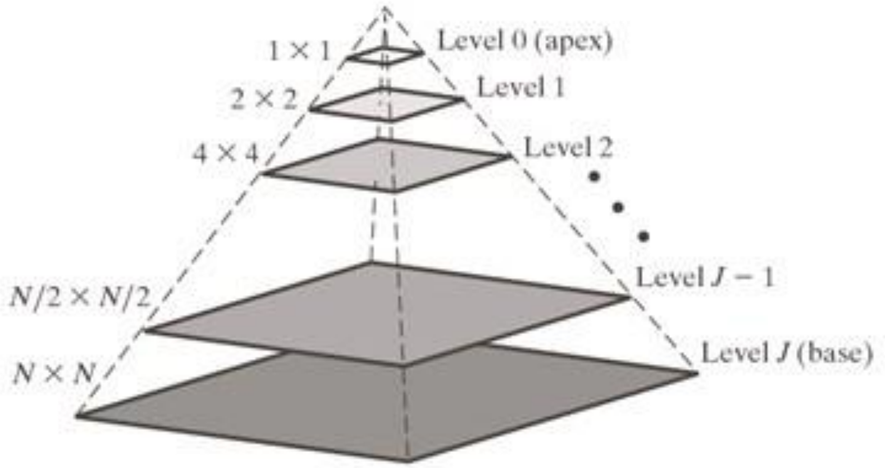


Image Pyramids(Contd..)

Prediction pyramid:

A prediction of each high resolution level is obtained by upsampling (inserting zeros) the previous low resolution level (prediction pyramid) and interpolation (filtering).

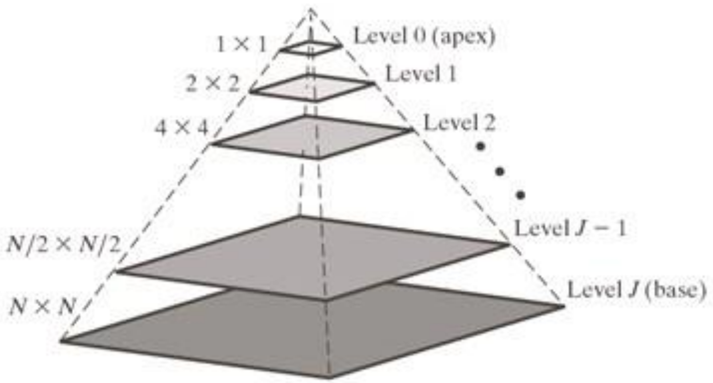
$$f_{2\uparrow}(n) = \begin{cases} f(n/2) & \text{if } n \text{ is even} \\ 0 & \text{otherwise} \end{cases}$$

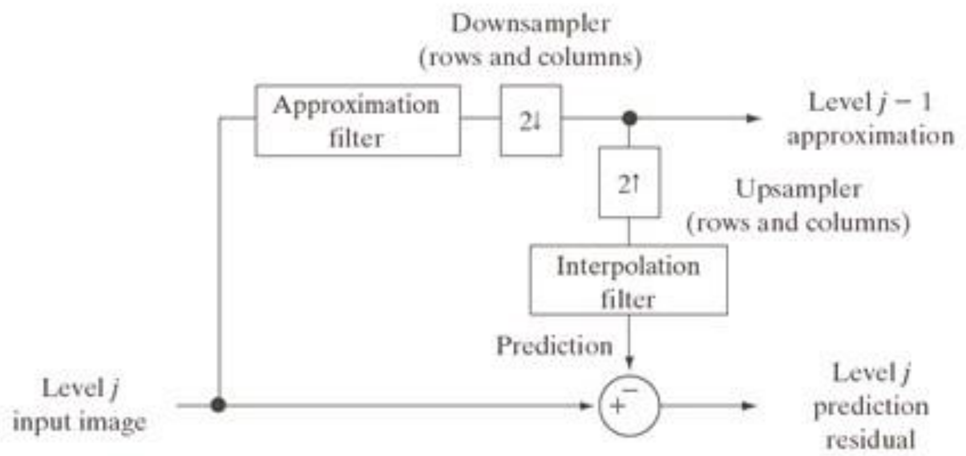


Prediction residual pyramid:

•At each resolution level, the prediction error is retained along with the lowest resolution level image.

•The original image may be reconstructed from this information.





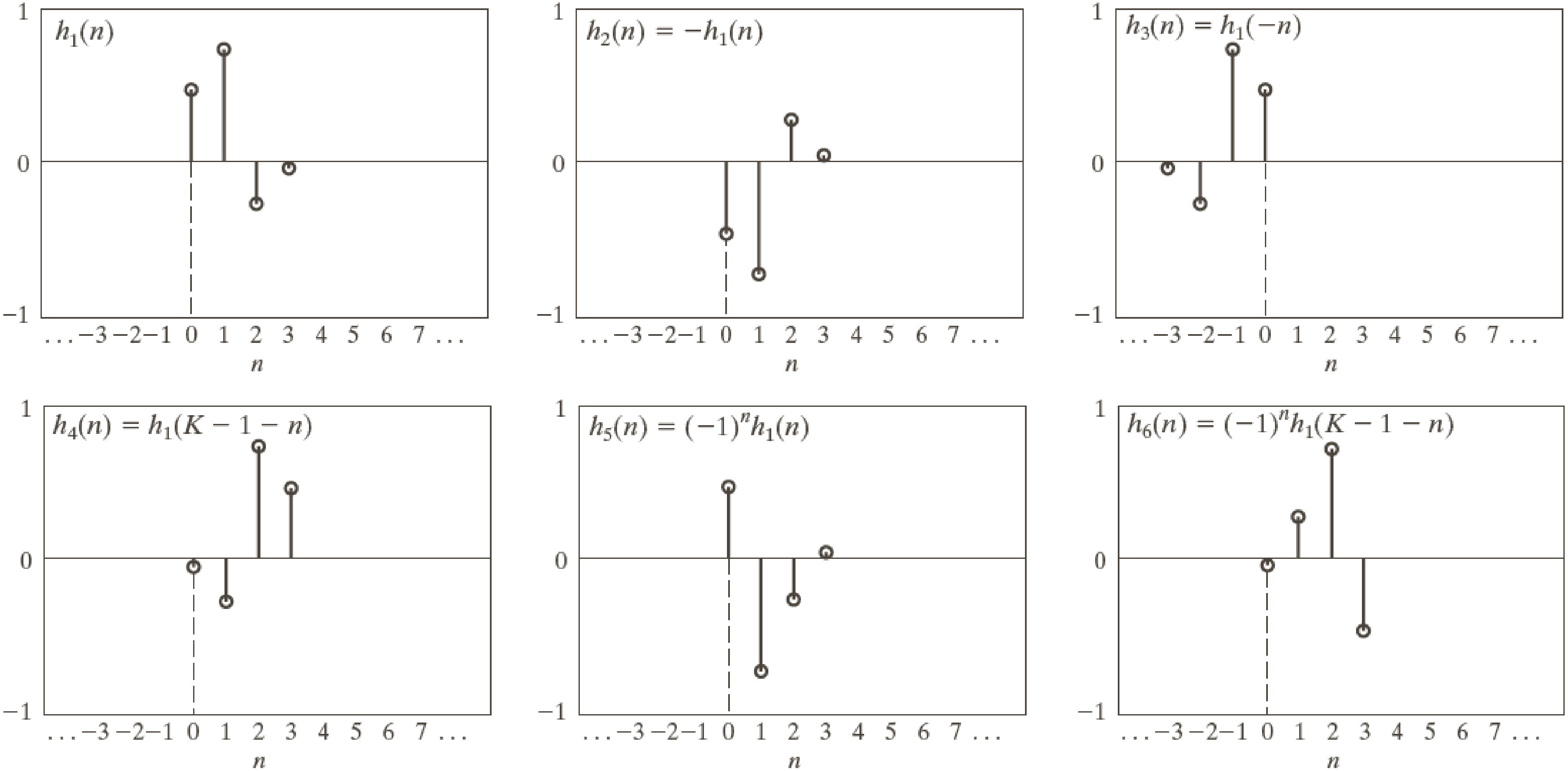
Subband Coding

•An image is decomposed to a set of bandlimited components (subbands).

•The decomposition is carried by filtering and downsampling.

•If the filters are properly selected the image may be reconstructed without error by filtering and upsampling.

Subband Coding(Contd..)

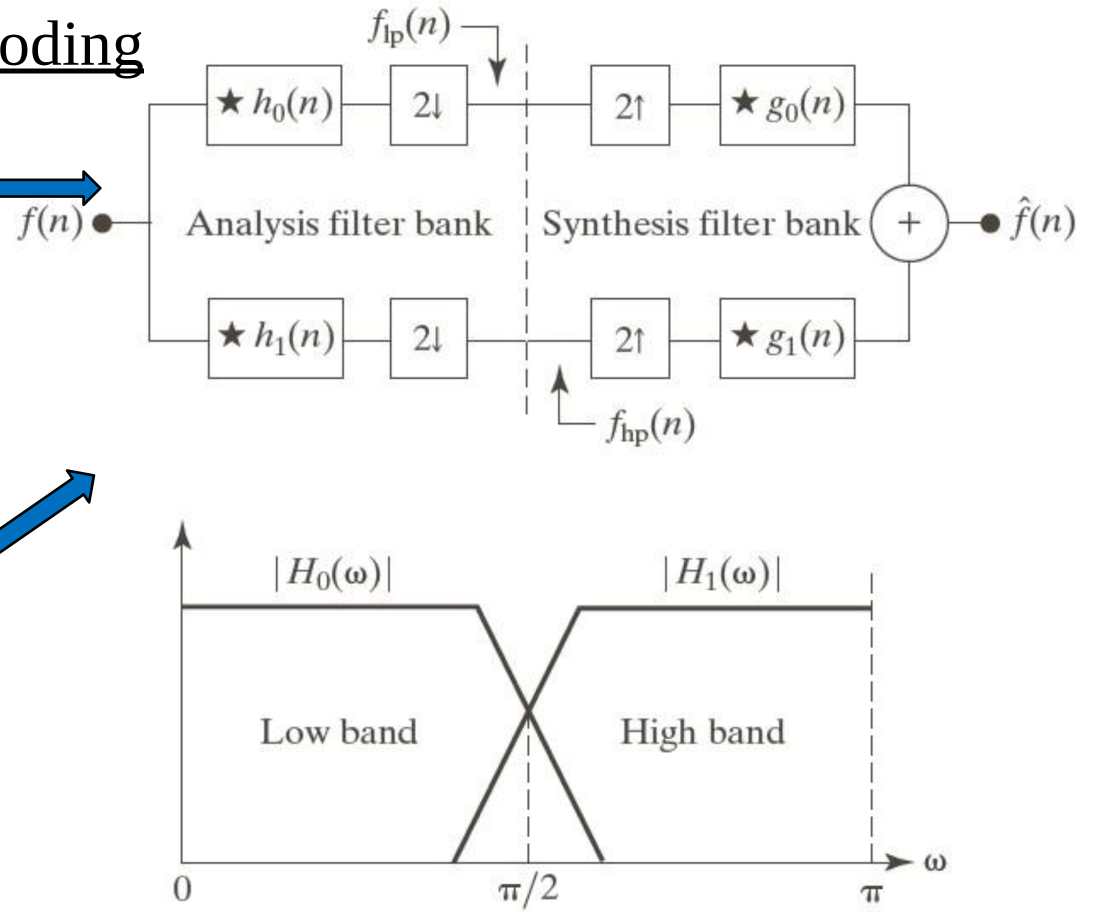


Subband Coding(Contd..)

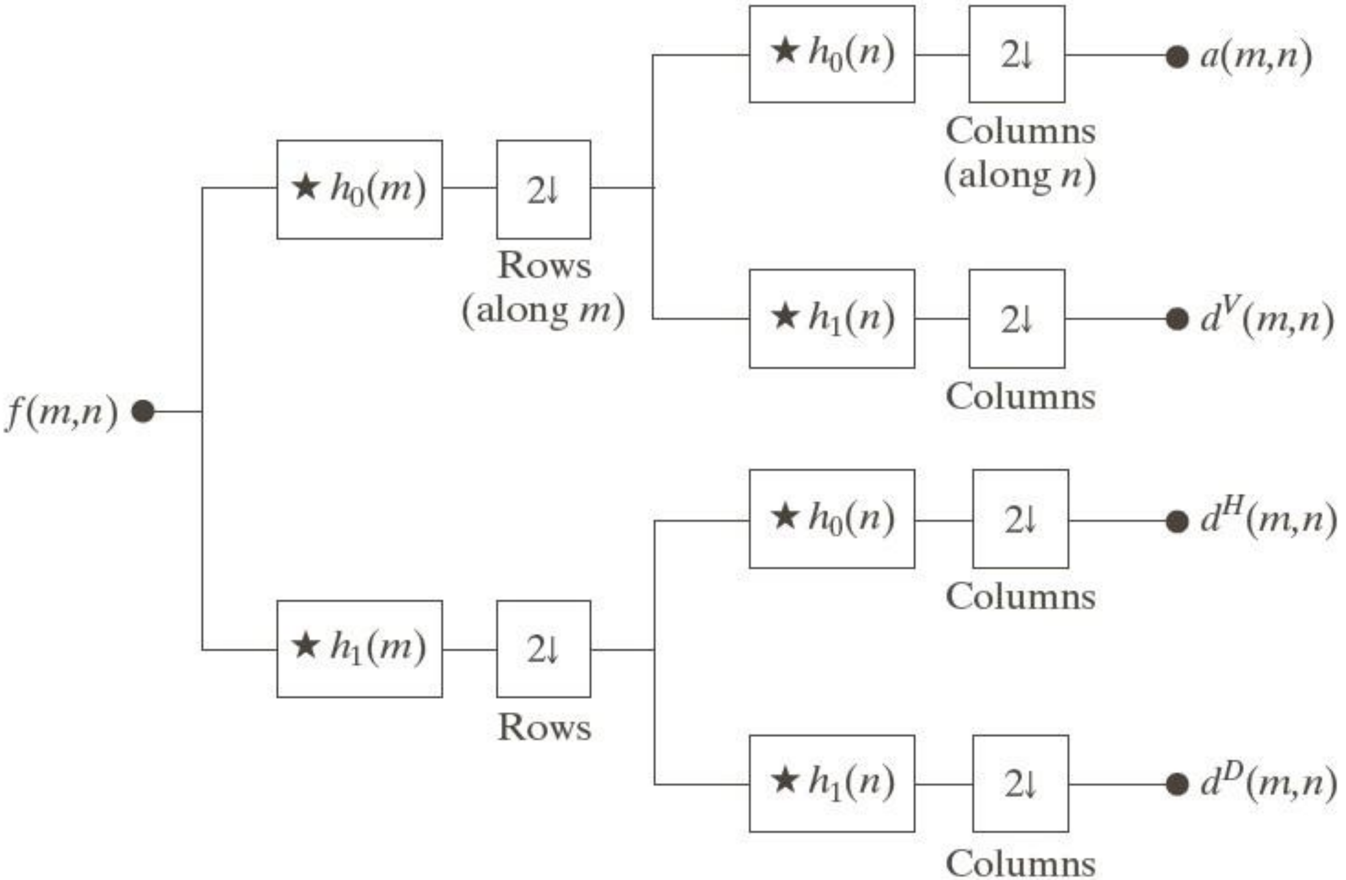
A two-band subband coding

Approximation filter (low pass)

Detail filter (high pass)



Subband Coding(Contd..)



Approximation subband

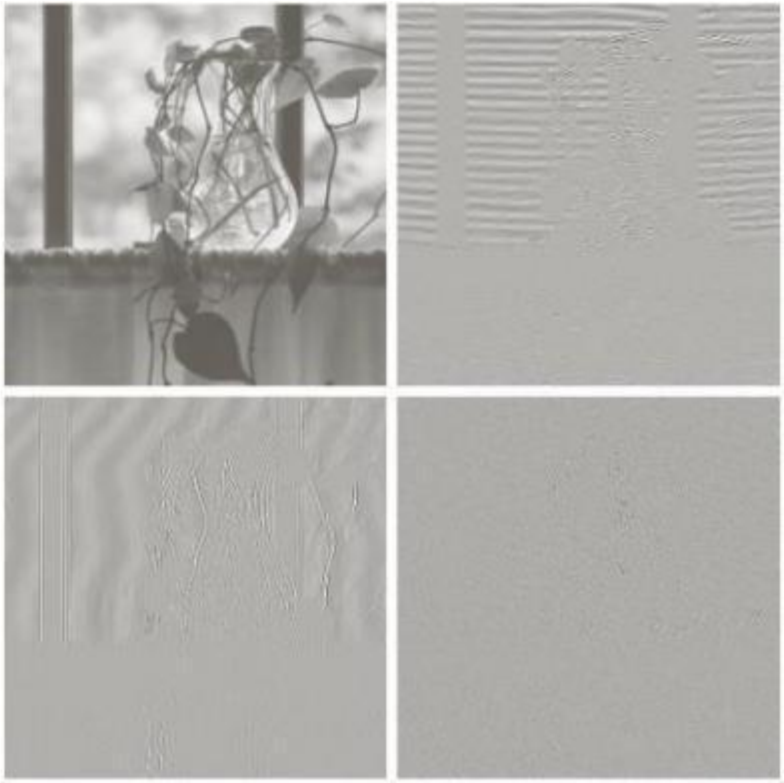
Vertical subband

Horizontal subband

Diagonal subband

Subband Coding(Contd..)

•The wavy lines are due to aliasing of the barely discernable window screen. Despite the aliasing, the image may be perfectly reconstructed.



| a | b |
| c | d |

FIGURE 7.9 A four-band split of the vase in Fig. 7.1 using the subband coding system of Fig. 7.7. The four subbands that result are the (a) approximation, (b) horizontal detail, (c) vertical detail, and (d) diagonal detail subbands.

The Haar Transform

•It is due to Alfred Haar [1910].
•Its basis functions are the simplest known orthonormal wavelets.
•The Haar transform is both separable and symmetric:

•$\mathbf{T}=\mathbf{HFH}$,

•**F** is a NxN image and **H** is the NxN transformation matrix and **T** is the NxN transformed image.
•Matrix **H** contains the Haar basis functions.

•The Haar basis functions $h_k(z)$ are defined for in $0\leq z \leq 1$, for $k=0,1,\ldots, N-1$, where $N=2^n$.

The Haar Transform(Contd..)

To generate **H**:

- we define the integer $k=2^p+q-1$, with $0\leq p \leq N-1$.
- if $p=0$, then $q=0$ or $q=1$.
- if $p\neq 0$, $1\leq q \leq 2^p$

For the above pairs of p and q, a value for k is determined and the Haar basis functions are computed.

$$h_0(z) = h_{00}(z) = \frac{1}{\sqrt{N}},\ z \in [0,1]$$

$$h_k(z) = h_{pq}(z) = \frac{1}{\sqrt{N}}, \begin{cases} 2^{p/2} & (q-1)/2^p \leq z \leq (q-0.5)/2^p \\ -2^{p/2} & (q-0.5)/2^p \leq z \leq q/2^p \\ 0 & \text{otherwise, } z \in [0,1] \end{cases}$$

The Haar Transform(Contd..)

The ith row of a NxN Haar transformation matrix contains the elements of $h_k(z)$ for z=0/N, 1/N, 2/N,…, (N-1)/N.

For instance, for N=4, p,q and k have the following values:

k	p	q
0	0	0
1	0	1
2	1	1
3	1	2

and the 4x4 transformation matrix is:

$$\mathbf{H}_4 = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

The Haar Transform(Contd..)

Similarly, for N=2, the 2x2 transformation matrix is:

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

•The rows of $\mathbf{H}_2$ are the simplest filters of length 2 that may be used as analysis filters $h_0(n)$ and $h_1(n)$ of a perfect reconstruction filter bank.

•Moreover, they can be used as scaling and wavelet vectors (defined in what follows) of the simplest and oldest wavelet transform.

Multi-resolution Expansions

- **Expansion of a signal $f(x)$:**

$$f(x)=\sum_k \alpha_k \phi_k(x)$$

α_k : real-valued expansion coefficients

$\phi_k(x)$: real-valued expansion functions

$$\alpha_k = \langle \tilde{\phi}_k(x), f(x) \rangle = \int \tilde{\phi}_k^*(x) f(x) dx$$

$\tilde{\phi}_k(x)$: the dual function of $\phi_k(x)$

- If $\{\phi_k(x)\}$ is an orthonormal basis for V, then $\phi_k(x) = \tilde{\phi}_k(x)$
- If the expansion is unique, the $\phi_k(x)$ are called basis functions.

- If $\{\phi_k(x)\}$ are not orthonormal but are an orthogonal basis for V,
then the basis funcitons and their duals are called biorthogonal.

$$\text{Biorthogonal: } \langle \phi_j(x), \tilde{\phi}_k(x) \rangle = \delta_{jk} = \begin{cases} 0 & , j \neq k \\ 1 & , j = k \end{cases}$$

Multi-resolution Expansions(Contd..)

- **Scaling function**

$$\phi_{j,k}(x) = 2^{j/2}\phi(2^j x - k), \quad \text{for } k \in \mathbf{Z} \text{ and } \phi(x) \in L^2(\mathbf{R})$$

- The subspace spanned over k for any j : $V_j = \underset{k}{span}\{\phi_{j,k}(x)\}$

- The scaling functions of any subspace can be built from double-resolution copies of themselves. That is,

$$\phi(x) = \sum_n h_\phi(n)\sqrt{2}\,\phi(2x - n)$$

where the coefficients are called scaling function coefficients.

Multi-resolution Expansions(Contd..)

Requirements of scaling function:

1. The scaling function is orthogonal to its integer translates.

2. The subspaces spanned by the scaling function at low scales are nested within those spanned at higher scales. That is

$$V_{-\infty} \subset \cdots \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_{\infty}$$

3. The only function that is common to all V_j is $f(x) = 0$. That is $V_{-\infty} = \{0\}$

4. Any function can be represented with arbitrary precision. That is,

$$V_{\infty} = \{L^2(\mathbf{R})\}$$

Wavelet function

•spans the difference between any two adjacent scaling subspaces V_j and V_{j+1}

$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$ for all $k \in \mathbf{Z}$ that spans the space W_j

where $W_j = \underset{k}{span}\{\psi_{j,k}(x)\}$

•The wavelet function can be expressed as a weighted sum of shifted, double-resolution scaling functions. That is,

$$\psi(x) = \sum_n h_\psi(n)\sqrt{2}\phi(2x - n)$$

where the $h_\psi(n)$ are called the wavelet function coefficients.

Wavelet series expansion

$$f(x)=\sum_{k} c_{j_0}(k)\phi_{j_0,k}(x)+\sum_{j=j_0}^{\infty}\sum_{k} d_j(k)\psi_{j,k}(x)$$

where j_0 is an arbitrary starting scale

$$c_{j_0}(k)=\left\langle f(x),\tilde{\phi}_{j_0,k}(x)\right\rangle=\int f(x)\tilde{\phi}_{j_0,k}(x)dx$$

called the approximation or scaling coefficients

$$d_j(k)=\left\langle f(x),\tilde{\psi}_{j,k}(x)\right\rangle=\int f(x)\tilde{\psi}_{j,k}(x)dx$$

called the detail or wavelet coefficients

Wavelet Transforms in One Dimension(Contd..)

Discrete Wavelet Transform

- The function *f*(*x*) is a sequence of numbers

$$f(x) = \frac{1}{\sqrt{M}} \sum_{k} W_{\phi}(j_0, k)\phi_{j_0,k}(x) + \frac{1}{\sqrt{M}} \sum_{j=j_0}^{\infty} \sum_{k} W_{\psi}(j,k)\psi_{j,k}(x)$$

where j_0 is an arbitrary starting scale

$$W_{\phi}(j_0,k) = \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x)\tilde{\phi}_{j_0,k}(x)$$

called the approximation or scaling coefficients

$$W_{\psi}(j,k) = \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x)\tilde{\psi}_{j,k}(x)$$

called the detail or wavelet coefficients

Fast Wavelet Transform

Fast Wavelet Transform (FWT)

- computationally efficient implementation of the DWT
- the relationship between the coefficients of the DWTat adjacent scales
- also called Mallat's herringbone algorithm
- resembles the twoband subband coding scheme

Fast Wavelet Transform(Contd..)

$$\phi(x) = \sum_n h_\phi(n)\sqrt{2}\phi(2x-n)$$

Scaling x by 2^j, translating it by k, and letting $m = 2k + n$

$$\phi(2^j x - k) = \sum_n h_\phi(n)\sqrt{2}\phi\left(2(2^j x - k) - n\right) = \sum_m h_\phi(m-2k)\sqrt{2}\phi\left(2^{j+1} x - m\right)$$

Similarity,

$$\psi(2^j x - k) = \sum_m h_\psi(m-2k)\sqrt{2}\phi(2^{j+1} x - m)$$

Consider the DWT. Assume $\tilde{\phi}(x) = \phi(x)$ and $\tilde{\psi}(x) = \psi(x)$

$$W_\phi(j_0, k) = \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x)\phi_{j_0,k}(x)$$

$$\phi(x) = \sum_n h_\phi(n)\sqrt{2}\phi(2x-n)$$

$$= \frac{1}{\sqrt{M}} \sum_m h_\psi(m-2k)\sqrt{2}\phi(2^{j_0} x - m)$$

Fast Wavelet Transform(Contd..)

$$W_\psi(j,k) = \frac{1}{\sqrt{M}} \sum_{x=0}^{M-1} f(x)\psi_{j,k}(x)$$

$$\psi_{j,k}(x) = 2^{j/2}\psi(2^j x - k)$$

$$= \frac{1}{\sqrt{M}} \sum_x f(x) 2^{j/2} \psi(2^j x - k)$$

$$= \frac{1}{\sqrt{M}} \sum_x f(x) 2^{j/2} \left[\sum_m h_\psi(m-2k)\sqrt{2}\phi(2^{j+1}x - m) \right]$$

$$= \sum_m h_\psi(m-2k) \left[\frac{1}{\sqrt{M}} \sum_x f(x) 2^{(j+1)/2} \phi(2^{j+1}x - m) \right]$$

$$= \sum_m h_\psi(m-2k) W_\phi(j+1, m)$$

Similarity,

$$W_\phi(j,k) = \sum_m h_\phi(m-2k) W_\phi(j+1, m)$$

Fast Wavelet Transform(Contd..)

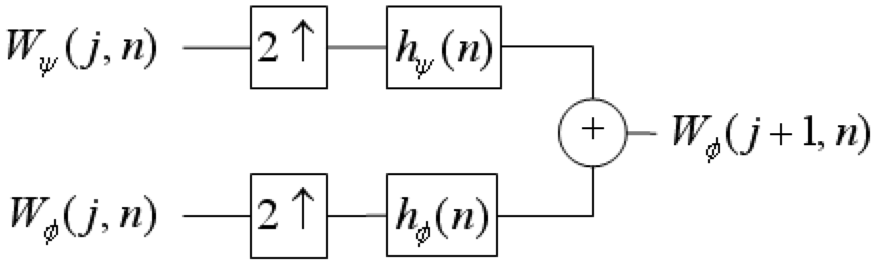


Figure: An FWT^{-1} synthesis filter bank.

•By subband coding theorem, perfect reconstrucion for two-band orthonormal filters requires $g_i(n) = h_i(-n)$ for $i = \{0, 1\}$.
That is, the synthesis and analysis filters must be time-reversed versions of one another. Since the FWT analysis filter are $h_0(n) = h_\phi(-n)$ and $h_1(n) = h_\psi(-n)$, the required FWT^{-1} synthesis filters are

$$g_0(n) = h_0(-n) = h_\phi(n) \qquad g_1(n) = h_1(-n) = h_\psi(n)$$

Wavelet Trasnforms in Two Dimensions

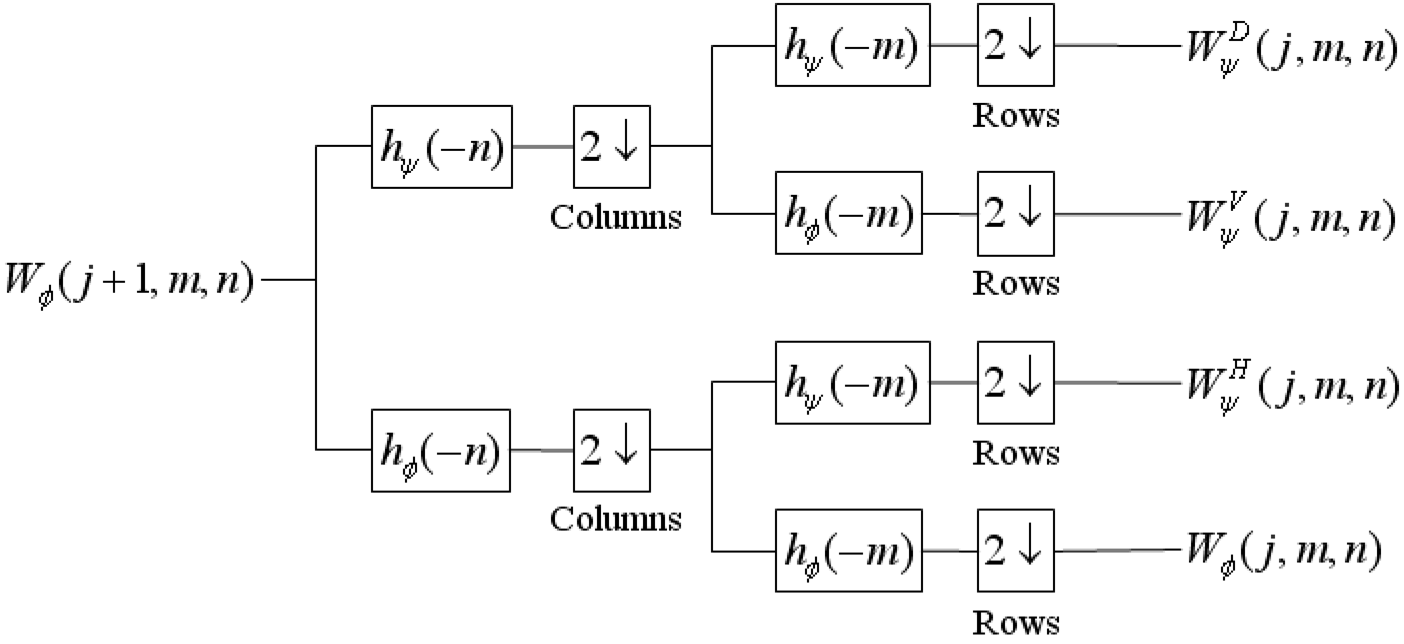
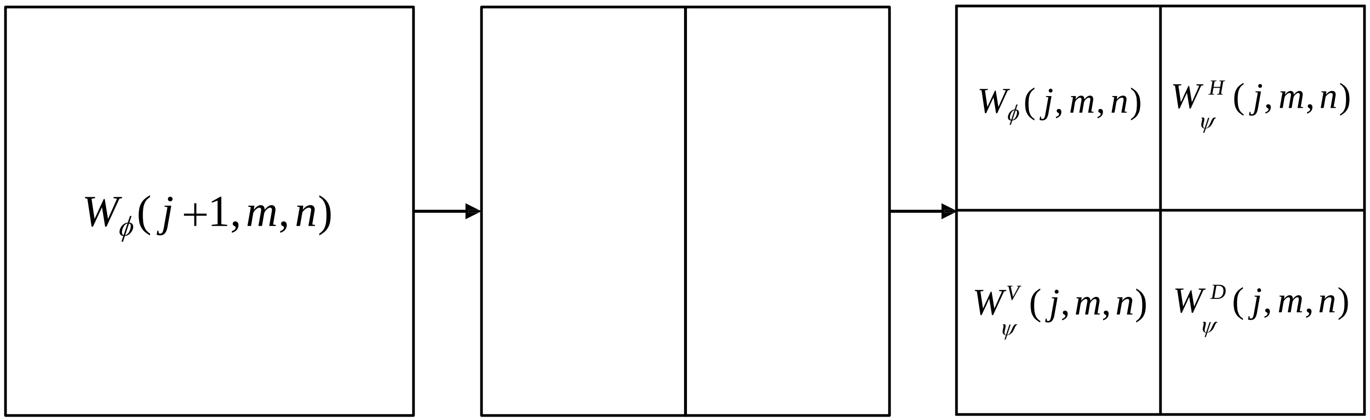


Figure : The two-dimensional FWT — the analysis filter.

Wavelet Trasnforms in Two Dimensions(Contd..)



two-dimensional decomposition

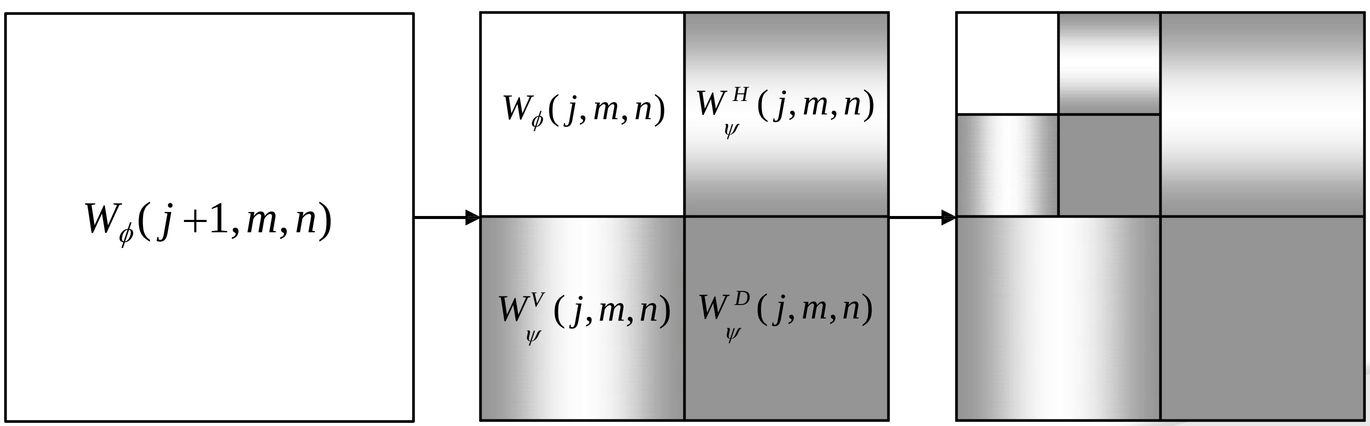
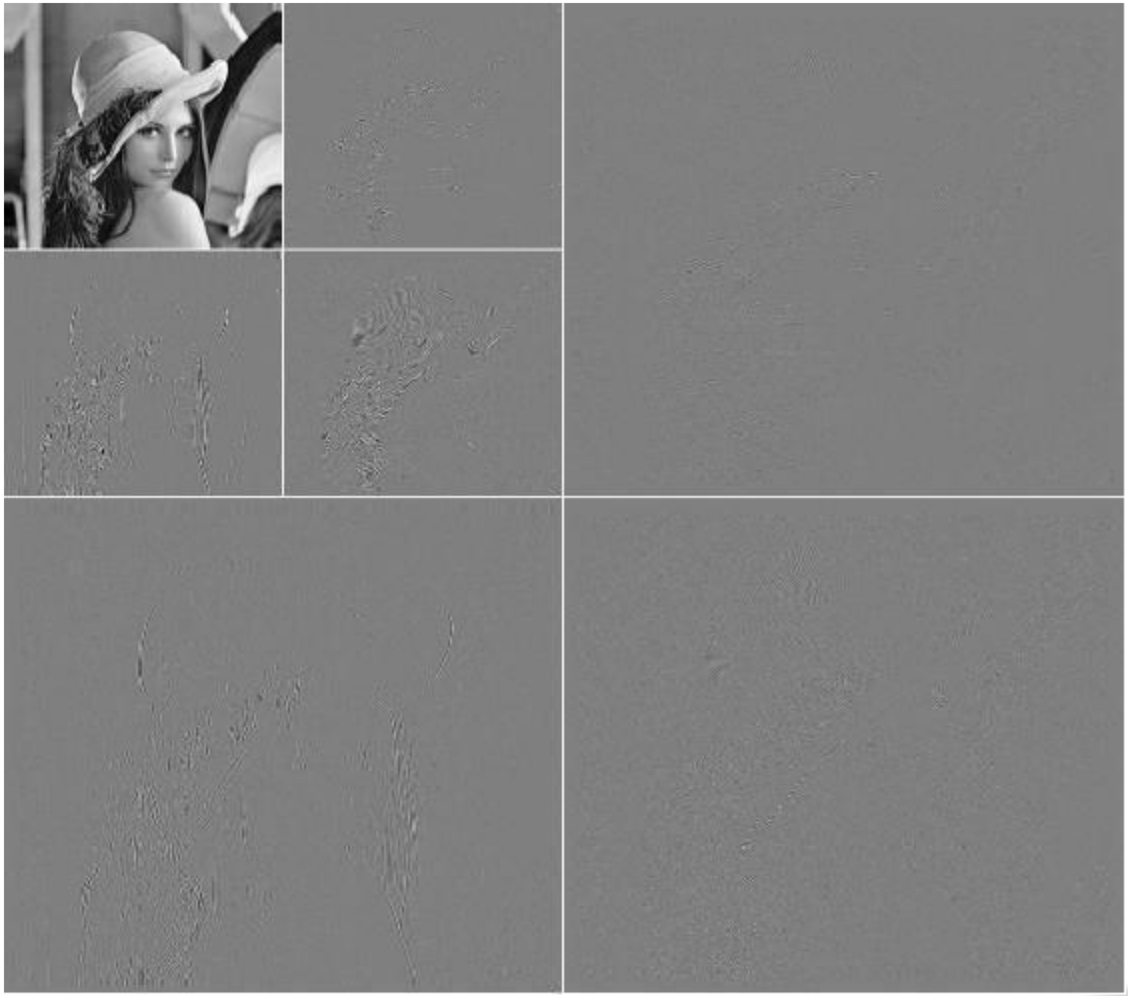


Figure : Two-scale of two-dimensional decomposition

Wavelet Trasnforms in Two Dimensions(Contd..)



Wavelet Trasnforms in Two Dimensions(Contd..)

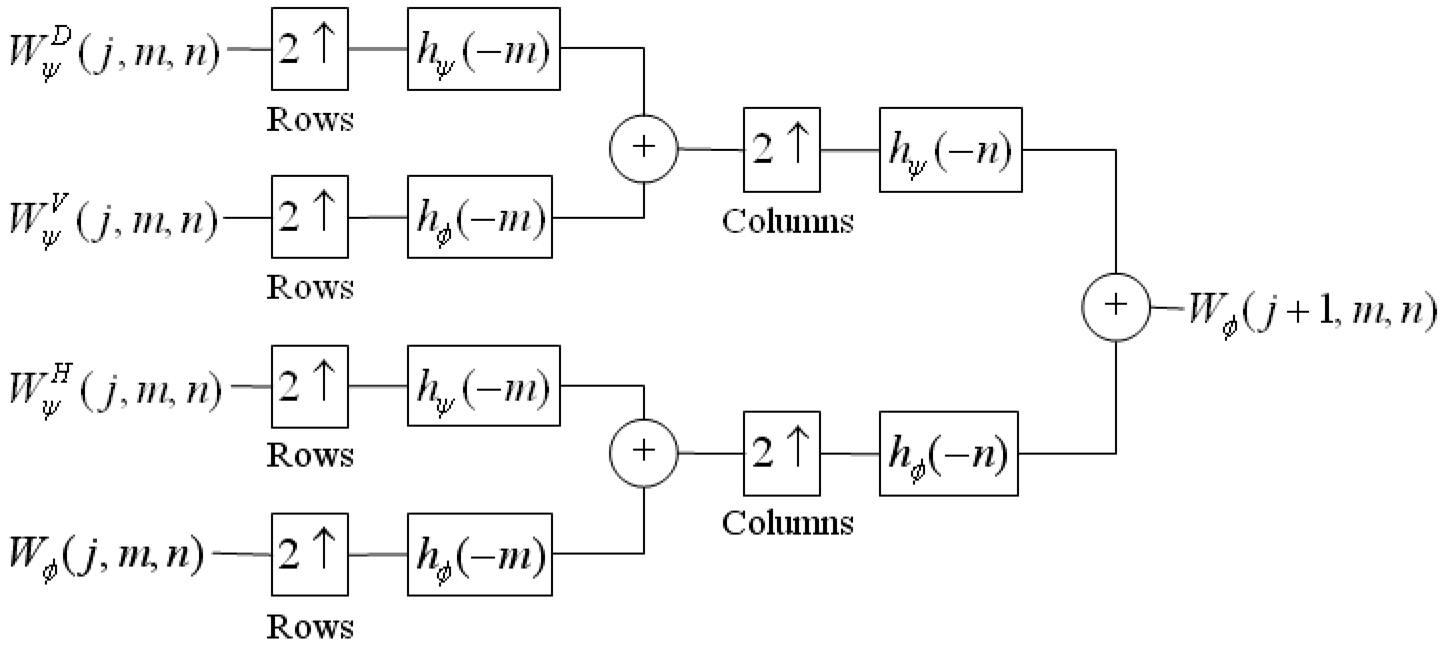
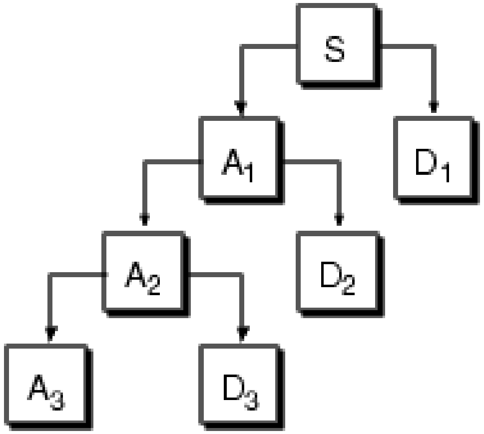


Figure : The two-dimensional FWT — the synthesis filter bank.

Wavelet Packets

- Generalization of wavelet decomposition
- Very useful for signal analysis

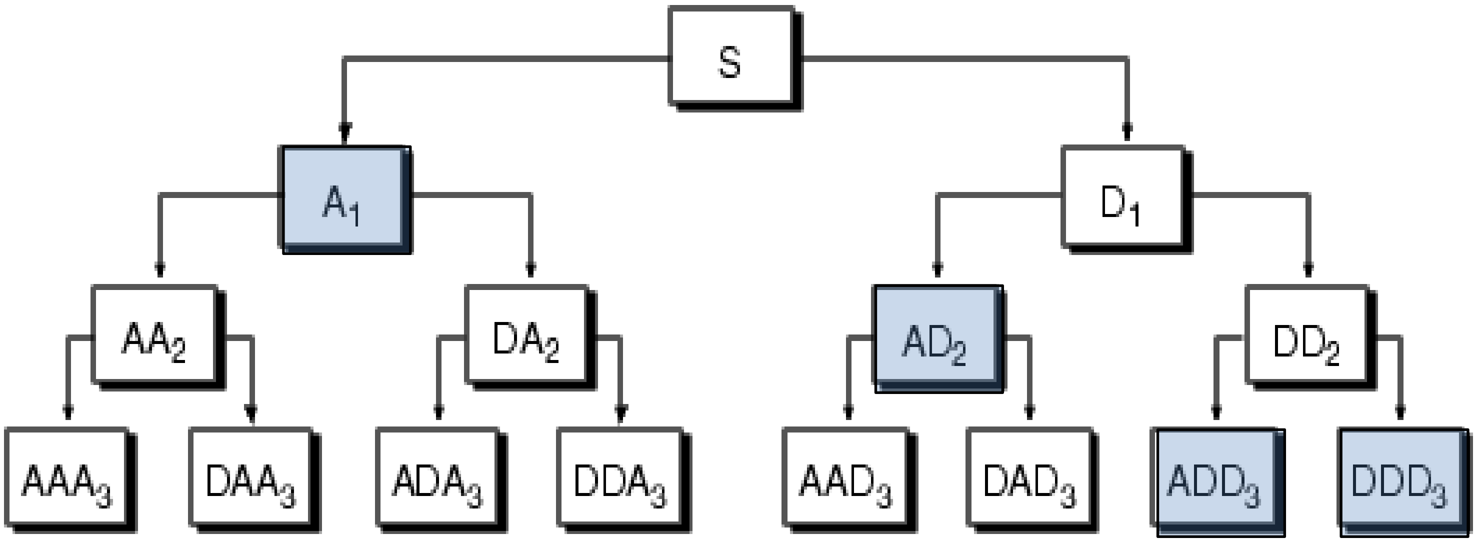


$$S = A_1 + D_1$$
$$= A_2 + D_2 + D_1$$
$$= A_3 + D_3 + D_2 + D_1$$

Wavelet analysis: n+1 (at level n) different ways to reconstuct S

Wavelet Packets(Contd..)

- We have a complete tree

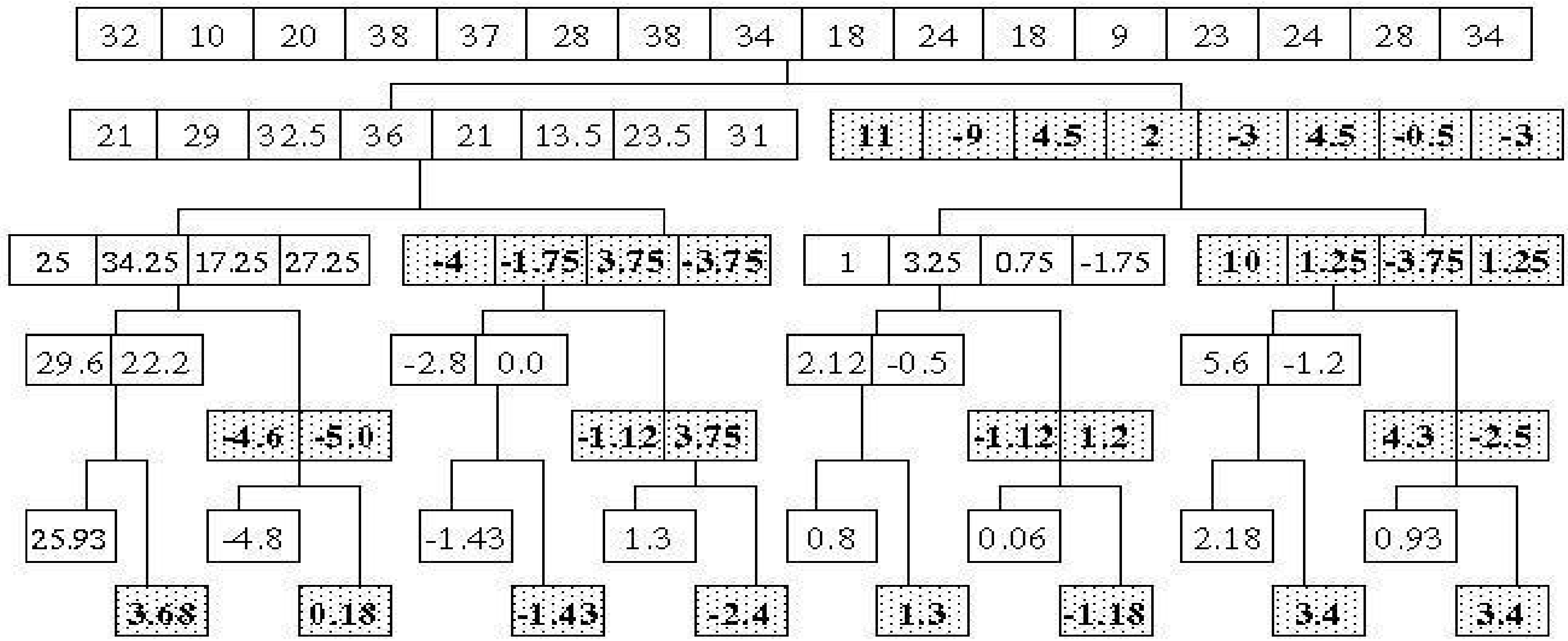


Wavelet packets: a lot of new possibilities to reconstruct S:

i.e. $S=A_1+AD_2+ADD_3+DDD_3$

Wavelet Packets(Contd..)

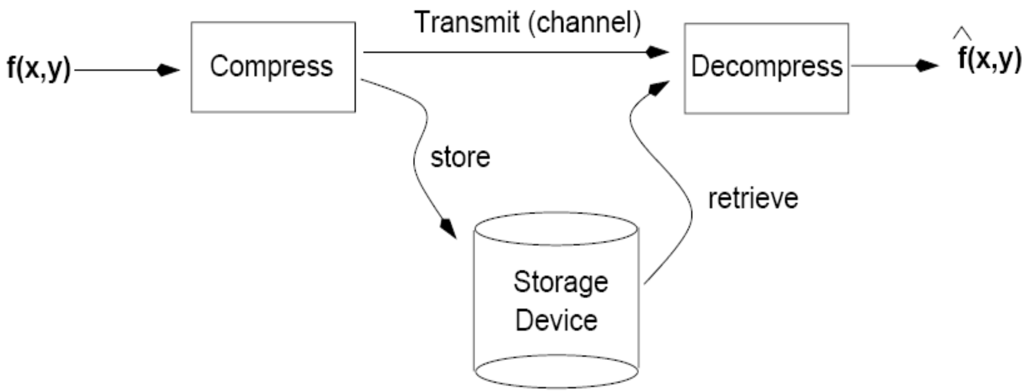
Wavelet Packet Transform example (Haar)



Wavelet Packet Tree

Image Compression

•The goal of image compression is to reduce the amount of data required to represent a digital image.



Types of Image Compression

Lossless

- Information preserving
- Low compression ratios

Lossy

- Not information preserving
- High compression ratios

Image Compression(Contd..)

Types of Data Redundancy

(1) Coding Redundancy
(2) Interpixel Redundancy
(3) Psychovisual Redundancy

•Data compression attempts to reduce one or more of these redundancy types.

Image Compression(Contd..)

1.Coding Redundancy

Case 1: $l(r_k)$ = constant length

Example:

r_k	$p_r(r_k)$	*Code 1*	$l_1(r_k)$
$r_0 = 0$	0.19	000	3
$r_1 = 1/7$	0.25	001	3
$r_2 = 2/7$	0.21	010	3
$r_3 = 3/7$	0.16	011	3
$r_4 = 4/7$	0.08	100	3
$r_5 = 5/7$	0.06	101	3
$r_6 = 6/7$	0.03	110	3
$r_7 = 1$	0.02	111	3

Assume an image with $L = 8$

Assume $l(r_k) = 3$, $L_{avg} = \sum_{k=0}^{7} 3P(r_k) = 3\sum_{k=0}^{7} P(r_k) = 3$ bits

Total number of bits: $3NM$

Image Compression(Contd..)

Case 2: $l(r_k)$ = variable length

Table 6.1 Variable-Length Coding Example

variable length

r_k	$p_r(r_k)$	Code 1	$l_1(r_k)$	Code 2	$l_2(r_k)$
$r_0 = 0$	0.19	000	3	11	2
$r_1 = 1/7$	0.25	001	3	01	2
$r_2 = 2/7$	0.21	010	3	10	2
$r_3 = 3/7$	0.16	011	3	001	3
$r_4 = 4/7$	0.08	100	3	0001	4
$r_5 = 5/7$	0.06	101	3	00001	5
$r_6 = 6/7$	0.03	110	3	000001	6
$r_7 = 1$	0.02	111	3	000000	6

$$L_{avg} = \sum_{k=0}^{7} l(r_k)P(r_k) = 2.7 \text{ bits}$$

$$C_R = \frac{3}{2.7} = 1.11 \text{ (about 10\%)}$$

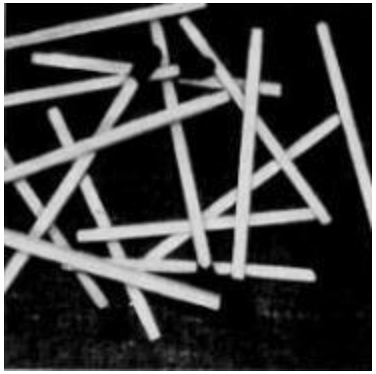
Total number of bits: 2.7NM

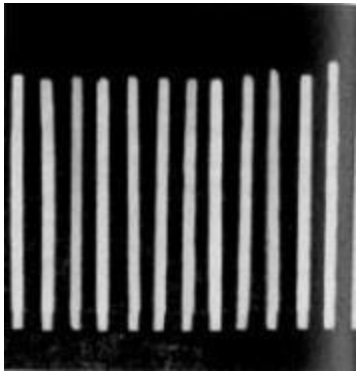
$$R_D = 1 - \frac{1}{1.11} = 0.099$$

Image Compression(Contd..)

2.Interpixel redundancy

•Interpixel redundancy implies that pixel values are correlated (i.e., a pixel value can be reasonably predicted by its neighbors).





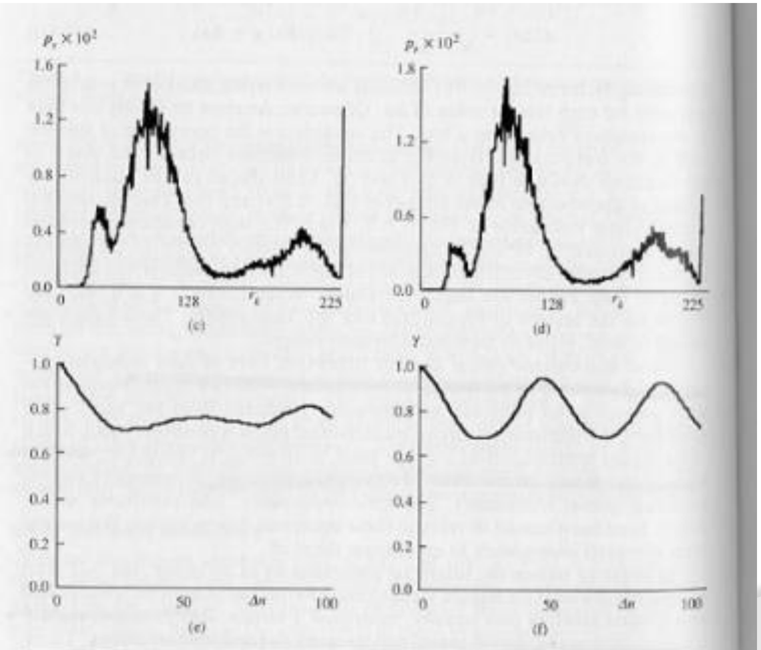


Figure 6.2 Two images and their gray-level histograms and normalized autocorrelation

$$f(x) \circ g(x) = \int_{-\infty}^{\infty} f(x)g(x+a)da$$

auto-correlation: f(x)=g(x)

Image Compression(Contd..)

3.Psychovisual redundancy

- The human eye is more sensitive to the lower frequencies than to the higher frequencies in the visual spectrum.
- <u>Idea:</u> discard data that is perceptually insignificant!

Example: quantization

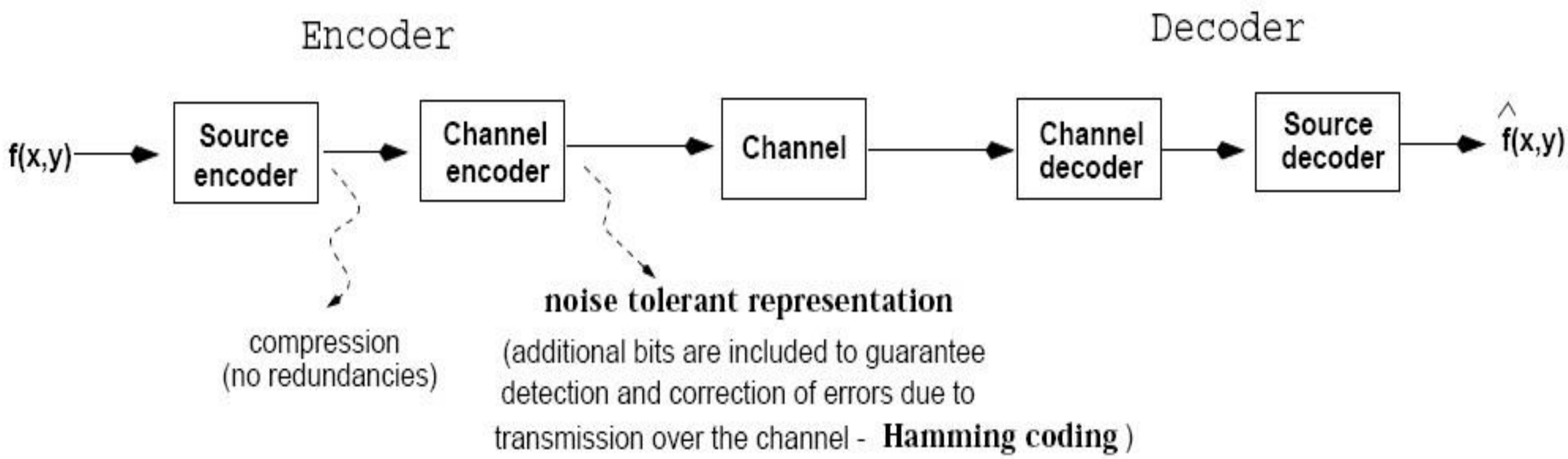
256 gray levels

16 gray levels

16 gray levels + random noise

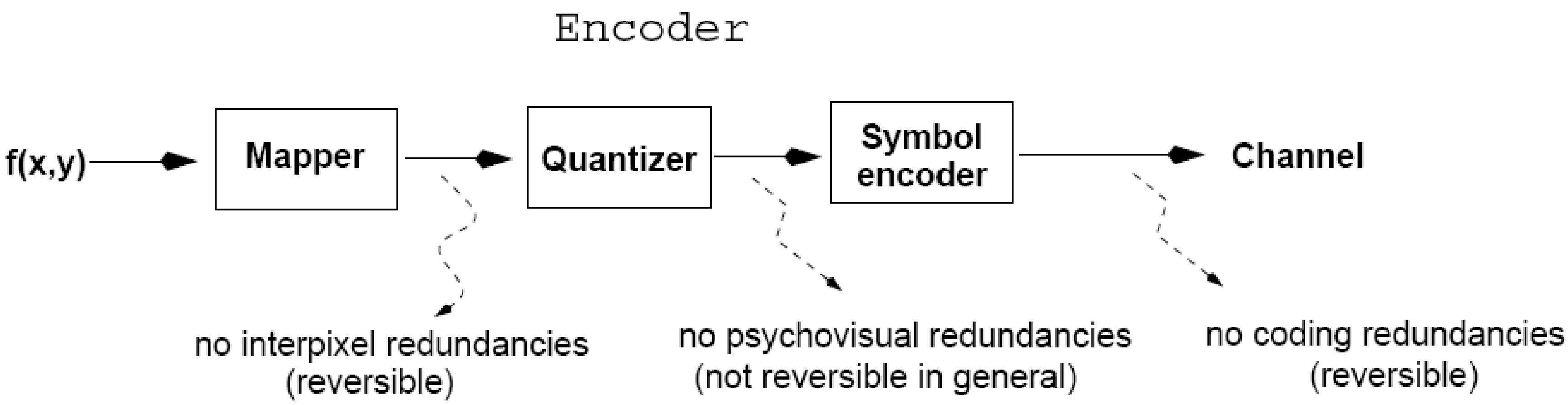
C=8/4 = 2:1

Image Compression Model



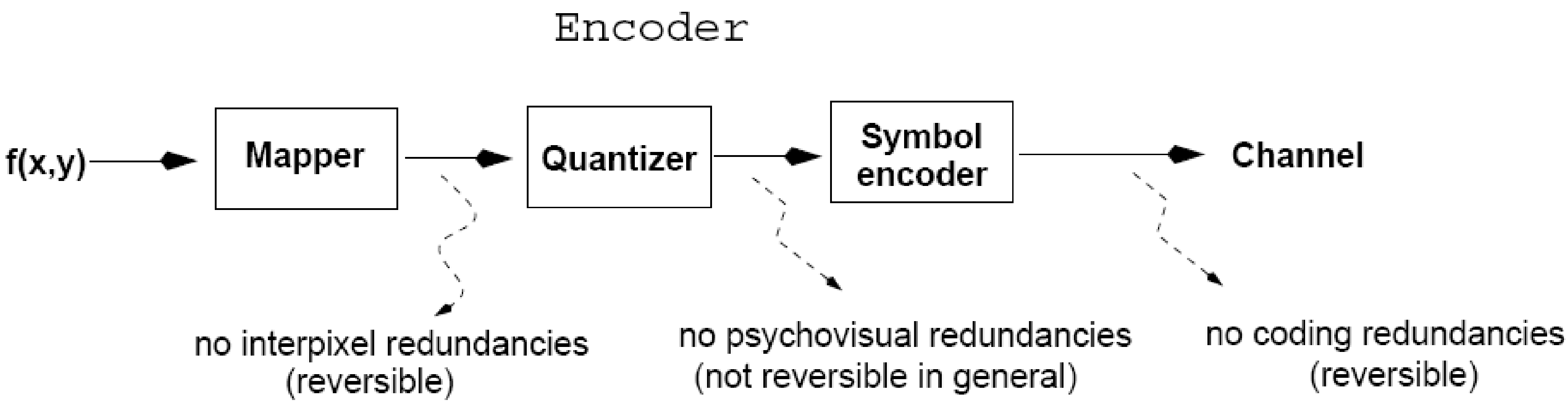
We will focus on the Source Encoder/Decoder only

Image Compression Model(Contd..)



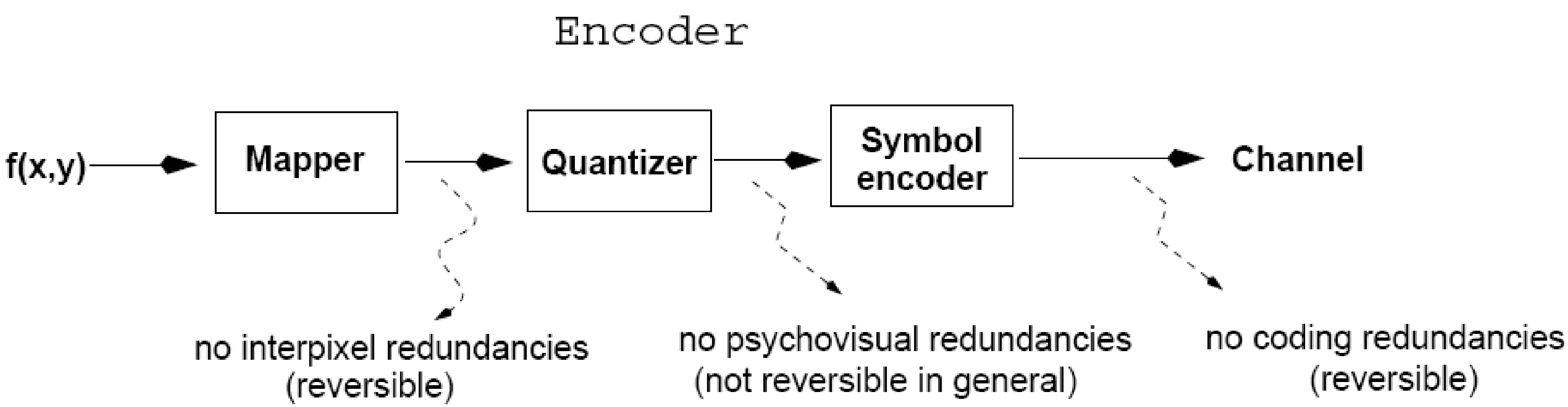
Mapper: transforms data to account for interpixel redundancies

Image Compression Model(Contd..)



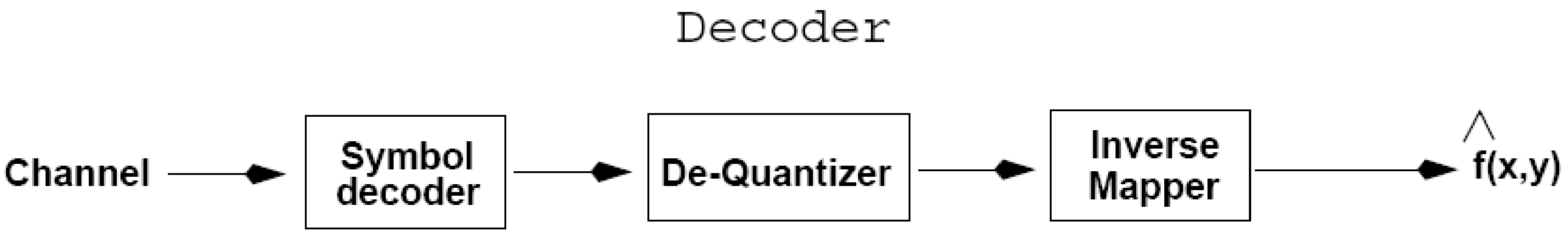
Quantizer: quantizes the data to account for psychovisual redundancies.

Image Compression Model(Contd..)



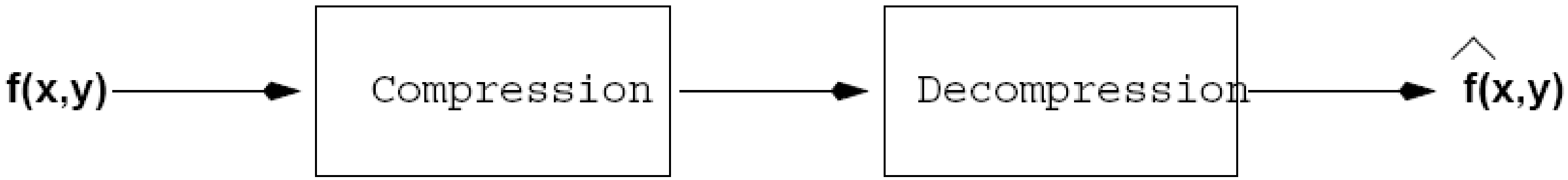
Symbol encoder: encodes the data to account for coding redundancies.

Image Compression Model(Contd..)



- The decoder applies the inverse steps.
- Note that quantization is irreversible in general.

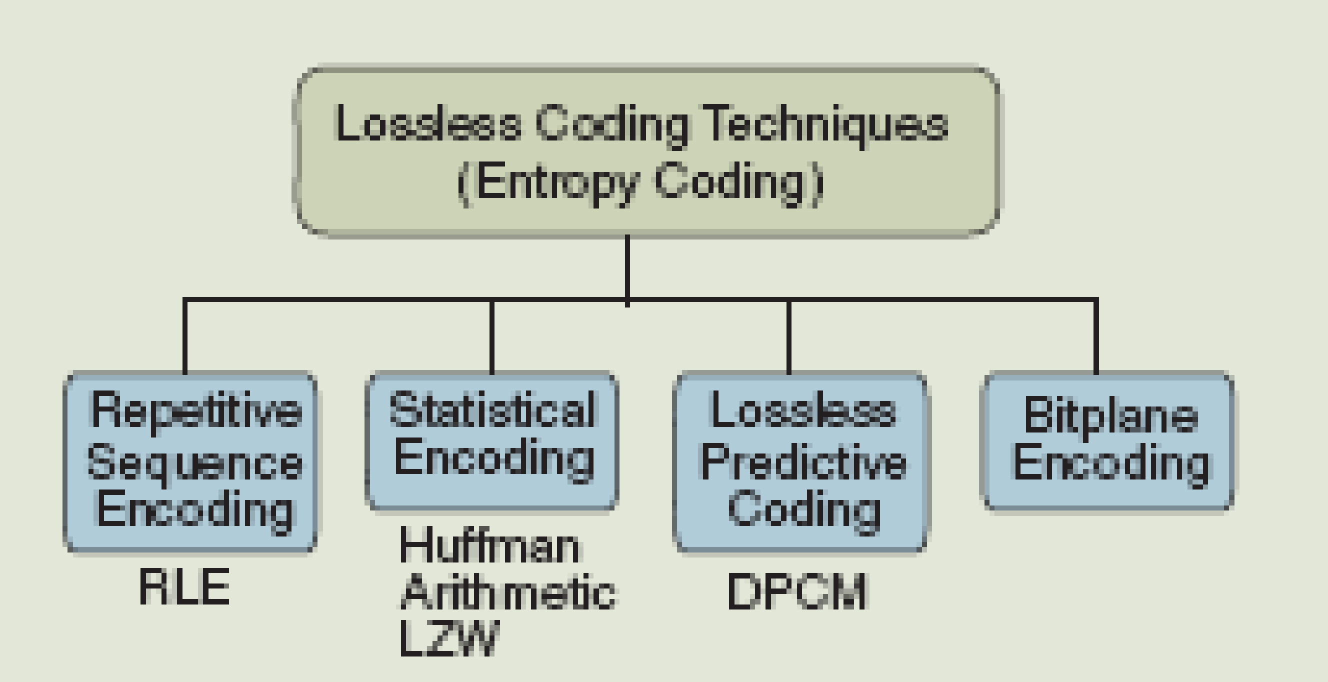
Error-free (Lossless)Compression



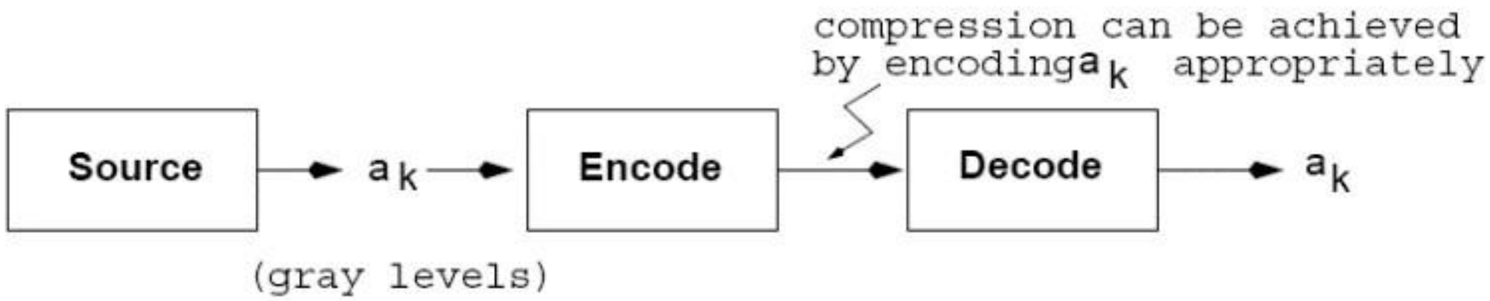
$$e(x,y) = \hat{f}(x,y) - f(x,y) = 0$$

Error-free (Lossless)Compression(Contd..)

Taxonomy of Lossless Methods



Huffman Coding (addresses coding redundancy)



- A variable-length coding technique.
- Source symbols are encoded one at a time!
    - There is a one-to-one correspondence between source symbols and code words.
- Optimal code - minimizes code word length per source symbol.

Error-free (Lossless)Compression(Contd..)

<u>LZW Coding (addresses interpixel redundancy</u>

- Requires <u>no prior knowledge</u> of symbol probabilities.

- Assigns fixed length code words to variable length symbol sequences.
  - There is no one-to-one correspondence between source symbols and code words.

- Included in GIF, TIFF and PDF file formats

Error-free (Lossless)Compression(Contd..)

LZW Coding (addresses interpixel redundancy

•A codebook (or dictionary) needs to be constructed.

•Initially, the first 256 entries of the dictionary are assigned to the gray levels 0,1,2,..,255 (i.e., assuming 8 bits/pixel)

Consider a 4x4, 8 bit image

39	39	126	126
39	39	126	126
39	39	126	126
39	39	126	126

Initial Dictionary

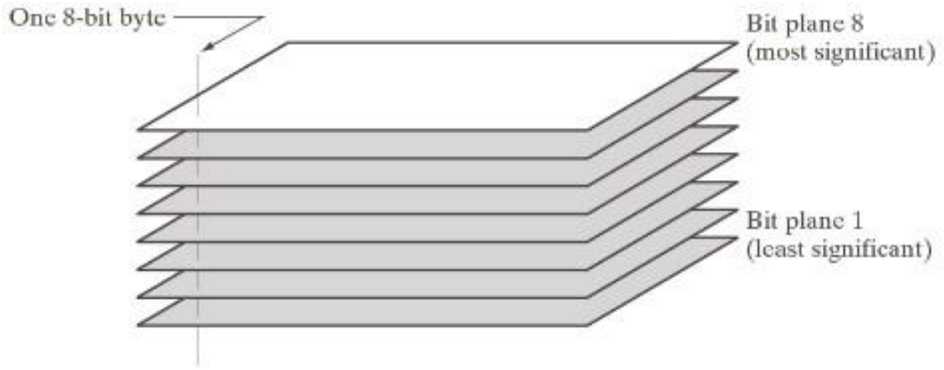
Dictionary Location	Entry
0	0
1	1
.	.
255	255
256	-
511	-

Error-free (Lossless)Compression(Contd..)

Bit-plane coding (addresses interpixel redundancy)

Process each bit plane individually.

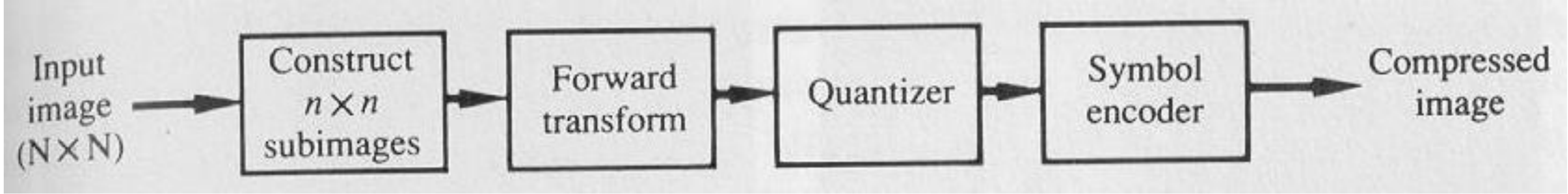
(1) Decompose an image into a series of binary images.
(2) Compress each binary image (e.g., using run-length coding)

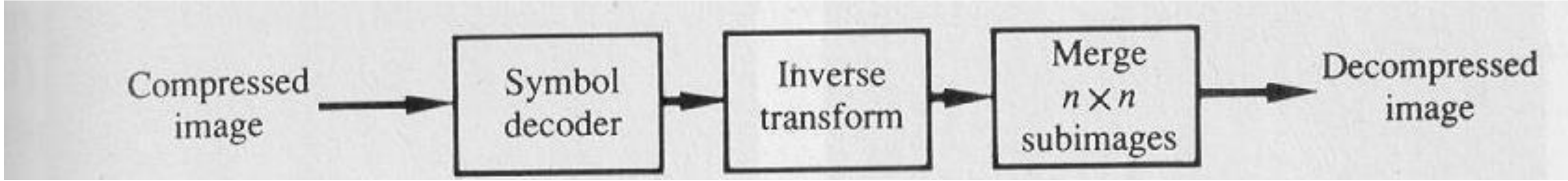


Lossy Compression

•Transform the image into some other domain to reduce interpixel redundancy.

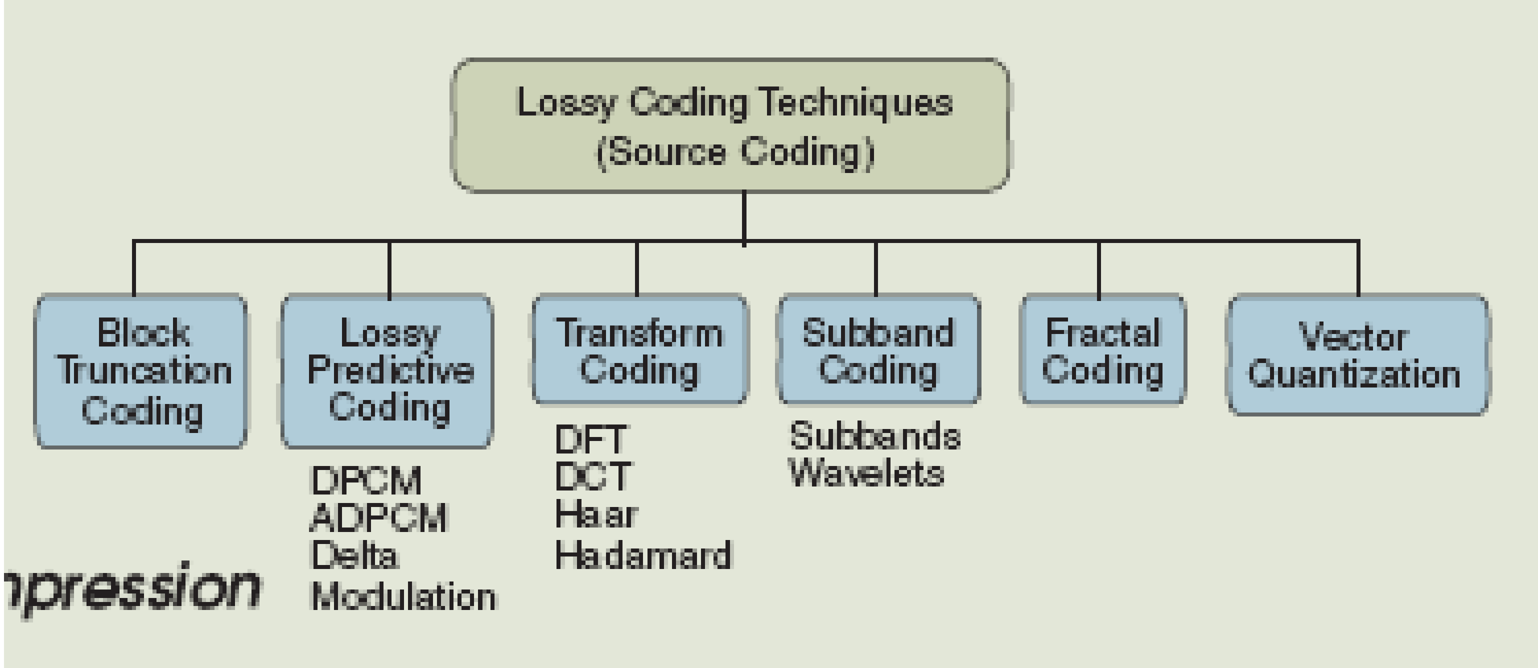
~ $(N/n)^2$ subimages





Lossy Compression(Contd..)

Lossy Methods - Taxonomy



UNIT – V

MORPHOLOGICAL IMAGE PROCESSING

Course Outcomes cont.

The course will enable the students to:	
CLO 18	Understand the basic multi-resolution techniques and segmentation methods
CLO 19	Explore on lossy/lossless compression models using wavelets
CLO 20	Use morphological operations like dilation and erosion to represent and describe regions, boundaries etc. in identification of the components in images.

Introduction

• **Morphology**: a branch of biology that deals with the form and structure of animals and plants

• Morphological image processing is used to extract image components for representation and description of region shape, such as boundaries, skeletons, and the convex hull

Preliminaries

- **Reflection**

The reflection of a set B, denoted B, is defined as

$$B = \{w \mid w = -b, \text{for } b \in B\}$$

- **Translation**

The translation of a set B by point $z = (z_1, z_2)$, denoted $(B)_Z$, is defined as

$$(B)_Z = \{c \mid c = b + z, \text{for } b \in B\}$$

Preliminaries(Contd..)

•**Structure elements (SE)**

Small sets or sub-images used to probe an image under study for properties of interest

Examples

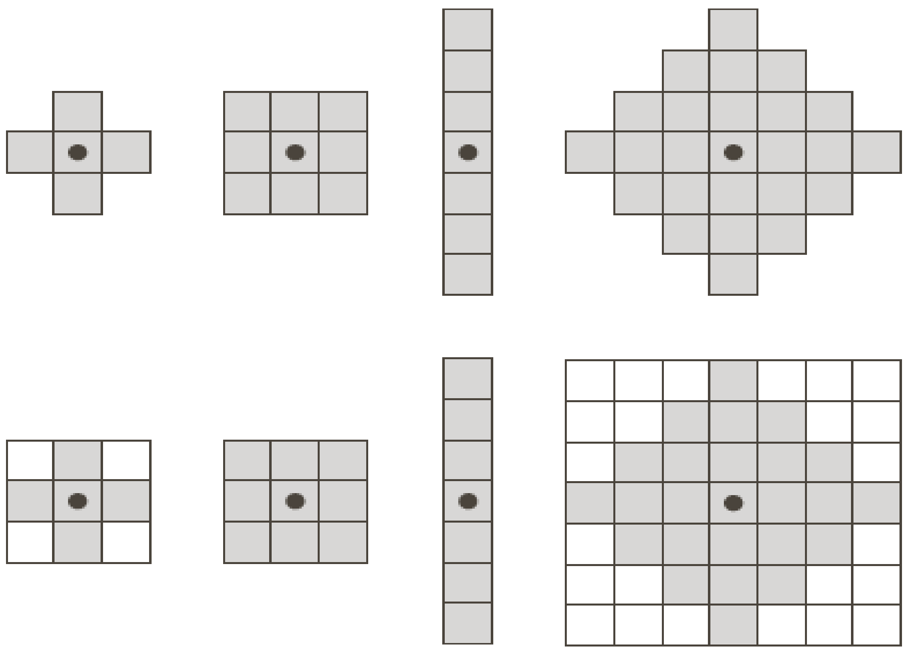
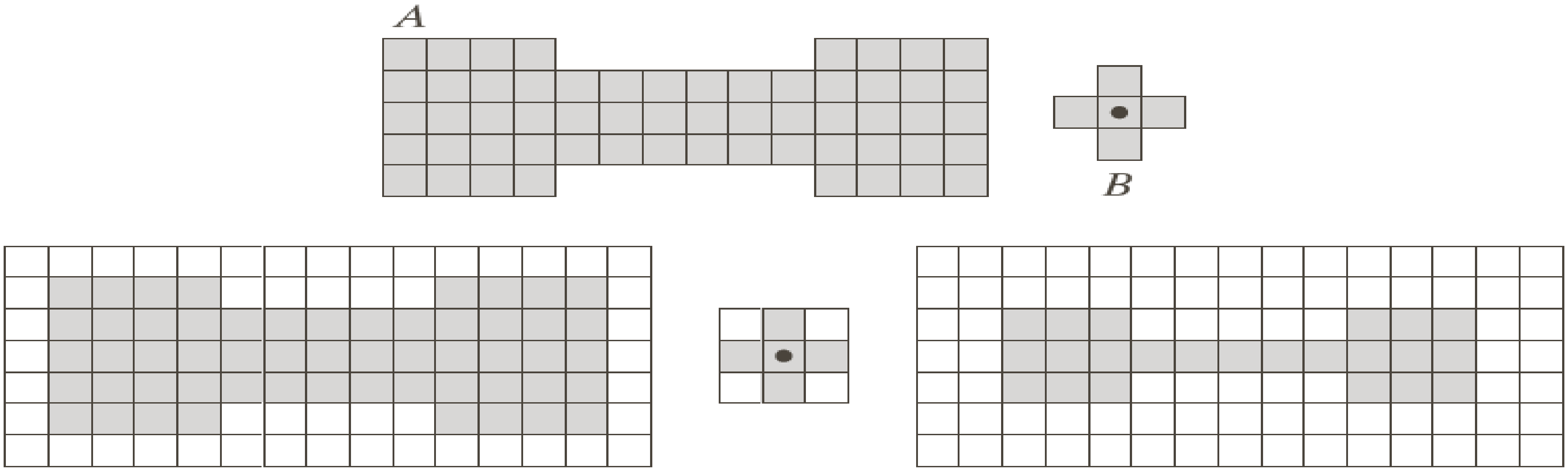


FIGURE 9.2 First row: Examples of structuring elements. Second row: Structuring elements converted to rectangular arrays. The dots denote the centers of the SEs.

Preliminaries(Contd..)



a b
c d e

FIGURE 9.3 (a) A set (each shaded square is a member of the set). (b) A structuring element. (c) The set padded with background elements to form a rectangular array and provide a background border. (d) Structuring element as a rectangular array. (e) Set processed by the structuring element.

Dilation

With A and B as sets in Z^2, the dilation of A by B, denoted $A \oplus B$, is defined as

$$A \oplus B = \left\{ z \mid \left(\hat{B} \right)_z \cap A \neq \varnothing \right\}$$

The set of all displacements z, the translated B and A overlap by at least one element.

$$A \oplus B = \left\{ z \mid \left[\left(\hat{B} \right)_z \cap A \right] \subseteq A \right\}$$

Dilation(Contd..)

Examples of Dilation

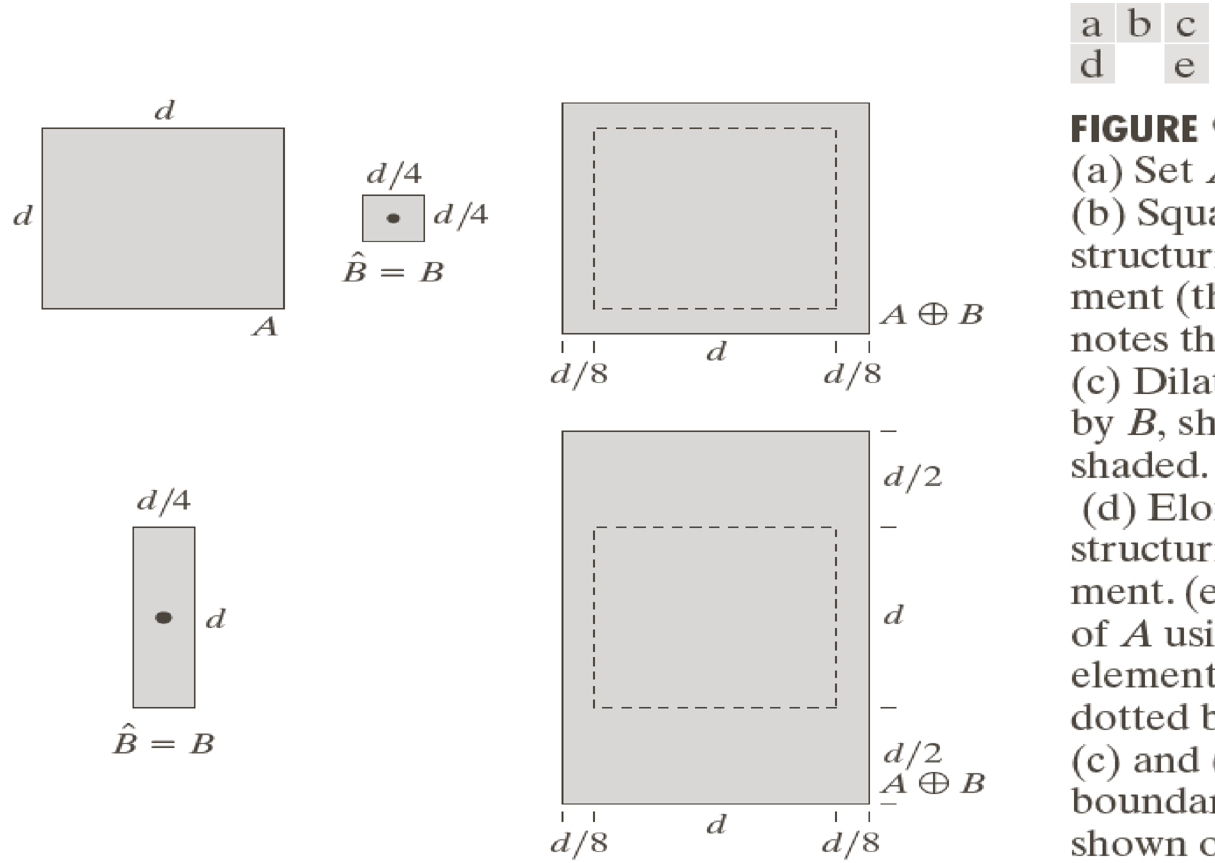
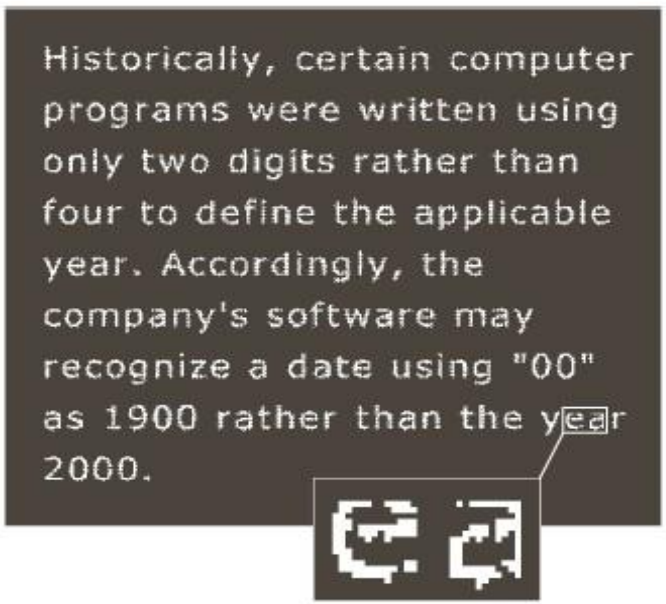


FIGURE 9.6
(a) Set A.
(b) Square structuring element (the dot denotes the origin).
(c) Dilation of A by B, shown shaded.
(d) Elongated structuring element. (e) Dilation of A using this element. The dotted border in (c) and (e) is the boundary of set A, shown only for reference

Dilation(Contd..)

Examples of Dilation



0	1	0
1	1	1
0	1	0

a c
b

FIGURE 9.7
(a) Sample text of poor resolution with broken characters (see magnified view). (b) Structuring element. (c) Dilation of (a) by (b). Broken segments were joined.

Erosion

With A and B as sets in Z^2, the erosion of A by B, denoted $A \quad B$, defined

$$A \quad B = \{z \mid (B)_Z \subseteq A\}$$

The set of all points z such that B, translated by z, is contained by A.

$$A \quad B = \{z \mid (B)_Z \cap A_{c} = \varnothing\}$$

Example of Erosion

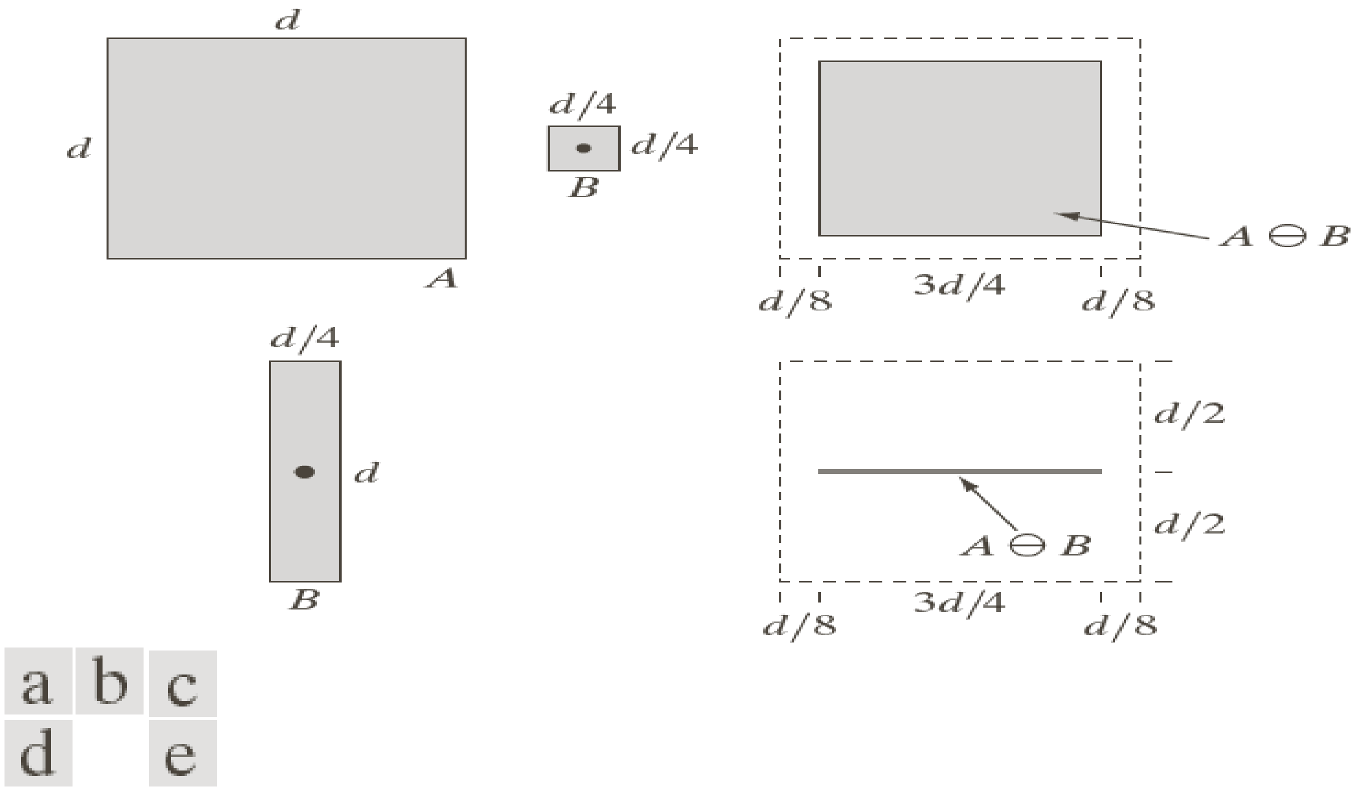
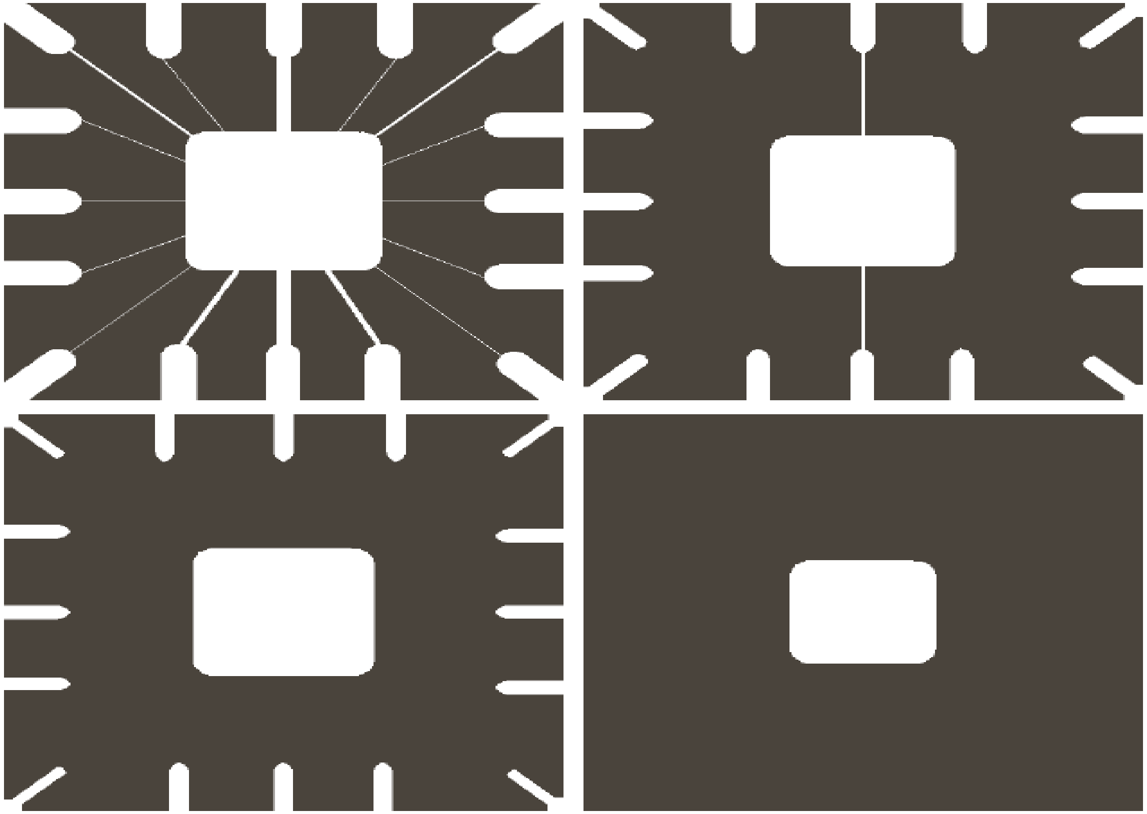


FIGURE 9.4 (a) Set A. (b) Square structuring element, B. (c) Erosion of A by B, shown shaded. (d) Elongated structuring element. (e) Erosion of A by B using this element. The dotted border in (c) and (e) is the boundary of set A, shown only for reference.

Erosion(Contd..)

Example of Erosion



a	b
c	d

FIGURE 9.5 Using erosion to remove image components. (a) A 486 × 486 binary image of a wire-bond mask. (b)–(d) Image eroded using square structuring elements of sizes 11 × 11, 15 × 15, and 45 × 45, respectively. The elements of the SEs were all 1s.

Duality

•Erosion and dilation are duals of each other with respect to set complementation and reflection

$$(A \ominus B)^c = A^c \oplus \hat{B}$$

and

$$(A \oplus B)^c = A^c \ominus \hat{B}$$

•Erosion and dilation are duals of each other with respect to set complementation and reflection

$$\begin{aligned}
(A \ominus B)^c &= \{z \mid (B)_Z \subseteq A\}^c \\
&= \{z \mid (B)_Z \cap A^c = \varnothing\}^c \\
&= \{z \mid (B)_Z \cap A^c \neq \varnothing\} \\
&= A^c \oplus \hat{B}
\end{aligned}$$

Opening and Closing

- Opening generally smoothes the contour of an object, breaks narrow isthmuses, and eliminates thin protrusions

- Closing tends to smooth sections of contours but it generates fuses narrow breaks and long thin gulfs, eliminates small holes, and fills gaps in the contour

The opening of set A by structuring element B, denoted $A \circ B$, is defined as

$$A \circ B = (A - B) \oplus B$$

The closing of set A by structuring element B, denoted $A \square B$, is defined as

$$A \square B = (A \oplus B) - B$$

Opening

The opening of set A by structuring element B, denoted $A \circ B$, is defined as

$$A \circ B = \bigcup \{ (B)_z \mid (B)_z \subseteq A \}$$

Example:

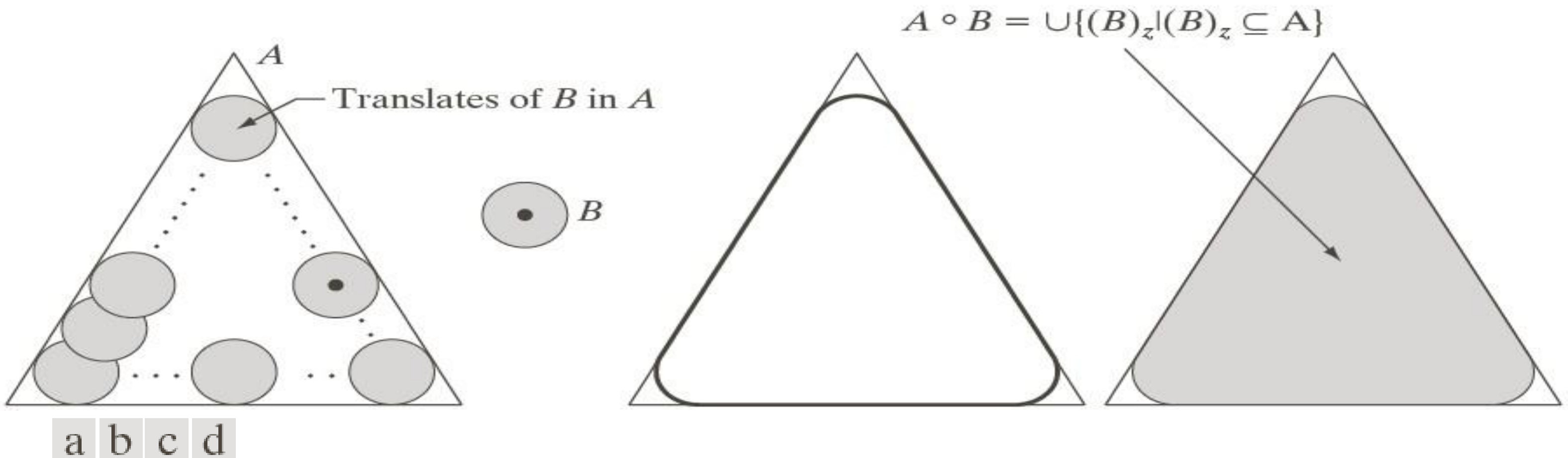


FIGURE 9.8 (a) Structuring element B "rolling" along the inner boundary of A (the dot indicates the origin of B). (b) Structuring element. (c) The heavy line is the outer boundary of the opening. (d) Complete opening (shaded). We did not shade A in (a) for clarity.

Closing

Example:

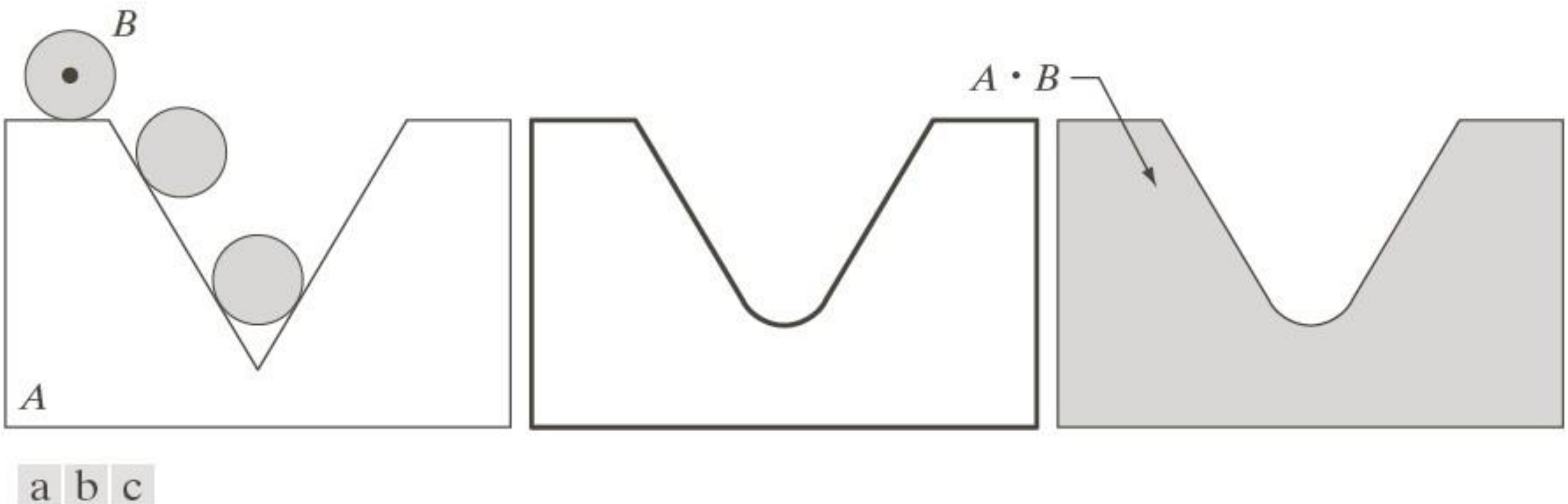
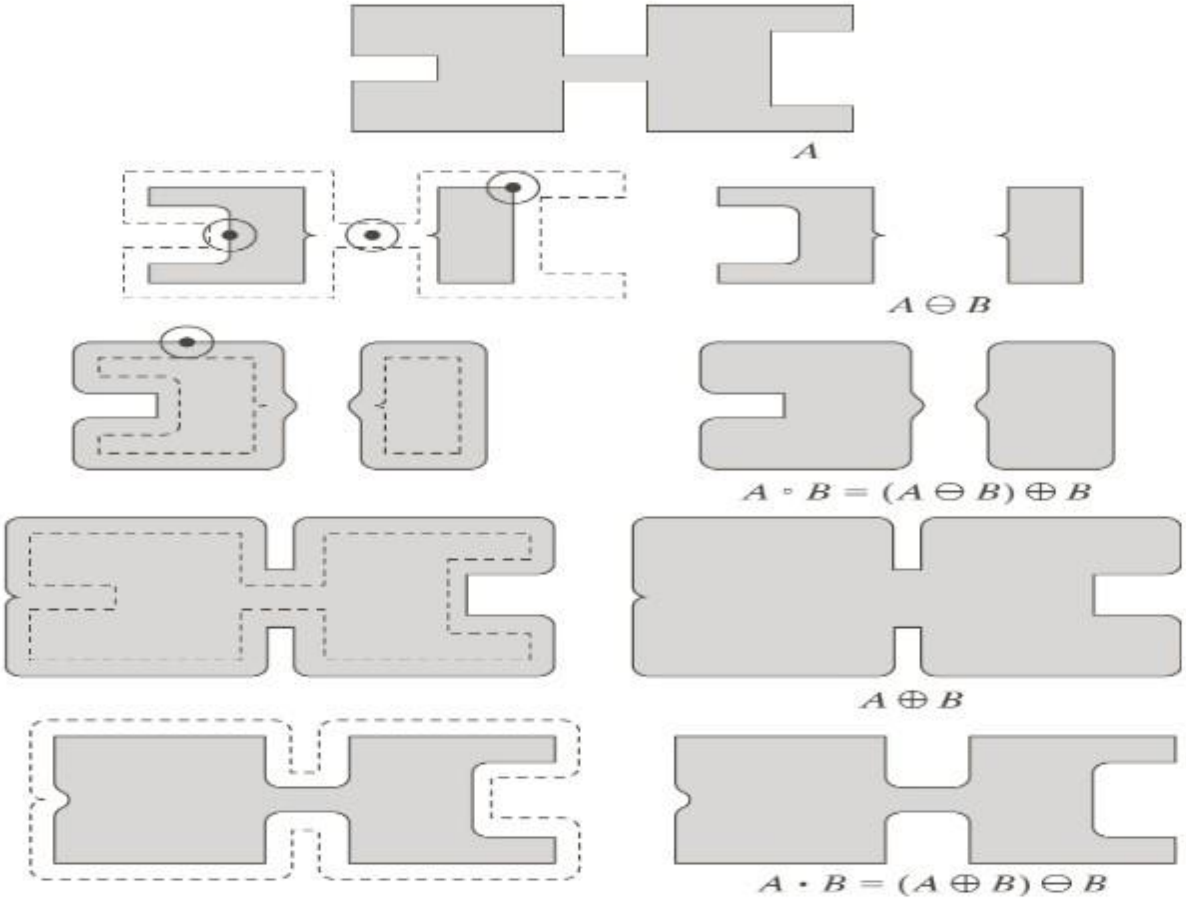


FIGURE 9.9 (a) Structuring element B "rolling" on the outer boundary of set A. (b) The heavy line is the outer boundary of the closing. (c) Complete closing (shaded). We did not shade A in (a) for clarity.

Opening and Closing(Contd..)



a	
b	c
d	e
f	g
h	i

FIGURE 9.10 Morphological opening and closing. The structuring element is the small circle shown in various positions in (b). The SE was not shaded here for clarity. The dark dot is the center of the structuring element.

Opening and Closing(Contd..)

Duality of Opening and Closing

•Opening and closing are duals of each other with respect to set complementation and reflection

$$\left(A \bullet B\right)^{c} = (A^{c} \circ B)$$

$$\left(A \circ B\right)^{c} = (A^{c} \bullet B)$$

Opening and Closing(Contd..)

The Properties of Opening and Closing

- Properties of Opening

(a) $A \circ B$ is a subset (subimage) of A

(b) if C is a subset of D, then $C \circ B$ is a subset of $D \circ B$

(c) $(A \circ B) \circ B = A \circ B$

- Properties of Closing

(a) A is subset (subimage) of $A \bullet B$

(b) If C is a subset of D, then $C \bullet B$ is a subset of $D \bullet B$

(c) $(A \bullet B) \bullet B = A \bullet B$

if B denotes the set composed of D and its background,the match (or set of matches) of B in A, denoted $A * B$,

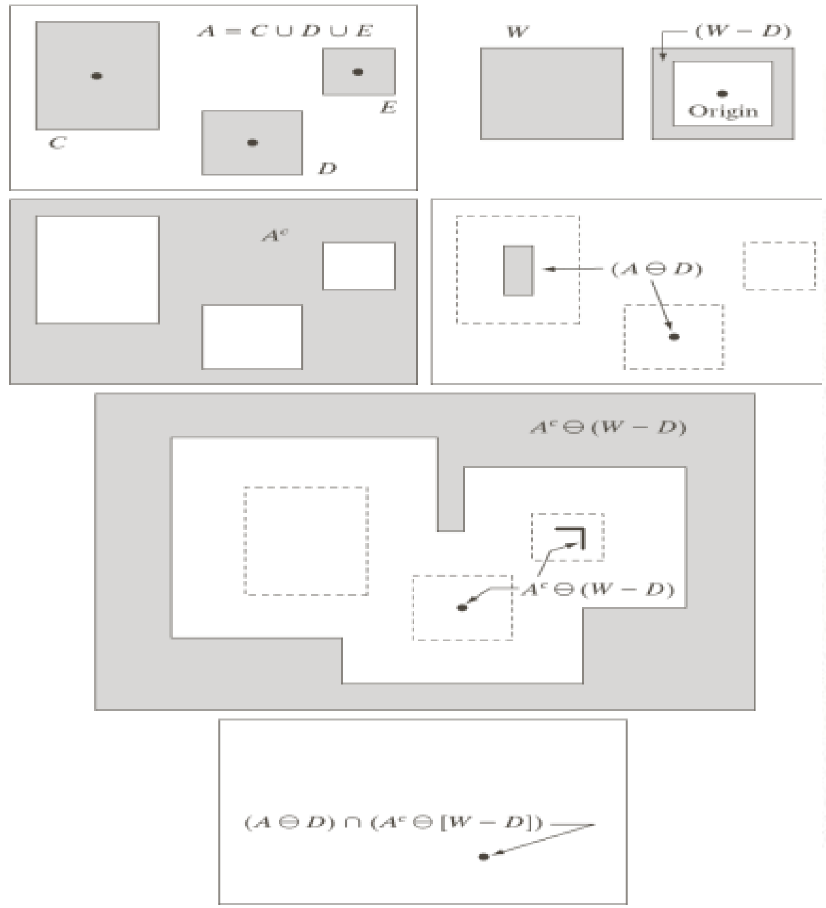
$$A * B = (A - D) \cap \left[A^c - (W - D)\right]$$

$$B = (B_1, B_2)$$

B_1: object

B_2: background

$$A * B = (A - B_1) \cap (A^c - B_2)$$



a b
c d
e
f

FIGURE 9.12 (a) Set A. (b) A window, W, and the local background of D with respect to W, $(W - D)$. (c) Complement of A. (d) Erosion of A by D. (e) Erosion of A^c by $(W - D)$. (f) Intersection of (d) and (e), showing the location of the origin of D, as desired. The dots indicate the origins of C, D, and E.

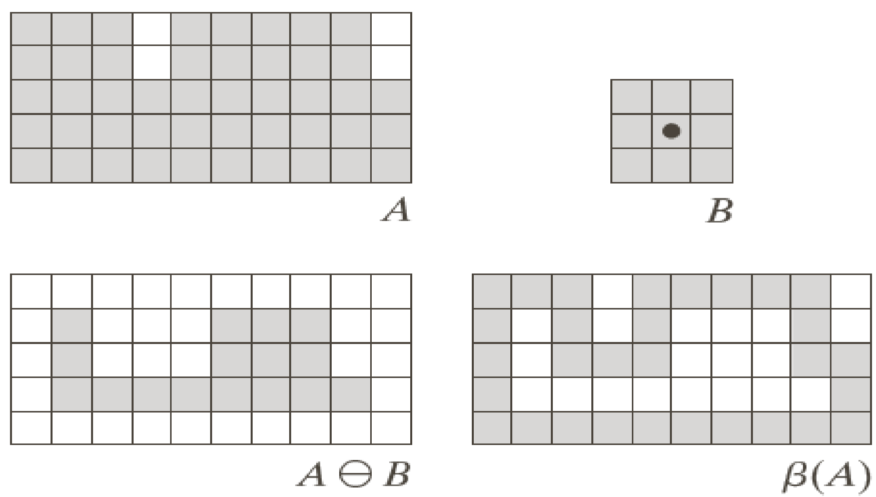
Algorithms

•Boundary Extraction

The boundary of a set A, can be obtained by first eroding A by B and then performing the set difference between A and its erosion.

$$\beta(A) = A - (A - B)$$

Example1:



a b
c d

FIGURE 9.13 (a) Set A. (b) Structuring element B. (c) A eroded by B. (d) Boundary, given by the set difference between A and its erosion.

Algorithms(Contd..)

Example2:

a b

FIGURE 9.14
(a) A simple binary image, with 1s represented in white. (b) Result of using Eq. (9.5-1) with the structuring element in Fig. 9.13(b).

Algorithms(Contd..)

Hole Filling

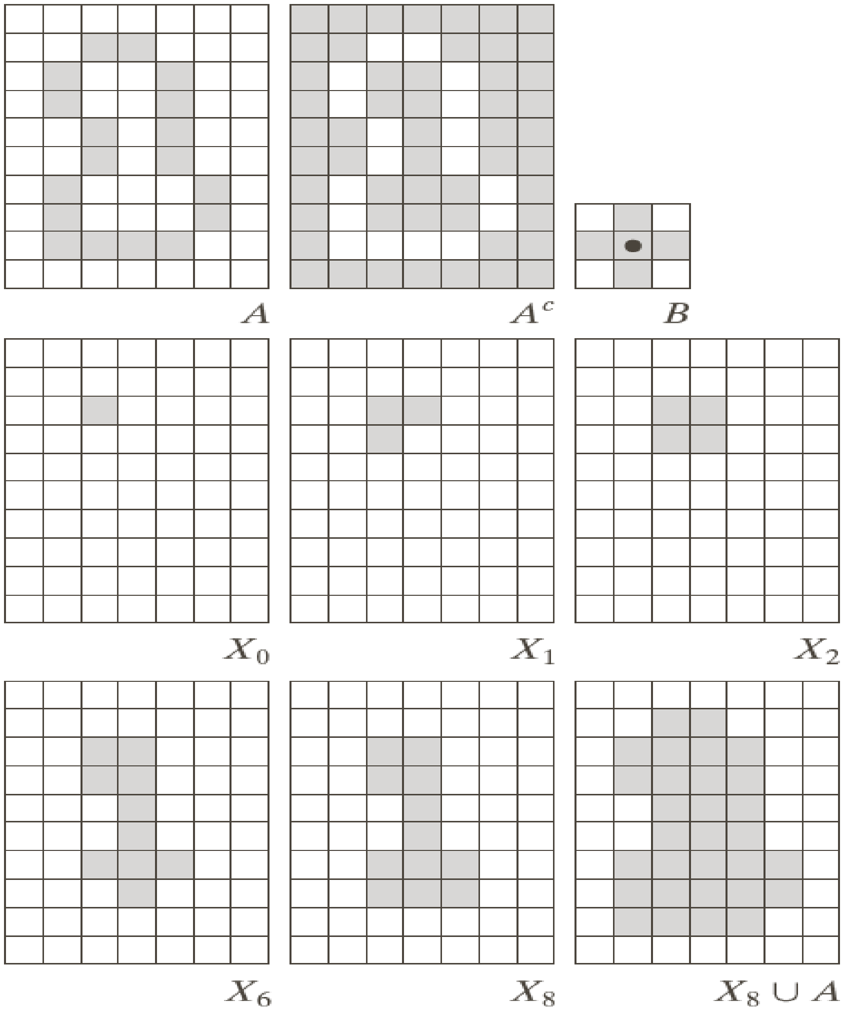
•A hole may be defined as a background region surrounded by a connected border of foreground pixels.

•Let A denote a set whose elements are 8-connected boundaries, each boundary enclosing a background region (i.e., a hole). Given a point in each hole, the objective is to fill all the holes with 1s.

•. Forming an array X_0 of 0s (the same size as the array containing A), except the locations in X_0 corresponding to the given point in each hole, which we set to 1.

- 2. $X_k = (X_{k-1} + B) \quad A^c \quad k=1,2,3,\ldots$
- Stop the iteration if $X_k = X_{k-1}$

Example:
Hole
Filling



a	b	c
d	e	f
g	h	i

FIGURE 9.15 Hole filling. (a) Set A (shown shaded). (b) Complement of A. (c) Structuring element B. (d) Initial point inside the boundary. (e)–(h) Various steps of Eq. (9.5-2). (i) Final result [union of (a) and (h)].

Algorithms(Contd..)

Extraction of Connected Components

- Central to many automated image analysis applications.

•Let A be a set containing one or more connected components, and form an array X_0 (of the same size as the array containing A) whose elements are 0s, except at each location known to correspond to a point in each connected component in A, which is set to 1.

•Central to many automated image analysis applications.

$$X_k = (X_{k-1} + B) \cap A$$

B : structuring element

until $X_k = X_{k-1}$

Example:Extraction of Connected Components

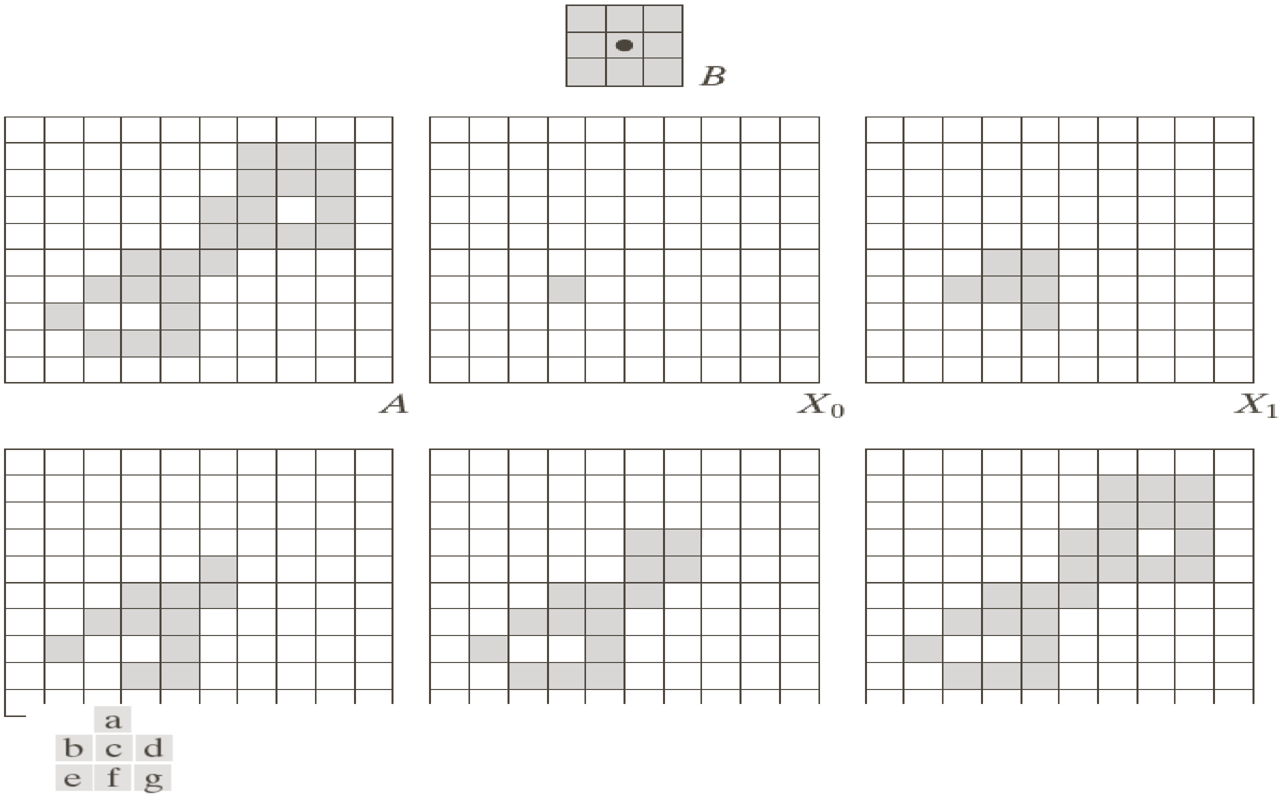


FIGURE 9.17 Extracting connected components. (a) Structuring element. (b) Array containing a set with one connected component. (c) Initial array containing a 1 in the region of the connected component. (d)–(g) Various steps in the iteration of Eq. (9.5-3).

Some Basic Morphological Algorithms(Contd..)

Convex Hull

•A set A is said to be ***convex*** if the straight line segment joining any two points in A lies entirely within A.

•The ***convex hull*** H or of an arbitrary set S is the smallest convex set containing S.

Let B^i, $i = 1, 2, 3, 4$, represent the four structuring elements.

The procedure consists of implementing the equation:

$$X_k^i = (X_{k-1} * B^i) \cup A$$

$$i = 1, 2, 3, 4 \text{ and } k = 1, 2, 3, \ldots$$

with $X_0^i = A$.

When the procedure converges, or $X_k^i = X_{k-1}^i$, let $D^i = X_k^i$,

the convex hull of A is

$$C(A) = \bigcup_{i=1}^{4} D^i$$

Example:Convex Hull

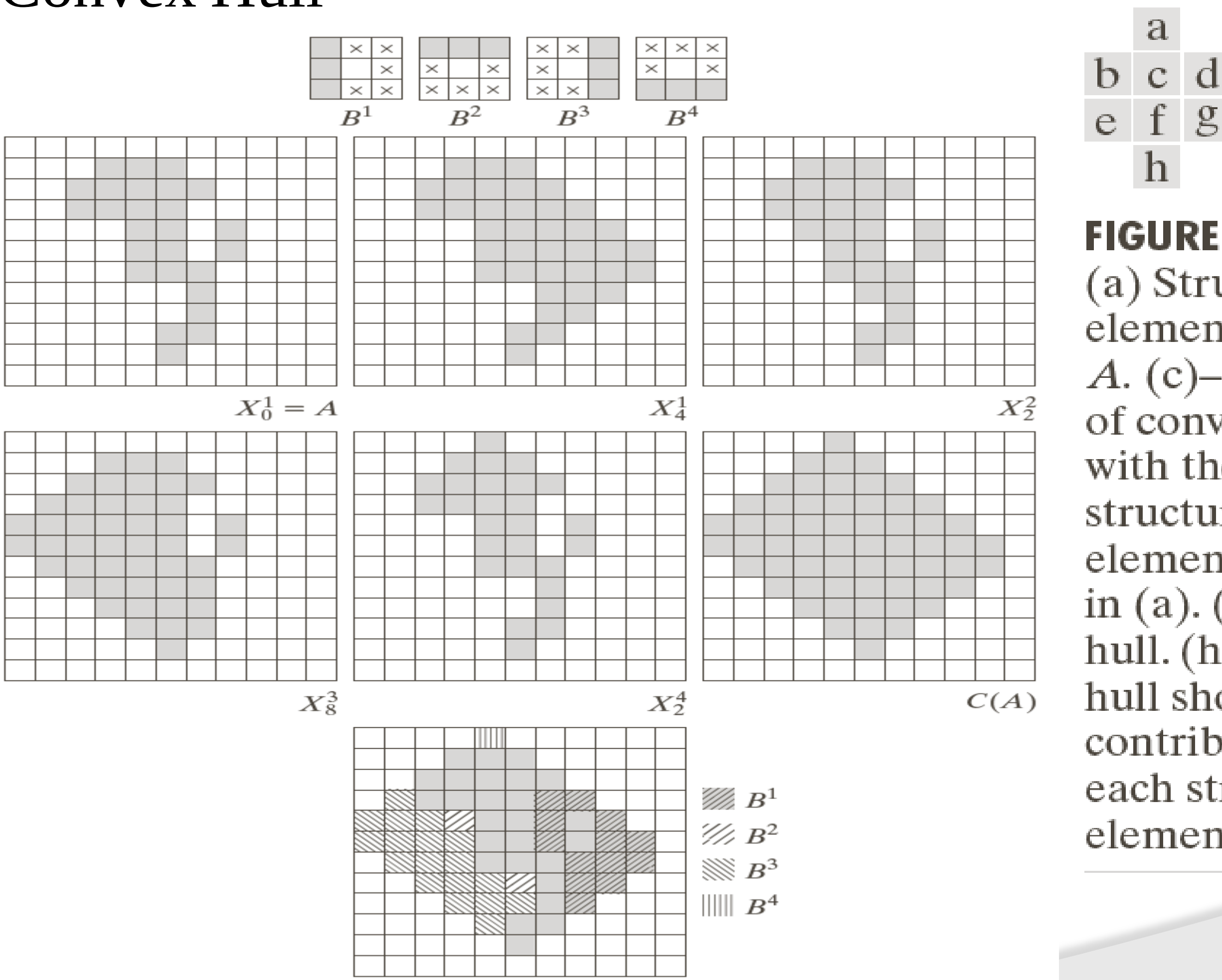


FIGURE 9.19 (a) Structuring elements. (b) Set A. (c)–(f) Results of convergence with the structuring elements shown in (a). (g) Convex hull. (h) Convex hull showing the contribution of each structuring element.

Algorithms(Contd..)

Thinning

- The thinning of a set A by a structuring element B, defined

$$A \otimes B = A - (A * B)$$
$$= A \cap (A * B)^c$$

•A more useful expression for thinning A symmetrically is based on a sequence of structuring elements:

$$\{B\} = \{B^1, B^2, B^3, \ldots, B^n\}$$

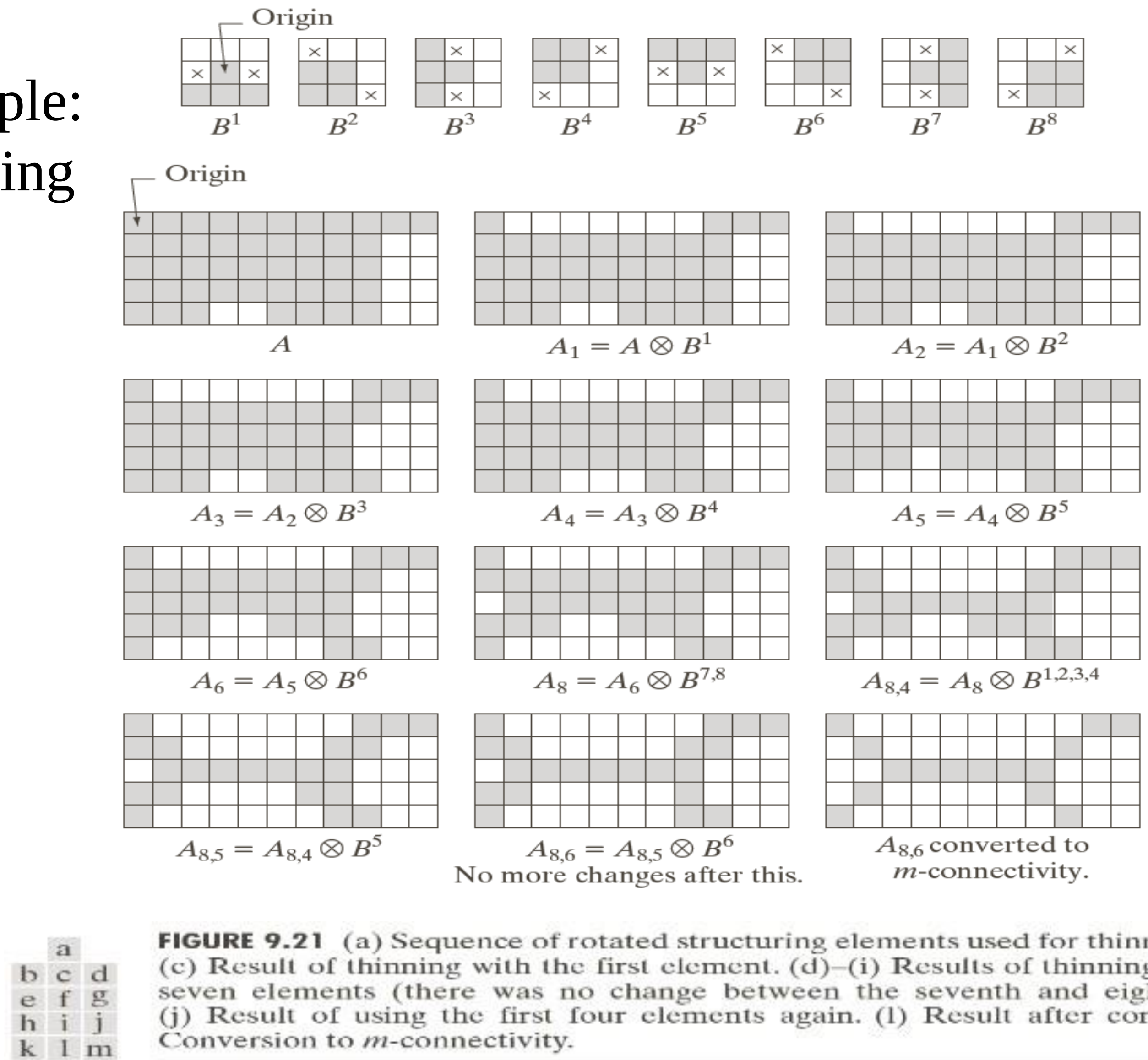
where B^i is a rotated version of B^{i-1}

The thinning of A by a sequence of structuring element $\{B\}$

$$A \otimes \{B\} = ((\ldots((A \otimes B^1) \otimes B^2)\ldots) \otimes B^n)$$

Algorithms(Contd..)

Example: Thinning



al

FIGURE 9.21 (a) Sequence of rotated structuring elements used for thinning. (b) Set A. (c) Result of thinning with the first element. (d)–(i) Results of thinning with the next seven elements (there was no change between the seventh and eighth elements). (j) Result of using the first four elements again. (l) Result after convergence. (m) Conversion to m-connectivity.

Algorithms(Contd..)

Thickening:

The thickening is defined by the expression

$$A \odot B = A \cup (A * B)$$

The thickening of A by a sequence of structuring element $\{B\}$

$$A \odot \{B\} = ((\ldots((A \odot B^1) \odot B^2)\ldots) \odot B^n)$$

In practice, the usual procedure is to thin the background of the set and then complement the result.

Algorithms(Contd..)

Example:Thickening

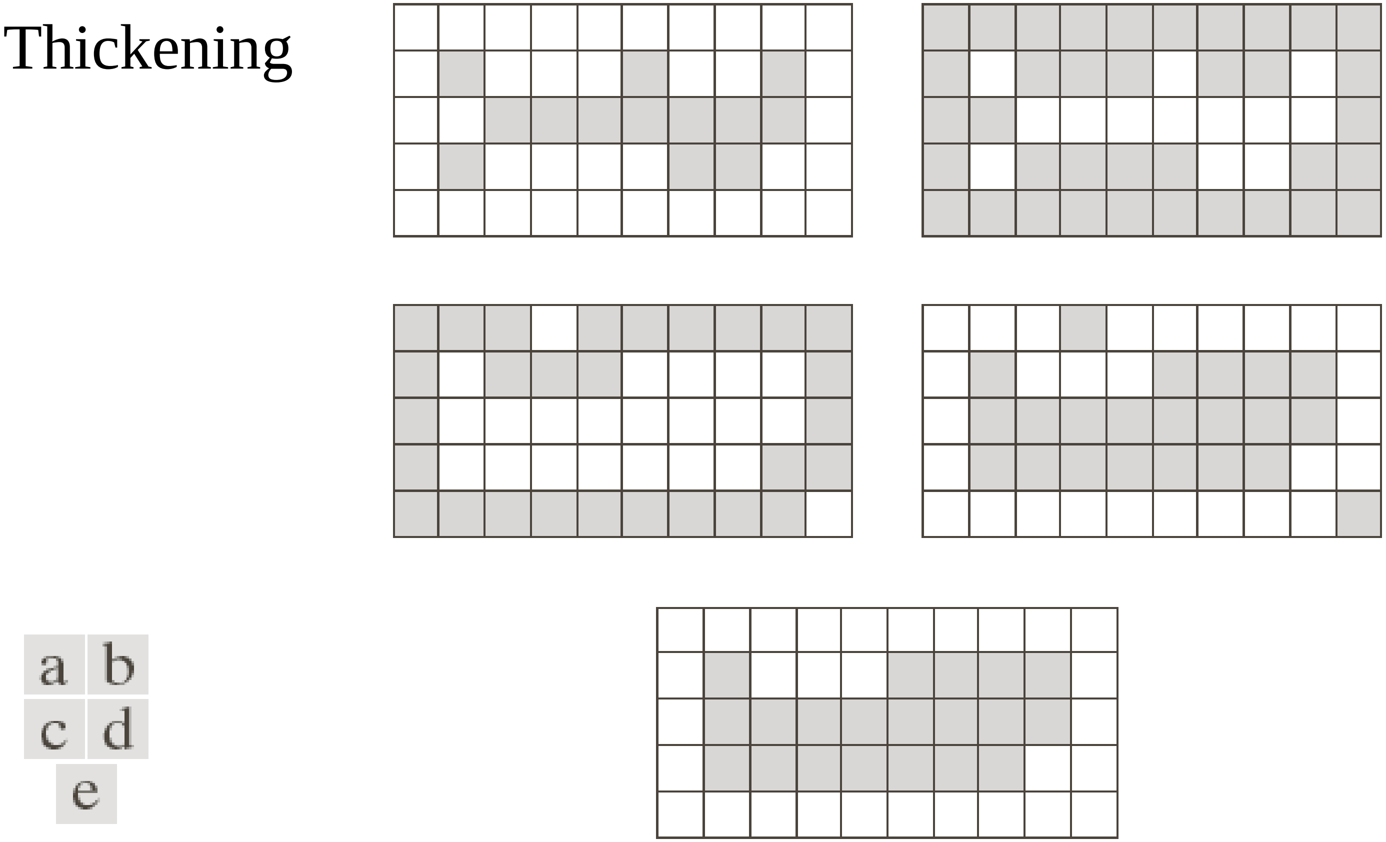


FIGURE 9.22 (a) Set A. (b) Complement of A. (c) Result of thinning the complement of A. (d) Thickened set obtained by complementing (c). (e) Final result, with no disconnected points.

Skeletons

A skeleton, $S(A)$ of a set A has the following properties

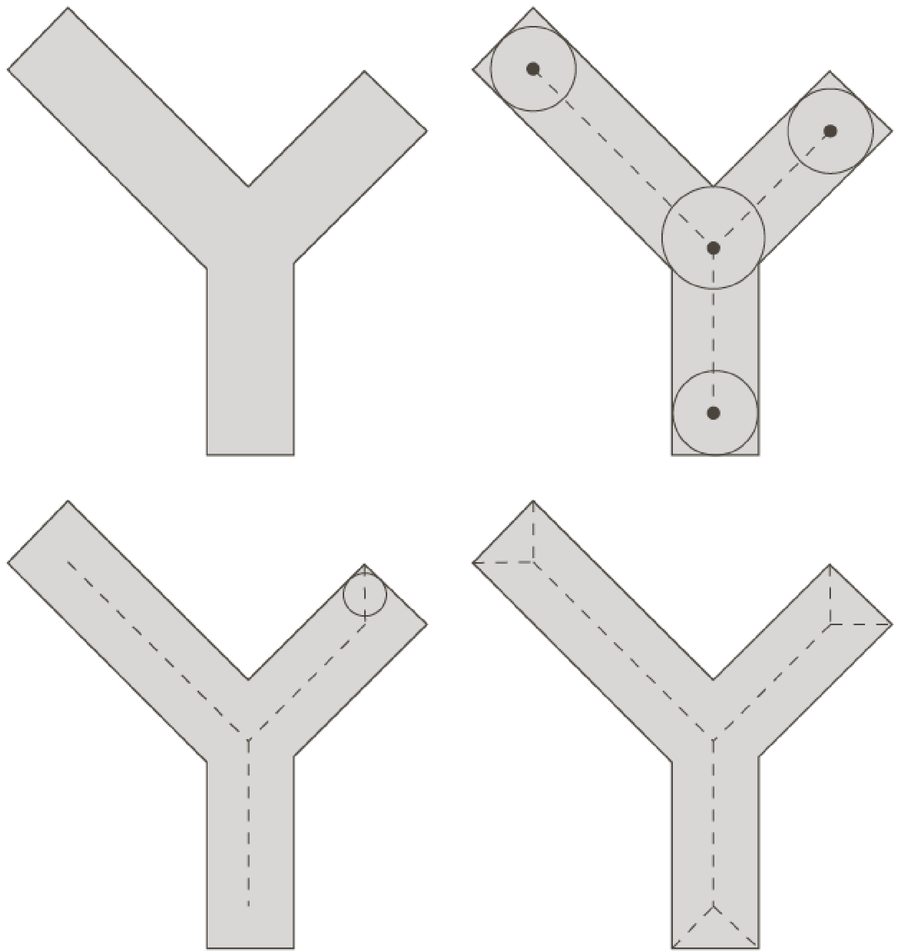
a. if z is a point of $S(A)$ and $(D)_z$ is the largest disk centered at z and contained in A, one cannot find a larger disk containing $(D)_z$ and included in A. The disk $(D)_z$ is called a maximum disk.

b. The disk $(D)_z$ touches the boundary of A at two or more different places.

Algorithms(Contd..)

Example:Skeleton

S



C

a	b
c	d

FIGURE 9.23
(a) Set A.
(b) Various positions of maximum disks with centers on the skeleton of A.
(c) Another maximum disk on a different segment of the skeleton of A.
(d) Complete skeleton.

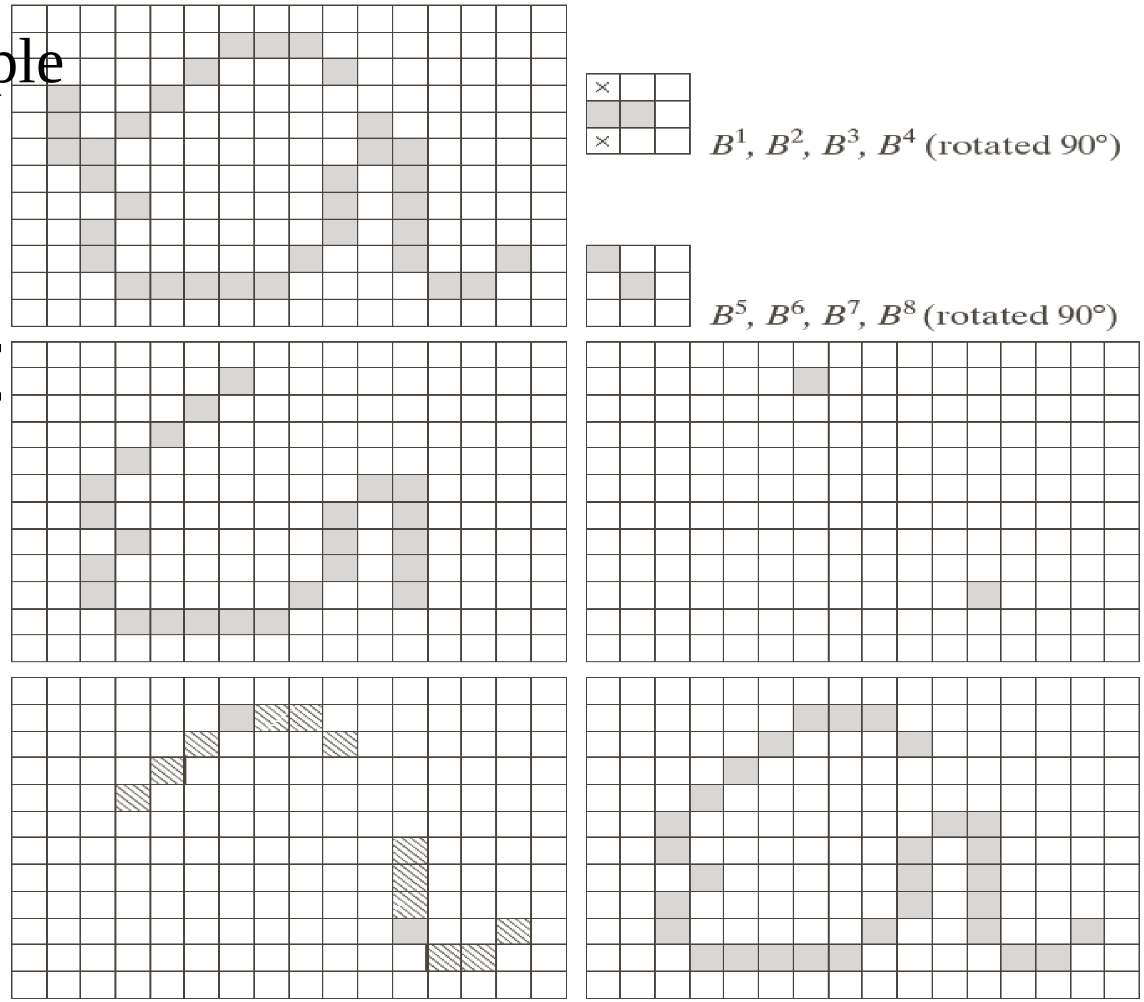
Algorithms(Contd..)

Pruning

a. Thinning and skeletonizing tend to leave parasitic components

b.Pruning methods are essential complement to thinning and skeletonizing procedures

Pruning: Example



a b
c
d e
f g

FIGURE 9.25 (a) Original image. (b) and (c) Structuring elements used for deleting end points. (d) Result of three cycles of thinning. (e) End points of (d). (f) Dilation of end points conditioned on (a). (g) Pruned image.

$$X_1 = A \otimes \{B\}$$

Some Basic Morphological Algorithms(Contd..)

Pruning: Example

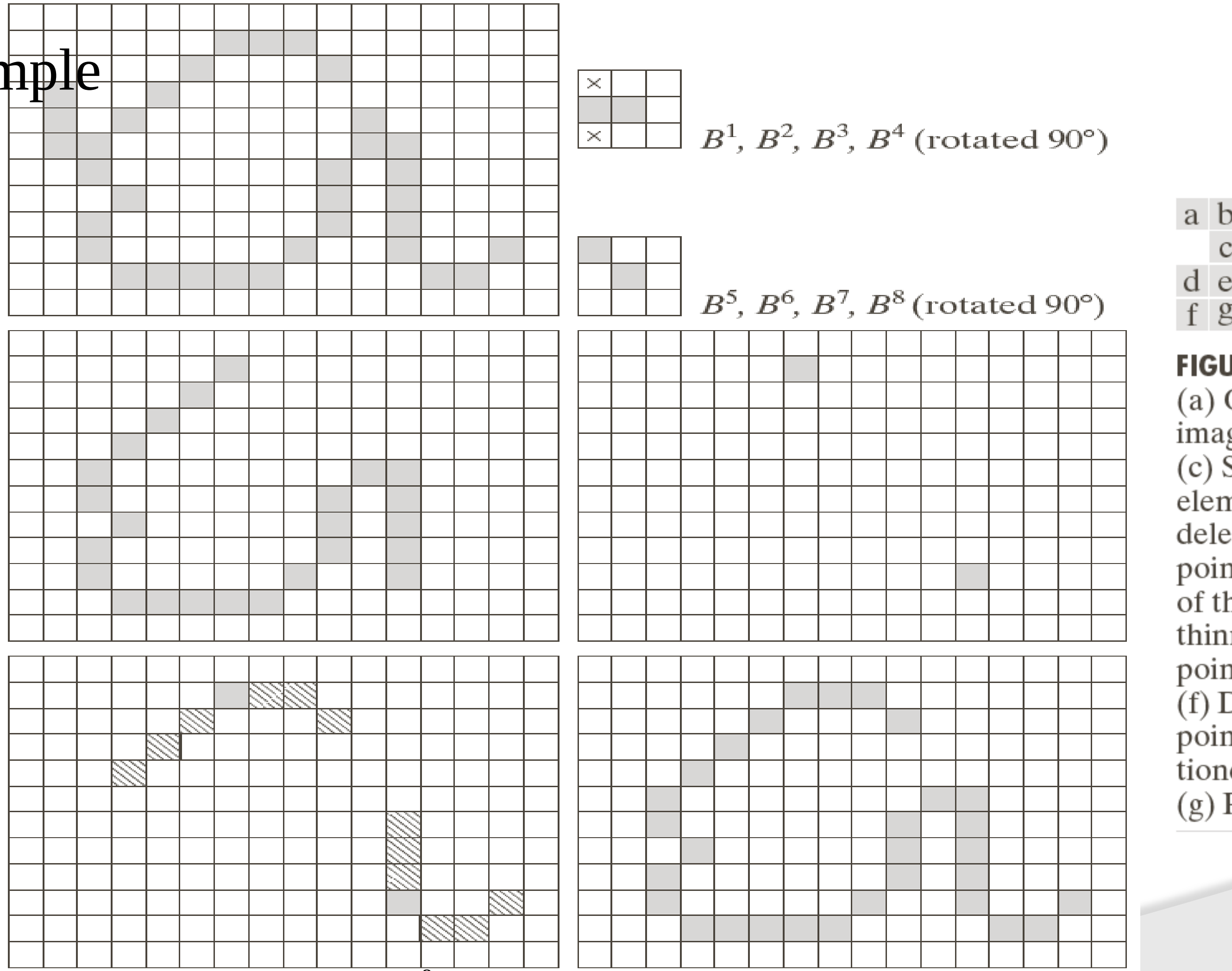


FIGURE 9.25 (a) Original image. (b) and (c) Structuring elements used for deleting end points. (d) Result of three cycles of thinning. (e) End points of (d). (f) Dilation of end points conditioned on (a). (g) Pruned image.

$$X_2 = \bigcup_{k=1}^{8} (X_1 * B^k)$$

Algorithms(Contd..)

Pruning: Example

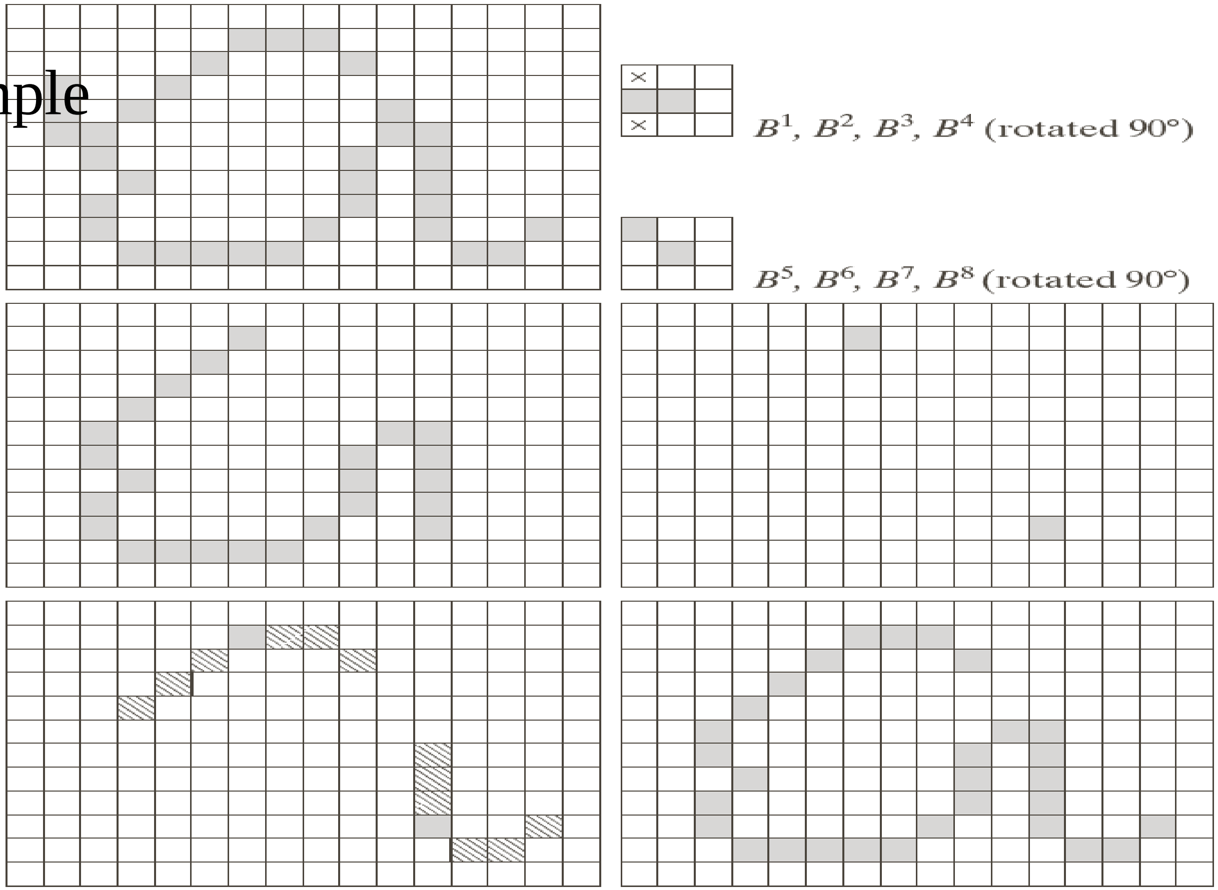


FIGURE 9.25 (a) Original image. (b) and (c) Structuring elements used for deleting end points. (d) Result of three cycles of thinning. (e) End points of (d). (f) Dilation of end points conditioned on (a). (g) Pruned image.

$$X_3 = (X_2 \oplus H) \cap A$$

H : 3×3 structuring element

Image Segmentation

- Image segmentation divides an image into regions that are connected and have some similarity within the region and some difference between adjacent regions.

- The goal is usually to find individual objects in an image.

- For the most part there are fundamentally two kinds of approaches to segmentation: discontinuity and similarity.

    -Similarity may be due to pixel intensity, color or texture.

    -Differences are sudden changes (discontinuities) in any of these, but especially sudden changes in intensity along a boundary line, which is called an edge.

Detection of Discontinuities

•There are three kinds of discontinuities of intensity: points, lines and edges.

•The most common way to look for discontinuities is to scan a small mask over the image.The mask determines which kind of discontinuity to look for.

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9 = \sum_{i=1}^{9} w_i z_i$$

FIGURE 10.1 A general 3×3 mask.

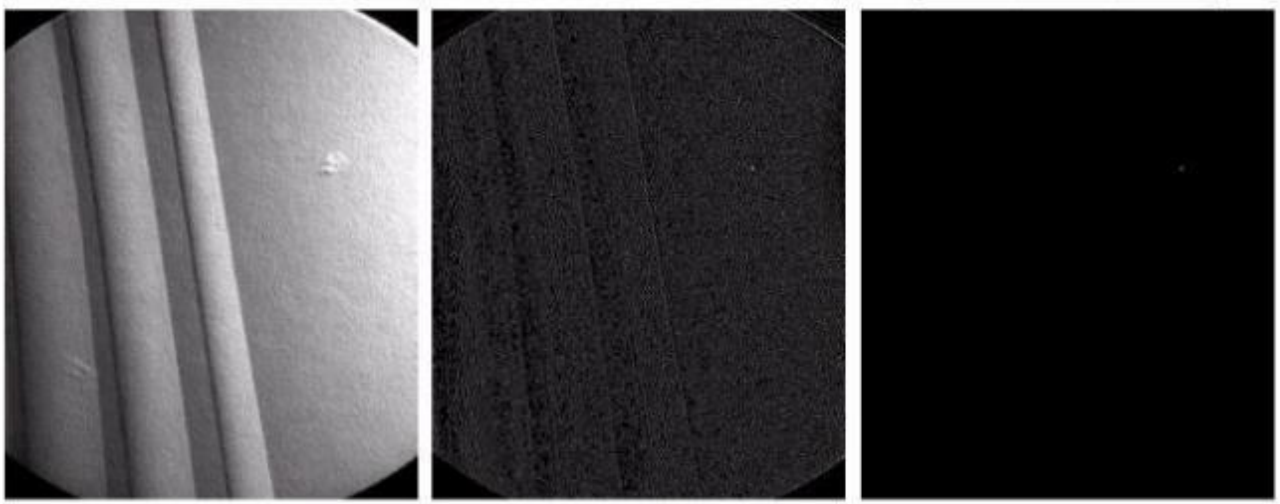
w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Point Detection

$$|R| \geq T$$

where T : a nonnegative threshold

−1	−1	−1
−1	8	−1
−1	−1	−1



a
b c d

FIGURE 10.2 (a) Point detection mask. (b) X-ray image of a turbine blade with a porosity. (c) Result of point detection. (d) Result of using Eq. (10.1-2). (Original image courtesy of X-TEK Systems Ltd.)

Detection of Discontinuities

•Only slightly more common than point detection is to find a one pixel wide line in an image.

•For digital images the only three point straight lines are only horizontal, vertical, or diagonal (+ or –45°).

FIGURE 10.3 Line masks.

Horizontal:

−1	−1	−1
2	2	2
−1	−1	−1

+45°:

−1	−1	2
−1	2	−1
2	−1	−1

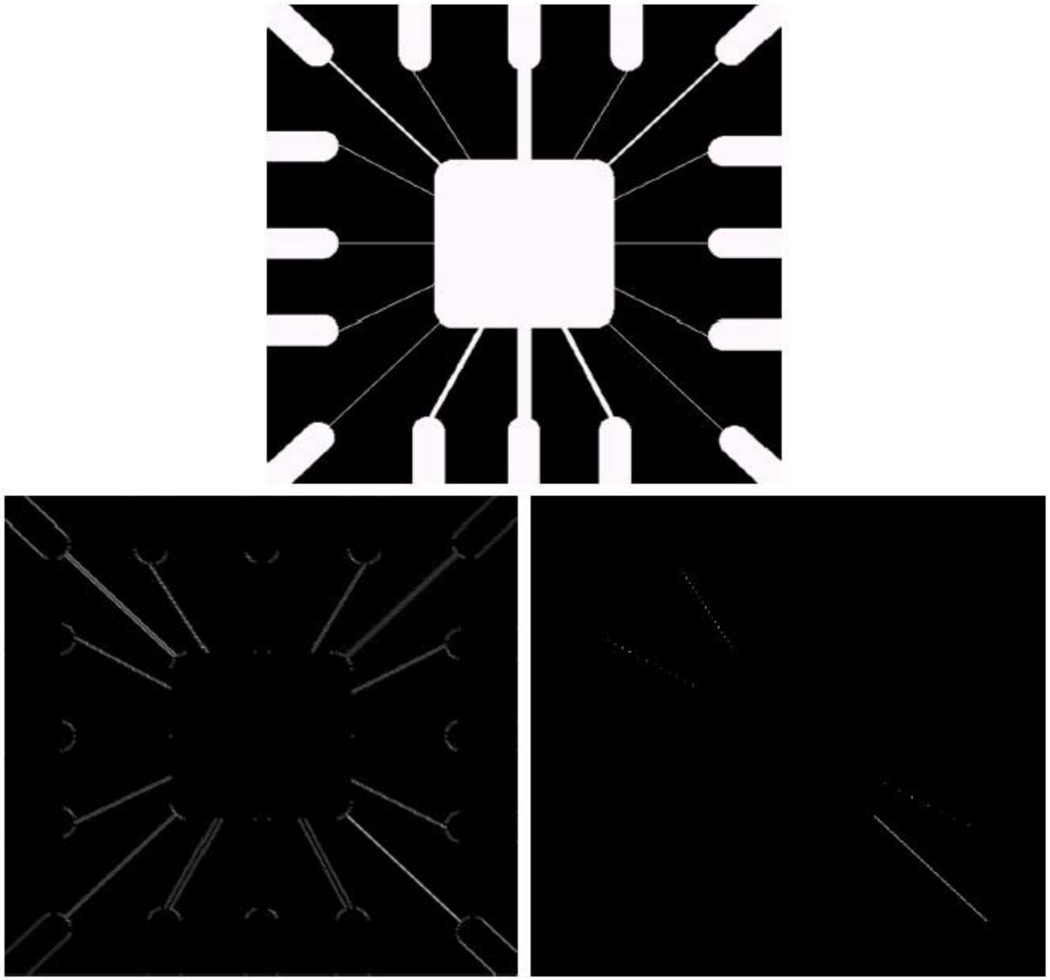
Vertical:

−1	2	−1
−1	2	−1
−1	2	−1

−45°:

2	−1	−1
−1	2	−1
−1	−1	2

Line Detection



a
b c

FIGURE 10.4 Illustration of line detection. (a) Binary wire-bond mask. (b) Absolute value of result after processing with −45° line detector. (c) Result of thresholding image (b).

Edge Detection

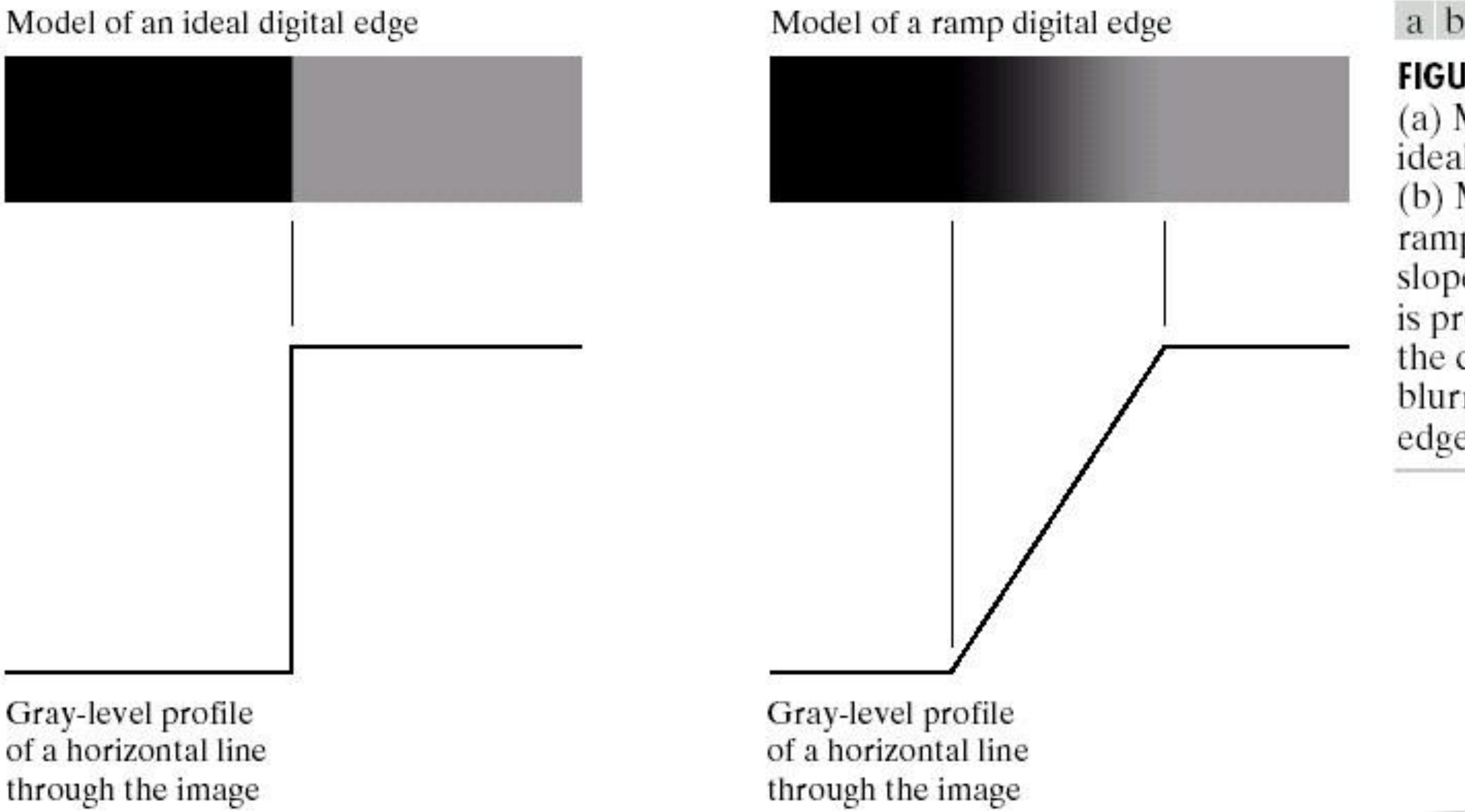
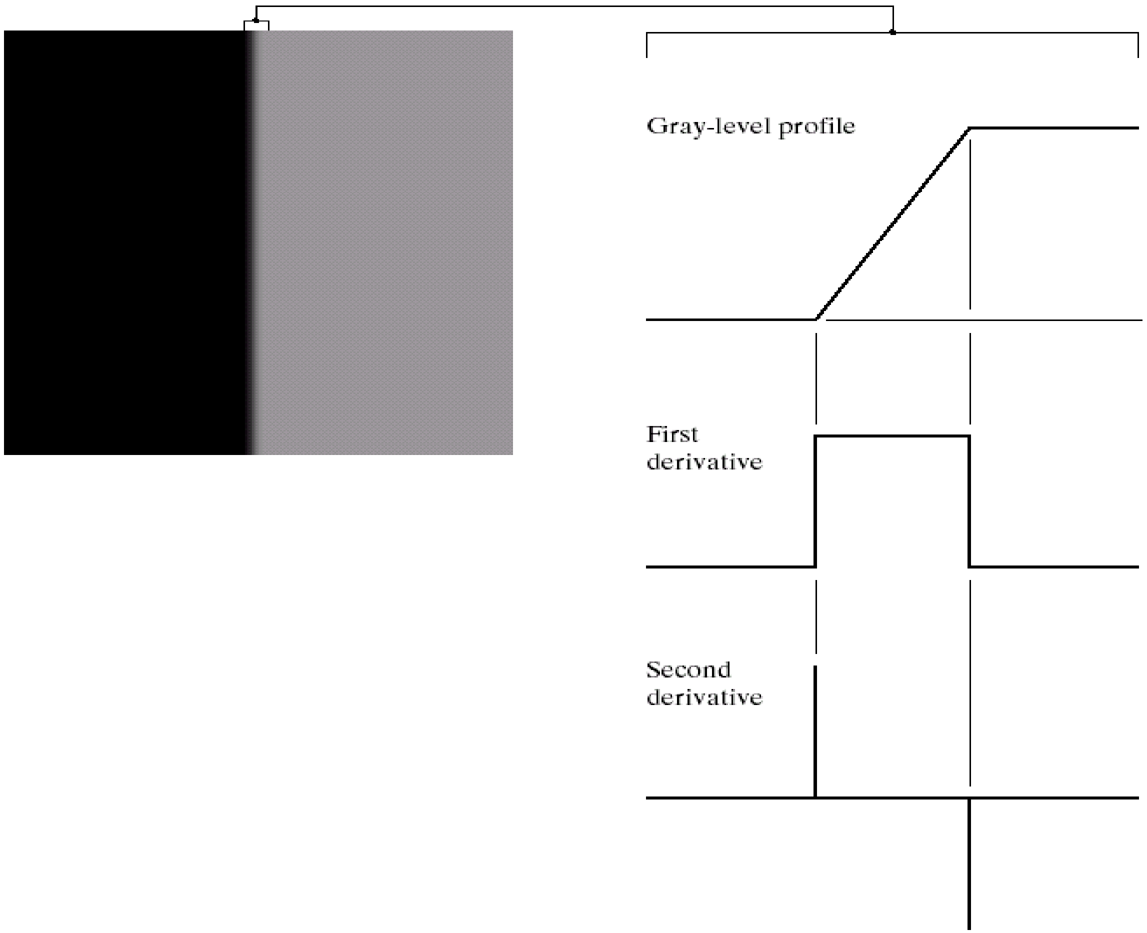


FIGURE 10.5 (a) Model of an ideal digital edge. (b) Model of a ramp edge. The slope of the ramp is proportional to the degree of blurring in the edge.

Edge Detection(Contd..)

a b

FIGURE 10.6 (a) Two regions separated by a vertical edge. (b) Detail near the edge, showing a gray-level profile, and the first and second derivatives of the profile.



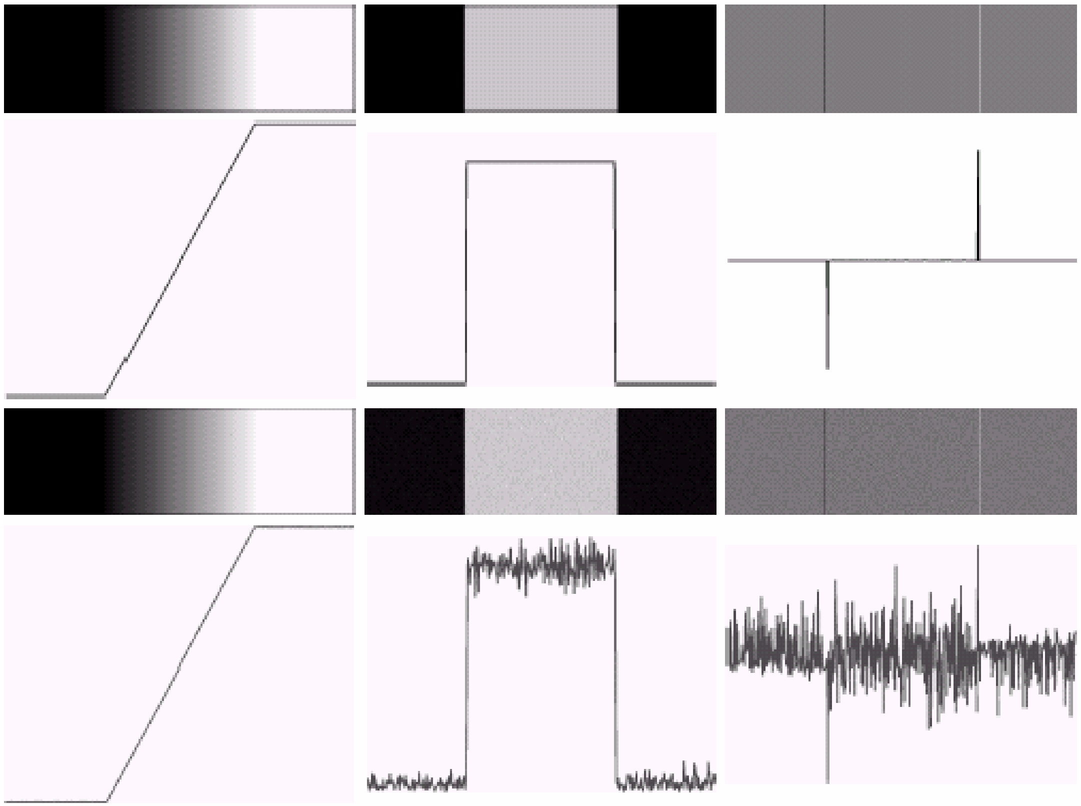


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and $\sigma = 0.0, 0.1, 1.0,$ and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

Edge Detection(Contd..)

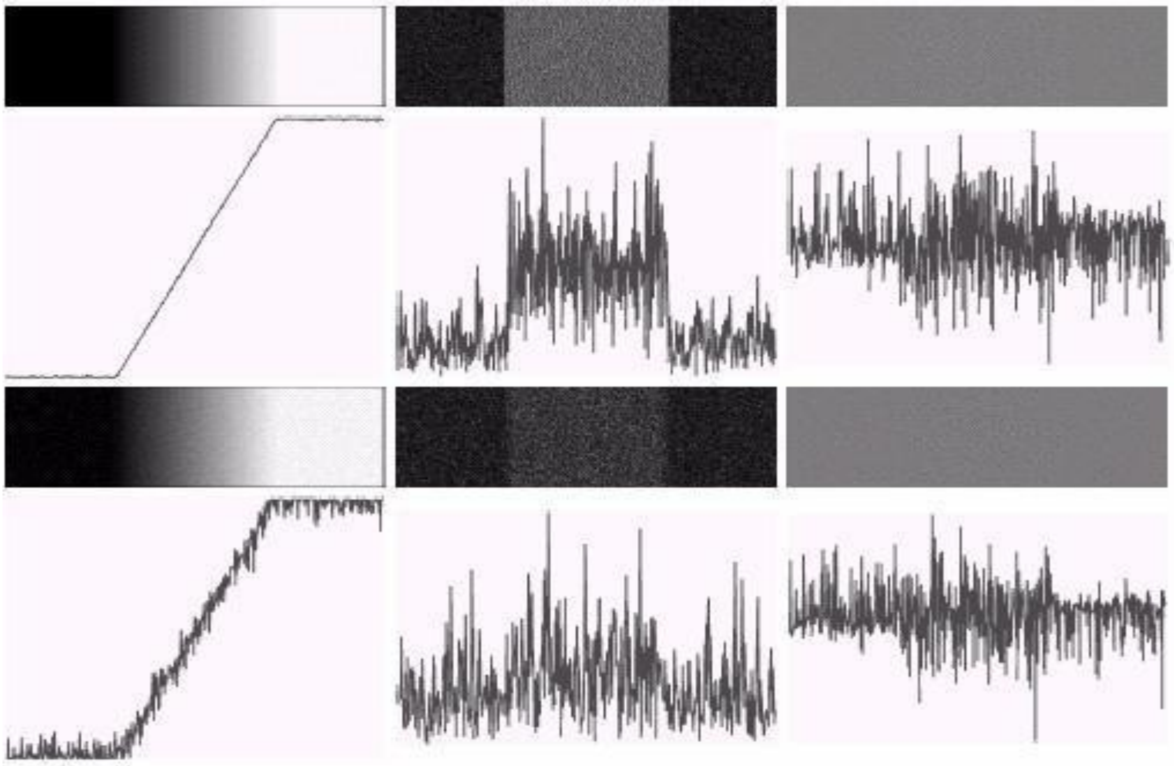


FIGURE 10.7 First column: images and gray-level profiles of a ramp edge corrupted by random Gaussian noise of mean 0 and $\sigma = 0.0, 0.1, 1.0$, and 10.0, respectively. Second column: first-derivative images and gray-level profiles. Third column: second-derivative images and gray-level profiles.

a
b
c
d

Gradient Operators

First-order derivatives:

- The gradient of an image *f*(*x*,*y*) at location (*x*,*y*) is defined as the vector:

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

- The magnitude of this vector: $\nabla f = \mathrm{mag}(\nabla \mathbf{f}) = \left[G_x^2 + G_y^2 \right]^{1/2}$

- The direction of this vector: $\alpha(x, y) = \tan^{-1}\left(\frac{G_x}{G_y} \right)$

Detection of Discontinuities

Roberts cross-gradient operators

−1	0		0	−1
0	1		1	0

Roberts

Prewitt operators

−1	−1	−1		−1	0	1
0	0	0		−1	0	1
1	1	1		−1	0	1

Prewitt

Sobel operators

−1	−2	−1		−1	0	1
0	0	0		−2	0	2
1	2	1		−1	0	1

Sobel

Detection of Discontinuities

Prewitt masks for detecting diagonal edges

0	1	1
−1	0	1
−1	−1	0

−1	−1	0
−1	0	1
0	1	1

Prewitt

Sobel masks for detecting diagonal edges

0	1	2
−1	0	1
−2	−1	0

−2	−1	0
−1	0	1
0	1	2

Sobel

a	b
c	d

FIGURE 10.9 Prewitt and Sobel masks for detecting diagonal edges.

Gradient Operators: Example

a b
c d

FIGURE 10.10 (a) Original image. (b) $|G_x|$, component of the gradient in the x-direction. (c) $|G_y|$, component in the y-direction. (d) Gradient image, $|G_x| + |G_y|$.

Gradient Operators(Contd..)

Second-order derivatives: (The Laplacian)

- The Laplacian of an 2D function $f(x,y)$ is defined as

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

- Two forms in practice:

FIGURE 10.13 Laplacian masks used to implement Eqs. (10.1-14) and (10.1-15), respectively.

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

Detection of Discontinuities

- Consider the function:

A Gaussian function

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}} \quad \text{where} \quad r^2 = x^2 + y^2$$

$$\text{and } \sigma : \text{the standard deviation}$$

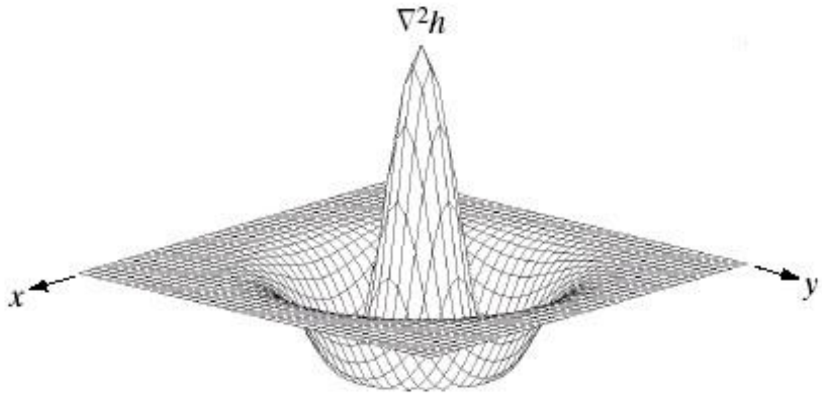
- The Laplacian of *h* is

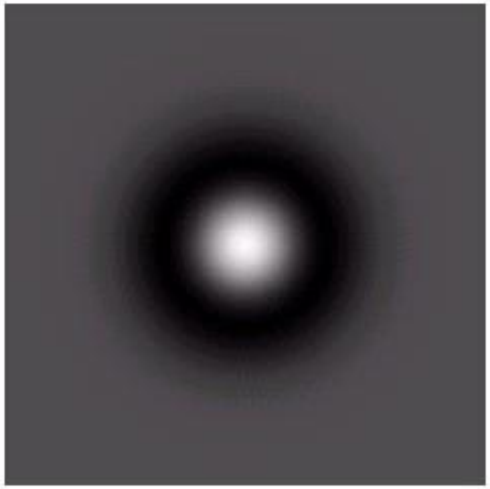
$$\nabla^2 h(r) = -\left[\frac{r^2 - \sigma^2}{\sigma^4}\right] e^{-\frac{r^2}{2\sigma^2}}$$

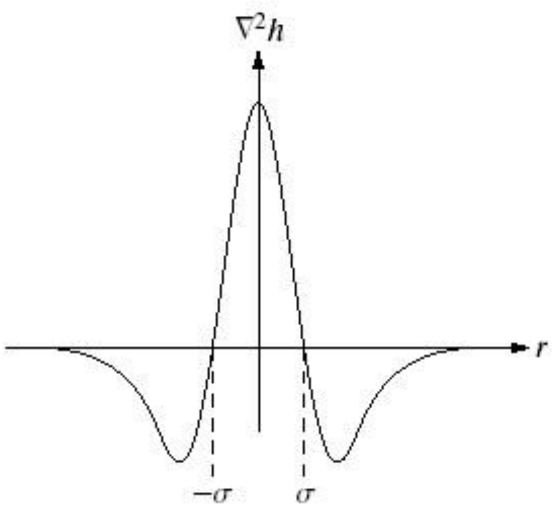
The Laplacian of a Gaussian (LoG)

- The Laplacian of a Gaussian sometimes is called the Mexican hat function. It also can be computed by smoothing the image with the Gaussian smoothing mask, followed by application of the Laplacian mask.

Detection of Discontinuities





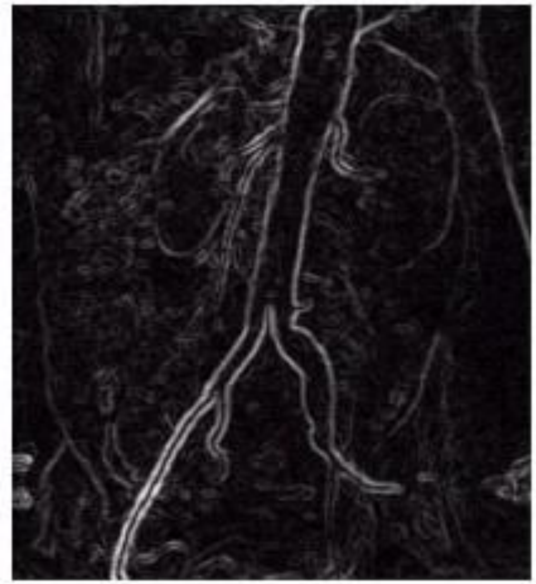


0	0	–1	0	0
0	–1	–2	–1	0
–1	–2	16	–2	–1
0	–1	–2	–1	0
0	0	–1	0	0

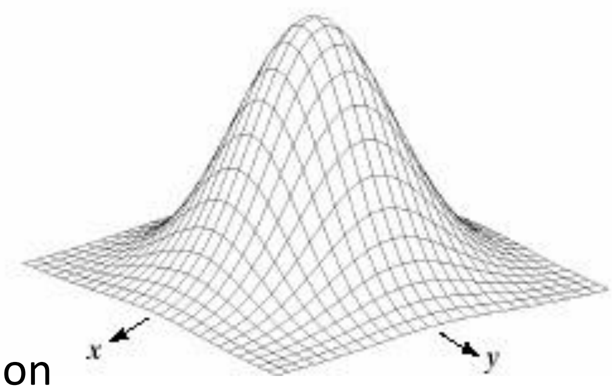
a	b
c	d

FIGURE 10.14 Laplacian of a Gaussian (LoG). (a) 3-D plot. (b) Image (black is negative, gray is the zero plane, and white is positive). (c) Cross section showing zero crossings. (d) 5 × 5 mask approximation to the shape of (a).

Detection of Discontinuities



Sobel gradient



Gaussian smooth function

–1	–1	–1
–1	8	–1
–1	–1	–1

Laplacian mask

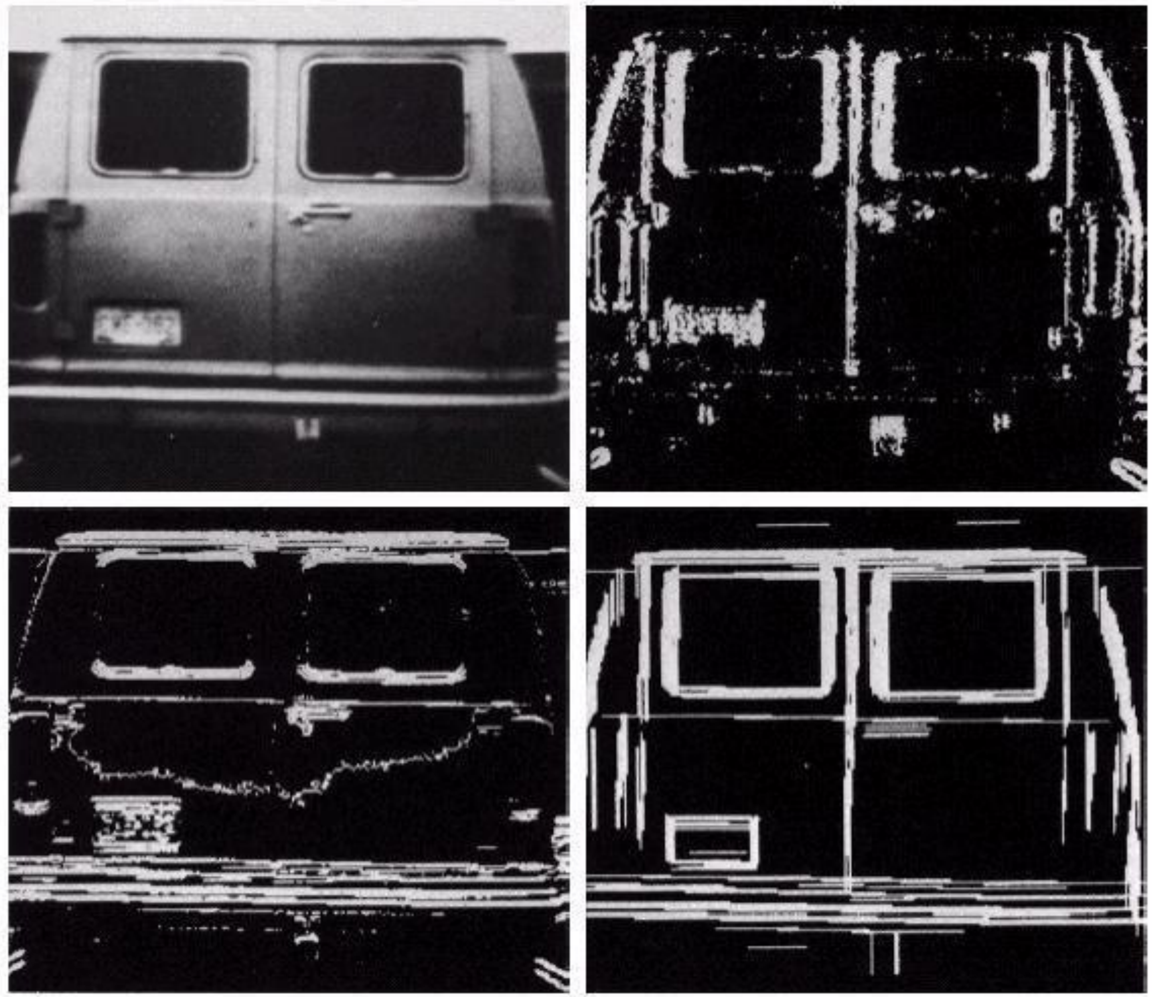
Edge Linking and Boundary Detection

- Two properties of edge points are useful for edge linking:
  - the strength (or magnitude) of the detected edge points
  - their directions (determined from gradient directions)
- This is usually done in local neighborhoods.
- Adjacent edge points with similar magnitude and direction are linked.
- For example, an edge pixel with coordinates (x_0,y_0) in a predefined neighborhood of (x,y) is similar to the pixel at (x,y) if

$$\left|\nabla f(x,y) - \nabla(x_0,y_0)\right| \le E, \quad E: \text{a nonnegative threshold}$$

$$\left|\alpha(x,y) - \alpha(x_0,y_0)\right| < A, \quad A: \text{a nonegative angle threshold}$$

Edge Linking and Boundary Detection



a	b
c	d

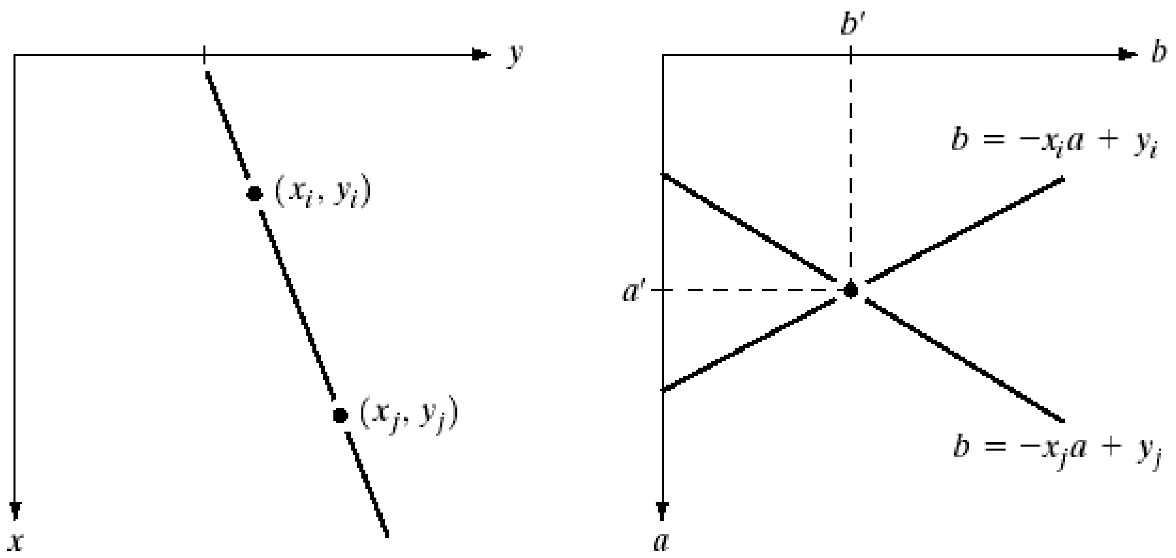
FIGURE 10.16 (a) Input image. (b) G_y component of the gradient. (c) G_x component of the gradient. (d) Result of edge linking. (Courtesy of Perceptics Corporation.)

In this example, we can find the license plate candidate after edge linking process.

Edge Linking and Boundary Detection

- Hough transform: a way of finding edge points in an image that lie along a straight line.
- Example: *xy*-plane v.s. *ab*-plane (parameter space)

$$y_i = ax_i + b$$

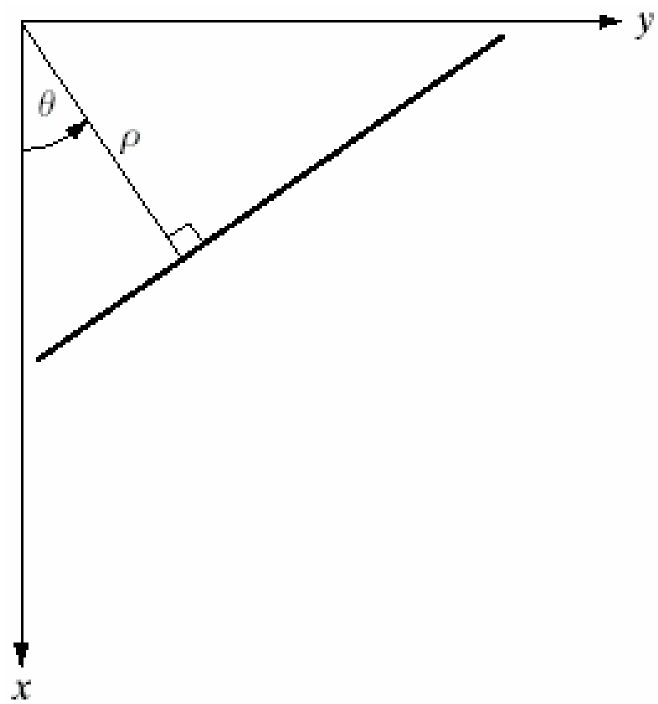


a b

FIGURE 10.17 (a) *xy*-plane. (b) Parameter space.

Edge Linking and Boundary Detection

- The Hough transform consists of finding all pairs of values of θ and ρ which satisfy the equations that pass through (x,y).
- These are accumulated in what is basically a 2-dimensional histogram.
- When plotted these pairs of θ and ρ will look like a sine wave. The process is repeated for all appropriate (x,y) locations.

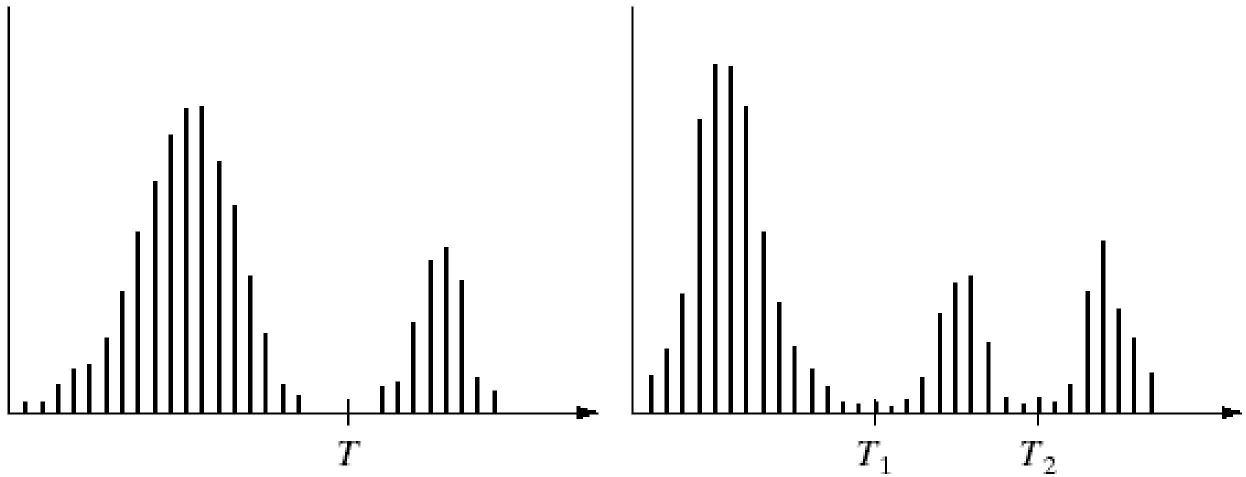


$$x\cos\theta + y\sin\theta = \rho$$

Thresholding

- Assumption: the range of intensity levels covered by objects of interest is different from the background.

$$g(x, y) = \begin{cases} 1 & \text{if} \quad f(x, y) > T \\ 0 & \text{if} \quad f(x, y) \le T \end{cases}$$



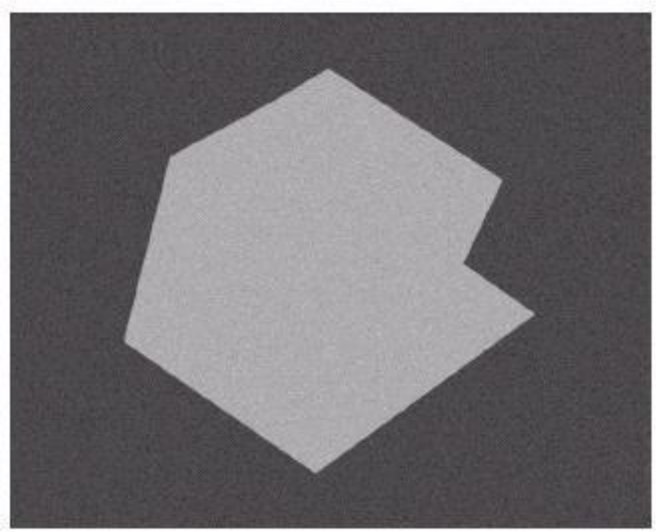
a b

Single threshold

Multiple threshold

FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

Thresholding
The Role of Illumination



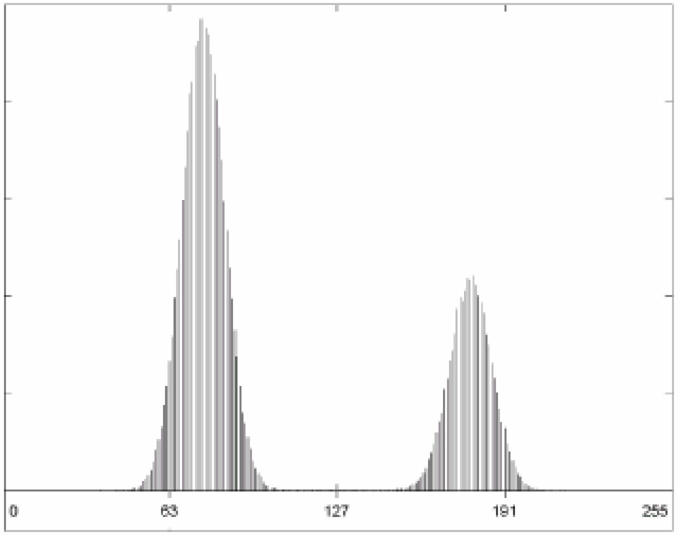
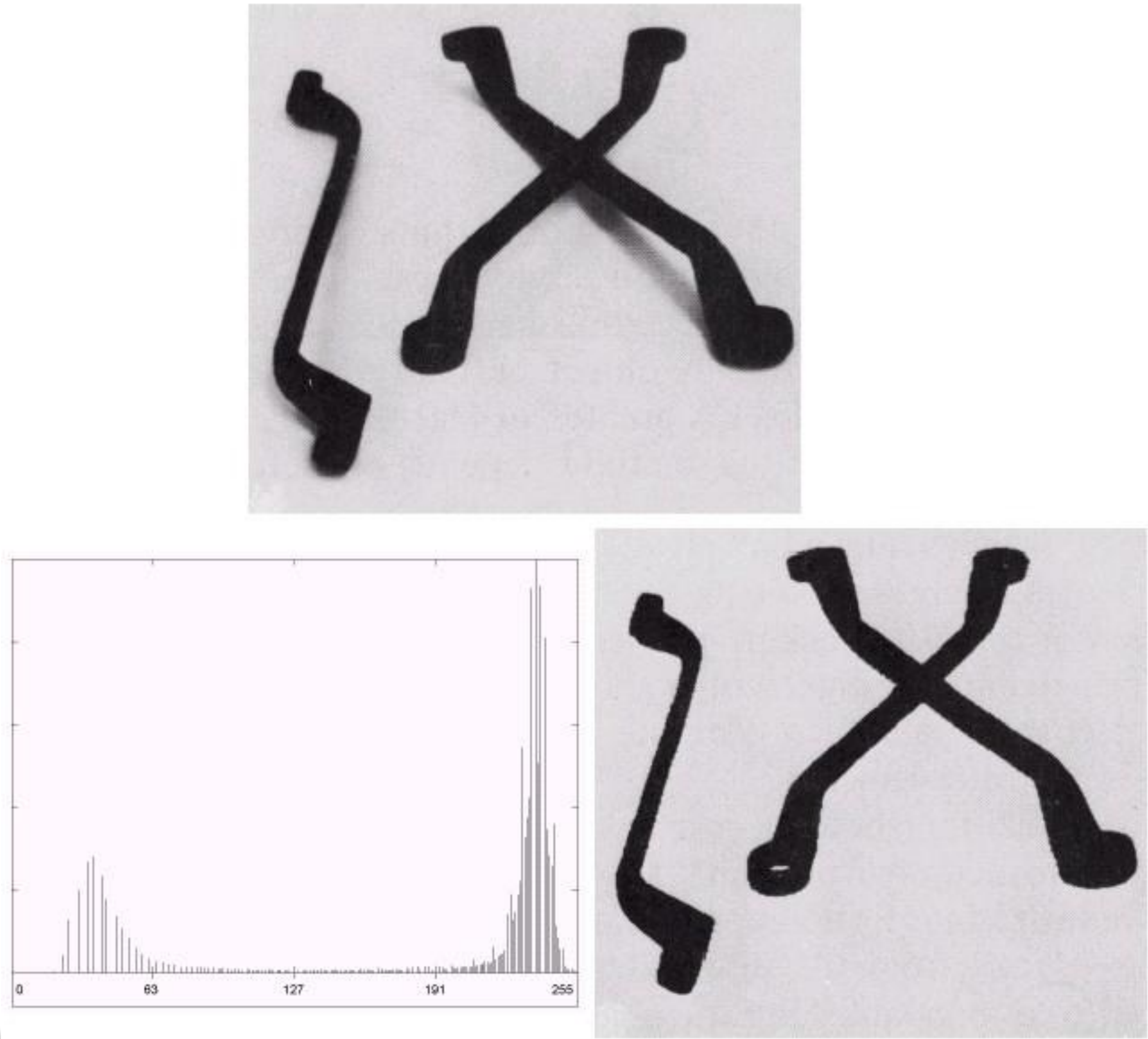


FIGURE 10.27
(a) Computer generated reflectance function.
(b) Histogram of reflectance function.

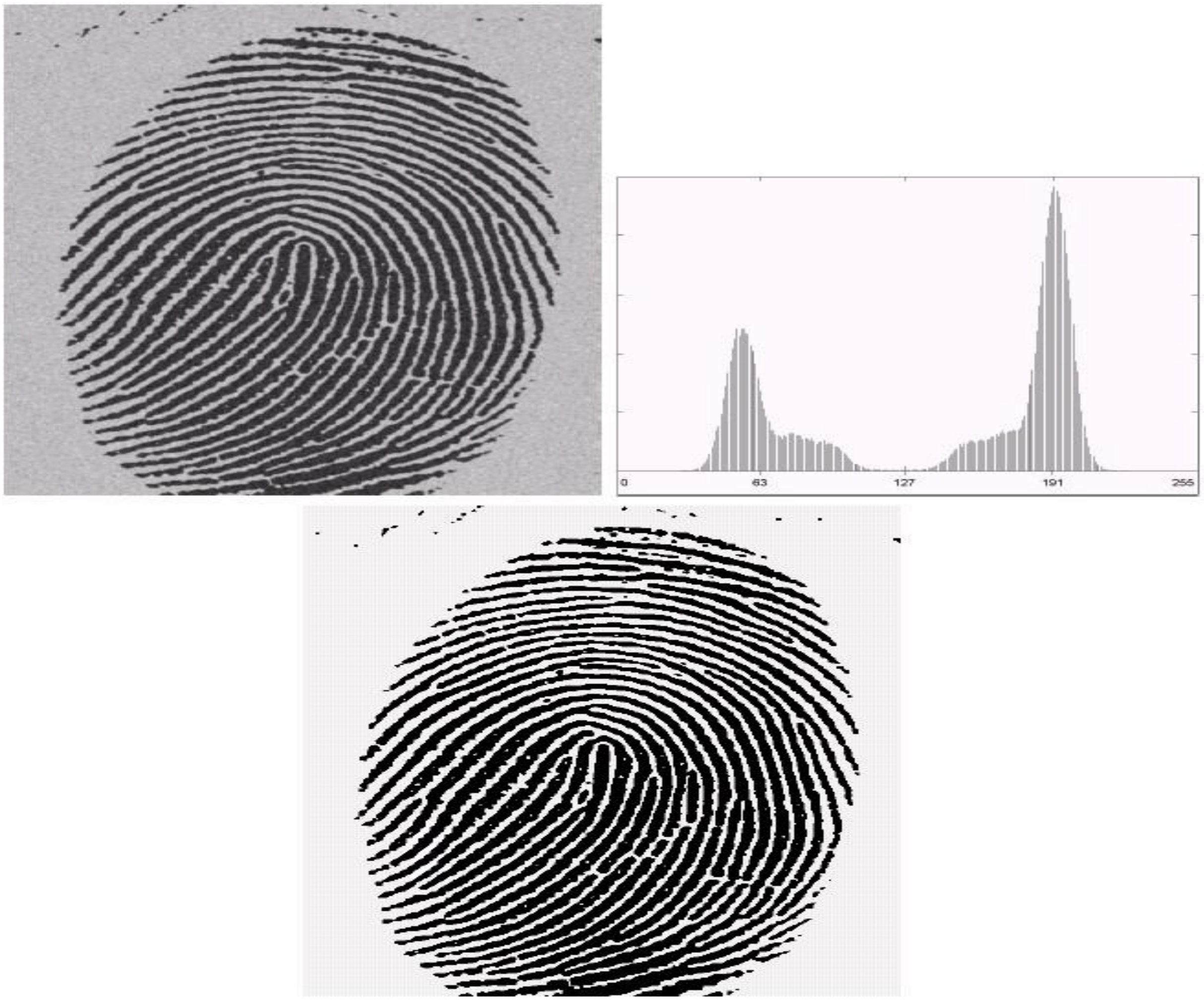
Basic Global Thresholding



a
b c

FIGURE 10.28 (a) Original image. (b) Image histogram. (c) Result of global thresholding with T midway between the maximum and minimum gray levels.

Basic Global Thresholding



a b
c

FIGURE 10.29 (a) Original image. (b) Image histogram. (c) Result of segmentation with the threshold estimated by iteration. (Original courtesy of the National Institute of Standards and Technology.)

Basic Adaptive Thresholding

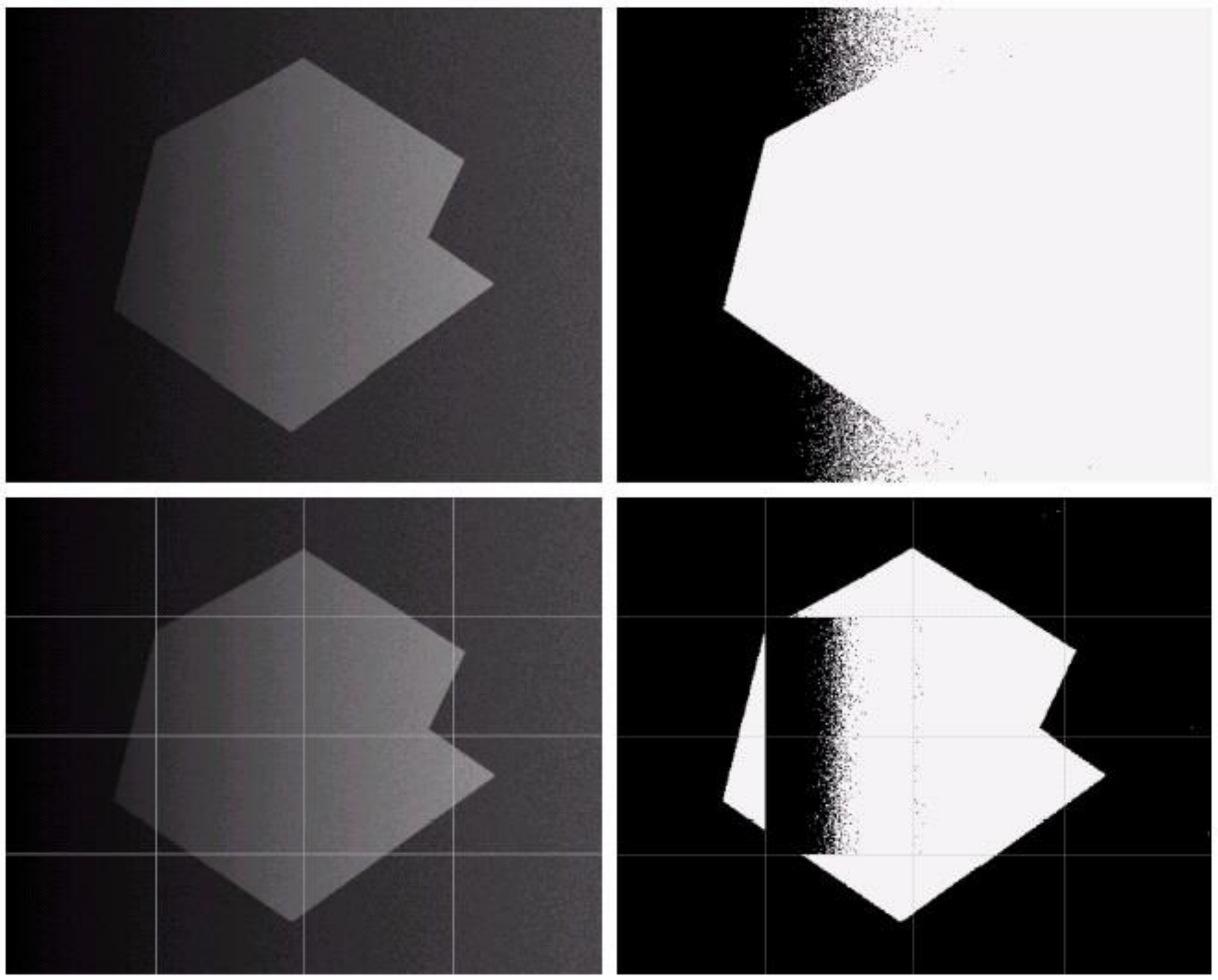
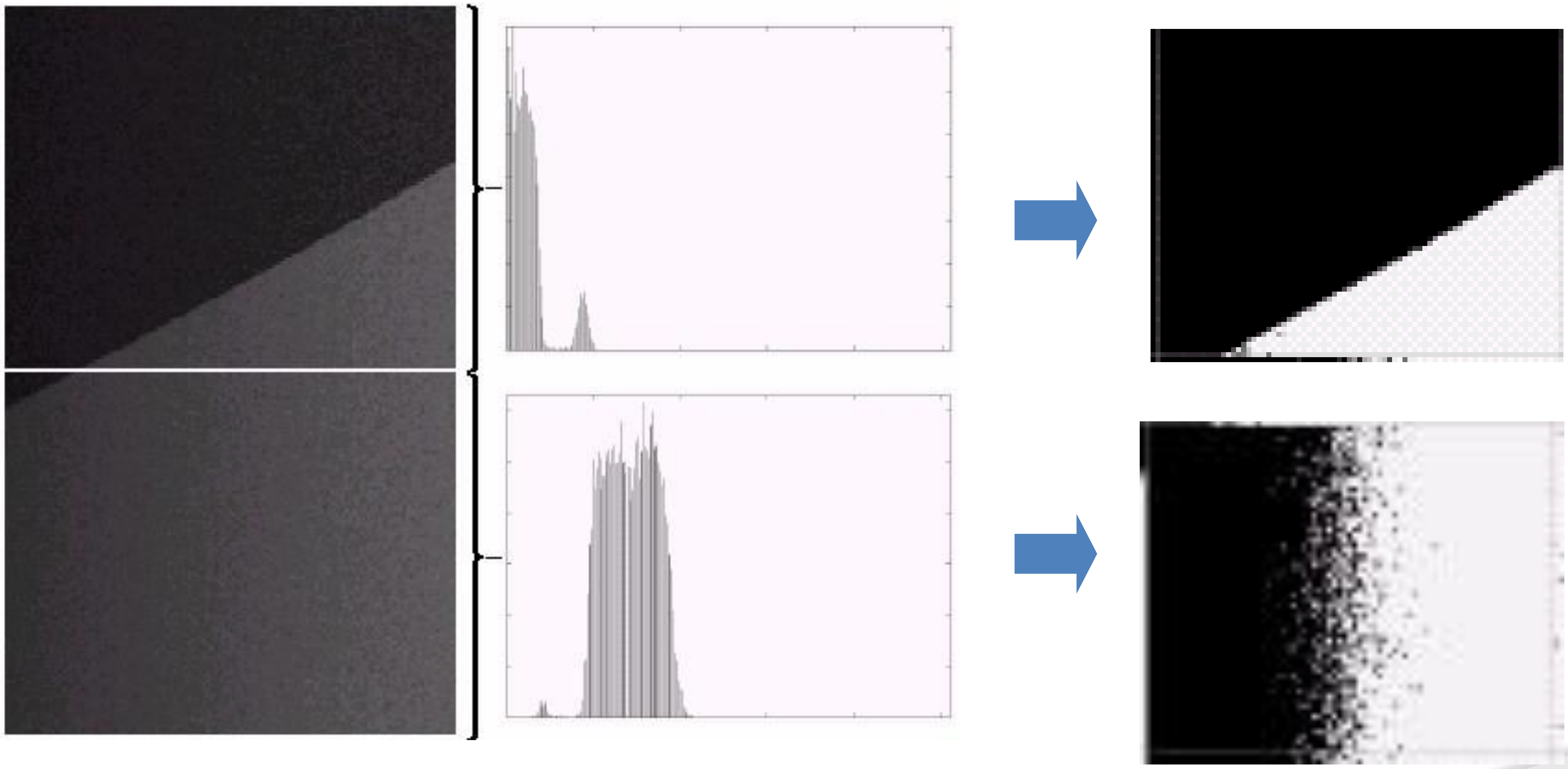


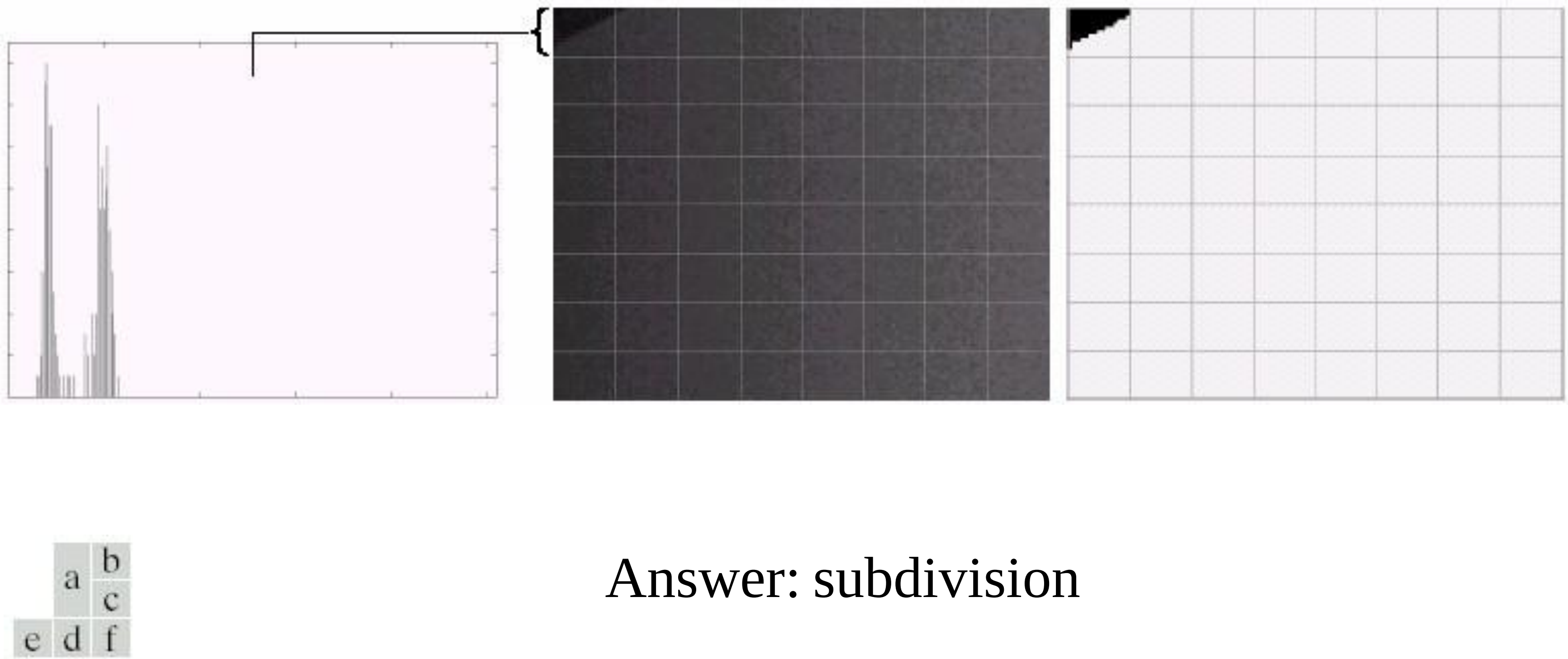
FIGURE 10.30 (a) Original image. (b) Result of global thresholding. (c) Image subdivided into individual subimages. (d) Result of adaptive thresholding.

Basic Adaptive Thresholding



How to solve this problem?

Basic Adaptive Thresholding



Answer: subdivision

FIGURE 10.31 (a) Properly and improperly segmented subimages from Fig. 10.30. (b)–(c) Corresponding histograms. (d) Further subdivision of the improperly segmented subimage. (e) Histogram of small subimage at top, left. (f) Result of adaptively segmenting (d).

Region-Based Segmentation

- Edges and thresholds sometimes do not give good results for segmentation.

- Region-based segmentation is based on the connectivity of similar pixels in a region.

  - Each region must be uniform.
  - Connectivity of the pixels within the region is very important.

- There are two main approaches to region-based segmentation: region growing and region splitting.

Region-Based Segmentation(Contd..)

- Let R represent the entire image region.
- Segmentation is a process that partitions R into sub regions, $R_1, R_2, \ldots, R_n$, such that

$$(a)\ \bigcup_{i=1}^{n} R_i = R$$

$$(b)\ R_i \text{ is a connected region, } i = 1,2,\ldots,n$$

$$(c)\ R_i \cap R_j = \phi \quad \text{for all } i \text{ and } j, i \neq j$$

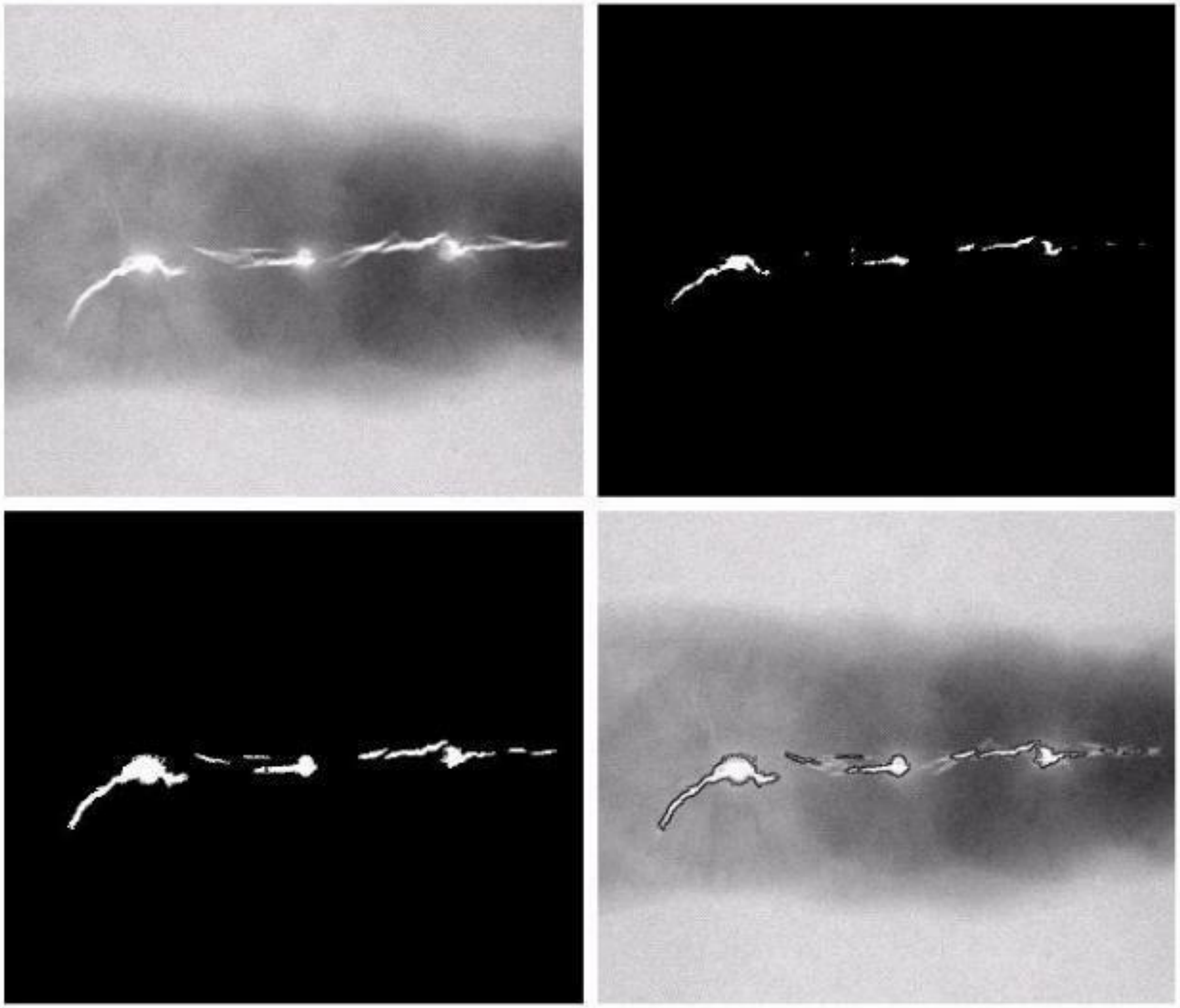
$$(d)\ P(R_i) = \text{TRUE for } i = 1,2,\ldots,n$$

$$(e)\ P(R_i \cup R_j) = \text{FALSE} \quad \text{for any adjacent regions } R_i \text{ and } R_j$$

- where $P(R_k)$: a logical predicate defined over the points in set R_k

For example: $P(R_k)$=TRUE if all pixels in R_k have the same gray level.

Region Growing



a b
c d

FIGURE 10.40 (a) Image showing defective welds. (b) Seed points. (c) Result of region growing. (d) Boundaries of segmented defective welds (in black). (Original image courtesy of X-TEK Systems, Ltd.).

Region Growing

- Fig. a) shows the histogram of Fig. b). It is difficult to segment the defects by thresholding methods. (Applying region growing methods are better in this case.)

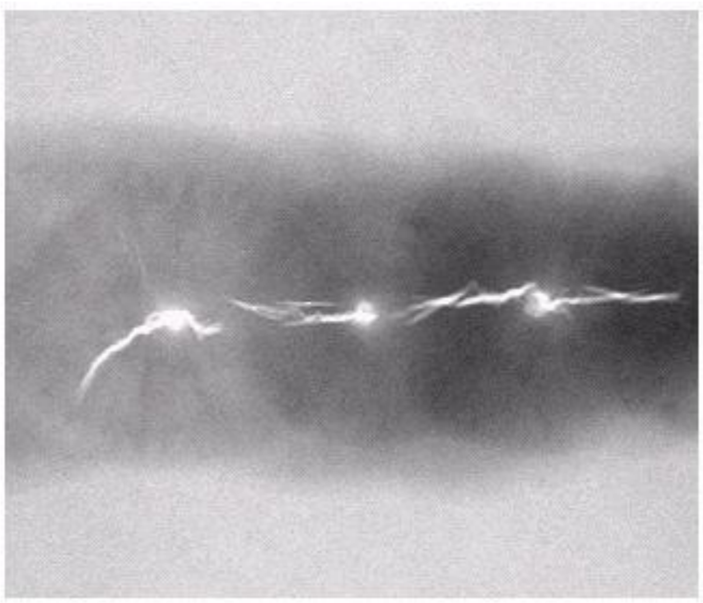


Figure b

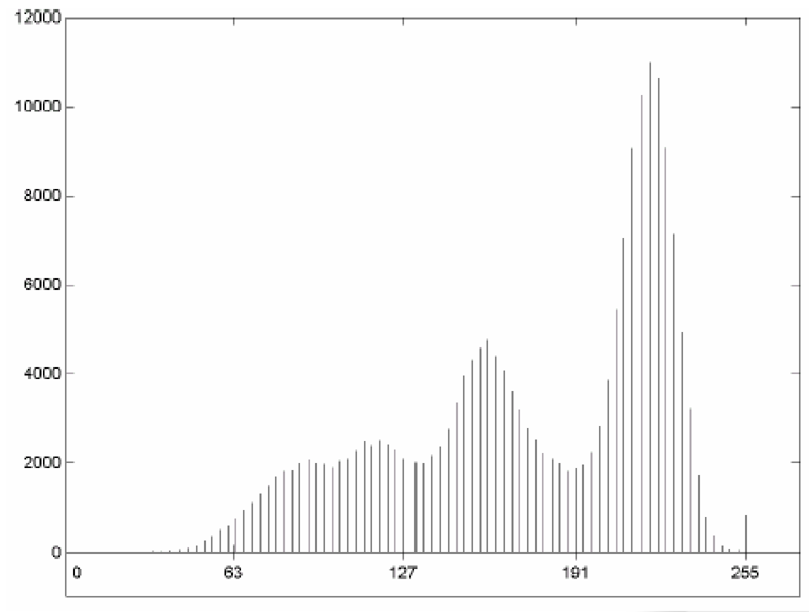


Figure a

Region-Based Segmentation Region Splitting and Merging

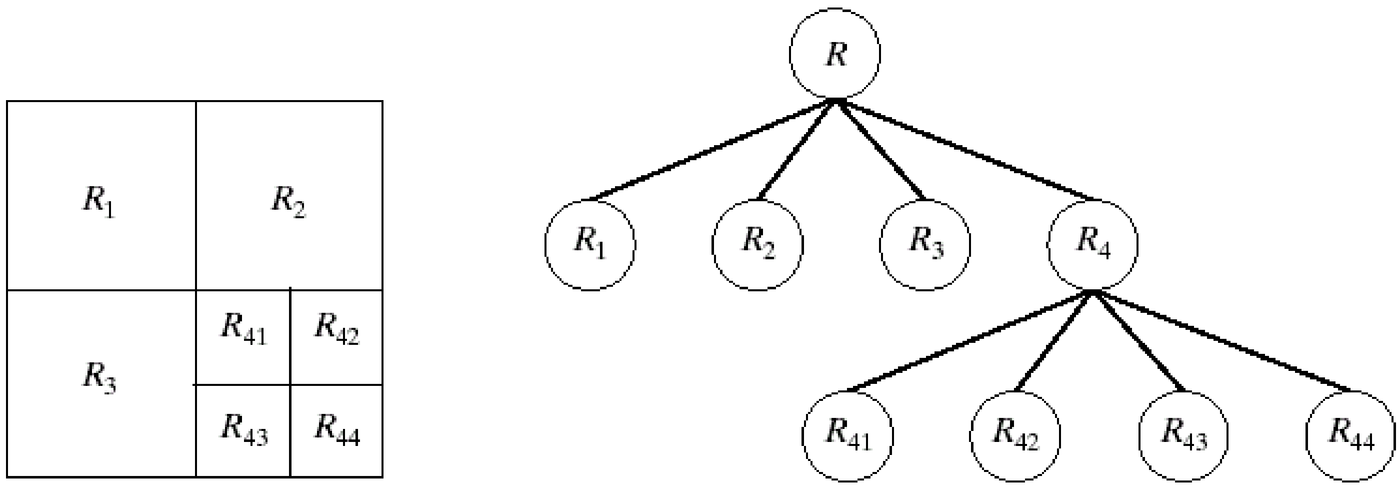
- Region splitting is the opposite of region growing.
  - ✓ First there is a large region (possible the entire image).
  - ✓ Then a predicate (measurement) is used to determine if the region is uniform.
  - ✓ If not, then the method requires that the region be split into two regions.
  - ✓ Then each of these two regions is independently tested by the predicate (measurement).
  - ✓ This procedure continues until all resulting regions are uniform.

Region Splitting

- The main problem with region splitting is determining where to split a region.
- One method to divide a region is to use a quadtree structure.
- Quadtree: a tree in which nodes have exactly four descendants.

a b

FIGURE 10.42
(a) Partitioned image.
(b) Corresponding quadtree.



Region Splitting and Merging

<u>The split and merge procedure:</u>

- Split into four disjoint quadrants any region R_i for which $P(R_i)$ = FALSE.
- Merge any adjacent regions R_j and R_k for which $P(R_j \cup R_k)$ = TRUE. (the quadtree structure may not be preserved)
- Stop when no further merging or splitting is possible.

a b c

FIGURE 10.43 (a) Original image. (b) Result of split and merge procedure. (c) Result of thresholding (a).

References

1. K. Jain, —Fundamentals of Digital Image Processing, Pearson, 3rd Edition, 2004.

- 2. Scott. E. Umbaugh,Digital Image Processing and Analysis, CRC Press, 2nd Edition, 2014.