18EID31

IV/IV B.Tech(Regular/ Supplementary) Degree examinations

November,2022 **Electronics and Instrumentation engineering**

Seventh Semester **Digital Image Processing**

Time:3 Hours Maximum Marks:50 Marks

****

1.	a)	What is need for image processing?	CO1	(BL1)
	b)	List significant advantages of image wavelet transforms?	CO1	(BL1)
	c)	Define image sampling	CO2	(BL1)
	d)	Draw the image negative transform curve?	CO2	(BL1)
	e)	Differentiate image enhancement and image restoration ?	CO2	(BL1)
	f)	What are the advantages of a Wiener filter over an inverse filter ?	CO1	(BL1)
	g)	Define compression ratio?	CO3	(BL1)
	h)	What is variable length code?	CO3	(BL1)
	i)	What is meant by image segmentation ?	CO4	(BL1)
	j)	Name any two boundary descriptors?	CO4	(BL1)
		UNIT-I		
2.	a)	List out the fundamental steps in digital image processing which can be applied to images.	CO1	(BL1)
	b)	Discus about 2D-Discrete Cosine Transform with relevant mathematical functions?	CO1	(BL1)
		(OR)		
3.	a)	Explain about image sampling and quantization process with proper steps?	CO1	(BL1)
	b)	Explain about the basic pixel relationships and distance measures between pixels in digital image	CO1	(BL1)
		UNIT-II		
4.	a)	What is histogram of a digital image .Explain histogram equalization process.	CO2	(BL1)
	b)	Develop image enhancement by point processing method.	CO2	(BL2)
		(OR)		
5.	a)	Discuss about any two types of sharpening filters in frequency domain along with the required expressions.	CO2	(BL1)
	b)	Explain the concept of Laplacian in frequency domain filtering of images	CO2	(BL1)
		UNIT-III		
6.	a)	What is redundancy in image compression ? Discuss the importance of data redundancies?	CO3	(BL1)
	b)	What are different noise models. Explain any three?	CO3	(BL1)
		(OR)		
7.	a)	Explain Error Free compression and lossy compression techniques	CO3	(BL1)
	b)	Explain the run length coding with respect to image compression ?	CO3	(BL1)
		UNIT-IV		
8.	a)	Explain about region based segmentation in detail	CO4	(BL2)
	b)	Analyze the basic Fourier descriptors?	CO4	(BL4)
		(OR)		
9.	a)	Explain the role of thresholding in segmentation	CO4	(BL1)
	b)	Describe shape numbers as boundary descriptors?	CO4	(BL3)

IV/IV B.Tech (Regular/ Supplementary) Degree examinations

Electronics and Instrumentation engineering

November,2022

Seventh Semester

Scheme of evaluation
of

Digital Image Processing

(18EID31)

1. a) What is need for image processing?

Ans Image processing is often viewed as **arbitrarily manipulating an image to achieve an aesthetic standard or to support a preferred reality**. However, image processing is more accurately defined as a means of translation between the human visual system and digital imaging devices.

b) List significant advantages of image wavelet transforms?

Ans The key advantage of the Wavelet Transform compared to the Fourier Transform is the **ability to extract both local spectral and temporal information**. A practical application of the Wavelet Transform is analyzing ECG signals which contain periodic transient signals of interest.

c) Define image sampling?

Ans A sampled image is **an array of sampled image values that are usually arranged in a row-column format**. Each of the indexed array elements is often called a picture element, or pixel for short.

d) Draw the image negative transform curve?

Ans **Image Negatives:**

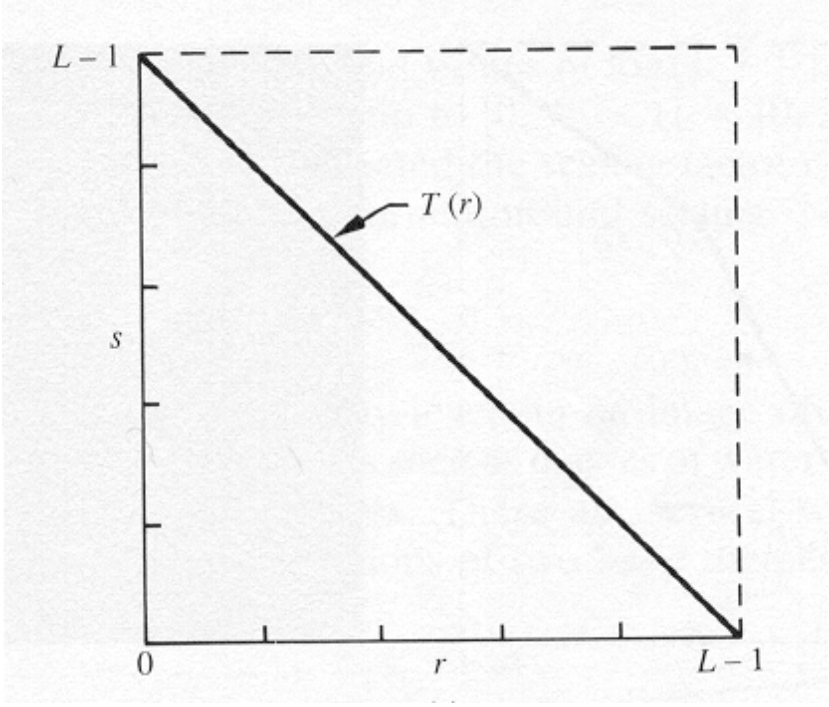
- Are obtained by using the transformation function s=T(r).

[0,L-1] the range of gray levels

$S = L\text{-}1\text{-}r$

Function reverses the order from black to white so that the intensity of the output image decreases as the intensity of the input increases.

Used mainly in medical images and to produce slides of the screen.



e) Differentiate image enhancement and image restoration?

Ans **State the difference between Enhancement and Restoration**

Image restoration and the image enhancement techniques aim at improving the image quality and both the techniques can be performed in both spatial and frequency domains. The difference between image enhancement and image restoration is given below:

Criterion	**Enhancement**	**Restoration**
Result Evaluation	Objective	Subjective
Modeling of Degradation	No	Yes

Use of Prior Knowledge	No	Yes

1. Image enhancement is largely a subjective process which means that it is a heuristic procedure designed to manipulate an image in order to achieve the pleasing aspects of a viewer. On the other hand image restoration involves formulating a criterion of goodness that will yield an optimal estimate of the desired result.

2. In image enhancement the degradation is not usually modeled. Image restoration attempts to reconstruct or recover an image that has been degraded by using the prior knowledge of the degradation. That is restoration techniques try to model the degradation and apply the inverse process in order to recover the original image.

f) What are the advantages of a Wiener filter over an inverse filter ?

Ans The Wiener filtering executes an optimal tradeoff between inverse filtering and noise smoothing. It **removes the additive noise and inverts the blurring simultaneously**. The Wiener filtering is optimal in terms of the mean square error.

g) Define compression ratio?

Ans Data compression ratio is defined as **the ratio between the uncompressed size and compressed size**: Thus, a representation that compresses a file's storage size from 10 MB to 2 MB has a compression ratio of 10/2 = 5, often notated as an explicit ratio, 5:1 (read "five" to "one"), or as an implicit ratio, 5/1.

h) What is variable length code?

Ans **variable-length code** is a code which maps source symbols to a *variable* number of bits.Variable-length codes can allow sources to be compressed and decompressed with *zero* error and still be read back symbol by symbol.

i) What is meant by image segmentation?

Ans Image segmentation is **a commonly used technique in digital image processing and analysis to partition an image into multiple parts or regions, often based on the characteristics of the pixels in the image**.

j) Name any two boundary descriptors?

Ans boundary descriptors, such as **boundary length, diameter, curvature**, etc. - regional descriptors, such as area, perimeter, compactness, mean value, etc.

UNIT-I

2 a) List out the fundamental steps in digital image processing which can be applied to images. 5M

fundamental steps in digital image processing 2M

Explanation 3M

Ans

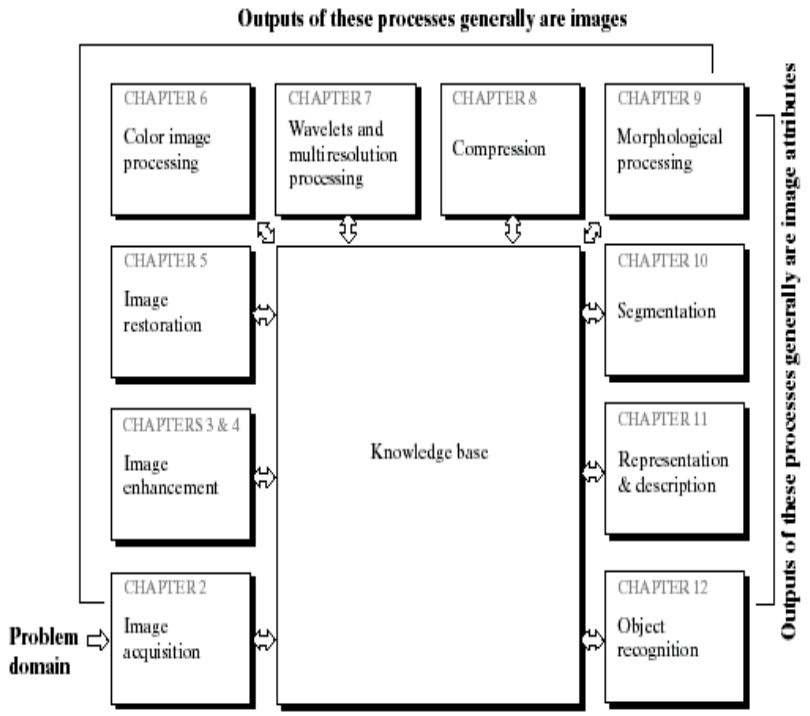


FIGURE 1.23 Fundamental steps in digital image processing.

Important Stages in Image Processing

- Image Acquisition
- Preprocessing
- Segmentation
- Representation and Description
- Recognition and Interpretation
- Knowledge base

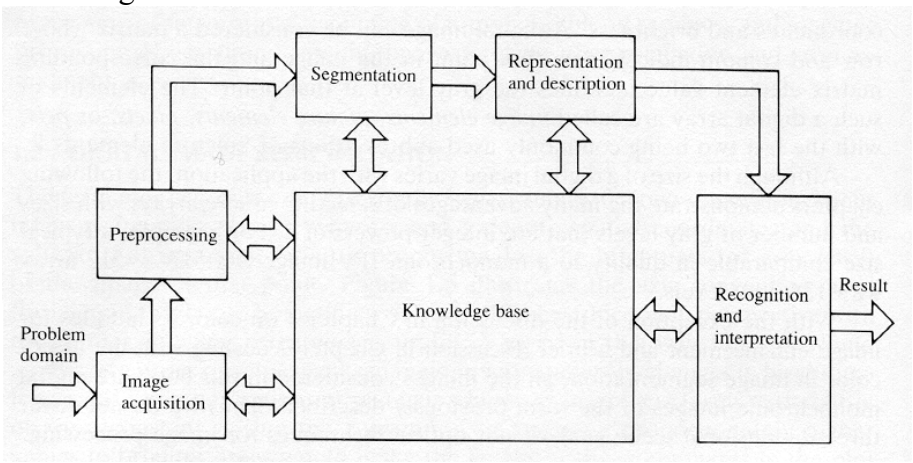


Image Acquistion:

- Imaging sensor & capability to digitize the signal collected by the sensor

- Video camera
- Digital camera
- Conventional camera & analog-to-digital converter.

Preprocessing

- To improve the image to ensure the success of further processes
  - e.g. enhancing contrast
    removing noise
    identifying information-rich areas

Segmentation

- To partition the image into its constituent parts (objects)
  - Autonomous segmentation (very difficult)
    - Can facilitate or disturb subsequent processes
  - Output (representation):
    - Raw pixel data, depicting either boundaries or whole regions (corners vs. texture for example)
    - Need conversion to a form suitable for computer processing
  - (Description)

Representation & Description

- Feature selection (description) deals with extracting:
  - features that result in quantitative information of interest or
  - features that are important for differentiating one class of objects from another

Recognition & Interpretation

- To assign a label to an object based on information provided by the descriptors
- To assign meaning to a group of recognized objects

Knowledge database

- Guides the operation of each processing module and controls the interaction between modules

2 b) Discus about 2D-Discrete Cosine Transform with relevant mathematical functions? 5M

Ans

2D-Discrete Cosine Transform 3M

2D inverse -Discrete Cosine Transform 2M

1-D DISCRETE COSINE TRANSFORM DCT

$$C(u) = a(u)\sum_{x=0}^{N-1} f(x)\cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

$$u = 0,1,\ldots,N-1$$

$$a(u) = \begin{cases} \sqrt{\frac{1}{N}} & u = 0 \\ \sqrt{\frac{2}{N}} & u = 1,\ldots,N-1 \end{cases}$$

Discrete Cosine Transform (DCT)

- DCT is proposed as an alternative approach to DFT
  - To reduce the complexity in computing complex number
- DCT -> DFT keeping only cosine term
- Types of DCT
  - 4 types of DCT are proposed in the literature
    - DCT-I, DCT-II, DCT-III, and DCT-IV

1-D INVERSE DISCRETE COSINE TRANSFORM
IDCT

$$f(x) = \sum_{u=0}^{N-1} a(u)C(u)\cos\left[\frac{(2x+1)u\pi}{2N}\right]$$

2-D DISCRETE COSINE TRANSFORM
DCT

$$C(u,v) = a(u)a(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} a(u)a(v)C(u,v)\cos\left[\frac{(2x+1)u\pi}{2N}\right]\cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

$$u,v = 0,1,\ldots,N-1$$

ADVANTAGES

- Notice that the DCT is a real transform.
- The DCT has excellent energy compaction properties.
- There are fast algorithms to compute the DCT similar to the FFT.

(OR)

3 a) Explain about image sampling and quantization process with proper steps? 5M

Ans

image sampling and quantization process 3M

quantization process 2M

The result of sampling and quantization is a matrix of real numbers .

Assume that an image f(x, y) is sampled so that the resulting digital image has M rows and N columns . The values of the coordinates (x, y) now become *discrete quantities. For notational* clarity and convenience , we shall use integer values for these discrete coordinates.

Thus, the values of the coordinates at the origin are (x, y)=(0, 0).

The next coordinate values along the first row of the image are represented as (x, y)=(0, 1). It is important to keep in mind that the notation (0, 1) is used to signify the second sample along the first row. It does *not mean that these are* the actual values of physical coordinates when the image was sampled.

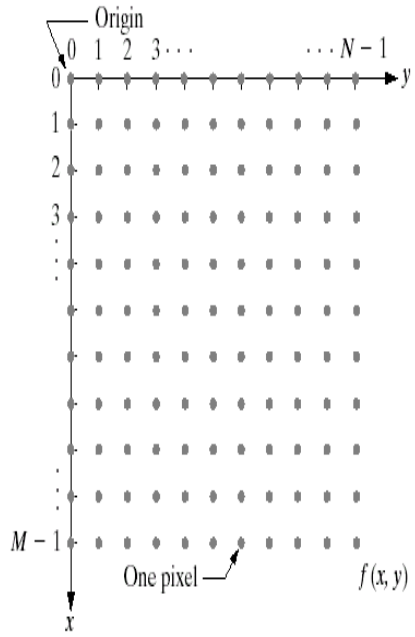


FIGURE 2.18
Coordinate convention used in this book to represent digital images.

$$\mathbf{A} = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}.$$

$$a_{ij} = f(x = i, y = j) = f(i, j)$$

Sampling is the principal factor determining the *spatial resolution of an image*

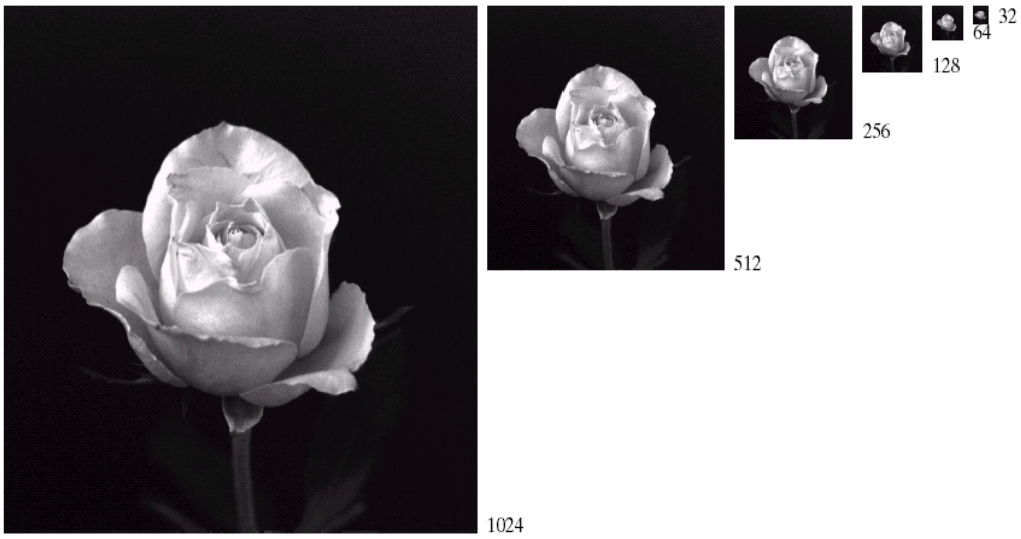


FIGURE 2.19 A 1024 × 1024, 8-bit image subsampled down to size 32 × 32 pixels. The number of allowable gray levels was kept at 256.

subsampling the 1024*1024 image. The subsampling was accomplished by deleting the appropriate number of rows and columns from the original image. For example, the 512*512 image was obtained by deleting every other row and column from the 1024*1024 image. The 256*256 image was generated by deleting every other row and column in the 512*512 image, and so on. The number of allowed gray levels was kept at 256.

Nonuniform Sampling & Quantization:

- An adaptive sampling scheme can improve the appearance of an image, where the sampling would consider the characteristics of the image.
  - i.e. fine sampling in the neighborhood of sharp gray-level transitions (e.g. boundaries)
  - Coarse sampling in relatively smooth regions
- Considerations: boundary detection, detail content
- Similarly: nonuniform quantization process
- In this case:
  - few gray levels in the neighborhood of boundaries
  - more in regions of smooth gray-level variations (reducing thus false contours)

3 b) Explain about the basic pixel relationships and distance measures between pixels in digital image? 5M

Ans

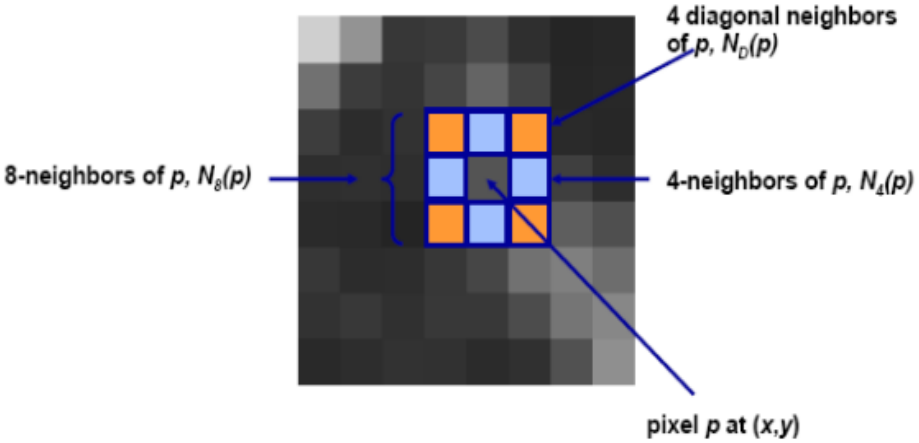
basic pixel relationships between pixels 3M

distance measures between pixels 2M

- f(x,y): digital image
- Pixels: p,q
- Subset of pixels of f(x,y): S
- A pixel p at (x,y) has 2 horizontal and 2 vertical neighbors:
- (x+1,y), (x-1,y), (x,y+1), (x,y-1)
- This set of pixels is called the 4-neighbors of p: $N_4(p)$

- Each pixel is a unit distance from (x, y), and some of the neighbours of p lie outside the digital image if (x, y) is on the border of the image.
- The 4 diagonal neighbors of p are: ($N_D(p)$)
  - (x+1,y+1), (x+1,y-1), (x-1,y+1), (x-1,y-1)
  - $N_4(p) + N_D(p) \rightarrow N_8(p)$: the 8-neighbors of p

Neighbors of a pixel



- **continuity**
- Two pixels are connected if:
  - They are neighbors (i.e. adjacent in some sense -- e.g. $N_4(p)$, $N_8(p)$, …)
  - Their gray levels satisfy a specified criterion of similarity (e.g. equality, …)
- V is the set of gray-level values used to define adjacency (e.g. V={1} for adjacency of pixels of value 1).
- **Adjacency**
- We consider three types of adjacency:
  - 4-adjacency: two pixels p and q with values from V are 4-adjacent if q is in the set $N_4(p)$
  - 8-adjacency : p & q are 8- adjacent if q is in the set $N_8(p)$
  - m-adjacency: p & q with values from V are m-adjacent if
    - q is in $N_4(p)$ or
    - q is in $N_D(p)$ and the set $N_4(p) \cap N_4(q)$ has no pixels with values from V

Let V: a set of intensity values used to define adjacency and connectivity.
p In a binary image, V = {1}, if we are referring to adjacency of pixels with value 1.
p In a gray-scale image, the idea is the same, but V typically contains more elements, for example, V = {180, 181, 182, …, 200}
p If the possible intensity values 0 – 255, V set can be any subset of these 256 values

Distance measures

- Euclidean distance between p and q:
  $De(p,q) = [(x-s)^2 + (y-t)^2]^{1/2}$
- $D4$ distance (also called *city-block distance*):
  $D4(p,q) = |x-s| + |y-t|$
- $D8$ distance (also called *chessboard distance*) :
  $D8(p,q) = \max(|x-s|, |y-t|)$

Example:

Compute the distance between the two pixels using the three distances :

q:(1,1)

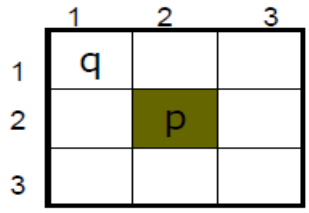
P: (2,2)

Euclidian distance : $((1-2)^2+(1-2)^2)^{1/2}$ = sqrt(2).

D4(City Block distance): |1-2| +|1-2| =2

D8(chessboard distance) : max(|1-2|,|1-2|)= 1

(because it is one of the 8-neighbors)



UNIT-II

4 a) What is histogram of a digital image .Explain histogram equalization process?. 5M
histogram processing 2M
Histogram equalization. 4M

Ans

Consider for a moment continuous functions, and let the variable r represent the gray levels of the image to be enhanced. We assume that r has been normalized to the interval [0, 1], with r=0 representing black and r=1 representing white. Later, we consider a discrete formulation and allow pixel values to be in the interval [0, L-1]. For any r satisfying the aforementioned conditions, we focus attention on transformations of the form

$$s=T(r) \qquad 0 \le r \le 1$$

that produce a level s for every pixel value r in the original image. For reasons that will become obvious shortly, we assume that the transformation function T(r) satisfies the

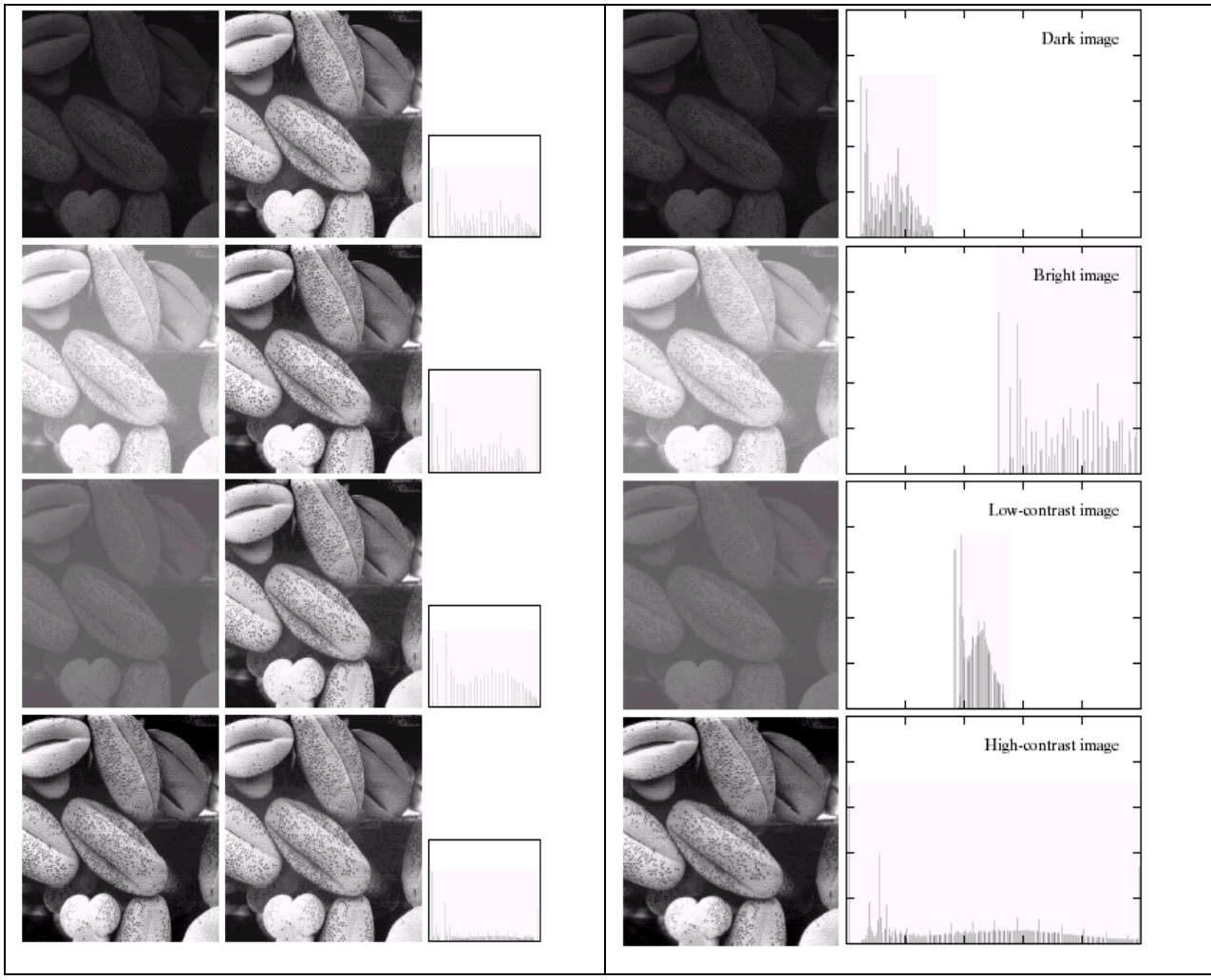
following conditions:
(a) T(r) is single-valued and monotonically increasing in the interval $0 \le r \le 1$; and

(b) $0 \le T(r) \le 1$ for $0 \le r \le 1$.

The requirement in (a) that T(r) be single valued is needed to guarantee that the inverse transformation will exist, and the monotonicity condition preserves the increasing order from black to white in the output image.A transformation function that is not monotonically increasing could result in at least a section of the intensity range being inverted, thus producing some inverted gray levels in the output image. Finally, condition (b) guarantees that the output gray levels will be in the same range as the input levels. Figure 4.1 gives an example of a transformation function that satisfies these two conditions.The inverse transformation from s back to r is denoted
$r=T^{-1}(s)$ for $0 \le s \le 1$.
It can be shown by example that even if T(r) satisfies conditions (a) and (b), it is possible that the corresponding inverse T^{-1} (s) may fail to be single valued.

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

Histogram equalization(HE) results are similar to contrast stretching but offer the advantage of full automation, since HE automatically determines a transformation function to produce a new image with a uniform histogram



4 b) Develop image enhancement by point processing method. 5M

Ans

Basic Gray Level Transformations 2M

Explanation 3M

- **Enhancement** at any point in an image depends only on the gray level at that point, techniques in this category often are referred to as ***point processing.***
- **Mask processing or filtering**: when the values of f in a predefined neighborhood of (x , y) determine the value of g at (x, y).
  - Through the use of masks (or kernels, templates, or windows, or filters).

Some Basic Gray Level Transformations

Three basic types of functions used frequently for image enhancement:

1. linear (negative and identity transformations),
2. logarithmic (log and inverse-log transformations), and
3. power-law (*nth power and nth root transformations).*

The identity function is the trivial case in which output intensities are identical to input intensities.

Some Simple Intensity Transformations:

Piecewise-Linear Transformation Functions:

Contrast stretching

Gray-level slicing

Bit-plane slicing

- Implemented via Look-Up Tables (LUT) where values of T are stored in a 1-D array (for 8-

bit, LUT will have 256 values)

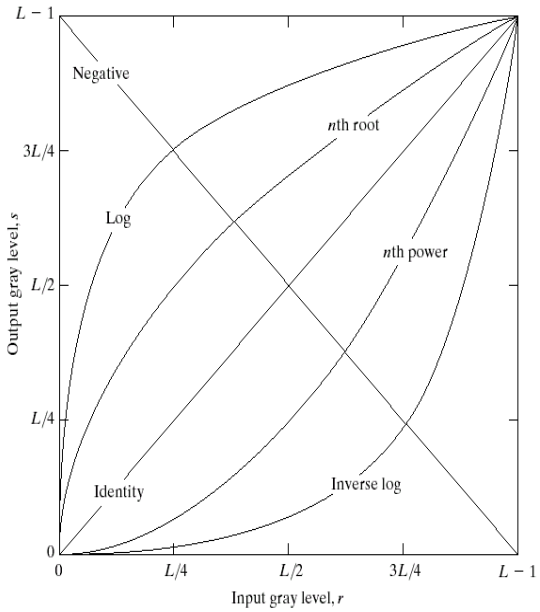


FIGURE 3.3 Some basic gray-level transformation functions used for image enhancement.

Image Negatives:

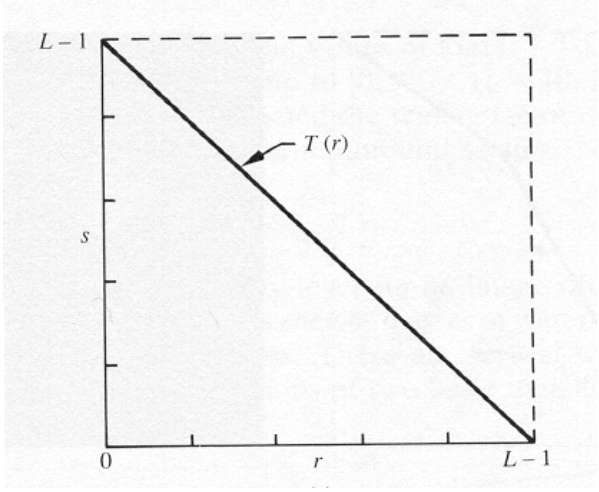
- Are obtained by using the transformation function s=T(r).

[0,L-1] the range of gray levels

$S = L\text{-}1\text{-}r$

Function reverses the order from black to white so that the intensity of the output image decreases as the intensity of the input increases.

Used mainly in medical images and to produce slides of the screen.



Log Transformations

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

c: constant

Assume r ≥ 0

- This transformation maps a narrow range of low gray-level values in the input image into a wider range of output levels . The opposite is true of higher values of input levels . We would use a transformation of this type to expand the values of dark pixels in an image while compressing the higher-level values .
- The opposite is true of the inverse log transformation.

Power-Law Transformations:

Gamma correction is important if displaying an image accurately on a computer screen is of concern. Images that are not corrected properly can look either bleached out, or, what is more

likely, too dark . Trying to reproduce colours accurately also requires some knowledge of gamma correction because varying the value of gamma correction changes not only the brightness, but also the ratios of red to green to blue. Gamma correction has become increasingly important in the past few years, as use of digital images for commercial purposes over the Internet has increased. It is not unusual that images created for a popular Web site will be viewed by millions of people, the majority of whom will have different monitors and/or monitor settings. Given these constraints, a reasonable approach when storing images in a Web site is to preprocess the images with a gamma that represents an "average" of the types of monitors and computer systems that one expects in the open market at any given point in time.

(OR)

5 a) Discuss about any two types of sharpening filters in frequency domain along with the required expressions. 5M

Ans Sharpening (Highpass) Filtering: 2M

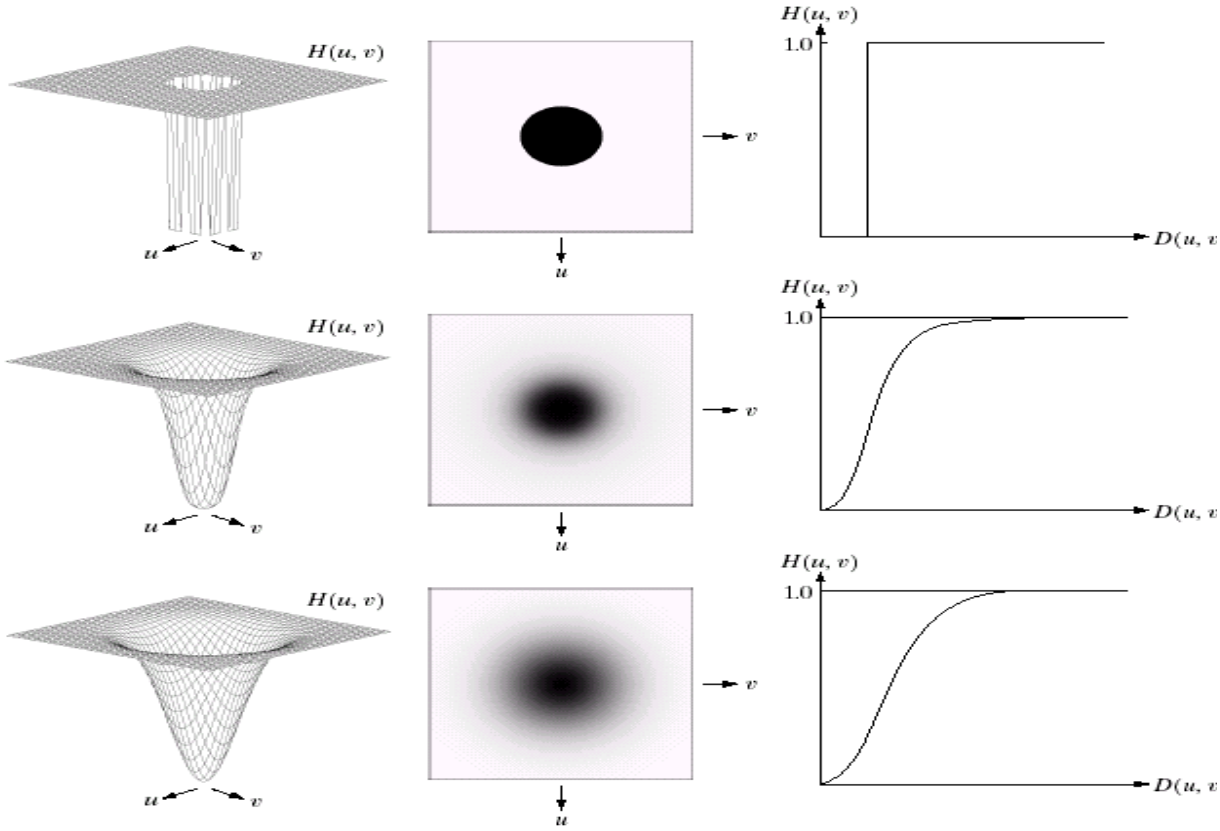
Butterworth Filter (Highpass) 3M

Sharpening (Highpass) Filtering:

- Image sharpening can be achieved by a highpass filtering process, which attenuates the low-frequency components without disturbing high-frequency information.

$$H_{hp}(u,v) = 1 - H_{lp}(u,v)$$

Zero-phase-shift filters: radially symmetric and completely specified by a cross section.



a b c
d e f
g h i

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.

Ideal Filter (Highpass):

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

- This filter is the opposite of the ideal low pass filter.

Effects of ideal high pass filters:

1. Ringing effect
2. Edge distortion (ie distorted ,thickend object boundaries)

Both effects are decreased as the cutoff frequency increases.

Butterworth Filter (Highpass)

High-frequency emphasis:

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

Adding a constant to a highpass filter to preserve the low-frequency components.

5 b) Explain the concept of Laplacian in frequency domain filtering of images? 5M

Laplacian in frequency domain filtering 2M

Explantion 3M

Ans

Laplacian (recap)

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

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

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

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

Laplacian in the FD

- It can be shown that:

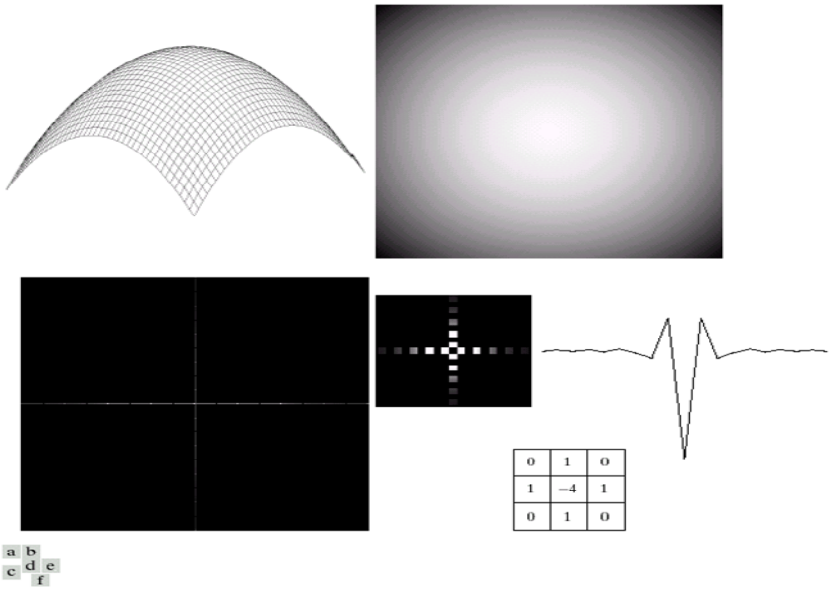
$$\Im\left[\nabla^2 f(x,y)\right] = -(u^2+v^2)F(u,v)$$

- The Laplacian can be implemented in the Fourier Domain by using the filter

$$H(u,v) = -(u^2+v^2)$$

- F T pair:

$$\nabla^2 f(x,y) \Leftrightarrow -[(u-M/2)^2 + (v-N/2)^2]F(u,v)$$



a b c d e f

FIGURE 4.27 (a) 3-D plot of Laplacian in the frequency domain. (b) Image representation of (a). (c) Laplacian in the spatial domain obtained from the inverse DFT of (b). (d) Zoomed section of the origin of (c). (e) Gray-level profile through the center of (d). (f) Laplacian mask used in Section 3.7.

UNIT-III

6 a) What is redundancy in image compression ? Discuss the importance of data redundancies? 5M
Coding redundancy 2M
Interpixel redundancy 3M

Ans Data redundancy is a central issue in digital image compression. It is not an abstract concept but a mathematically quantifiable entity. If n_1 and n_2 denote the number of information-carrying units in two data sets that represent the same information, the relative data redundancy R_D of the first data set (the one characterized by n_1) can be defined as

$$R_D = 1 - (1/C_R)$$
$$C_R = n_1/n_2$$

For the case $n_2 = n_1$, $C_R = 1$ and $R_D = 0$, indicating that (relative to the second data set) the first representation of the information contains no redundant data. When $n_2 \ll n_1$, $C_R \rightarrow \infty$ and $R_D \rightarrow 1$, implying significant compression and highly redundant data. Finally, when $n_2 \gg n_1$, $C_R \rightarrow 0$ and $R_D \rightarrow \infty$, indicating that the second data set contains much more data than the original representation. This, of course, is the normally undesirable case of data expansion. In general, C_R and R_D lie in the open intervals $(0, \infty)$ and $(-\infty, 1)$, respectively. A practical compression ratio, such as 10 (or 10:1), means that the first data set has 10 information carrying units (say, bits) for every 1 unit in the second or compressed data set. The corresponding redundancy of 0.9 implies that 90% of the data in the first data set is redundant.

In digital image compression, three basic data redundancies can be identified and exploited: **coding redundancy**, **interpixel redundancy**, and **psychovisual redundancy**. Data compression is achieved when one or more of these redundancies are reduced or eliminated.

Coding Redundancy:

In this, we utilize formulation to show how the gray-level histogram of an image also can provide a great deal of insight into the construction of codes to reduce the amount of data used to represent it.

Let us assume, once again, that a discrete random variable r_k in the interval [0, 1] represents the gray levels of an image and that each r_k occurs with probability $p_r(r_k)$.

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

where L is the number of gray levels, n_k is the number of times that the kth gray level appears in the image, and n is the total number of pixels in the image. If the number of bits used to represent each value of r_k is $l(r_k)$, then the average number of bits required to represent each pixel is

$$L_{avg} = \sum_{k=0}^{L-1} l(r_k) p_r(r_k).$$

That is, the average length of the code words assigned to the various gray-level values is found by summing the product of the number of bits used to represent each gray level and the probability that the gray level occurs. Thus the total number of bits required to code an M X N image is MNL_{avg}.

Interpixel Redundancy:

Consider the images shown in Figs. 1.1(a) and (b). As Figs. 1.1(c) and (d) show, these images have virtually identical histograms. Note also that both histograms are trimodal, indicating the presence of three dominant ranges of gray-level values. Because the gray levels in these images are not equally probable, variable-length coding can be used to reduce the coding redundancy that would result from a straight or natural binary encoding of their pixels. The coding process, however, would not alter the level of correlation between the pixels within the images. In other words, the codes used to represent the gray levels of each image have nothing to do with the correlation between pixels. These correlations result from the structural or geometric relationships between the objects in the image.

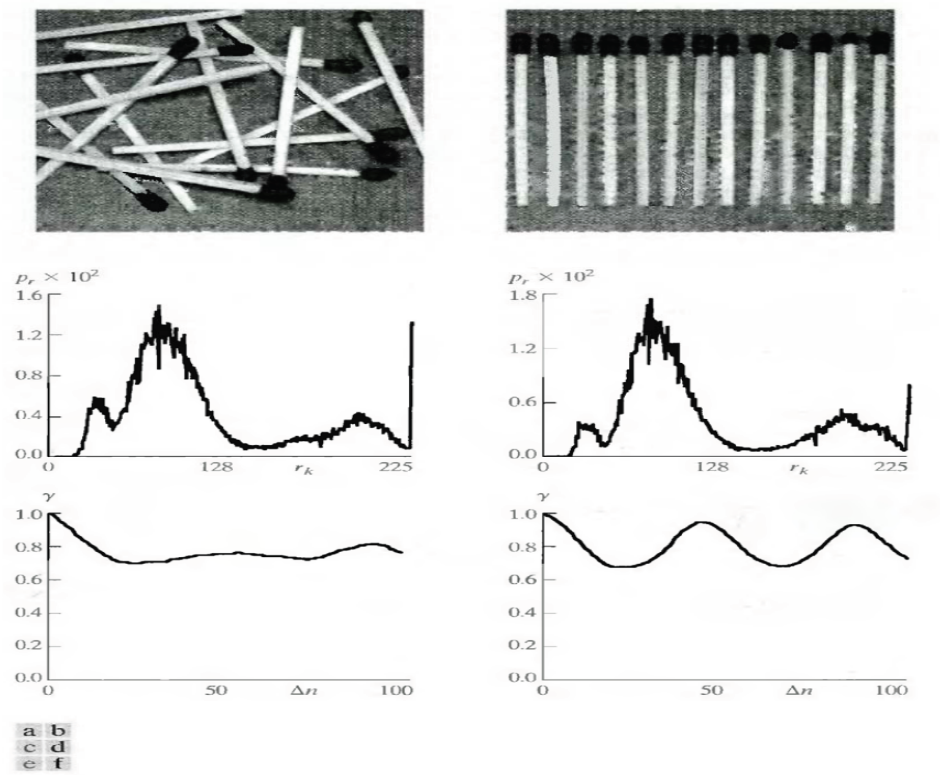


Fig.1.1 Two images and their gray-level histograms and normalized autocorrelation coefficients along one line.

Figures 1.1(e) and (f) show the respective autocorrelation coefficients computed along one line of each image.

$$\gamma(\Delta n) = \frac{A(\Delta n)}{A(0)}$$

where

$$A(\Delta n) = \frac{1}{N - \Delta n} \sum_{y=0}^{N-1-\Delta n} f(x, y) f(x, y + \Delta n).$$

The scaling factor in Eq. above accounts for the varying number of sum terms that arise for each integer value of Δn. Of course, Δn must be strictly less than N, the number of pixels on a line. The variable x is the coordinate of the line used in the computation. Note the dramatic difference between the shape of the functions shown in Figs. 1.1(e) and (f). Their shapes can be qualitatively related to the structure in the images in Figs. 1.1(a) and (b).This relationship is particularly noticeable in Fig. 1.1 (f), where the high correlation between pixels separated by 45 and 90 samples can be directly related to the spacing between the vertically oriented matches of Fig. 1.1(b). In addition, the adjacent pixels of both images are highly correlated. When Δn is 1, γ is 0.9922 and 0.9928 for the images of Figs. 1.1 (a) and (b), respectively. These values are typical of most properly sampled television images.

Psychovisual Redundancy:

The brightness of a region, as perceived by the eye, depends on factors other than simply the light reflected by the region. For example, intensity variations (Mach bands) can be perceived in an area of constant intensity. Such phenomena result from the fact that the eye does not respond with equal sensitivity to all visual information. Certain information simply has less relative importance than other information in normal visual processing. This information is said to be psychovisually redundant. It can be eliminated without significantly impairing the quality of image perception.

That psychovisual redundancies exist should not come as a surprise, because human perception of the information in an image normally does not involve quantitative analysis of every pixel value in the image. In general, an observer searches for distinguishing features such as edges or textural regions and mentally combines them into recognizable groupings. The brain then correlates these groupings with prior knowledge in order to complete the image interpretation process. Psychovisual redundancy is fundamentally different from the redundancies discussed earlier. Unlike coding and interpixel redundancy, psychovisual redundancy is associated with real or quantifiable visual information. Its elimination is possible only because the information itself is not essential for normal visual processing. Since the elimination of psychovisually redundant data results in a loss of quantitative information, it is commonly referred to as quantization.

6 b) What are different noise models. Explain any three? 5M

Ans Noise models 2M

Explanation 3M

Noise models

Spatial noise descriptor: statistical behaviour of the intensity values in the noise component. Random variables characterized by a Probability Density Function (PDF)

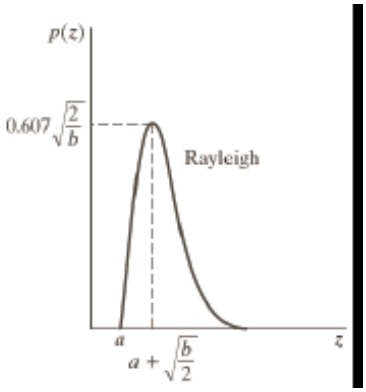
⇒ **Some important noise probability density function.**

PDF of a Gaussian random variable z: $p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\bar{z})^2/2\sigma^2}$

Rayleigh Noise

$$p(z) = \begin{cases} \frac{2}{b}(z-a)e^{(z-a)^2/b} & \text{for } z \geq a \\ 0 & \text{for } z < a \end{cases}$$

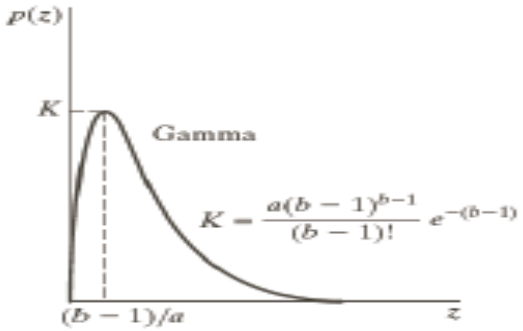
Mean: $\bar{z} = a + \sqrt{\pi b/4}$ Variance: $\sigma^2 = \frac{b(4-\pi)}{4}$



Erlang (Gamma) Noise

$$p(z) = \begin{cases} \frac{a^b z^{b-1}}{(b-1)!} e^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases} \quad a > 0, \text{ b positive integer}$$

Mean: $\bar{z} = \frac{b}{a}$ Variance: $\sigma^2 = \frac{b}{a^2}$

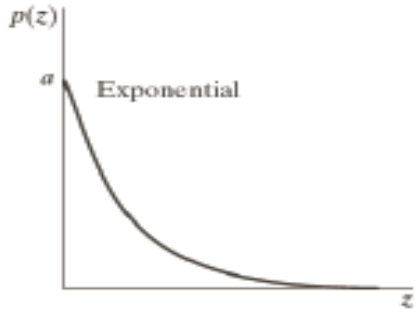


Exponential Noise

$$p(z) = \begin{cases} ae^{-az} & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases}$$

(cf. Erlang noise with $b = 1$)

$a > 0$



(OR)

7	**a)**	**Explain Error Free compression and lossy compression techniques**	**5M**
		Explain Error Free compression	**3M**
		lossy compression techniques	**2M**

Ans **Lossless Predictive Coding:**

The error-free compression approach does not require decomposition of an image into a collection of bit planes. The approach, commonly referred to as lossless predictive coding, is based on eliminating the interpixel redundancies of closely spaced pixels by extracting and coding only the new information in each pixel. The new information of a pixel is defined as the difference between the actual and predicted value of that pixel.

Figure 8.1 shows the basic components of a lossless predictive coding system. The system consists of an encoder and a decoder, each containing an identical predictor. As each successive pixel of the input image, denoted f_n, is introduced to the encoder, the predictor generates the anticipated value of that pixel based on some number of past inputs. The output of the predictor is then rounded to the nearest integer, denoted $\hat{f}_n$ and used to form the difference or prediction error which is coded using a variable-length code (by the symbol encoder) to generate the next element of the compressed data stream.

$$e_n = f_n - \hat{f}_n,$$

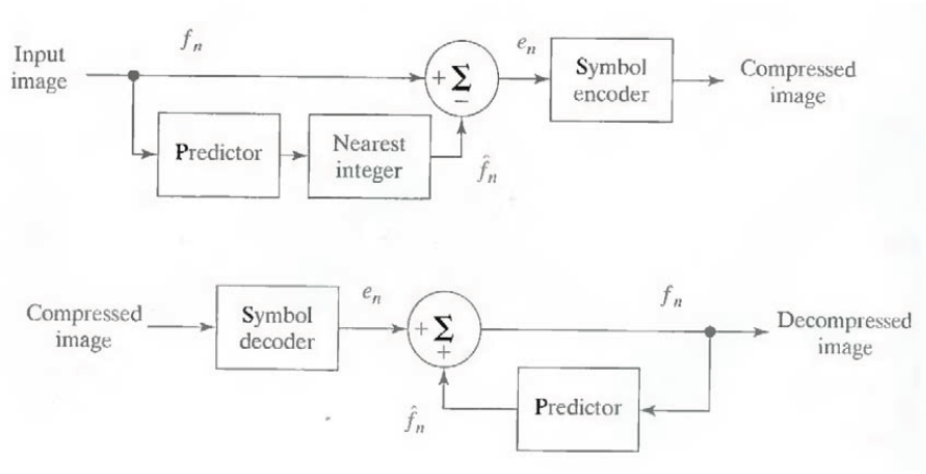


Fig.8.1 A lossless predictive coding model: (a) encoder; (b) decoder

The decoder of Fig. 8.1 (b) reconstructs en from the received variable-length code words and performs the inverse operation

$$f_n = e_n + \hat{f}_n.$$

Various local, global, and adaptive methods can be used to generate $\hat{f}_n$. In most cases, however, the prediction is formed by a linear combination of m previous pixels. That is,

$$\hat{f}_n = \text{round}\left[\sum_{i=1}^{m} \alpha_i f_{n-i}\right]$$

where m is the order of the linear predictor, round is a function used to denote the rounding or nearest integer operation, and the α_i for i = 1,2,..., m are prediction coefficients. In raster scan applications, the subscript n indexes the predictor outputs in accordance with their time of occurrence. That is, f_n, $f^\wedge_n$ and e_n in Eqns. above could be replaced with the more explicit notation f (t), f^(t), and e (t), where t represents time. In other cases, n is used as an index on the spatial coordinates and/or frame number (in a time sequence of images) of an image. In 1-D linear predictive coding, for example, Eq. above can be written as

$$\hat{f}_n(x, y) = \text{round}\left[\sum_{i=1}^{m} \alpha_i f(x, y - i)\right]$$

LOSSY COMPRESSION

Lossless compression algorithms do not deliver compression ratios that are high enough. Hence, most multimedia compression algorithms are lossy.

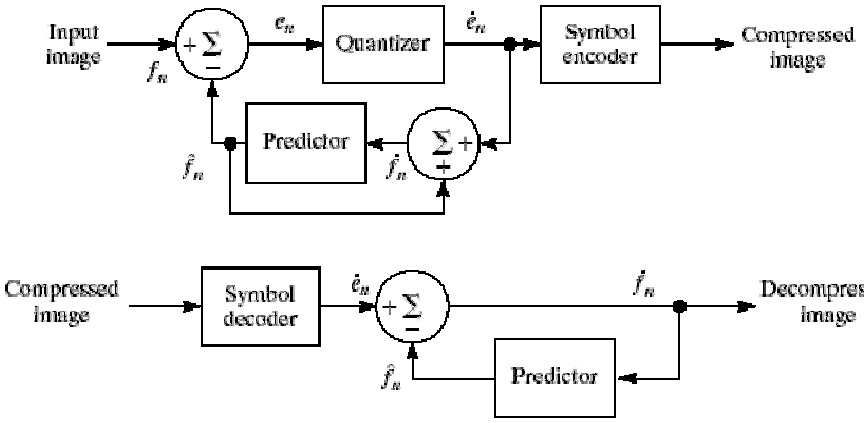
What is lossy compression ?

The compressed data is not the same as the original data, but a close approximation of it. – Yields a much higher compression ratio than that of lossless compression.

Lossy Compression:

1. Spatial domain methods
2. Transform coding

Lossy Predictive Coding:



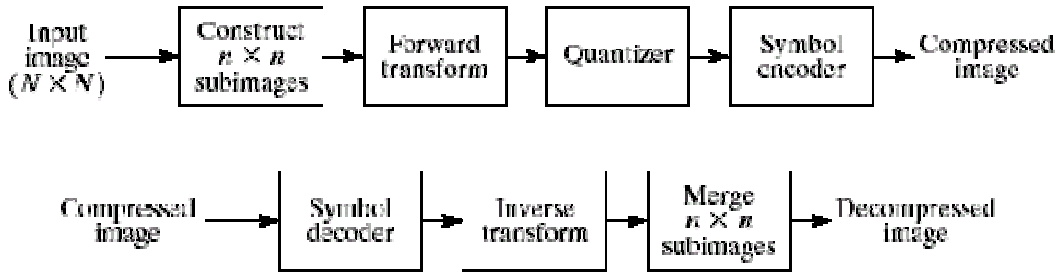
a
b

FIGURE 8.21 A lossy predictive coding model: (a) encoder and (b) decoder.

$$\dot{f}_n = \dot{e}_n + \hat{f}_n$$

Transform Coding

In transform coding ,a reversible ,linear transform (such as Fourier)is used to map the image in to a set of transform coefficients.



a
b

FIGURE 8.28 A transform coding system: (a) encoder; (b) decoder.

Any or all of the transform encoding steps can be adapted to local image content ,called adaptive transform coding or fixed for all sub images ,called non adaptive transform coding.
Transform selection:
The choice of transformation selection depends on reconstruction error and computational resources available.
Forward discrete transform T(u,v) can be expressed as:

$$T(u,v) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} f(x,y)g(x,y,u,v) \quad u,v = 0,1,\cdots,N-1$$

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} T(u,v)h(x,y,u,v) \quad x,y = 0,1,\cdots,N-1$$

In these equations g(x,y,u,v) and h(x,y,u,v) are called forward and inverse transformation Kernals .Forward kernel is Separable if:

$$g(x,y,u,v) = g_1(x,u).g_2(y,v)$$

Forward kernel is Symmetric if:

$$g_1 = g_2 \Rightarrow g(x,y,u,v) = g_1(x,u).g_1(y,v)$$

7 b) Explain the run length coding with respect to image compression ? 5M
run length coding 2M
Example 3M

Ans

Run-Length Coding
A=[5 2 2 2 3 3 3 3 4 4 1 1 1 1 1 1 1 6 6 4 4]
Result =

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

EXPLANATION:
5 has occurred 1 times, 2 has occurred 3 times, 3 has occurred 4 times, 4 has occurred 2 times, 1 has occurred 7 times, 6 has occurred 2 times and 4 has occurred 2 times.
Consider a matrix A with 15 elements,
A= [10 10 9 9 9 9 4 0 0 0 0 0 10 10 10]
In the given example,
10 has occurred 2 times, 9 has occurred 4 times, 4 has occurred once, 0 has occurred 5 times and 10 has occurred 3 times.
After Run length encoding, we obtain the matrix without any repetition in the adjacent elements, [10 9 4 0 10]. And the occurrences of each element [2 4 1 5 3]
Thus the matrix is reduced to 10 elements from 15 elements.

UNIT-IV

8 a) Explain about region based segmentation in detail? 5M
Region Growing 3M
Region merging 2M

Ans

Region-Based Segmentation:

The objective of segmentation is to partition an image into regions. We approached this problem by finding boundaries between regions based on discontinuities in gray levels, whereas segmentation was accomplished via thresholds based on the distribution of pixel properties, such as gray-level values or color.

Basic Formulation:

Let R represent the entire image region. We may view segmentation as a process that partitions R into n subregions, $R_1, R_2..., R_n$, such that

(a) $\bigcup_{i=1}^{n} R_i = R.$
(b) R_i is a connected region, $i = 1, 2, \ldots, n.$
(c) $R_i \cap R_j = \varnothing$ for all i and $j, i \neq j.$
(d) $P(R_i) =$ TRUE for $i = 1, 2, \ldots, n.$
(e) $P(R_i \cup R_j) =$ FALSE for $i \neq j.$

Here, P (R_i) is a logical predicate defined over the points in set R_i and Ø` is the null set. Condition (a) indicates that the segmentation must be complete; that is, every pixel must be in a region. Condition (b) requires that points in a region must be connected in some predefined sense. Condition (c) indicates that the regions must be disjoint. Condition (d) deals with the properties that must be satisfied by the pixels in a segmented region—for example P (R_i) = TRUE if all pixels in R_i, have the same gray level. Finally, condition (c) indicates that regions R_i and R_j are different in the sense of predicate P.

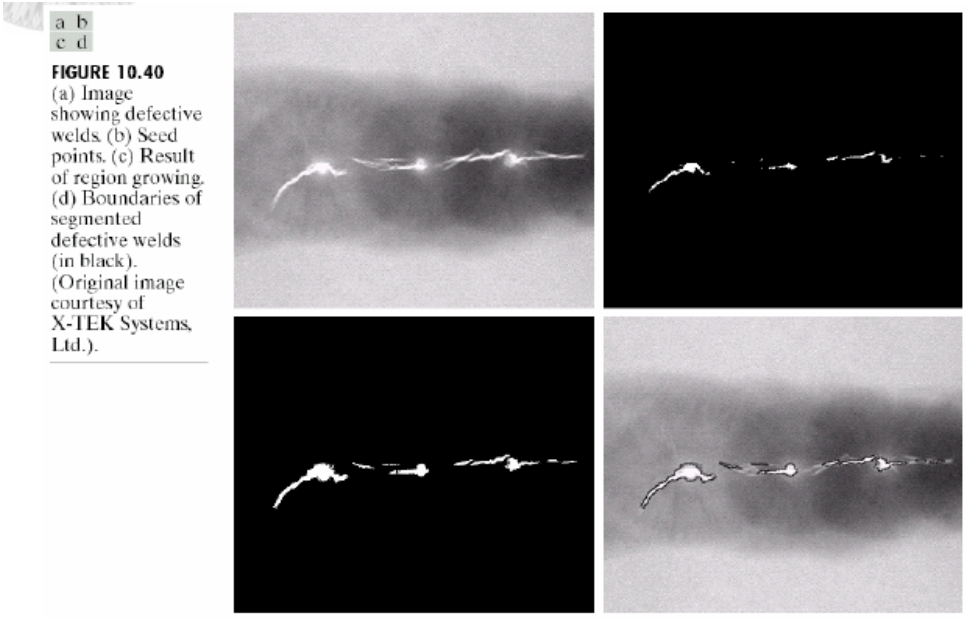
Region Growing:

As its name implies, region growing is a procedure that groups pixels or subregions into larger regions based on predefined criteria. The basic approach is to start with a set of "seed" points and from these grow regions by appending to each seed those neighboring pixels that have properties similar to the seed (such as specific ranges of gray level or color). When a priori information is not available, the procedure is to compute at every pixel the same set of properties that ultimately will be used to assign pixels to regions during the growing process. If the result of these computations shows clusters of values, the pixels whose properties place them near the centroid of these clusters can be used as seeds.

The selection of similarity criteria depends not only on the problem under consideration, but also on the type of image data available. For example, the analysis of land-use satellite imagery depends heavily on the use of color. This problem would be significantly more difficult, or even impossible, to handle without the inherent information available in color images. When the images are monochrome, region analysis must be carried out with a set of descriptors based on gray levels and spatial properties (such as moments or texture).

Basically, growing a region should stop when no more pixels satisfy the criteria for inclusion in that region. Criteria such as gray level, texture, and color, are local in nature and do not take into account the "history" of region growth. Additional criteria that increase the power of a region-growing algorithm utilize the concept of size, likeness between a candidate pixel and the pixels grown so far (such as a comparison of the gray level of a candidate and the average gray level of the grown region), and the shape of the region being grown. The use of these types of descriptors is based on the assumption that a model of expected results is at least partially available.

Figure 7.1 (a) shows an X-ray image of a weld (the horizontal dark region) containing several cracks and porosities (the bright, white streaks running horizontally through the middle of the image). We wish to use region growing to segment the regions of the weld failures. These segmented features could be used for inspection, for inclusion in a database of historical studies, for controlling an automated welding system, and for other numerous applications.



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.).

The first order of business is to determine the initial seed points. In this application, it is known that pixels of defective welds tend to have the maximum allowable digital value B55 in this case). Based on this information, we selected as starting points all pixels having values of 255. The points thus extracted from the original image are shown in Fig. 10.40(b). Note that many of the points are clustered into seed regions.

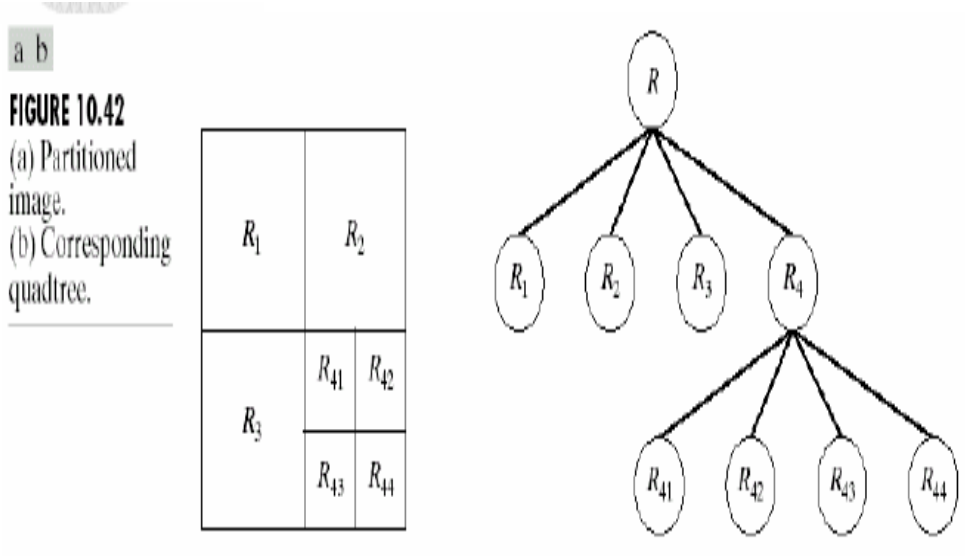
The next step is to choose criteria for region growing. In this particular example we chose two criteria for a pixel to be annexed to a region: (1) The absolute gray-level difference between any pixel and the seed had to be less than 65. This number is based on the histogram shown in Fig. 7.2 and represents the difference between 255 and the location of the first major valley to the left, which is representative of the highest gray level value in the dark weld region. (2) To be included in one of the regions, the pixel had to be 8-connected to at least one pixel in that region.

If a pixel was found to be connected to more than one region, the regions were merged. Figure 7.1 (c) shows the regions that resulted by starting with the seeds in Fig. 7.2 (b) and utilizing the criteria defined in the previous paragraph. Superimposing the boundaries of these regions on the original image [Fig. 7.1(d)] reveals that the region-growing procedure did indeed segment the defective welds with an acceptable degree of accuracy. It is of interest to note that it was not necessary to specify any stopping rules in this case because the criteria for region growing were sufficient to isolate the features of interest.

Region Splitting and Merging:

The procedure just discussed grows regions from a set of seed points. An alternative is to subdivide an image initially into a set of arbitrary, disjointed regions and then merge and/or split the regions in an attempt to satisfy the conditions. A split and merge algorithm that iteratively works toward satisfying these constraints is developed.

Let R represent the entire image region and select a predicate P. One approach for segmenting R is to subdivide it successively into smaller and smaller quadrant regions so that, for any region R_i, $P(R_i)$ = TRUE. We start with the entire region. If P(R) = FALSE, we divide the image into quadrants. If P is FALSE for any quadrant, we subdivide that quadrant into subquadrants, and so on. This particular splitting technique has a convenient representation in the form of a so-called quadtree (that is, a tree in which nodes have exactly four descendants), as illustrated in Fig. 7.3. Note that the root of the tree corresponds to the entire image and that each node corresponds to a subdivision. In this case, only R_4 was subdivided further.



If only splitting were used, the final partition likely would contain adjacent regions with identical properties. This drawback may be remedied by allowing merging, as well as splitting. Satisfying the constraints, requires merging only adjacent regions whose combined pixels satisfy the predicate P. That is, two adjacent regions R_j and R_k are merged only if $P(R_j \cup R_k) = TRUE$.

The preceding discussion may be summarized by the following procedure, in which, at any step we

1. Split into four disjoint quadrants any region R_i, for which $P(R_i) = FALSE$.

2. Merge any adjacent regions R_j and R_k for which $P(R_j \cup R_k) = TRUE$.

3. Stop when no further merging or splitting is possible.

Several variations of the preceding basic theme are possible. For example, one possibility is to split the image initially into a set of blocks. Further splitting is carried out as described previously, but merging is initially limited to groups of four blocks that are descendants in the quadtree representation and that satisfy the predicate P. When no further mergings of this type are possible, the procedure is terminated by one final merging of regions satisfying step 2. At this point, the merged regions may be of different sizes. The principal advantage of this approach is that it uses the same quadtree for splitting and merging, until the final merging step.

8 b) Analyze the basic Fourier descriptors? 5M

Fourier descriptors 3M

Coefficients 2M

Ans **Fourier Descriptors**

$(x_0,y_0), (x_1,y_1), (x_2,y_2) \ldots\ldots\ldots\ldots (x_{k-1},y_{k-1})$

$x(k)=x_k$ and $y(k)=y_k$

$s(k)=[x(k),y(k)]$,for k=0,1,2,…….K-1

$S(k)=x(k)+jy(k)$

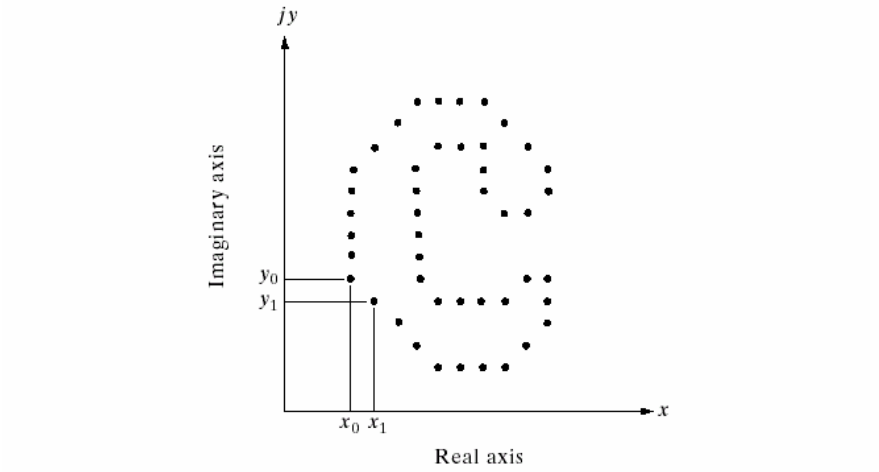


FIGURE 11.13 A digital boundary and its representation as a complex sequence. The points (x_0, y_0) and (x_1, y_1) shown are (arbitrarily) the first two points in the sequence.

An N-point DFT is performed to the sequence and the complex coefficients obtained are called the *Fourier descriptors of the boundary.*

In general, only the first few coefficients are of significant magnitude and are pretty enough to describe the general shape of the boundary.

Fourier descriptors are not directly insensitive to geometrical changes such as translation, rotation and scale changes, but the changes can be related to simple transformations on the descriptors.

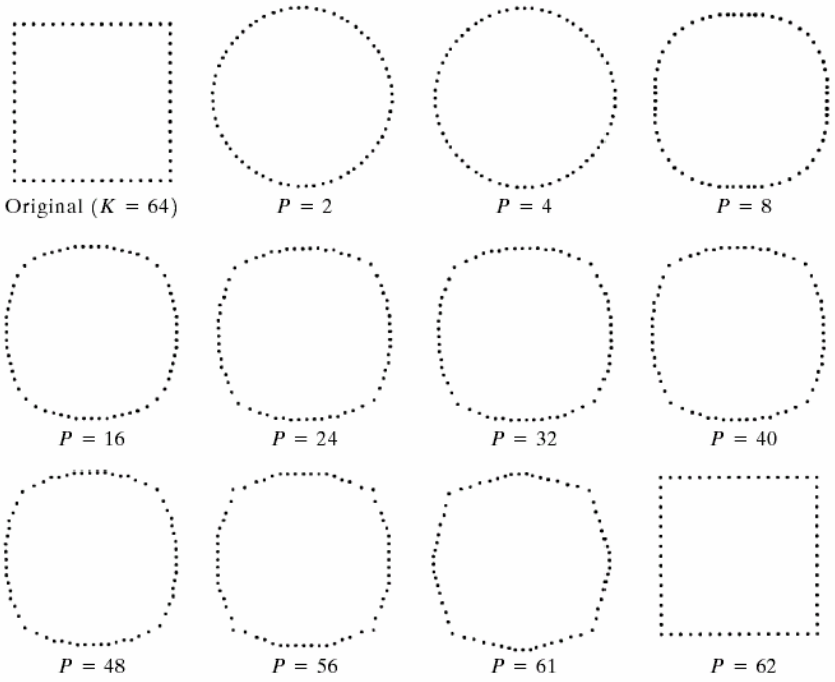


FIGURE 11.14 Examples of reconstruction from Fourier descriptors. P is the number of Fourier coefficients used in the reconstruction of the boundary.

(OR)

9 a) Explain the role of thresholding in segmentation **5M**

Global thresholding **2M**

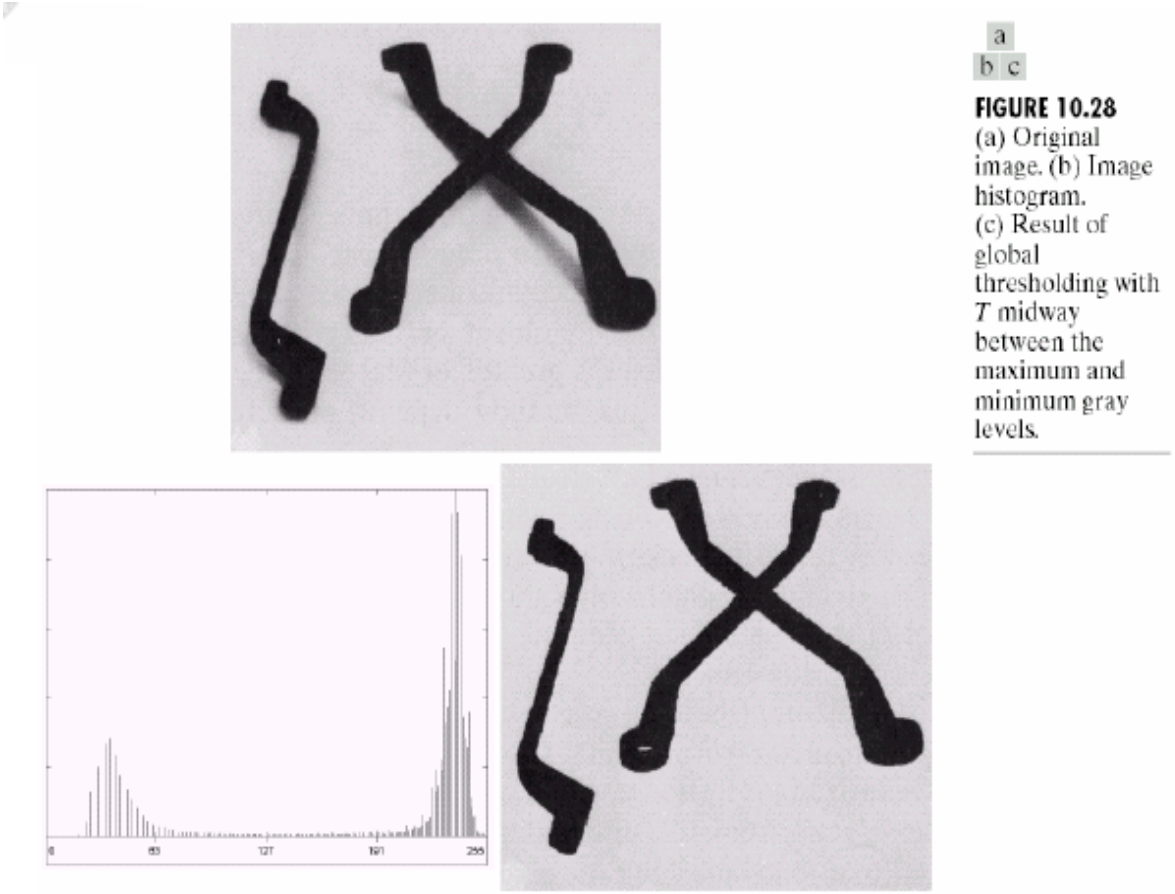
Adoptive Thresholding **3M**

Ans **Thresholding:**

Because of its intuitive properties and simplicity of implementation, image thresholding enjoys a central position in applications of image segmentation.

Global Thresholding:

The simplest of all thresholding techniques is to partition the image histogram by using a single global threshold, T. Segmentation is then accomplished by scanning the image pixel by pixel and labeling each pixel as object or back-ground, depending on whether the gray level of that pixel is greater or less than the value of T. As indicated earlier, the success of this method depends entirely on how well the histogram can be partitioned.



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.

Figure 4.1(a) shows a simple image, and Fig. 4.1(b) shows its histogram. Figure 4.1(c) shows the result of segmenting Fig. 4.1(a) by using a threshold T midway between the maximum and minimum gray levels. This threshold achieved a "clean" segmentation by eliminating the shadows and leaving only the objects themselves. The objects of interest in this case are darker than the background, so any pixel with a gray level $\leq$ T was labeled black (0), and any pixel with a gray level $\geq$ T was labeled white (255).The key objective is merely to generate a binary image, so the black-white relationship could be reversed. The type of global thresholding just described can be expected to be successful in highly controlled environments. One of the areas in which this often is possible is in industrial inspection applications, where control of the illumination usually is feasible.

The threshold in the preceding example was specified by using a heuristic approach, based on visual inspection of the histogram. The following algorithm can be used to obtain T automatically:

1. Select an initial estimate for T.

2. Segment the image using T. This will produce two groups of pixels: G_1 consisting of all pixels with gray level values >T and G_2 consisting of pixels with values < T.

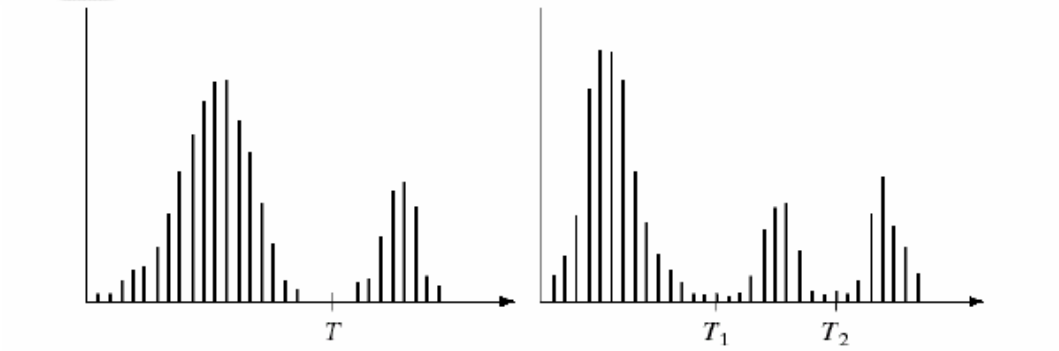
3. Compute the average gray level values μ_1 and μ_2 for the pixels in regions G_1 and G_2.

4. Compute a new threshold value:

$$T = \frac{1}{2}(\mu_1 + \mu_2).$$

5. Repeat steps 2 through 4 until the difference in T in successive iterations is smaller than a predefined parameter T_o.

When there is reason to believe that the background and object occupy comparable areas in the image, a good initial value for T is the average gray level of the image. When objects are small compared to the area occupied by the background (or vice versa), then one group of pixels will dominate the histogram and the average gray level is not as good an initial choice. A more appropriate initial value for T in cases such as this is a value midway between the maximum and minimum gray levels. The parameter T_o is used to stop the algorithm after changes become small in terms of this parameter. This is used when speed of iteration is an important issue.



a b

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

- Thresholding may be viewed as an operation that involves tests against a function T of the form $T = T[x, y, p(x, y), f(x, y)]$ where $f(x,y)$ is the gray level of point (x,y) and $p(x,y)$ denotes some local property of this point.
- When T depends only on $f(x,y)$ (that is, only on gray-level values) the threshold is called *global*.
- If T depends on both $f(x,y)$ and $p(x,y)$, the threshold is called *local*.
- If, in addition, T depends on the spatial coordinates x and y, the threshold is called *dynamic* or *adaptive*.

9 b) Describe shape numbers as boundary descriptors? **5M**

Ans Shape number of a boundary is defined as the first difference of a chain code of the smallest magnitude. · The order n of a shape number is the number of digits in its representation.
· The following figures shows all shapes of order 4 and 6 in a 4-directional chain code:
The dots indicates the starting point.

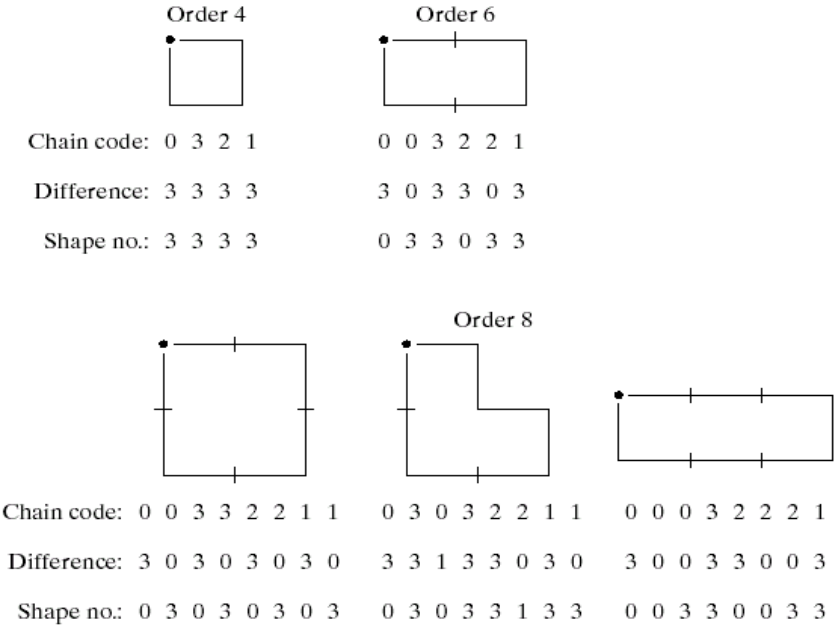


FIGURE 11.11 All shapes of order 4, 6, and 8. The directions are from Fig. 11.1(a), and the dot indicates the starting point.