DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

IMPORTANT QUESTIONS AND ANSWERS

Subject Code: **IT6005**

Subject Name: **DIGITAL IMAGE PROCESSING**

Sem / Year: **VII/IV**

Regulation: 2013

IT6005 DIGITAL IMAGE PROCESSING L T P C

3 0 0 3

OBJECTIVES:

The student should be made to:

- ✓ Learn digital image fundamentals.
- ✓ Be exposed to simple image processing techniques.
- ✓ Be familiar with image compression and segmentation techniques.
- ✓ Learn to represent image in form of features.

UNIT I DIGITAL IMAGE FUNDAMENTALS 8

Introduction – Origin – Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships between pixels - color models.

UNIT II IMAGE ENHANCEMENT 10

Spatial Domain: Gray level transformations – Histogram processing – Basics of Spatial Filtering– Smoothing and Sharpening Spatial Filtering – Frequency Domain: Introduction to Fourier Transform – Smoothing and Sharpening frequency domain filters – Ideal, Butterworth and Gaussian filters.

UNIT III IMAGE RESTORATION AND SEGMENTATION 9

Noise models – Mean Filters – Order Statistics – Adaptive filters – Band reject Filters – Band pass Filters – Notch Filters – Optimum Notch Filtering – Inverse Filtering – Wiener filtering Segmentation: Detection of Discontinuities– Edge Linking and Boundary detection – Region based segmentation- Morphological processing- erosion and dilation.

UNIT IV WAVELETS AND IMAGE COMPRESSION 9

Wavelets – Subband coding - Multi resolution expansions - Compression: Fundamentals – Image Compression models – Error Free

Compression – Variable Length Coding – Bit-Plane Coding – Lossless Predictive Coding – Lossy Compression – Lossy Predictive Coding – Compression Standards.

UNIT V IMAGE REPRESENTATION AND RECOGNITION 9

Boundary representation – Chain Code – Polygonal approximation, signature, boundary segments – Boundary description – Shape number – Fourier Descriptor, moments- Regional Descriptors – Topological feature, Texture - Patterns and Pattern classes - Recognition based on matching.

TOTAL: 45 PERIODS

OUTCOMES:

Upon successful completion of this course, students will be able to:

- ✓ Discuss digital image fundamentals.
- ✓ Apply image enhancement and restoration techniques.
- ✓ Use image compression and segmentation Techniques.
- ✓ Represent features of images.

TEXT BOOK:

1. Rafael C. Gonzales, Richard E. Woods, "Digital Image Processing", Third Edition, Pearson Education, 2010.

REFERENCES:

1. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, "Digital Image Processing Using MATLAB", Third Edition Tata Mc Graw Hill Pvt. Ltd., 2011.

2. Anil Jain K. "Fundamentals of Digital Image Processing", PHI Learning Pvt. td., 2011.

3. Willliam K Pratt, "Digital Image Processing", John Willey, 2002.

4. Malay K. Pakhira, "Digital Image Processing and Pattern Recognition", First Edition, PHI Learning Pvt. Ltd., 2011.

5. http://eeweb.poly.edu/~onur/lectures/lectures.html.

6. http://www.caen.uiowa.edu/~dip/LECTURE/lecture.html

IT6005 DIGITAL IMAGE PROCESSING L T P C

3 0 0 3

1. Aim and Objective of the Subject

Aim:

To study the fundamental concepts in Digital Image Processing and also discuss the operation of Digital Image Processing and its components required for build different type of Processing signals.

OBJECTIVES:

- ✓ Learn digital image fundamentals.
- ✓ Be exposed to simple image processing techniques.
- ✓ Be familiar with image compression and segmentation techniques.
- ✓ Learn to represent image in form of features.

2. Need and Importance for Study of the Subject

Need for Study of the Subject:

- ➢ Makes it possible to design the simple image processing concept and experiments that are: – Image compression, Image Enhancement, Image Restoration ect.
- ➢ Allows students to upgrade their knowledge in Digital Image Processing field for their research and project
- ➢ Helps students/engineers in touch with the latest signal processing technologies.(3G, 4G, 5G, smart phones, Wifi, encoding and decoding concepts, Signal transmission ect).

Importance for Study of the Subject:

At the end of the course, the student should be able to:

- ✓ Discuss digital image fundamentals.
- ✓ Apply image enhancement and restoration techniques.
- ✓ Use image compression and segmentation Techniques.

- ✓ Represent features of images.
- ✓ Studying this course is essential to understand how these communication signals are employed in various applications include Area monitoring, Health care monitoring, Environmental/Earth sensing, Air pollution monitoring, Forest fire detection, Landslide detection, Water quality monitoring and Natural disaster monitoring and prevention, signal transmission and detection etc.

3. Industry Connectivity and Latest Developments

Industry Connectivity:

- ➢ The following companies (Industries) are connectivity to Digital Image Processing: CICSO, IBM, SIEMENS, NORTEL.

Latest Developments:

- LIFI if fast data transfer in communication signals networking.
- New revolution in 5G, 6G in all the smart phones, smart televisions, etc.
- Satellite signal transmission and restoration.

4. Industrial Visit (Planned if any): -NO-

Department of Electronics and Communication Engineering

Detailed Lesson Plan

Name of the Subject& Code: IT 6005 - Digital Image Processing

Name of the Faculty: Dr. N. Muthukumaran, Mr. S. Esakki Rajavel, Mr. S. Allwin Devaraj.

TEXT BOOK:

1. Rafael C. Gonzales, Richard E. Woods, “Digital Image Processing”, Third Edition, Pearson Education, 2010.

REFERENCES:

1. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, “Digital Image Processing Using MATLAB”, Third Edition Tata Mc Graw Hill Pvt. Ltd., 2011.
2. Anil Jain K. “Fundamentals of Digital Image Processing”, PHI Learning Pvt. Ltd., 2011.
3. Willliam K Pratt, “Digital Image Processing”, John Willey, 2002.
4. Malay K. Pakhira, “Digital Image Processing and Pattern Recognition”, First Edition, PHI Learning Pvt. Ltd., 2011.
5. http://eeweb.poly.edu/~onur/lectures/lectures.html.
6. http://www.caen.uiowa.edu/~dip/LECTURE/lecture.html

Lect. No.	Unit No	Topics to be Covered	Text/ Reference	Pages	Week No.
		DIGITAL IMAGE FUNDAMENTALS			
1	I	Introduction about the Digital Image Processing	T1, R1	15-17, 10-12	1
2		Origin of Digital Image Processing	T1	17-21	
3		Steps in Digital Image Processing – Components	T1	39-42	
4		Elements of Visual Perception	T1	34-38	

5		Image Sensing and Acquisition	T1	45-50	
6,7		Image Sampling and Quantization	T1	52-64	
8		Relationships between pixels	T1	66-69	2
9		Color models	T1	289-295	
		IMAGE ENHANCEMENT			
10		Introduction about Image Enhancement	T1, R1	75-75, 54-56	
11		Spatial Domain: Gray level transformations	T1	78-85	
12		Histogram processing	T1	88-94	3
13		Basics of Spatial Filtering	T1	116-119	
14, 15	II	Smoothing and Sharpening Spatial Filtering	T1	119-128	
16		Frequency Domain: Introduction to Fourier Transform	T1	149-156	
17, 18		Smoothing and Sharpening frequency domain filters	T1	167-178	
19		Ideal filters	T1	182	4
20		Butterworth filters	T1	183	
21		Gaussian filters	T1	184	
		IMAGE RESTORATION AND SEGMENTATION			
22		Introduction about Image Restoration and Segmentation Noise models	T1	220-221, 248 - 249	
23		Mean Filters	T1	231	5
24		Order Statistics	T1	233	
25	III	Adaptive filters	T1	237	
26		Band reject Filters – Band pass Filters	T1	244-245	
27		Notch Filters – Optimum Notch Filtering	T1	246-248	
28		Inverse Filtering	T1	261	6
29		Wiener filtering Segmentation:	T1	262	
30, 31		Detection of Discontinuities–Edge Linking and Boundary detection	T1	568-572	

32		Region based segmentation	T1	612-617	
33, 34		Morphological processing- erosion and dilation	R1	519-528	
	IV	**WAVELETS AND IMAGE COMPRESSION**			
35		Introduction about Wavelets and Image Compression	T1, R2	349-351, 158-162	7
36		Wavelets – Sub band coding	T1	351-360	
37		Multi resolution expansions	T1	363-369	
38		Fundamentals – Image Compression models	T1	411-419	
39		Error Free Compression	T1	440-442	
40		Variable Length Coding – Bit-Plane Coding	T1	442-448	8
41		Lossless Predictive Coding	T1	456-459	
42, 43		Lossy Compression, Lossy Predictive Coding	T2	459-486	
44		Compression Standards.	T1	492-510	
	V	**IMAGE REPRESENTATION AND RECOGNITION**			
45		Introduction about Image Representation and Recognition	T1	643-644	9
46		Boundary representation	T1	644-645	
47		Chain Code	T1	644	
48		Polygonal approximation, signature, boundary segments	T1	646	
49		Boundary description – Shape number	T1	653-655	
50		Fourier Descriptor, moments- Regional Descriptors	T1	655-659	10
51		Topological feature	T1	661-663	
52. 53		Texture - Patterns and Pattern classes	T1	665-672	
54		Recognition based on matching.	T1	698-704	

Faculty In charge HoD

UNIT – I

DIGITAL IMAGE FUNDAMENTALS

Introduction – Origin – Steps in Digital Image Processing – Components – Elements of Visual Perception – Image Sensing and Acquisition – Image Sampling and Quantization – Relationships between pixels - color models.

PART -A (2 Marks)

1. **Define Image?** **[AUC NOV 2012]**

An image may be defined as two dimensional light intensity function f(x, y)where x and y denote spatial co-ordinate and the amplitude or value of f at any point (x, y) is called intensity or gray scale or brightness of the image at that point.

2. **Define Brightness?** **[AUC NOV2011]**

Brightness of an object is the perceived luminance of the surround. Two objects with different surroundings would have identical luminance but different brightness.

3. What do you meant by Color model? **[AUC APR 2013]**

A Color model is a specification of 3D-coordinates system and a subspace within that system where each color is represented by a single point.

4.**List the hardware oriented color models?** **[AUC APR 2012]**

RGB model

CMY model

YIQ model

HSI model

5. **What is Hue of saturation?** **[AUC NOV 2012, APR 2013]**

Hue is a color attribute that describes a pure color where saturation gives a measure of the degree to which a pure color is diluted by white light.

6. **List the applications of color models?** **[AUC NOV 2011]**

1. RGB model--- used for color monitor & color video camera
2. CMY model---used for color printing

3. HIS model----used for color image processing
4. 4. YIQ model---used for color picture transmission

7. **Define Resolutions? [AUC NOV 2012]**

Resolution is defined as the smallest number of discernible detail in an image. Spatial resolution is the smallest discernible detail in an image and gray level resolution refers to the smallest discernible change is gray level.

8. **What is meant by pixel? [AUC NOV 2009]**

A digital image is composed of a finite number of elements each of which has a particular location or value. These elements are referred to as pixels or image elements or picture elements or pels elements.

9. **Define Digital image? [AUC NOV 2010]**

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

10. What are the steps involved in DIP? [AUC NOV 2013]

1. Image Acquisition
2. Preprocessing
3. Segmentation
4. Representation and Description
5. Recognition and Interpretation

11. What is recognition and Interpretation? [AUC NOV 2010]

Recognition means is a process that assigns a label to an object based on the information provided by its descriptors.

Interpretation means assigning meaning to a recognized object.

12. Specify the elements of DIP system? [AUC APR 2011]

1. Image Acquisition
2. Storage
3. Processing
4. Display

13. **Define sampling and quantization [AUC NOV 2013, MAY 2017]**

Sampling means digitizing the co-ordinate value (x, y).

Quantization means digitizing the amplitude value.

PART B

16 Marks

1. What are the fundamental steps in Digital Image Processing? [AUC NOV 2012]

The fundamental steps in digital image processing are:

- Image acquisition
- Image enhancement
- Image restoration
- Image compression
- Image segmentation
- Image representation and description

Image Acquisition is the first step which involves capturing of images using cameras. It also involves steps like preprocessing and scaling.

Image Enhancement is the process of highlighting certain features of interest in an image.

Image Restoration deals with improving the appearance of an image.

Color image processing involves the processing of images which are in color rather than in binary or gray. It finds applications in the use of digital images in the internet.

Wavelets are foundation for representing images in various degrees of resolution.

Image compression deals with the techniques for reducing the size of the image for storage and reducing the bandwidth for transmitting.

Morphological processing deals with the tools for extracting the image components that are useful in the representation and description of shape.

Segmentation partition an image into its constituent parts or objects.

Representation transforms raw data into a suitable form subsequent for computer processing. ***Description*** deals with extracting attributes that result in some quantitative information of interest or are basic for differentiating one class of objects from another.

Recognition is the process that assigns label to an object based on its descriptors.

Knowledge about the problem domain is coded into the image processing system in the form of a knowledge database.

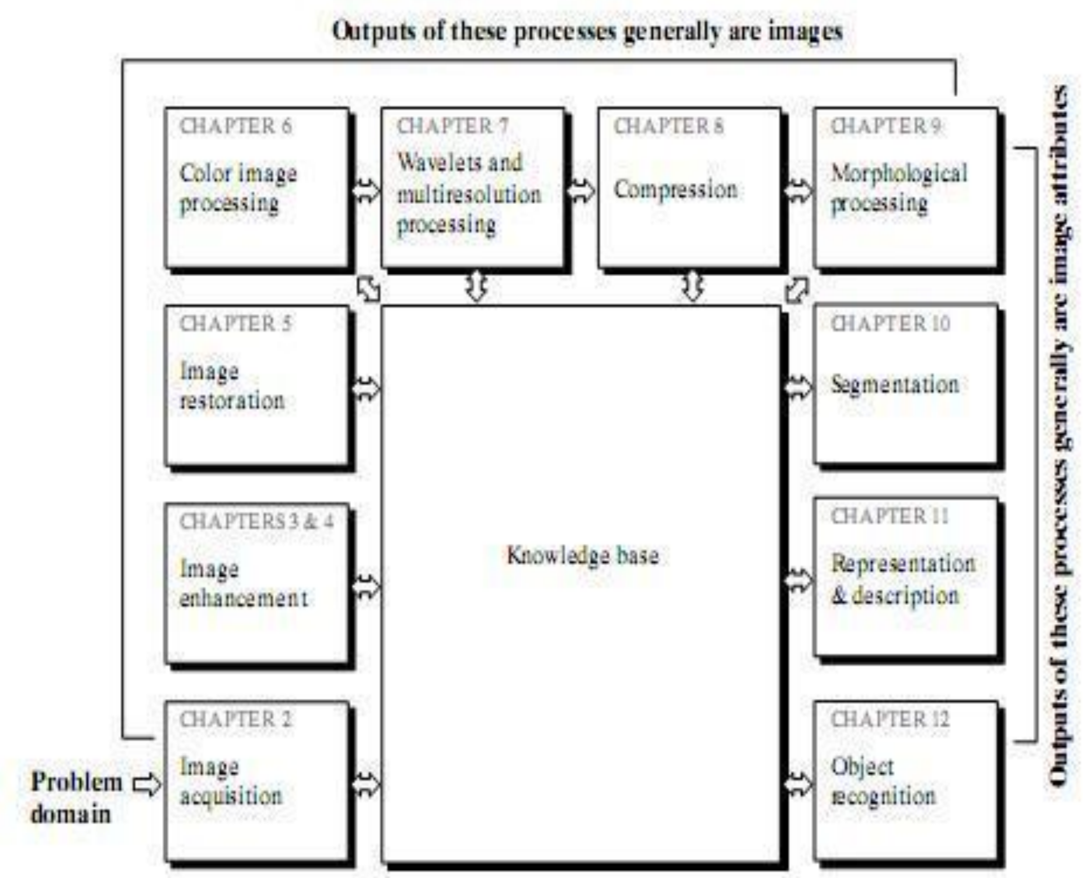


Fig: Fundamental steps in Digital Image Processing

2. What are the components of an Image Processing System?

[AUC NOV 2012 , APR 2013]

Briefly discuss about the elements of a Digital Image Processing System. [MAY/JUNE 2014, MAY/JUNE 2017]

1. Image Sensors

Image sensing or image acquisition is used to acquire i.e., to get digital images. It requires two elements, which are,

a)A physical device which is use to sense the object.

b)The second device is the digitizer which is used to convert the output of the physical sensing device into digital form.

2. Specialized image processing hardware

This hardware usually consists of digitizer plus ALU that performs some primitive operations like arithmetic and logical.

3. Computer

The computer in an image processing system is a general purpose computer and can range from PC to a supercomputer.

4. Software

The software for image processing system consists of specialized modules that can perform specific tasks. Some software packages have the facility for the user to write code using specialized modules.

5. Mass storage

Mass storage capability is needed if the image is not compressed.

There are three principal categories.

- Short term storage for use during processing, example: computer memory, frame buffers. Frame buffers are specialized boards that can store one or more images and can be accessed rapidly at video rates. This method allows instantaneous image zoom, scroll (vertical shifts) and pan (horizontal shifts also.
- On- line storage for relatively fast recall, example; magnetic disk or optical media. This type of storage gives frequent access to the storage data.
- Archival storage characterized by frequent access, example: magnetic tapes and optical disks. It requires large amount of storage space and the stored data is accessed infrequently.

6. Image displays

Image displays are color TV monitors. These monitors are driven by the output of image and graphics display cards which are a part of the computer system.

7. Hard copy

Hard copy devices are used for recording images. These devices include laser printers, film cameras, heat sensitive devices, inkjet printers and digital units such as optical and CD ROM disks.

8. Networking

Networking is useful for transmitting images. It includes optical fiber and other broad band technologies.

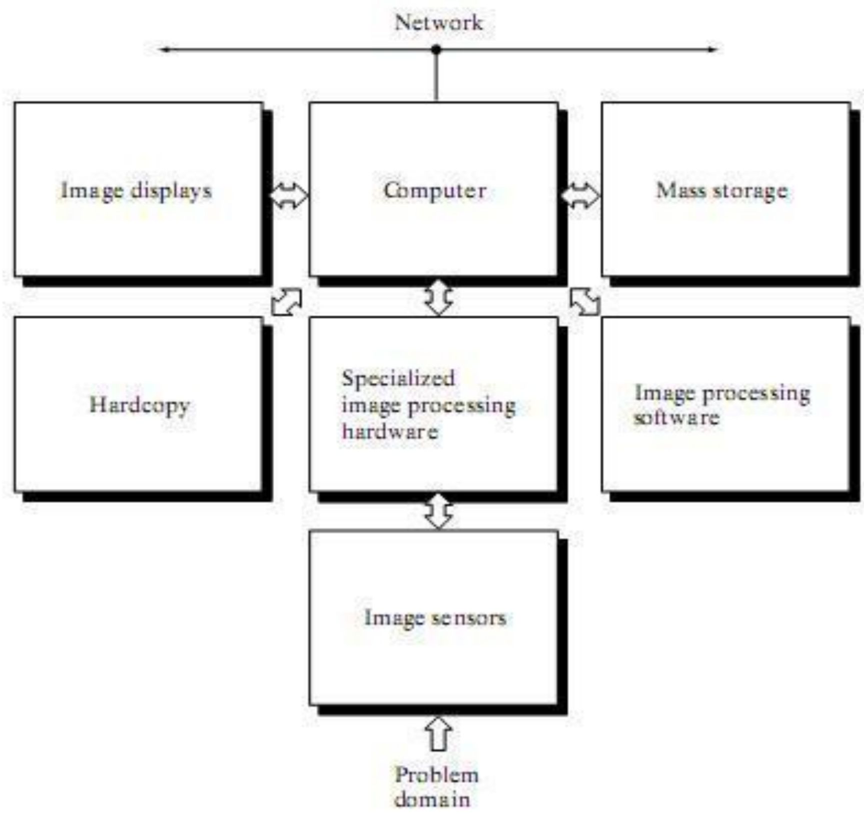


Fig: components of digital image processing systems

3. Explain about elements of visual perception. [AUC NOV 2013]

OR

Describe the elements of visual perception with suitable diagram.

[MAY/JUNE 2016]

EYE Characteristics

- Nearly spherical
- Approximately 20 mm in diameter
- Three membranes

Cornea &sclera outer cover.

The cornea is a tough, transparent tissue that covers the anterior i.e., front surface of eye. The sclera is an opaque membrane that is continuous with the cornea and encloses the remaining portion of the eye.

Choroid

It is located directly below the sclera. It contains network of blood vessels which provides nutrition to the eye. Slight injury in choroid can lead to severe eye damage as it causes restriction of blood flow. The outer cover of the choroid is heavily pigmented to reduce amount of extraneous light entering the eye. Also contains the iris diaphragm and ciliary body.

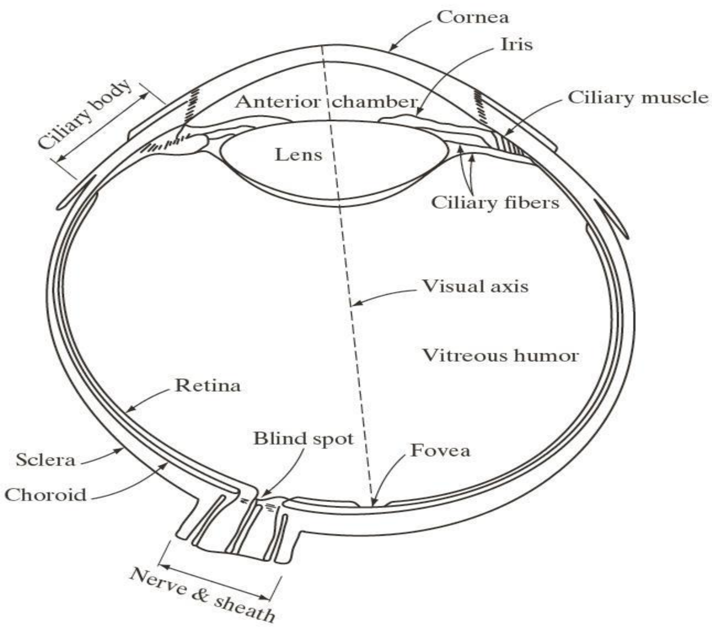


Fig: Structure of Human eye

Iris diaphragm

It contracts and expands to control the amount of light enteringinto the eye. The central opening of the iris is known as pupil whose diameter varies from 2-8 mm.

Lens

It is made up of many layers of fibrous cells. It is suspended and is attached to the ciliary body. It contains 60% to 70% water and 6% fat and more protein. The lens is colored by a slightly yellow pigmentation. This coloring increases with age, which leads to clouding of lens. Excessive clouding of lens happens in extreme cases which are known as **cataracts**. This leads to poor color discrimination and loss of clear vision.

Retina

It is the inner most membrane, objects are imaged on the surface. The central portion of retina is called the fovea.

There are two types of receptors in the retina.

- The rods are long slender receptors
- The cones are generally shorter and thicker in structure

The rods and cones are not distributed evenly around the retina. Rods and cones operate differently.

Cones

Cones are highly sensitive to color and are located in the fovea. There are 6 to 7 million cones. Each cone is connected with its own nerve end. Therefore humans can resolve fine details with the use of cones. Cones respond to higher levels of illumination; their response is called ***photopic vision or bright light vision***

Rods

Rods are more sensitive to low illumination than cones. There are about 75 to 159 million rods. Many numbers of rods are connected to a common, single nerve. Thus the amount of detail recognizable is less. Therefore rods provide only a general overall picture of the field of view. Due to stimulation of rods the objects that appear color in daylight will appear colorless in moon light. This phenomenon is called ***scotopic vision or dim light vision.***

- There are three basic types of cones in the retina
- These cones have different absorption characteristics as a function of wavelength with peak absorptions in the red, green, and blue regions of the optical spectrum. Most of the cones are at the fovea. Rods are spread just about everywhere except the fovea
- There is a relatively low sensitivity to blue light. There is a lot of overlap

- Distribution is radially symmetric about the fovea
- The area where there is absence of receptors is called the blind spot
- Receptor density measured in degrees from the fovea (the angle formed between the visual axis and a line extending from the center of the lens to the retina)

Image Formation in the EYE

- The lens of eye is flexible, whereas the optical lens is not.
- The radius of curvature of the anterior surface of the lens is greater than the radius of its posterior surface.
- The tension in the fibers of the ciliary body controls the shape of the lens
- To focus distant object greater than 3m the lens is made flattened by the controlling muscles and it will have lowest refractive index

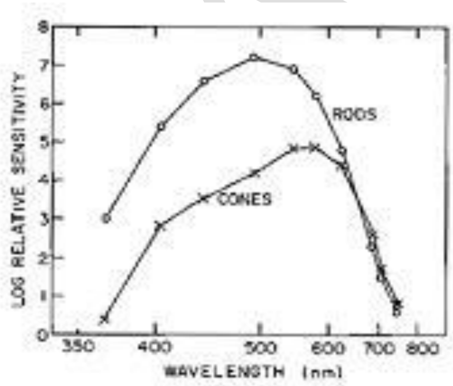


Fig: Sensitivity of rods and cones

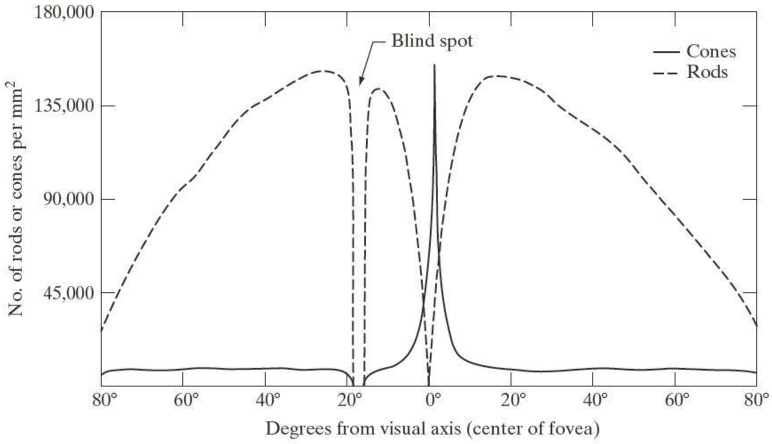


Fig: Distribution of rods and cones in retina

- To focus nearer objects the muscles allow the lens to become thicker, and strongest refractive index.

- The distance between the centre of the lens and the retina is called focal length.
- It ranges from 14mm to 17mm as the refractive power decreases from maximum to minimum.

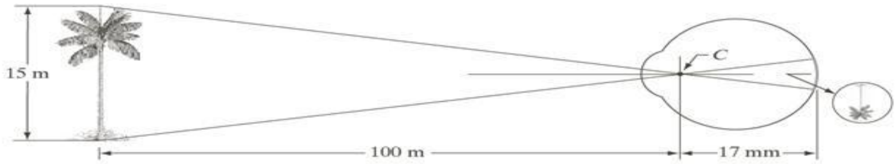


Fig: Image formation in the eye

Brightness adaptation and discrimination

The range of light intensity levels to which the human visual system can adapt is enormous from scotopic threshold to the glare limit. Subjective brightness is a logarithmic function of the light intensity incident on the eye. The visual system cannot operate over long range simultaneously; rather it accomplishes this large variation by changes in its overall sensitivity. This phenomenon is known as brightness adaptation. For any given set of conditions the current sensitivity level of the visual system is called the brightness adaptation level. Fig shows the plot of light intensity versus subjective brightness.

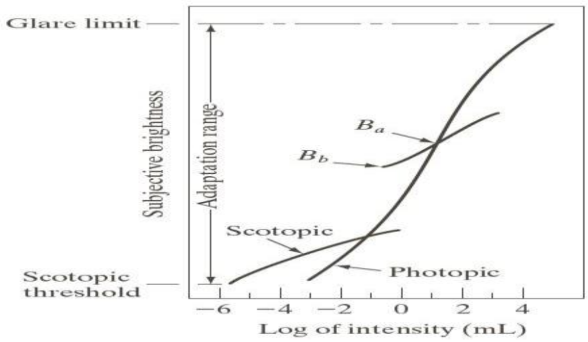


Fig: Range of subjective brightness sensation showing a particular adaptation level

Contrast Sensitivity

- The response of the eye to changes in the intensity of illumination is nonlinear

- Consider a patch of light of intensity i+dI surrounded by a background intensity I as shown in the following figure
- Over a wide range of intensities, it is found that the ratio dI/I, called the **Weber fraction**, is nearly constant at a value of about 0.02.
- This does not hold at very low or very high intensities
- Furthermore, contrast sensitivity is dependent on the intensity of the surround.

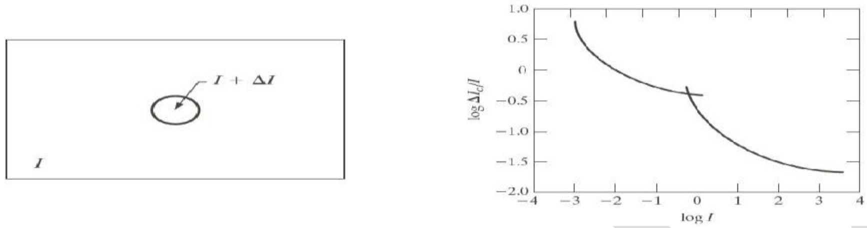


Fig: Weber ratio as a function of intensity

Perceived brightness and intensity

In actual case, the perceived brightness is not a function of intensity. This can be explained with the use of two phenomena namely

1. Simultaneous contrast
2. Mach band effect

1. Simultaneous contrast

- The small squares in each image are the same intensity.
- Because the different background intensities, the small squares do not appear equally bright.
- Perceiving the two squares on different backgrounds as different, even though they are in fact identical, is called the simultaneous contrast effect.
- Psychophysically, we say this effect is caused by the difference in the backgrounds.

Fig: Simultaneous contrast

2. Mach band effect

It shows that the human visual system tends to undershoot or overshoot around the boundary regions of different intensities.

- The Mach band effect is illustrated in the figure below.
- The intensity is uniform over the width of each bar.
- However, the visual appearance is that each strip is darker at its right side than its left.
- A bright bar appears at position B and a dark bar appears at D.
- The visual system perceives a brightness pattern which is strongky scalloped. These scalloped bands are called Mach bands.

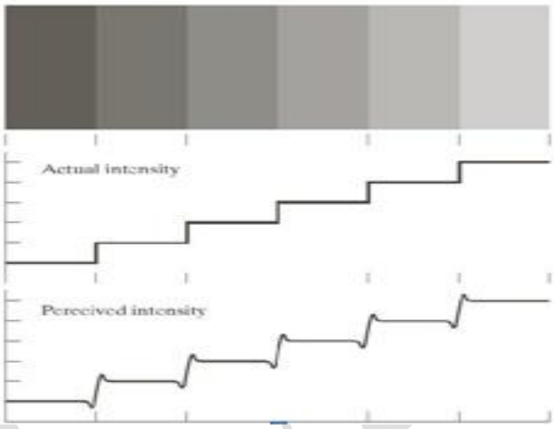
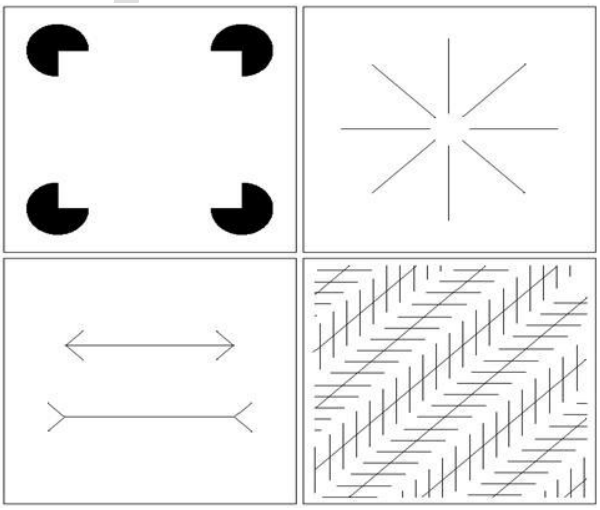


Fig: Example for match band effect

optical illusion

This is one in which the eye fills in nonexisting information or wrongly perceives geometrical properties of object.



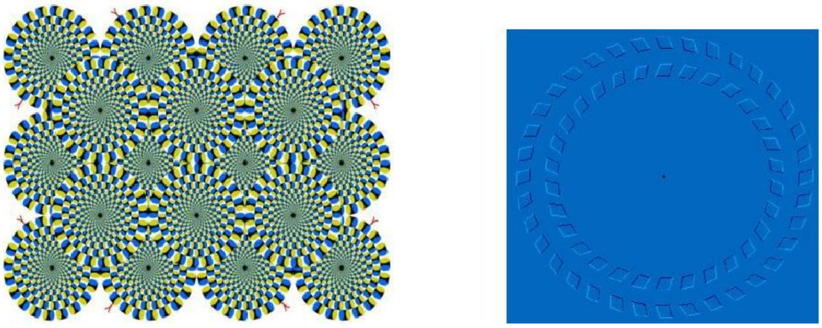


Fig: Examples for optical illusion

4. Explain the process of image acquisition.

Image Sensing and Acquisition:

The types of images in which we are interested are generated by the combination of an “illumination” source and the reflection or absorption of energy from that source by the elements of the “scene” being imaged.

We enclose illumination and scene in quotes to emphasize the fact that they are considerably more general than the familiar situation in which a visible light source illuminates a common everyday 3-D (three-dimensional) scene.

For example, the illumination may originate from a source of electromagnetic energy such as radar, infrared, or X-ray energy. But, as noted earlier, it could originate from less traditional sources, such as ultrasound or even a computer-generated illumination pattern.

Similarly, the scene elements could be familiar objects, but they can just as easily be molecules, buried rock formations, or a human brain. We could even image a source, such as acquiring images of the sun.

Depending on the nature of the source, illumination energy is reflected from, or transmitted through, objects. An example in the first category is light reflected from a planar surface.

An example in the second category is when X-rays pass through a patient’s body for the purpose of generating a diagnostic X-ray film.

In some applications, the reflected or transmitted energy is focused onto a photo converter (e.g., a phosphor screen), which converts the energy into visible light. Electron microscopy and some applications of gamma imaging use this approach.

Figure 4.1 shows the three principal sensor arrangements used to transform illumination energy into digital images.

The idea is simple: Incoming energy is transformed into a voltage by the combination of input electrical power and sensor material that is responsive to the particular type of energy being detected.

The output voltage waveform is the response of the sensor(s), and a digital quantity is obtained from each sensor by digitizing its response.

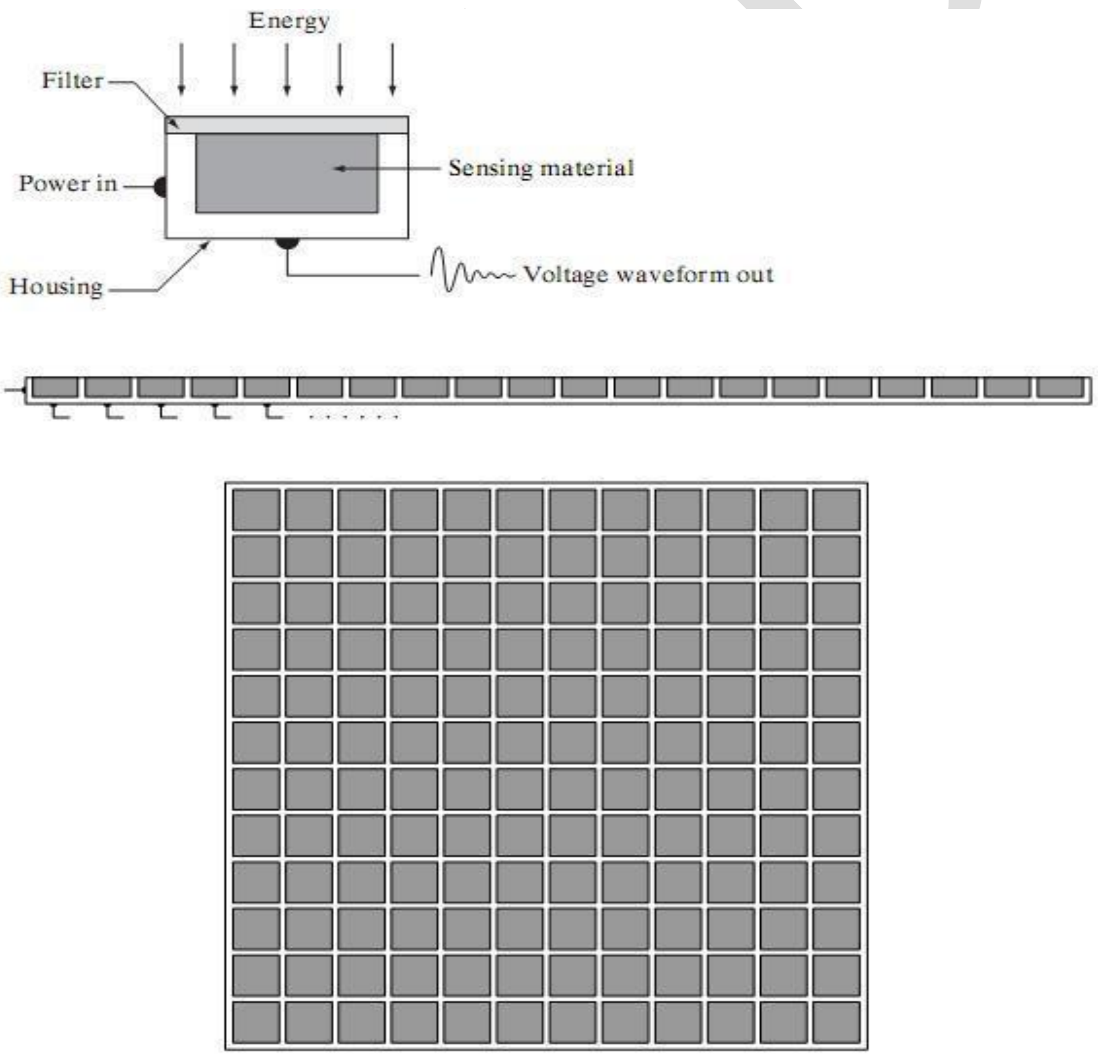


Fig.4.1 (a) Single imaging Sensor (b) Line sensor (c) Array sensor

(1)Image Acquisition Using a Single Sensor:

Figure 4.1 (a) shows the components of a single sensor. Perhaps the most familiar sensor of this type is the photodiode, which is constructed of silicon materials and whose output voltage waveform is proportional to light. The use of a filter in front of a sensor improves selectivity. For example, a green (pass) filter in front of a light sensor favors light in the green band of the color spectrum. As a consequence, the sensor output will be stronger for green light than for other components in the visible spectrum.

In order to generate a 2-D image using a single sensor, there has to be relative displacements in both the x- and y-directions between the sensor and the area to be imaged. Figure 4.2 shows an arrangement used in high-precision scanning, where a film negative is mounted onto a drum whose mechanical rotation provides displacement in one dimension. The single sensor is mounted on a lead screw that provides motion in the perpendicular direction. Since mechanical motion can be controlled with high precision, this method is an inexpensive (but slow) way to obtain high-resolution images. Other similar mechanical arrangements use a flat bed, with the sensor moving in two linear directions. These types of mechanical digitizers sometimes are referred to as micro densitometers.

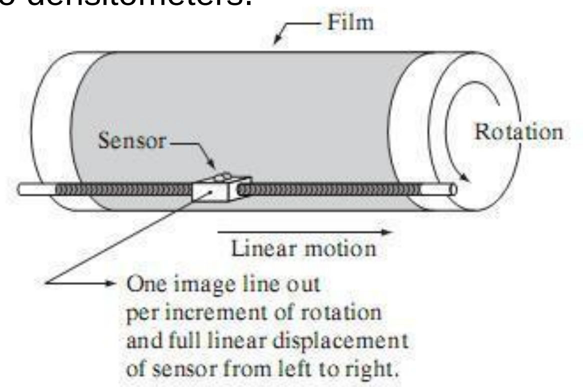


Fig.4.2. Combining a single sensor with motion to generate a 2-D image

(2) Image Acquisition Using Sensor Strips:

A geometry that is used much more frequently than single sensors consists of an in-line arrangement of sensors in the form of a sensor strip, as Fig. 4.1 (b) shows. The strip provides imaging elements in one direction. Motion perpendicular to the strip provides imaging in the other direction, as shown in Fig. 4.3 (a).This is the type of arrangement used in most flat bed scanners. Sensing devices with 4000 or more in-line sensors are possible. In-line sensors are used routinely in airborne imaging applications, in which the imaging system is mounted on an aircraft that flies at a constant altitude and speed over the geographical area to be imaged. One-dimensional imaging sensor strips that respond to various bands of the electromagnetic spectrum are mounted perpendicular to the direction of flight. The imaging strip gives one line of an image at a time, and the motion of the strip completes the other dimension of a two-dimensional image. Lenses or other focusing schemes are used to project the area to be scanned onto the sensors.

Sensor strips mounted in a ring configuration are used in medical and industrial imaging to obtain cross-sectional ("slice") images of 3-D objects, as Fig. 4.3 (b) shows. A rotating X-ray source provides illumination and the portion of the sensors opposite the source collect the X-ray energy that pass through the object (the sensors obviously have to be sensitive to X-ray energy).This is the basis for medical and industrial computerized axial tomography (CAT). It is important to note that the output of the sensors must be processed by reconstruction algorithms whose objective is to transform the sensed data into meaningful cross-sectional images.

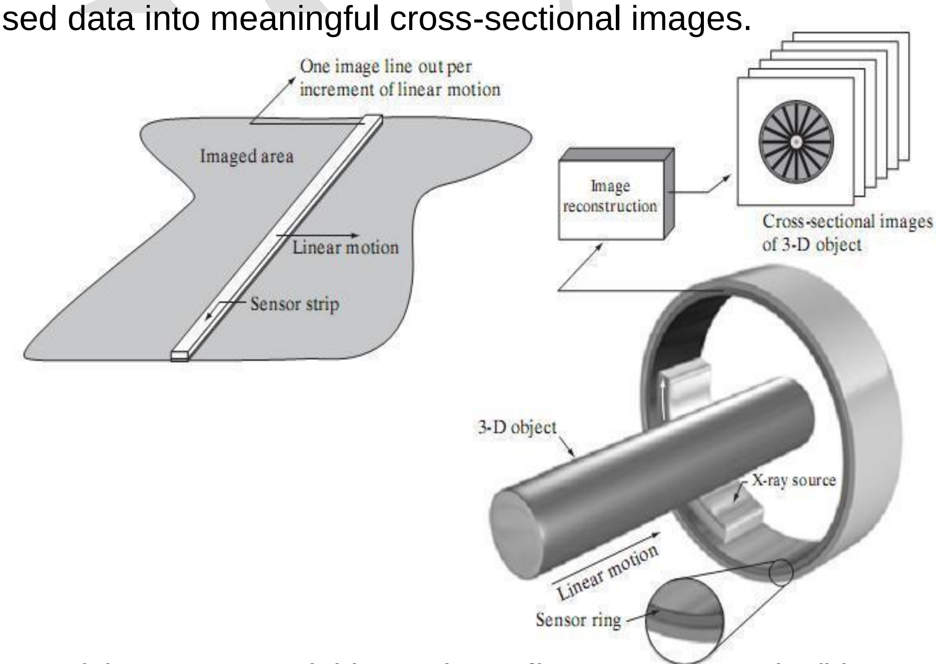


Fig.4.3 (a) Image acquisition using a linear sensor strip (b) Image acquisition using a circular sensor strip.

In other words, images are not obtained directly from the sensors by motion alone; they require extensive processing. A 3-D digital volume consisting of stacked images is generated as the object is moved in a direction perpendicular to the sensor ring. Other modalities of imaging based on the CAT principle include magnetic resonance imaging (MRI) and positron emission tomography (PET).The illumination sources, sensors, and types of images are different, but conceptually they are very similar to the basic imaging approach shown in Fig. 4.3 (b).

(3) Image Acquisition Using Sensor Arrays:

Figure 4.1 (c) shows individual sensors arranged in the form of a 2-D array. Numerous electromagnetic and some ultrasonic sensing devices frequently are arranged in an array format.

This is also the predominant arrangement found in digital cameras. A typical sensor for these cameras is a CCD array, which can be manufactured with a broad range of sensing properties and can be packaged in rugged arrays of 4000 * 4000 elements or more.

CCD sensors are used widely in digital cameras and other light sensing instruments.

The response of each sensor is proportional to the integral of the light energy projected onto the surface of the sensor, a property that is used in astronomical and other applications requiring low noise images.

Noise reduction is achieved by letting the sensor integrate the input light signal over minutes or even hours.

Since the sensor array shown in Fig. 4.4 (c) is two dimensional, its key advantage is that a complete image can be obtained by focusing the energy pattern onto the surface of the array.

The principal manner in which array sensors are used is shown in Fig.4.4. This figure shows the energy from an illumination source being reflected from a scene element, but, as mentioned at the beginning of this section, the energy also could be transmitted through the scene elements.

The first function performed by the imaging system shown in Fig.4.4 (c) is to collect the incoming energy and focus it onto an image plane.

If the illumination is light, the front end of the imaging system is a lens, which projects the viewed scene onto the lens focal plane, as Fig. shows.

The sensor array, which is coincident with the focal plane, produces outputs proportional to the integral of the light received at each sensor.

Digital and analog circuitry sweep these outputs and converts them to a video signal, which is then digitized by another section of the imaging system.

The output is a digital image, as shown diagrammatically in Fig. 4.4 (e).

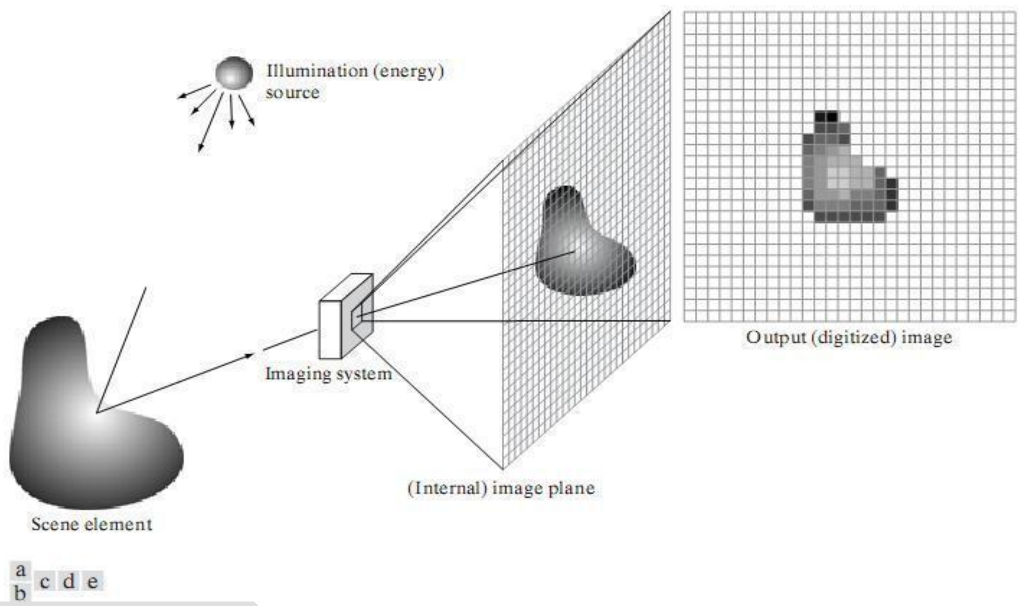


Fig.4.4 An example of the digital image acquisition process (a) Energy ("illumination") source (b) An element of a scene (c) Imaging system (d) Projection of the scene onto the image plane (e) Digitized image

5. Explain about image sampling and quantization process.

[AUC APR 2013]

Describe how the image is digitized by sampling and quantization and explain about checker board effect and false contouring with neat sketch. [APRIL/MAY 2015]

Describe how the image is digitized by sampling and quantization and

explain about checker board effect and false contouring with neat sketch. [APRIL/MAY 2015, APRIL/MAY 2017]

To be suitable for computer processing an image, f(x, y) must be digitized both spatially and in amplitude. Digitizing the spatial coordinates is called *image sampling.* Amplitude digitization is called gray-level quantization. The one dimensional function shown in fig.5.1.(a). Fig.5.1 (b) is a plot of amplitude values of the continuous image along the line segment AB in fig.5.1(a). the random variations are due to image noise. To sample this function, we take equally spaced samples along line AB, as shown in fig.5.1(c).

The location of each sample is given by vertical tick mark in the bottom part of the figure. The samples are shown as white square box superimposed on the function. The set of these discrete locations gives the sampled function.

In order to form a digital function, the gray values must be quantized into digital values. The right side of fig.5.1(c) shows the gray level scale divided into eight discrete levels, ranging from black to white. The vertical tick marks indicate the specific value assigned to each of the eight gray levels. The continuous values are quantized simply by assigning one of the eight gray levels to each sample. The digital sample resulting from sampling and quantization is shown in fig.5.1(d). Starting from the top of the image and carrying out this procedure line by line produces a two dimensional image.

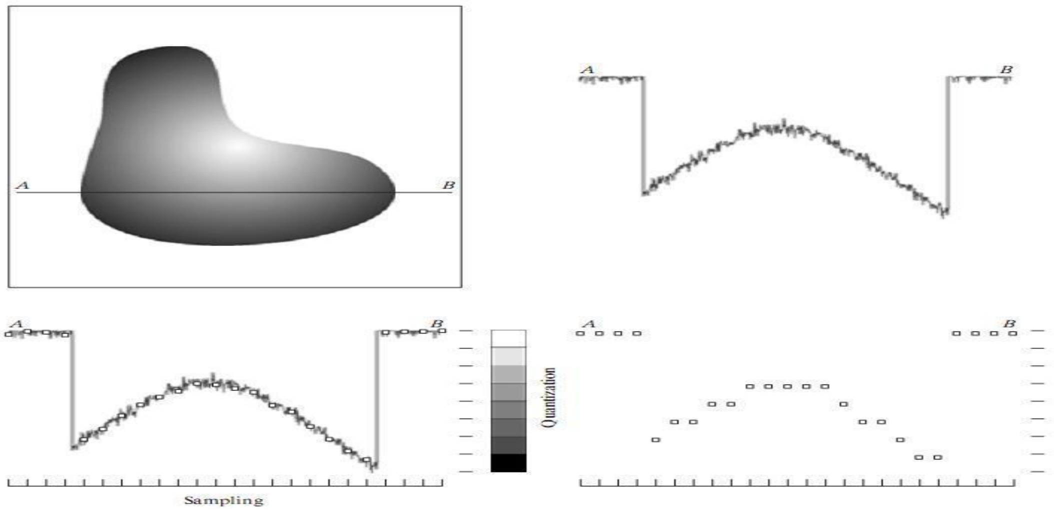


Fig.5.1. Generating a digital image. (a) continuous image (b)A scan line from A to B in the continuous image (c) sampling and quantization (d) Digital scan line.

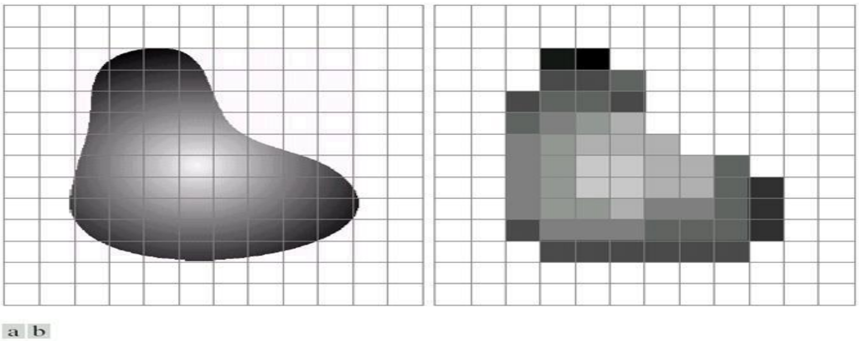


Fig.5.2 (a) Continuous image. (b) Result of sampling and quantization

Representing digital images

The result of sampling and quantization is a matrix of real numbers. Assume that 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. The values of the coordinates at the origin are (x, y) = (0, 0). Fig.5.3 shows the coordinate convention.

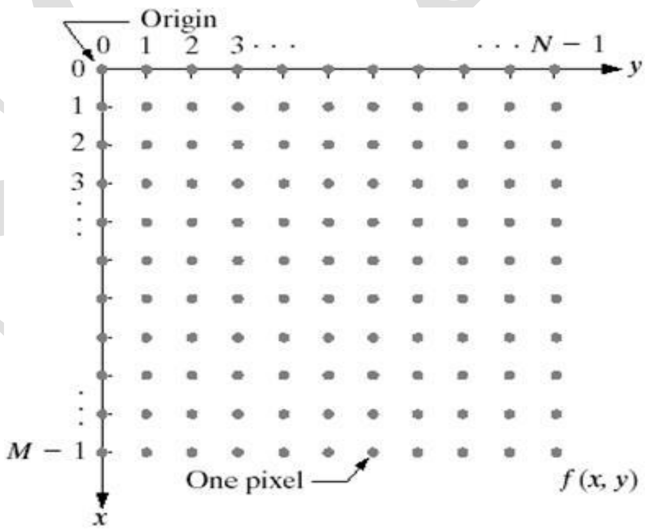


Fig.5.3 Coordinate convention to represent digital images

The complete digital image in matrix form can be represented as

$$\begin{matrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ \vdots & & \ddots & \vdots \\ f(N-1,0) & f(N-1,1) & \cdots & f(N-1,M-1) \end{matrix}$$

Each element of this matrix array is called an image element, picture element, pixel or pel. Common practice is to let N and M be powers of two; $N=2^n$ and $M=2^k$ And $L=2^m$ where L denotes the number of gray levels.

The range of values spanned by the gray scale is called dynamic range. It is given by the interval [0,L-1]

The assumption here is that gray levels are equally space in the interval [0,L]

The number of bits, b, necessary to store the image is then

$$b = N*M*m$$

or if N = M

$$b = N^2 m$$

For example, a 128x128 image with 64 gray levels would require 98,304 bits of storage.

Spatial and Gray level resolution:

Spatial resolution:

It is the smallest discernible detail in an Image. The more pixels in a fixed range, the higher the resolution.

Aliasing and Moiré patterns:

The Shannon's sampling theorem tells that, if a function is sampled at a rate equal to or greater than twice its higher frequency it is possible to recover completely the original functions from the samples. If the function is under sampled, then a phenomenon called aliasing corrupts the sampled image, the corruption in the form of additional frequency components being introduced into the sampling function. These are called aliased frequencies. The effect of aliased frequencies can be seen under right conditions in the form of Moiré patterns. Aliasing effect can be decreased by reducing the high frequency components. This is done by blurring or smoothing the image before sampling.

6. Explain about the Color Model? [APRIL/MAY 2017, NOV/DEC 2016]

Color models provide a standard way to specify a particular color, by defining a 3D coordinate system, and a subspace that contains all constructible colors within a particular model. Any color that can be specified using a model will correspond to a single point within the subspace it defines. Each color model is oriented towards either specific hardware (RGB, CMY, YIQ), or image processing applications (HSI).

Hardware oriented models:

- **RGB** (red, green, blue): Monitor video camera.
- **CMY** (cyan, magenta, yellow), **CMYK** (CMY, black) model for color printing.
- **HSI** model, which corresponds closely with the way humans describe and interpret color.

Application oriented models

- These models are used in applications where color manipulation is a goal
- One example is the creation of color graphics for animation

The RGB Model

In the RGB model, an image consists of three independent image planes, one in each of the primary colors: red, green and blue. (The standard wavelengths for the three primaries are as shown in figure). Specifying a particular color is by specifying the amount of each of the primary components present. Figure 6.1 shows the geometry of the RGB color model for specifying colors using a Cartesian coordinate system. The grayscale spectrum, i.e. those colors made from equal amounts of each primary, lies on the line joining the black and white vertices.

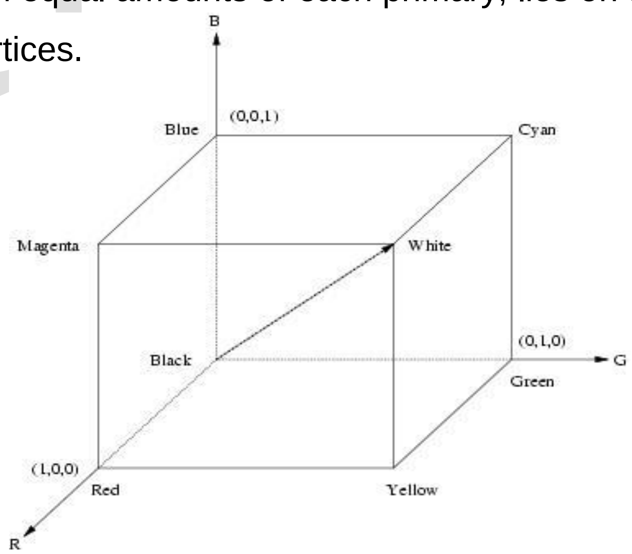


Fig.6.1 The RGB color cube. The gray scale spectrum lies on the line joining the black and white vertices.

This is an *additive* model, i.e. the colors present in the light add to form new colors, and is appropriate for the mixing of colored light for example. The image on the left of figure 6.2 shows the additive mixing of red, green and blue primaries to form the three secondary colors yellow (red + green), cyan (blue + green) and magenta (red + blue), and white ((red + green + blue). The RGB model is used for color monitors and most video cameras.

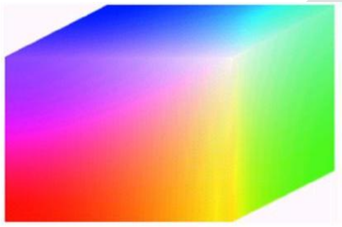


Fig.6.2 RGB 24 bit color cube

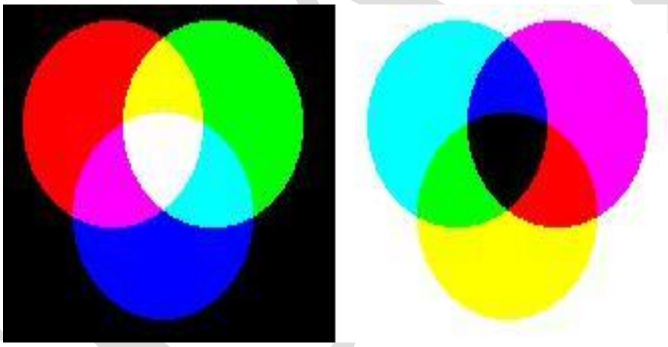


Fig.6.3 The figure on the left shows the additive mixing of red, green and blue primaries to form the three secondary colors yellow (red + green), cyan (blue + green) and magenta (red + blue), and white ((red + green + blue). The figure on the right shows the three subtractive primaries, and their pair wise combinations to form red, green and blue, and finally black by subtracting all three primaries from white.

Pixel Depth:

The number of bits used to represent each pixel in the RGB space is called the pixel depth. If the image is represented by 8 bits then the pixel depth of each RGB color pixel = 3*number of bits/plane=3*8=24

A full color image is a 24 bit RGB color image. Therefore total number of colors in a full color image = $(2^8)^3$ = 16,777,216

Safe RGB colors:

Many applications use only a few hundred or fewer colors. Therefore most of the system in use today is limited to 256 colors. Such systems can have a subset of colors which can be reproduced without depending on the hardware capabilities of the system. This subset of colors is called the set of RGB colors or the set of all systems safe colors. In internet applications, these colors are called the safe web colors or safe browser colors.

Standard safe colors:

It is assumed that a minimum number of 256 colors can be reproduced faithfully by any system. Among these, 40 colors are found to be processed differently by different operating system. Therefore the remaining 216 colors are accepted to be the standard safe colors.

Component values of safe colors:

Each of the 216 safe colors can be formed from three RGB component values. But each component value should be selected only from the set of values {0, 51,102,153,204,255}, in which the successive numbers are obtained by adding 51 and are divisible by 3 therefore total number of possible values= 6*6*6=216

Hexadecimal representation

The component values in RGB model should be represented using hexadecimal number system. The decimal numbers 1,2,….14,15 correspond to the hex numbers 0,1,2,….9,A,B,C,D,E,F. the equivalent representation of the component values is given in table 1.1

Table 1.1 Valid values of RGB components

Decimal	Hexadecimal
0	00
51	33
102	66
153	99
204	CC
255	FF

Applications:

- Color monitors, Color video cameras

Advantages:

- Image color generation
- Changing to other models such such as CMY is straight forward
- It is suitable for hardware implementation
- It is based on the strong perception of human vision to red, green and blue primaries.

Disadvantages:

- It is not acceptable that a color image is formed by combining three primary colors.
- This model is not suitable for describing colors in a way which is practical for human interpretation.

The CMY Model

The CMY (cyan-magenta-yellow) model is a *subtractive* model appropriate to absorption of colors, for example due to pigments in paints. Whereas the RGB model asks what is added to black to get a particular color, the CMY model asks what is subtracted from white. In this case, the primaries are cyan, magenta and yellow, with red, green and blue as secondary colors.

When a surface coated with cyan pigment is illuminated by white light, no red light is reflected, and similarly for magenta and green, and yellow and blue. The relationship between the RGB and CMY models is given by:

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

The CMY model is used by printing devices and filters.

The HSI Model

As mentioned above, color may be specified by the three quantities hue, saturation and intensity which is similar to the way of human interpretation.

Hue: It is a color attribute that describes a pure color.

Saturation: It is a measure of the degree to which a pure color is diluted by white light.

Intensity:

It is a measureable and interpretable descriptor of monochromatic images, which is also called the gray level.

(1) To find Intensity:

The intensity can be extracted from an RGB image because an RGB color image is viewed as three monochrome intensity images.

Intensity Axis:

A vertical line joining the black vertex (0, 0, 0) and white vertex(1,1, 1) is called the intensity axis. The intensity axis represents the gray scale.

Determining intensity component

The intensity component of any color is determined as

- A plane which is perpendicular to the intensity axis and containing the color point is passed through the cube.
- The point at which the plane intersects the intensity axis gives the intensity value.
- The intensity value will be in the range [0, 1]

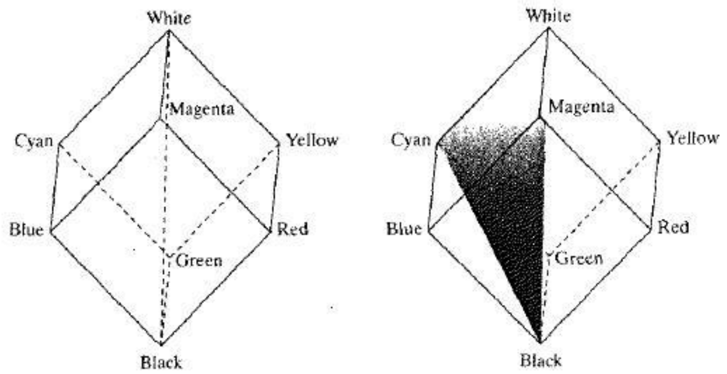


Fig.6.4 The HIS Model

(2) To find saturation:

- All points on the intensity axis are gray which means that the saturation i.e., purity of points on the axis is zero.
- When the distance of a color from the intensity axis increases, the saturation of that color also increases.

(3) To Find Hue:

The hue of a color can also be determined from the RGB color cube because, it is clear from the RGB color model that, if three points namely black, white and any one color are joined, a triangle is formed. All the points inside the triangle will have the same hue. This is due to the fact that black and white components cannot change the hue. But, intensity and saturation of points inside the triangle will be different.

(4) HSI color space

The HIS color space is represented by

- A vertical intensity axis and
- The locus of color points that lie on planes perpendicular to the axis

The shape of the cube is defined by the intersecting points of these planes with the faces of cube. As the planes move up and down along the intensity axis, the shape can either be a triangle or a hexagon. The hexagon shaped plane and triangle plane is shown in fig. 1.23

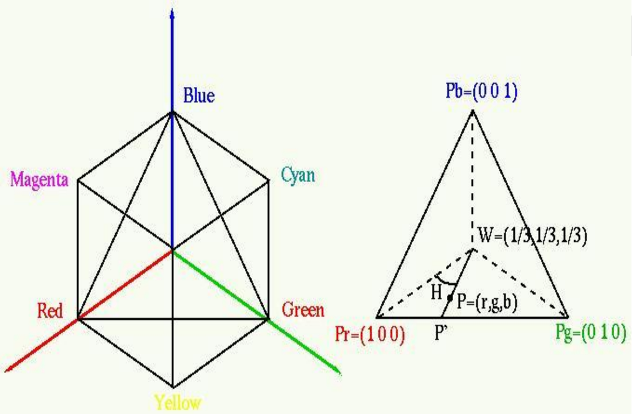


Fig.6.5 HIS color space

In HSI space,

- The primary colors are separated by 120°.
- The secondary colors are also separated by 120°.
- The angle between the secondary's and primaries is 60°.

Representation of Hue:

- The hue of a color point is determined by an angle from some reference point.
- Almost in all cases, the red axis is selected as the reference.
- Therefore, if the angle between the point and the red axis is 0°, it represents zero hue.

- The hue increases if the angle from red axis increases in the counter clock wise direction

Representation of saturation

- The saturation is described as the length from the vertical axis.
- In the HSI space, it is represented by the length of the vector from the origin to the color point.
- If the length is more the saturation is high and vice versa.

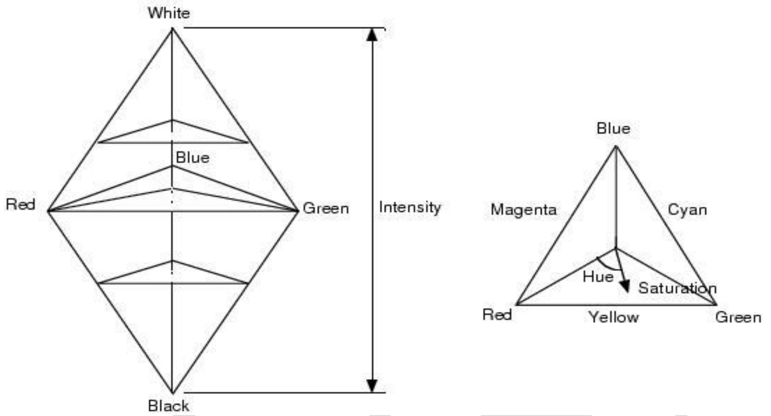


Fig.6.6 The HSI model, showing the saturation and Hue calculation.

Fig. 6.6 shows the HSI model, with HSI solid on the left, and the HSI triangle on the right, formed by taking a horizontal slice through the HSI solid at a particular intensity. Hue is measured from red, and saturation is given by distance from the axis. Colors on the surface of the solid are fully saturated, i.e. pure colors, and the grayscale spectrum is on the axis of the solid. For these colors, hue is undefined.

Components of HSI color space

- The vertical intensity axis
- The length of the vector S from the origin to a color point.
- The angle H between the vector and the red axis.

Advantages of HSI model:

- It describes colors in terms that are suitable for human interpretation.
- The model allows independent control over the color describing quantities namely hue saturation and intensity.
- It can be used as an ideal tool for developing image processing algorithms based on color descriptions.

UNIT-II

IMAGE ENHANCEMENT

Spatial Domain: Gray level transformations – Histogram processing – Basics of Spatial Filtering– Smoothing and Sharpening Spatial Filtering – Frequency Domain: Introduction to Fourier Transform – Smoothing and Sharpening frequency domain filters – Ideal, Butterworth and Gaussian filters.

PART A

2 Marks

1. Specify the objective of image enhancement technique. [APR/ MAY 2017]

The objective of enhancement technique is to process an image so that the result is more suitable than the original image for a particular application.

2. Name the different types of derivative filters?

1. Perwitt operators
2. Roberts cross gradient operators
3. Sobel operators

3. What is contrast stretching?

Contrast stretching reduces an image of higher contrast than the original by darkening the levels below m and brightening the levels above m in the image.

4. What is grey level slicing?

Highlighting a specific range of grey levels in an image often is desired. Applications include enhancing features such as masses of water in satellite imagery and enhancing flaws in x-ray images.

5. Define image subtraction.

The difference between 2 images f(x,y) and h(x,y) expressed as,

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

is obtained by computing the difference between all pairs of corresponding pixels from f and h

6. What is the purpose of image averaging

An important application of image averaging is in the field of astronomy, where imaging with very low light levels is routine, causing sensor noise frequently to render single images virtually useless for analysis.

7. What is meant by masking? [NOV DEC 2016]

Mask is the small 2-D array in which the values of mask co-efficient determines the nature of process. The enhancement technique based on this type of approach is referred to as mask processing.

8. Give the formula for negative and log transformation.

Negative: S=L-1-r

Log: S = c log(1+r)

Where c-constant and r=0

9. What is meant by bit plane slicing? [NOV DEC 2016]

Instead of highlighting gray level ranges, highlighting the contribution made to total image appearance by specific bits might be desired. Suppose that each pixel in an image is represented by 8 bits. Imagine that the image is composed of eight 1-bit planes, ranging from bit plane 0 for LSB to bit plane-7 for MSB.

10. Define histogram.

The histogram of a digital image with gray levels in the range [0, L-1] is a discrete function h(rk)=nk.

rk-kth gray level

nk-number of pixels in the image having gray level rk.

PART B

16 MARKS

1) Discuss in detail about Histogram Processing in detail.

OR

Explain the histogram equalization method of image enhancement. [MAY/JUNE 2016, NOV DEC 2016]]

Definition of the histogram of an image.

The histogram of a digital image with gray levels in the range [0.L-1] is a discrete function $h(r_k)= n_k$ where r_k is the k^{th} gray level and n_k is the number of pixels in the image having gray level r_k

Normalized histogram is given by

$$P(r_k) = n_k/n, \quad k=0,1,\ldots.L-1.$$

$P(r_k)$ is the estimate of probability of occurrence of gray level r_k

The horizontal axis of each histograms corresponds to gray level values r_k

The vertical axis corresponds to values of $h(r_k)= n_k$ or $P(r_k) = n_k/n$

By processing (modifying) the histogram of an image we can create a new image with specific desired properties.

For dark image, the components of histogram are concentrated on the low or dark side of gray scale.

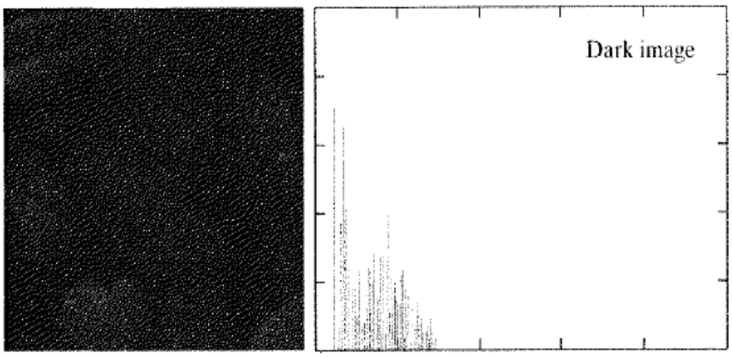


Fig Dark image and its histogram

For bright image, the components of histogram are concentrated on the high side of gray scale.

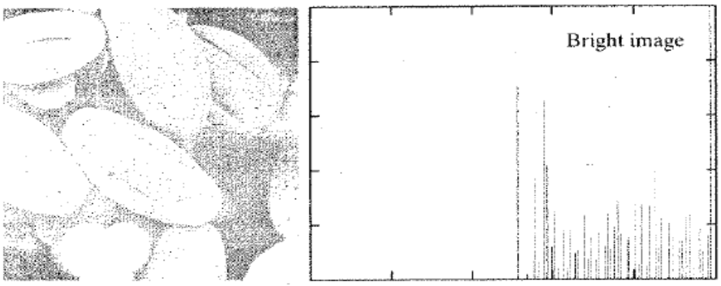


Fig: Bright image and its histogram

For low contrast image, the histogram is narrow and is at the middle.

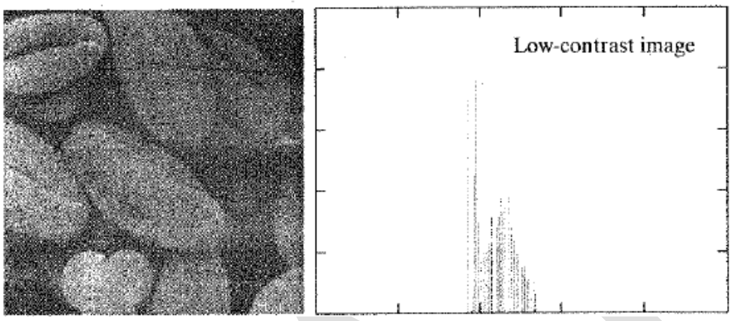


Fig: Low contrast image and its histogram

For high contrast image, histogram extends a broad range and is uniform throughout.

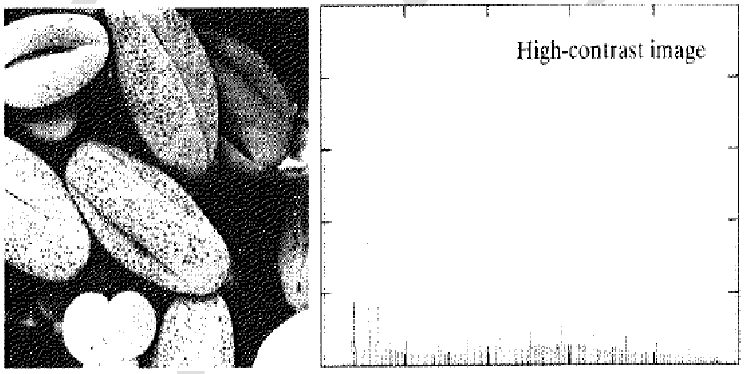
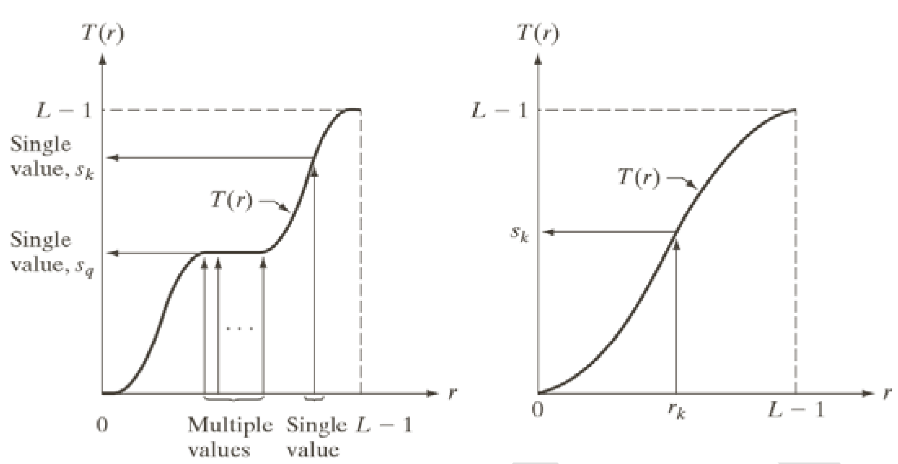


Fig: High contrast image and its histogram

Global histogram equalization

In this section we will assume that the image to be processed has a continuous intensity that lies within the interval [0, L-1]. Suppose we divide the image intensity with its maximum valueL-1. Let the variable 'r' represent the new grey levels (image intensity) in the image, where now $0 \le r \le 1$ and let $p_r(r)$ denote the probability density function (pdf) of the variable r . We now apply the following transformation function to the intensity.

$$s = T(r) = \int_0^r p_r(w)dw, \quad 0 \le r \le 1$$



Equlization Transformation Functin

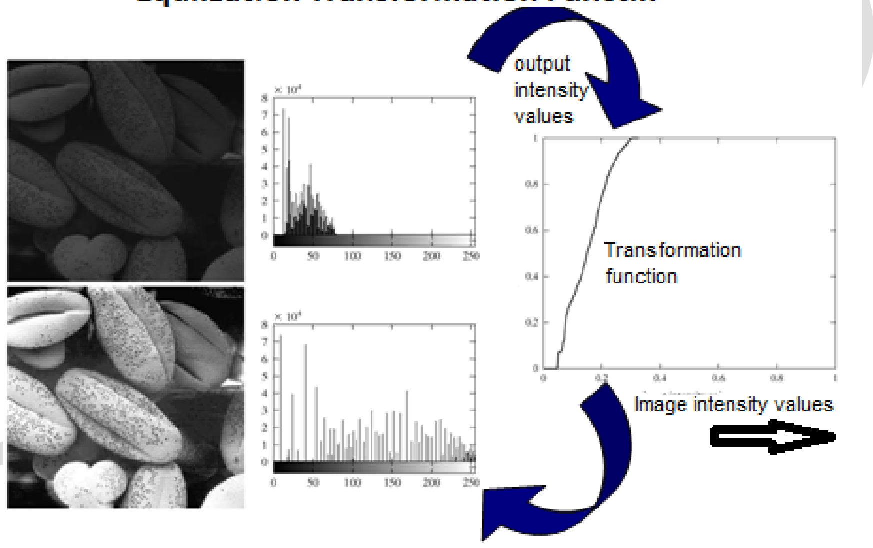


Fig Transformation function

By observing the transformation of equation (1) we immediately see that it possesses the following properties:

(i) T(r) should be single valued and monotonically increasing in the interval

$0 \le r \le 1$. Single valued function guarantees that the inverse transformation will also exist. Monotonicity condition preserves the increasing order of gray levels from black to white in the output image.

(ii) $0 \le T(r) \le 1$, for, $0 \le r \le 1$ i.e., the input and output gray levels are same.

Inverse transformation:

The inverse transformation from s to r is denoted by $r=T^{-1}(s)$, $0 \le s \le 1$

Probability Density Function(PDF)

Suppose that $P_r(r)$, $P_s(s)$ are the probability distribution functions (PDF's) of the variables r and s respectively.

If $p_r(r)$ and $T(r)$ are known and $T^{-1}(s)$ satisfies condition (i), then the PDF of the transformed variable s is obtained by,

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

Cumulative Distribution Function(CDF)

The CDF of the random variable r is given by,

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

w is dummy variable for integration

To find ps(s)

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

$$= p_r(r)$$

Substituting (5) in (3) gives,

$$p_s(s) = \left[p_r(r)\frac{1}{p_r(r)}\right]_{r=T^{-1}(s)} = 1,\ 0 \le s \le 1$$

From the above analysis it is obvious that the transformation of equation (1) converts the original image into a new image with uniform probability density function. Unfortunately, in a real life scenario we must deal with digital images. The discrete form of histogram equalization is given by the relation

$$s_k = T(r_k) = \sum_{j=0}^{k}\frac{n_j}{n} = \sum_{j=0}^{k} p_r(r_j),\ 0 \le r_k \le 1,\ k = 0,1,\ldots,L-1$$

Thus mapping each pixel with level rk in the input image into a corresponding pixel with level sk in the output image using eqn.(7) will produce an enhanced image. The plat of $p_r(r_k)$ versus rk is known as a histogram and the transformation or mapping given in eqn.(7) is called histogram equalization or histogram linearization. The inverse transformation is given by,

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

Advantages:

- It is easy to implement
- The information it needs can be obtained directly from the given image and no additional parameter specifications are required.
- The results from this technique are predictable and It is fully automatic.

Local histogram equalization

Global histogram equalization is suitable for overall enhancement. It is often necessary to enhance details over small areas. The number of pixels in these areas may have negligible influence on the computation of a global transformation, so the use of this type of transformation does not necessarily guarantee the desired local enhancement. The solution is to devise transformation functions based on the grey level distribution – or other properties – in the neighborhood of every pixel in the image. The histogram processing technique previously described is easily adaptable to local enhancement. The procedure is to define a square or rectangular neighborhood and move the centre of this area from pixel to pixel. At each location the histogram of the points in the neighborhood is computed and a histogram equalization transformation function is obtained. This function is finally used to map the grey level of the pixel centered in the neighborhood. The centre of the neighbourhood region is then moved to an adjacent pixel location and the procedure is repeated. Since only one new row or column of the neighborhood changes during a pixel-to-pixel translation of the region, updating the histogram obtained in the previous location with the new data introduced at each motion step is possible quite easily. This approach has obvious advantages over repeatedly computing the histogram over all pixels in the neighborhood region each time the region is moved one pixel location. Another approach often used to reduce computation is to utilize non overlapping regions, but this methods usually produces an undesirable checkerboard effect.

Histogram specification or histogram matching:

The method used to generate a processed image that has a specified histogram is called histogram matching or histogram specification. Suppose we want to specify a particular histogram shape (not necessarily uniform) which is capable of highlighting certain grey levels in the image.

Let us suppose that:

$p_r(r)$ is the original probability density function

$p_z(z)$ is the desired probability density function

Suppose that histogram equalization is first applied on the original image r

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

Suppose that the desired image z is available and histogram equalization is applied as well

$$v = G(z) = \int_0^z p_z(t)dt = s$$

Get the inverse transformation function G^{-1}

Apply the following equation to all the pixels in the input image to get the output image. The inverse process

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

Therefore, the process of histogram specification can be summarized in the following steps.

(i) We take the original image and equalize its intensity using the relation

$$s = T(r) = \int_0^r p_r(w)dw.$$

(ii)From the given probability density function$P_Z(Z)$ we specify the probability distribution function G(Z).

(iii) We apply the inverse transformation function

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

2) Discuss in detail about Gray level transformations.

How color image is enhanced and compare it with gray scale processing? [APRIL/MAY 2015]

How color image is enhanced and compare it with gray scale processing? [APRIL/MAY 2015]

Here, T is a transformation that maps a pixel value r into a pixel value s. Since we are concerned with digital data, the transformation can generally be implemented with a simple lookup table.

Three basic types of transformations

- Linear (negative and identity transformations)
- Logarithmic (log and inverse-log transformations)
- Power-law (nth power and nth root transformations)

(i)Image Negative

The negative of an image with intensity levels in the range [0,L-1] can be described by: $s = L - 1 - r$

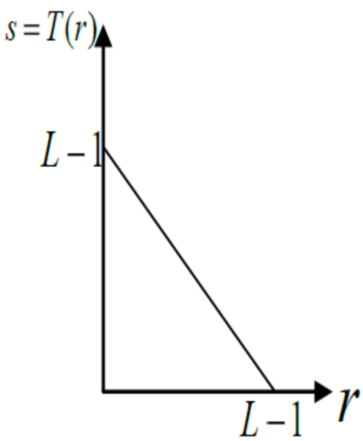


Fig: Image Negative

(ii)Log Transformations

General form: s= *c* log(1 / *r*) *c* is a constant and r≥0 Maps a narrow range of low intensity values in input to a wider output range. The opposite is true for high intensity input values. Compresses the dynamic range of images with large variations in pixel values.

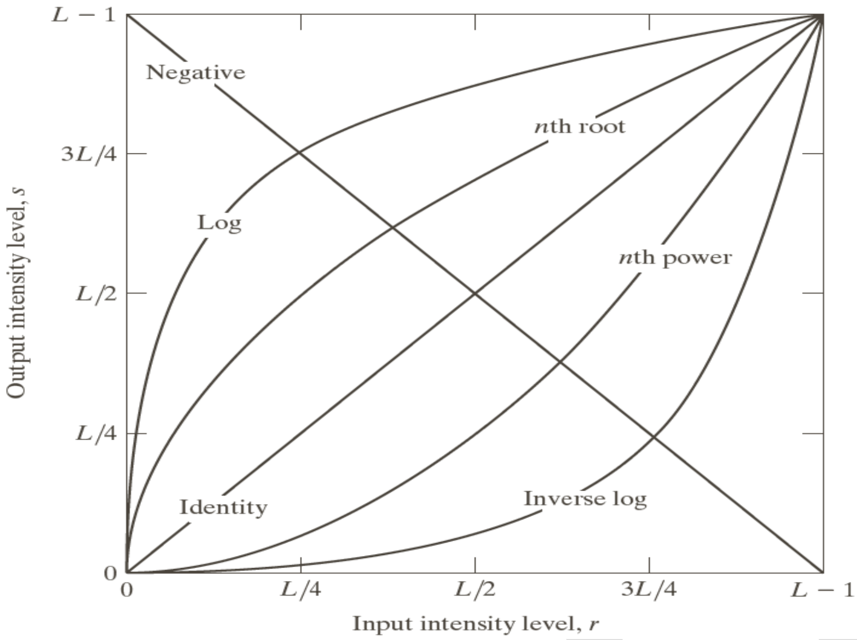


Fig. Basic gray level transformation functions

(iii)Power-Law (Gamma) Transformations:

Basic form $S=cr^{\gamma}$

Where *c* and γ are positive constants. Power- law curves with fractional values of γ map a narrow range of dark input values to a wider range of output values. The opposite is true for higher values of input levels exists a family of possible transformation curves by varying γ

Power-Law Transformation Curves Gamma Correction

Many devices used for image capture, display and printing respond according to a power law The exponent in the power-law equation is referred to as *gamma.* The process of correcting for the power-law response is referred to as *gamma correction*

Example:

CRT devices have an intensity-to-voltage response that is a power function (exponents typically range from 1.8-2.5). Gamma correction in this case could be achieved by applying the transformation $s= r^{1/2.5}= r^{0.4}$

(iv)Piecewise-Linear Transformations

Piecewise functions can be arbitrarily complex. A disadvantage is that their specification requires significant user input.

Example functions: Contrast stretching, Intensity-level slicing, Bit-plane slicing.

Contrast Stretching

Contrast stretching expands the range of intensity levels in an image so it spans a given (full) intensity range Control points (r1,s1) and (r2,s2) control the shape of the transform T(r) r1=r2, s1=0 and s2=L-1 yields a *thresholding function*

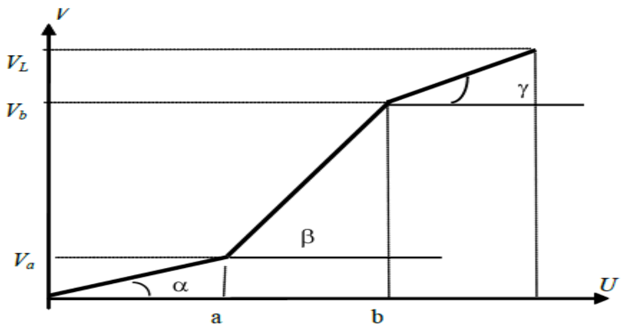
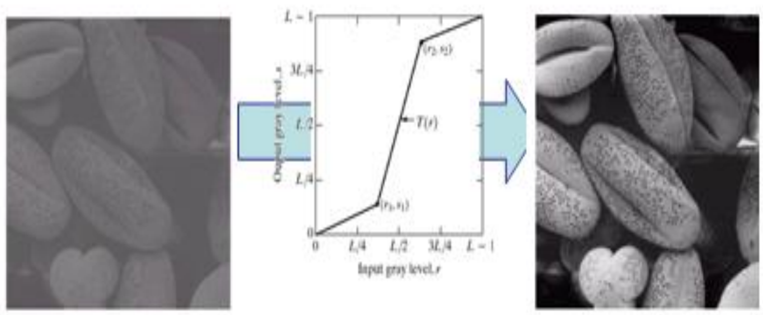


Fig. Contrast Stretching



Low Contrast Image Contrast stretched image

Intensity-level Slicing

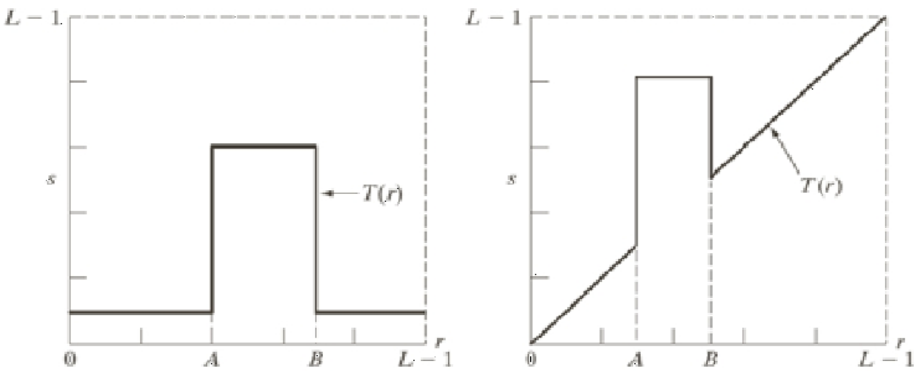


Fig. Intensity-level Slicing

Used to highlight a specific range of intensities in an image that might be of interest. Two common approaches. Set all pixel values within a range of interest to one value (white) and all others to another value (black), Produces a binary image. Brighten (or darken) pixel values in a range of interest and leave all others unchanged.

Bit plane Slicing

Instead of highlighting gray level images, highlighting the contribution made to total image appearance by specific bits might be desired. Suppose that each pixel in an image is represented by 8 bits. Imagine the image is composed of 8, 1-bit planes ranging from bit plane1-0 (LSB)to bit plane 7 (MSB). In terms of 8-bits bytes, plane 0 contains all lowest order bits in the bytes comprising the pixels in the image and plane 7 contains all high order bits. Separating a digital image into its bit planes is useful for analyzing the relative importance played by each bit of the image, implying, it determines the adequacy of numbers of bits used to quantize each pixel , useful for image compression.

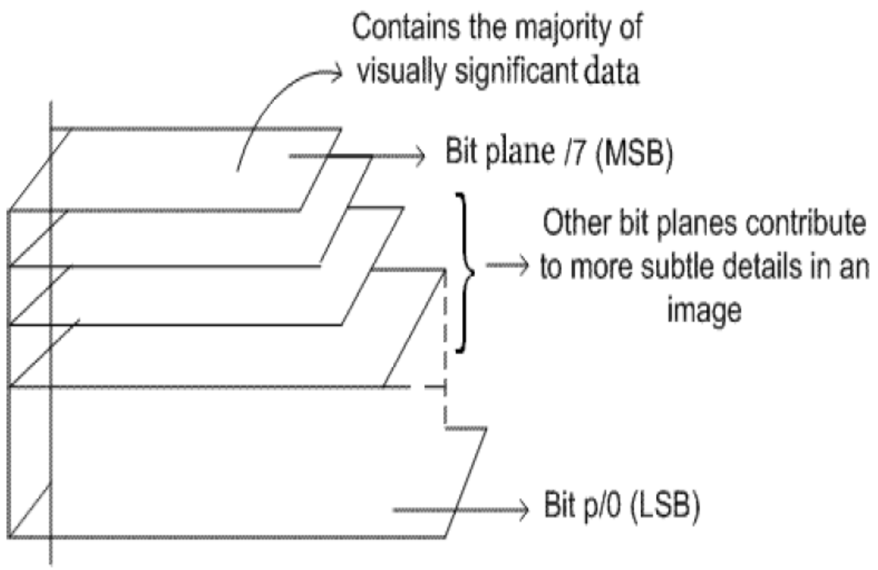


Fig: Bit plane representation

In terms of bit-plane extraction for a 8-bit image, it is seen that binary image for bit plane 7 is obtained by proceeding the input image with a thresholding gray-level transformation function that maps all levels between 0 and 127 to one level (e.g 0)and maps all levels from 129 to 253 to another (eg. 255).

3) What is meant by Spatial Filtering? Discuss in detail about Smoothing and Sharpening Spatial Filtering? [APR MAY 2017, NOV DEC 2016]

or

What are the spatial transformation techniques used for image restoration? Explain them in detail. [MAY/JUNE 2016]

- Generally involves operations over the entire image
- Operations take place involving pixels within a neighborhood of a point of interest
- Also involves a predefined operation called a spatial filter
- The spatial filter is also commonly referred to as:
  - ➢ Spatial mask, Kernel, Template, Window

Linear Filters and Non linear filters based on the operation performed on the image. Filtering means accepting (passing) or rejecting some frequencies. Mechanics of spatial filtering

$$f(x, y) \rightarrow \boxed{\text{Filter}} \rightarrow g(x, y)$$

	Col -1	*Col 0*	*Col 1*
Row -1	*-1,-1*	*-1,0*	*-1,-1*
Row 0	*0,-1*	*0,0*	*0,1*
Row 1	*1,-1*	*1,0*	*1,1*

At any point (x,y) in the image, the response g(x,y) of the filter is the sum of products of the filter coefficients and the image response and the image pixels encompassed by the filter.

Observe that he filter w(0,0) aligns with the pixel at location(x,y)

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

1) Smoothing Spacial Filter

Smoothing filter are used for blurring and for noise detection.

a) Smoothing Linear Filter

The output (response) of a smoothing, linear spatial filter is simply the average of the pixels contained in the neighborhood of the filter mask. These filters sometimes are called averaging filters. They also are referred to a lowpass filters.

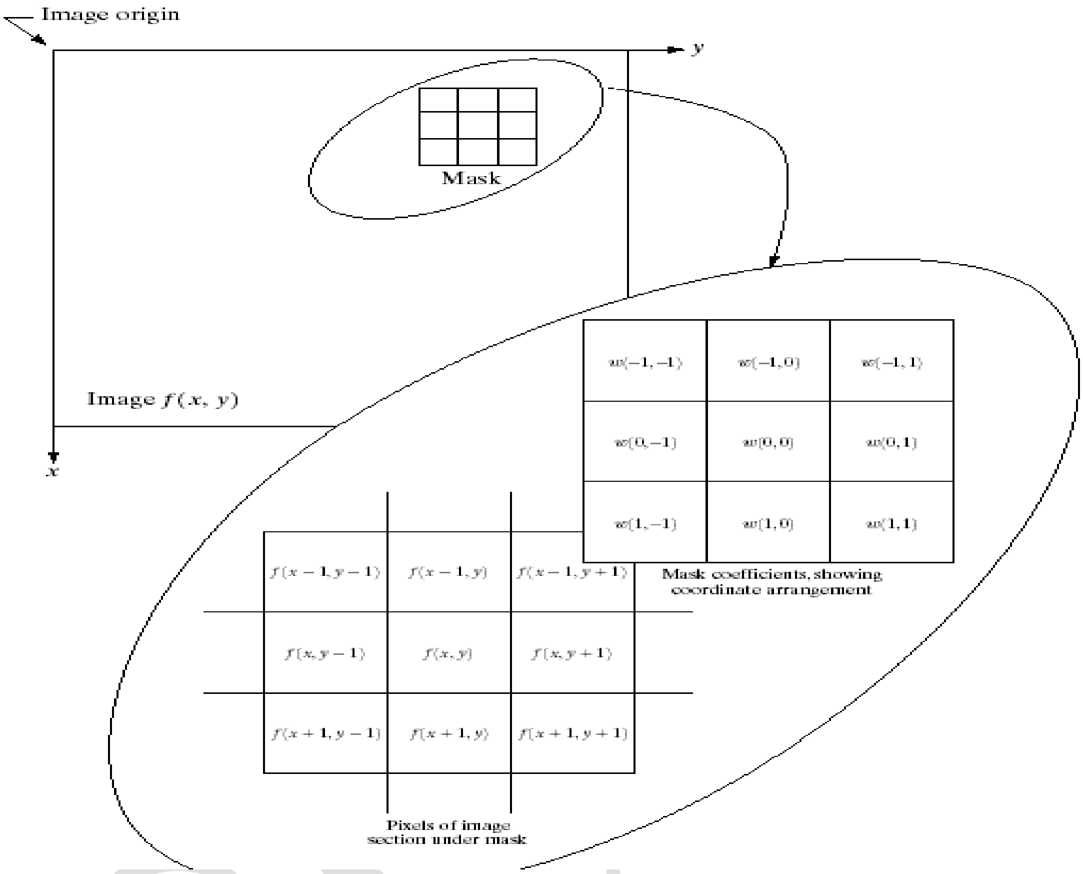


Fig: The mechanism of spacial filtering of 3x3 Mask

$$\frac{1}{9}\times\begin{bmatrix}1 & 1 & 1\\ 1 & 1 & 1\\ 1 & 1 & 1\end{bmatrix} \qquad \frac{1}{16}\times\begin{bmatrix}1 & 2 & 1\\ 2 & 4 & 2\\ 1 & 2 & 1\end{bmatrix}$$

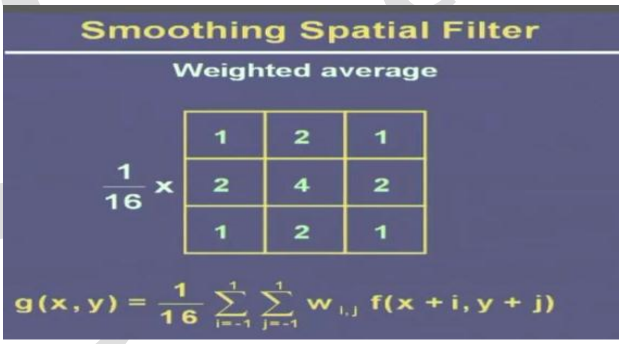
Fig: Two 3 X 3 smoothing filter masks. The constant multiplier is equal to the sum of the values of its coefficients as is required to compute an average

The idea behind smoothing filters is straightforward. By replacing the value of every pixel in an image by the average of the gray levels in the neighborhood defined by the filter mask, this process results in an image with reduced "sharp" transitions in gray levels. Because random noise typically consists of sharp transitions in gray levels, the most obvious application of smoothing is noise reduction.

Weighted Average mask: Central pixel usually have higher value. Weight age is inversely proportional to the distance of the pixel from centre of the mask. *T* the general implementation for filtering an MxN image with a weighted averaging filter of size m x n (m and *n* odd) is given by the expression, m=2a+1 and n=2b+1,where a and b are nonnegative integers.

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

An important application of spatial averaging is to blur an image for the purpose getting a gross representation of objects of interest, such that the intensity of smaller objects blends with the background; after filtering and thresholding.



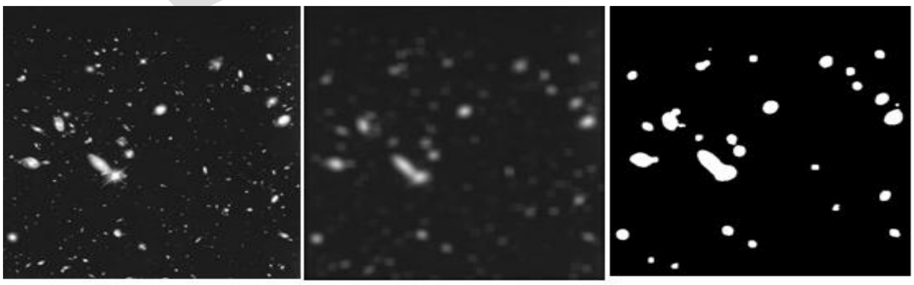


Fig: (a) Image from the Hubble Space Telescope (b) Image Processed by a 15 x15 averaging mask. (c) Results of the Thresholding

Examples of Low Pass Masks (Local Averaging)

$$\frac{1}{9}\times\begin{array}{|c|c|c|}\hline 1&1&1\\\hline 1&1&1\\\hline 1&1&1\\\hline\end{array}\qquad \frac{1}{25}\times\begin{array}{|c|c|c|c|c|}\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline\end{array}\qquad \frac{1}{49}\times\begin{array}{|c|c|c|c|c|c|c|}\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline\end{array}$$

Popular techniques for lowpass spatial filtering

Uniform filtering

The most popular masks for low pass filtering are masks with all their coefficients positive and equal to each other as for example the mask shown below. Moreover, they sum up to 1 in order to maintain the mean of the image.

$$\frac{1}{9}\times\begin{array}{|c|c|c|}\hline 1&1&1\\\hline 1&1&1\\\hline 1&1&1\\\hline\end{array}$$

Gaussian filtering

The two dimensional Gaussian mask has values that attempts to approximate the continuous function.

b) Order-Statistics (non linear)Filters

The best-known example in this category is the M*edian filter*, which, as its name implies, replaces the value of a pixel by the median of the gray levels in the neighborhood of that pixel (the original value of the pixel is included in the computation of the median). Order static filter / (non-linear filter) / median filter Objective: Replace the valve of the pixel by the median of the intensity values in the neighbourhood of that pixel

Although the median filter is by far the most useful order-statistics filter in image processing, it is by no means the only one. The median represents the 50th percentile of a ranked set of numbers, but the reader will recall from basic statistics that ranking lends itself to many other possibilities. For example, using the 100th percentile results in the so-called *max filter*, which is useful in finding the brightest points in an image. The response of a 3*3 max filter is given by R=max [zk| k=1, 2, ,… 9]

The 0th percentile filter is the *min filter*, used for the opposite purpose.

2) Sharpening Spatial Filters

In the last section, we saw that image blurring could be accomplished in the spatial domain by pixel averaging in a neighborhood.

Since averaging is analogous to integration, it is logical to conclude that sharpening could be accomplished by spatial differentiation. This, in fact, is the case, and the discussion in this section deals with various ways of defining and implementing operators for sharpening by digital differentiation.

Fundamentally, the strength of the response of a derivative operator is proportional to the degree of discontinuity of the image at the point at which the operator is applied. Thus, image differentiation enhances edges and other discontinuities (such as noise) and deemphasizes areas with slowly varying gray-level values.

For first derivative (1) must be zero in flat segments (areas of constant gray-level values); (2) must be nonzero at the onset of a gray-level step or ramp; and (3) must be nonzero along ramps. Similarly, any definition of a second derivative (1) must be zero in flat areas; (2) must be nonzero at the onset and end of a gray-level step or ramp; and (3) must be zero along ramps of constant slope.

A basic definition of the first-order derivative of a one-dimensional function f(x) is the difference.

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

Similarly, we define a second-order derivative as the difference

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

Let us consider the properties of the first and second derivatives as we traverse the profile from left to right. First, we note that the first-order derivative is nonzero along the entire ramp, while the second-order derivative is nonzero only at the onset and end of the ramp. Because edges in an image resemble this type of transition, we conclude that first-order derivatives produce “thick” edges and second-order derivatives, much finer ones

A second-order derivative to enhance fine detail (including noise) much more than a first-order derivative.

(1) First-order derivatives generally produce thicker edges in an image.

(2) Second-order derivatives have a stronger response to fine detail, such as thin lines and isolated points.

(3) First order derivatives generally have a stronger response to a gray-level step.

(4) Second- order derivatives produce a double response at step changes in gray level.

Use of Second Derivatives for Enhancement–The Laplacian

We are interested in *isotropic* filters, whose response is independent of the direction of the discontinuities in the image to which the filter is applied. In other words, isotropic filters are *rotation invariant*, in the sense that rotating the image and then applying the filter gives the same result as applying the filter to the image first and then rotating the result.

Development of the method

It can be shown (Rosenfeld and Kak [1982]) that the simplest isotropic derivative operator is the *Laplacian*, which, for a function (image) f(x, y) of two variables, is defined as

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

Because derivatives of any order are linear operations, the Laplacian is a linear operator.

In order to be useful for digital image processing, this equation needs to be expressed in discrete form.

We use the following notation for the partial second-order derivative in the *x*-direction:

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

and, similarly in the *y*-direction, as

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

The digital implementation of the two-dimensional Laplacian is obtained by summing these two components:

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

This equation can be implemented using the mask gives an isotropic result for rotations in increments of 90°. The diagonal directions can be incorporated in the definition of the digital Laplacian by adding two more terms, one for each of the two diagonal directions. Since each diagonal term also contains a –2f(x, y) term, the total subtracted from the difference terms now would be –8f(x, y).

Laplacian operator

The main disadvantage of the Laplacian operator is that it produces double edges. Because the Laplacian is a derivative operator, its use highlights gray-level discontinuities in an image and deemphasizes regions with slowly varying gray levels. This will tend to produce images that have grayish edge lines and other discontinuities, all superimposed on a dark, featureless background. Background features can be "recovered" while still preserving the sharpening effect of the Laplacian operation simply by adding the original and Laplacian images. As noted in the previous paragraph, it is important to keep in mind which definition of the Laplacian is used. If the definition used has a negative center coefficient, then we *subtract*, rather than add, the Laplacian image to obtain a sharpened result. Thus, the basic way in which we use the Laplacian for image enhancement is as follows

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

(a) Filter mask used to implement the digital Laplacian,

(b) Mask used to implement an extension that includes the diagonal neighbors.

(c) and(d) Two other implementations of the Laplacian.

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

Unsharp masking and high-boost filtering

A process used for many years in the publishing industry to sharpen images consists of subtracting a blurred version of an image from the image itself. This process, called *un sharp masking*, is expressed as

$$fs(x, y) = f(x, y) - b(x, y)$$

where *f*s(x, y) denotes the sharpened image obtained by un sharp masking, and b(x, y) is a blurred version of f(x, y).

A slight further generalization of un sharp masking is called *high-boost* *filtering*. A high- boost filtered image, *f*hb, is defined at any point (x, y) as

$$fhb(x, y) = Af(x, y) – b(x, y)$$

where A _ 1 and, as before, b is a blurred version of f. This equation may be written as

$$fhb(x, y) = (A - 1)f(x, y) + f(x, y) - b(x, y).$$

$$fhb(x, y) = (A - 1)f(x, y) + fs(x, y)$$

as the expression for computing a high-boost-filtered image.

One of the principal applications of boost filtering is when the input image is darker than desired. By varying the boost coefficient, it generally is possible to obtain an overall increase in average gray level of the image, thus helping to brighten the final result.

4) What is meant by Frequency Filtering? Discuss in detail about Smoothing and Sharpening frequency Filtering?

or

Compare the various filters available under frequency domain for image enhancement. [MAY/JUNE 2016]

Edges and other sharp transitions (such as noise) in the gray levels of an image contribute significantly to the high-frequency content of its Fourier transform. Hence smoothing (blurring is achieved in the frequency domain by attenuating a specified range of high-frequency components in the transform of a given image.

Our basic “model” for filtering in the frequency domain is

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

where F(u, v) is the Fourier transform of the image to be smoothed. The objective is to select a filter transfer function H(u, v) that yields G(u, v) by attenuating the high-frequency components of F(u, v).

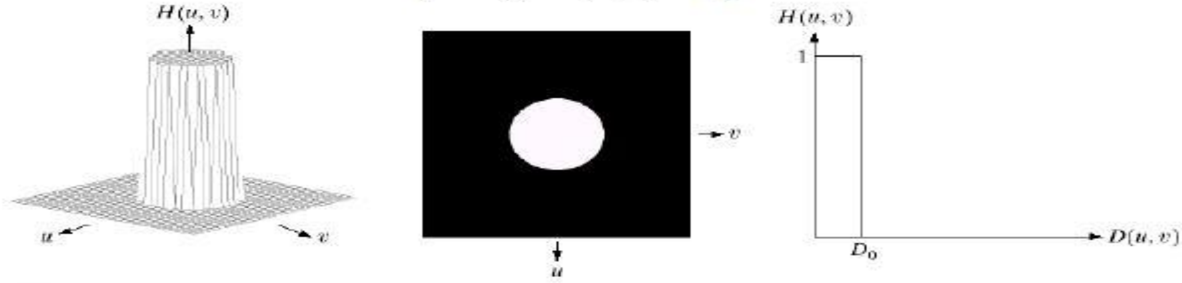
Ideal Lowpass Filters

The simplest lowpass filter we can envision is 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 (centered) transform. Such a filter is called a two-dimensional (2-D) ideal lowpass filter

(**ILPF**) and has the transfer function

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

D(u,v) is the distance from the origin
D_0 is the cutoff frequency

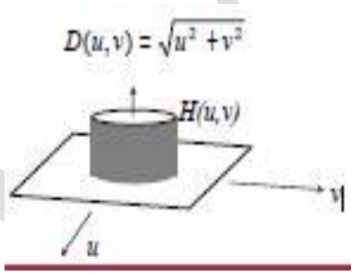


a b c

(a) Perspective plot of an ideal lowpass filter transfer function. (b) Filter displayed as an image. (c) Filter radial cross section.

where Do is a stated nonnegative quantity (the cutoff frequency) and D(u,v) is the distance from the point (u,v) to the center of the frequency plane.If the image in question is of size M x N, we know that its transform also is of this size, so the center of the frequency rectangle is at (u, v) = (M/2, N/2) due to the fact that the transform has been centered. In this case, the distance from any point (u, v) to the center (origin) of the Fourier transform is given by

$$D(u, v)= [(u – M/2)^2 + (v – N/2)^2]^{½}$$



For an ideal lowpass filter cross section, the point of transition between H (u, v) = 1 and H (u, v) = 0 is called the cutoff frequency.

One way to establish a set of standard cutoff frequency loci is to compute circles that enclose specified amounts of total image power P_T .This quantity is obtained by summing the components of the power spectrum at each point (u, v), for u = 0,1,2,…,M – 1 and v = 0,1,2,…,N -1; that is,

$$P_T = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} P(u,v) \qquad u = 0, 1, ..., M-1 \quad , v = 0, 1, ..., N-1$$

For the centered transform, a circle of radius *r* encompasses β percent of the power, where

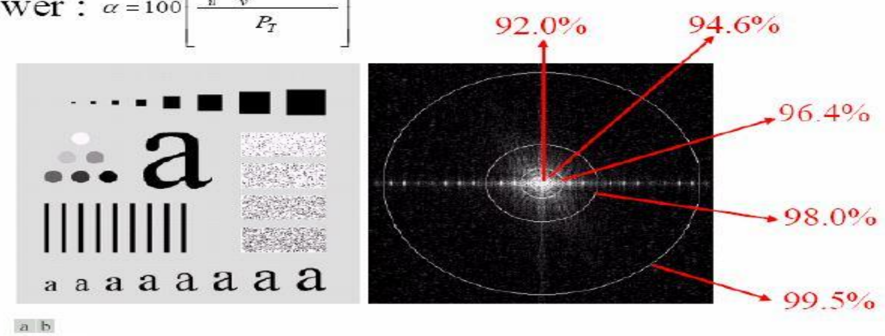
$$\alpha = 100\left[\sum_u \sum_v P(u,v) / P_T\right] \qquad u \le radius(D_o) \quad , v \le radius(D_o)$$

and the summation is taken over the values of (u, v) that lie inside the circle or on its boundary. The blurring and ringing properties of the ILPF can be explained by reference to the convolution theorem. The Fourier transforms of the original image f(x, y) and the blurred image g(x, y) are related in the frequency domain by the equation.

$$G(u, v) = H(u, v)\, F(u, v)$$

Total image power : $P_T = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} P(u,v)$

Enclosed power : $\alpha = 100\left[\frac{\sum_u \sum_v P(u,v)}{P_T}\right]$



(a) An image of size 500 × 500 pixels and (b) its Fourier spectrum. The superimposed circles have radii values of 5, 15, 30, 80, and 230, which enclose 92.0, 94.6, 96.4, 98.0, and 99.5% of the image power, respectively.

where, as before, h(u, v) is the filter function and F and G are the Fourier transforms of the two images just mentioned. The convolution theorem tells us that the corresponding process in the spatial domain is

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

where h(x, y) is the inverse Fourier transform of the filter transfer function H(u, v).

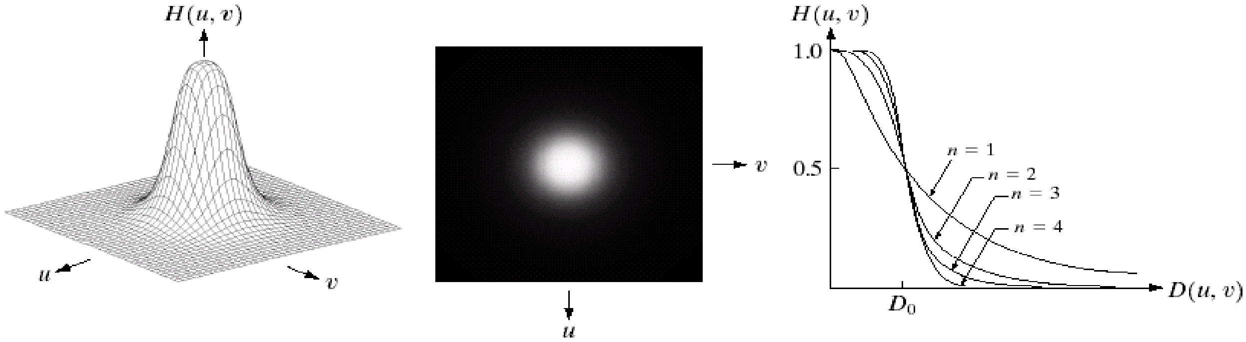
The spatial filter function h(x, y) was obtained in the standard way: (1) H(u, v) was multiplied by $(-1)^{u+v}$ for centering; (2) this was followed by the inverse DFT; and (3) the real part of the inverse DFT was multiplied by $(-1)^{x+y}$.

Butterworth Lowpass Filters

The transfer function of a Butterworth lowpass filter (BLPF) of order n, and with cutoff frequency at a distance D_0 from the origin, is defined as

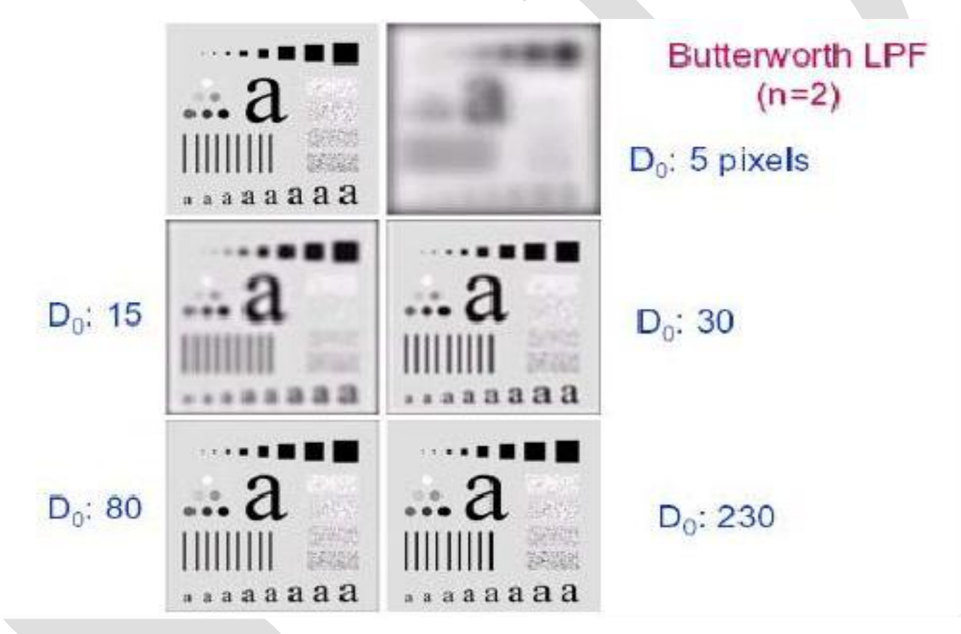
$$H(u,v) = \frac{1}{1+[D(u,v)/D_0]^{2n}}$$

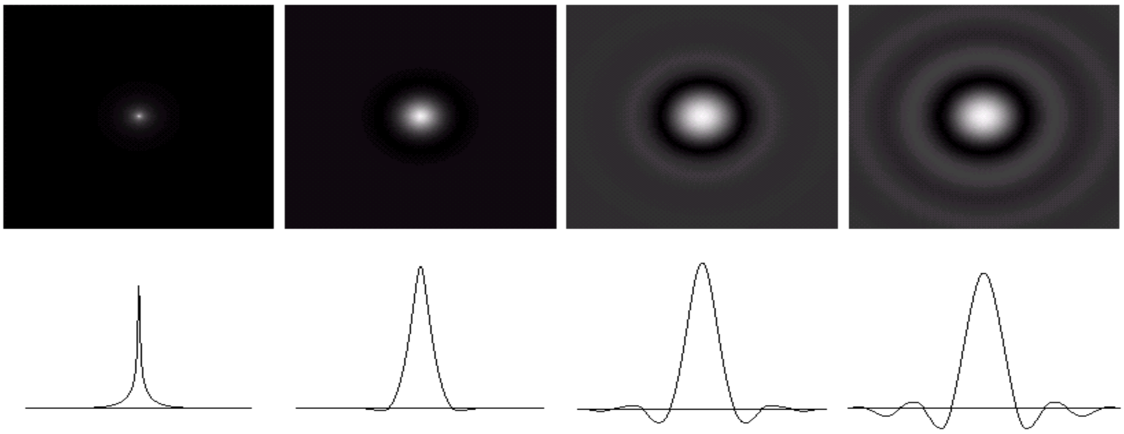
D(u,v) is the distance from the origin
D_0 is the cutoff frequency.
n is the order of the filter



a b c

(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.





a b c d

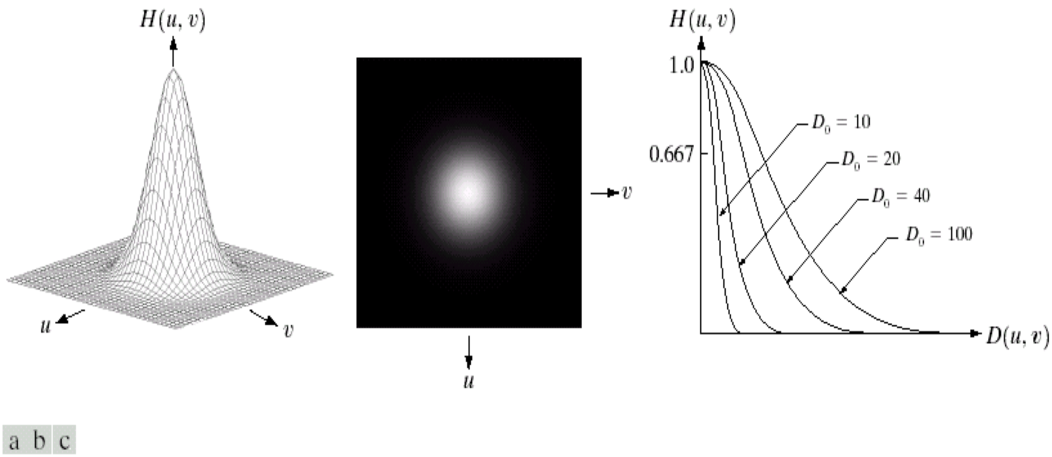
(a)–(d) Spatial representation of BLPFs of order 1, 2, 5, and 20, and corresponding gray-level profiles through the center of the filters (all filters have a cutoff frequency of 5). Note that ringing increases as a function of filter order.

Unlike the **ILPF**, the **BLPF** transfer function does not have a sharp discontinuity that establishes a clear cutoff between passed and filtered frequencies. A Butterworth filter of order 1 has no ringing. Ringing generally is imperceptible in filters of order 2, but can become a significant factor in filters of higher order. The filter of order 2 does so mild ringing and small negative values, but they certainly are less pronounced than in the **ILPF**. A Butterworth filter of order 20 already exhibits the characteristics of the **ILPF**. In general, **BLPF**s of order 2 are a good compromise between effective lowpass filtering and acceptable ringing characteristics.

Gaussian Lowpass Filters

The form of these filters in two dimensions is given by

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



(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.

D(u, v) is the distance form of the origin of the Fourier transform.is a measure of the spread of the Gaussian curve. By letting = D_0.

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

where D_0 is the cutoff frequency. When D(u, v) = D_0, the filter is down to 0.607 of its maximum value. The inverse Fourier transform of the Gaussian lowpass filter also is Gaussian.

Sharpening Frequency Domain Filters

The high-frequency components are: edges and sharp transitions such as noise. Sharpening can be achieved by high pass filtering process, which attenuates low frequency components without disturbing the high-frequency information in the frequency domain.

The filter transfer function, $H_{hp}(u,v)$, of a high pass filter is given by:

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

We will consider only 3 types of sharpening high pass filters :

- Ideal Highpass filters,
- Butterworth Highpass filter
- Gaussian Highpass filters

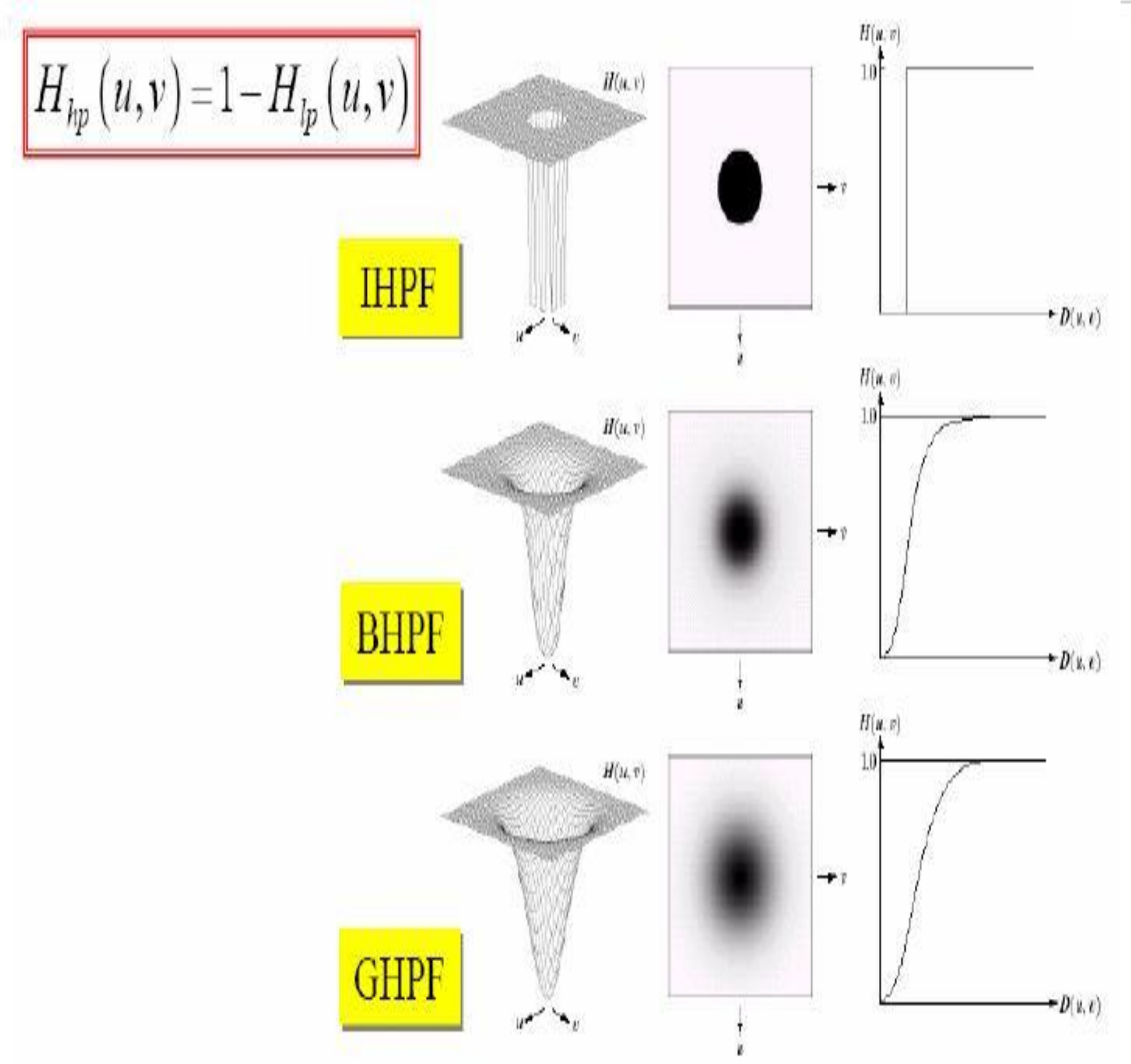


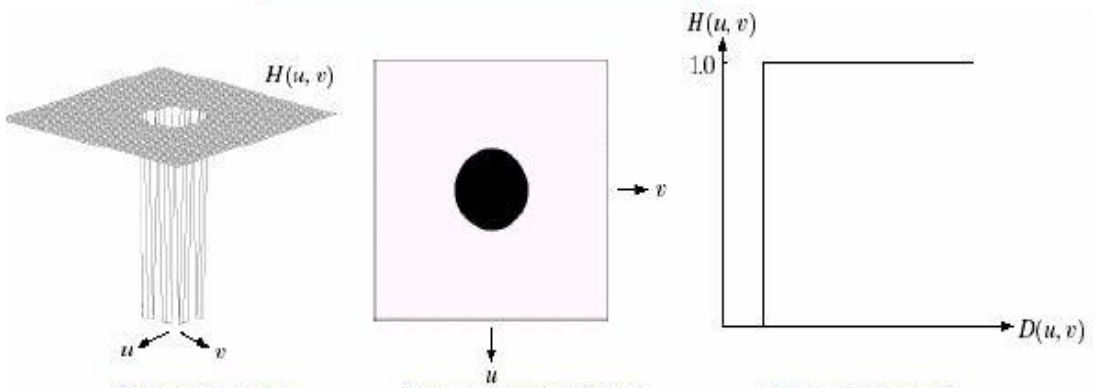
Fig: Top Row- Perspective plot, Image Representation and Cross Section of typical Ideal High Pass Filter, Second Row- Perspective plot, Image Representation and Cross Section of typical Butterworth High Pass Filter, Bottom Row- Perspective plot, Image Representation and Cross Section of typical Gaussian High Pass Filter

Ideal High pass Filter (IHPF):

A 2-D ideal highpass filter (IHPF) is defined as

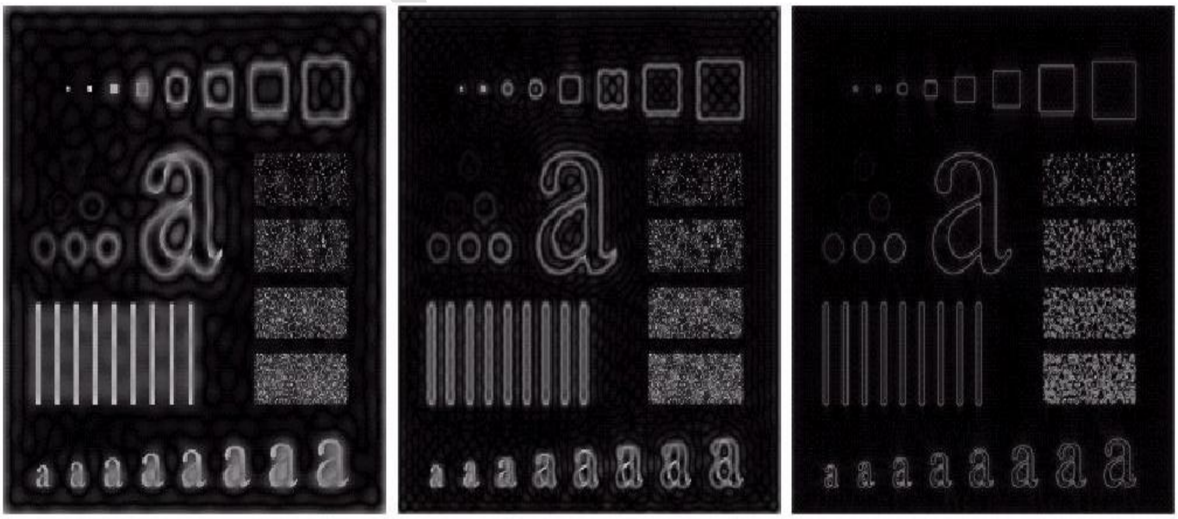
$$\boxed{H(u,v) = \begin{cases} 0 & \text{if } D(u,v) \le D_0 \\ 1 & \text{if } D(u,v) > D_0 \end{cases}}$$

$D(u,v)$ is the distance from the origin
D_0 is the cutoff frequency



Perspective plot of IHPF *Image representation of IHPF* *Cross section of IHPF*

where D_0 is the cutoff distance measured from the origin of the frequency rectangle. This filter is opposite of the ideal low-pass filter in the sense that it sets to zero all frequencies inside a circle of radius D_0 while passing, without attenuation, all frequencies outside the circle. As in the case of the ideal lowpass filter, **IHPF** is physically realizable with electronic components.

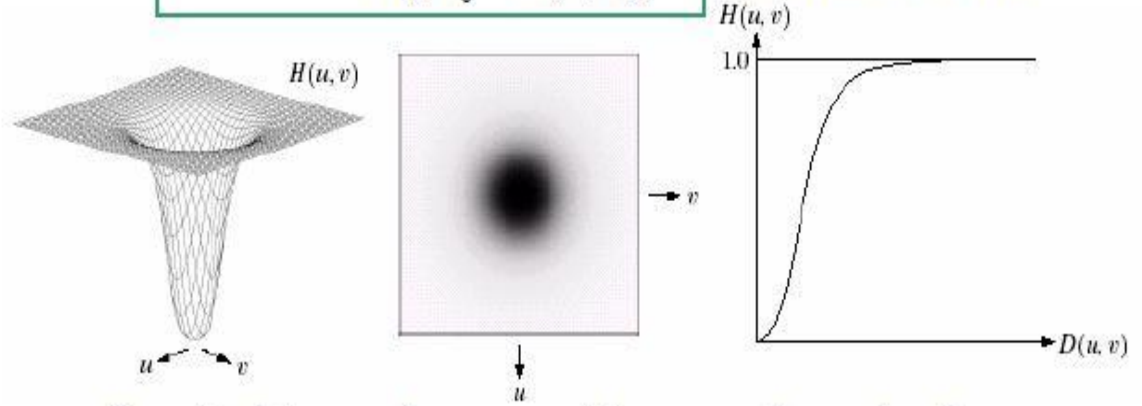


Butterworth Highpass Filters

The transfer function of the Butterworth highpass filter (BHPF) of order n and with cutoff frequency locus at a distance D_0 from the origin is given by

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

D(u,v) is the distance from the origi
D_0 is the cutoff frequency.
n is the order of the filter



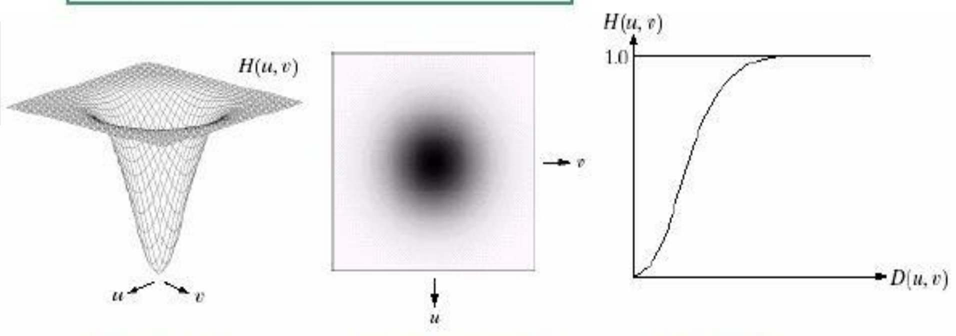
Perspective plot of BHPF | Image representation of BHPF | Cross section of BHPF

Gaussian Highpass Filters

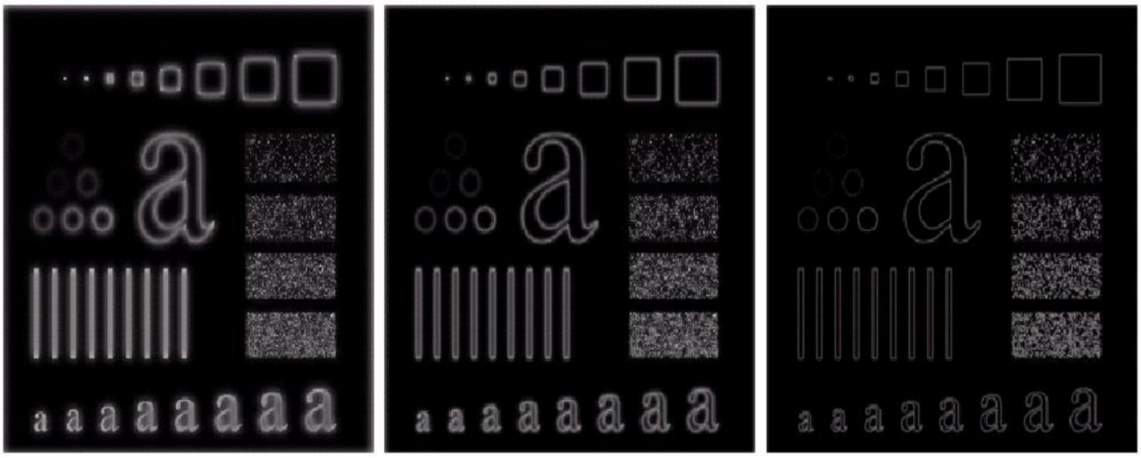
The transfer function of the Gaussian Highpass Filters (GHPF) with cutoff frequency locus at a distance D_0 from the origin is given by

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

D(u,v) is the distance from the origin
D_0 is the cutoff frequency.



Perspective plot of BHPF | Image representation of BHPF | Cross section of BHPF



Spatial Representation of High pass filters

IHPF **BHPF** **GHPF**

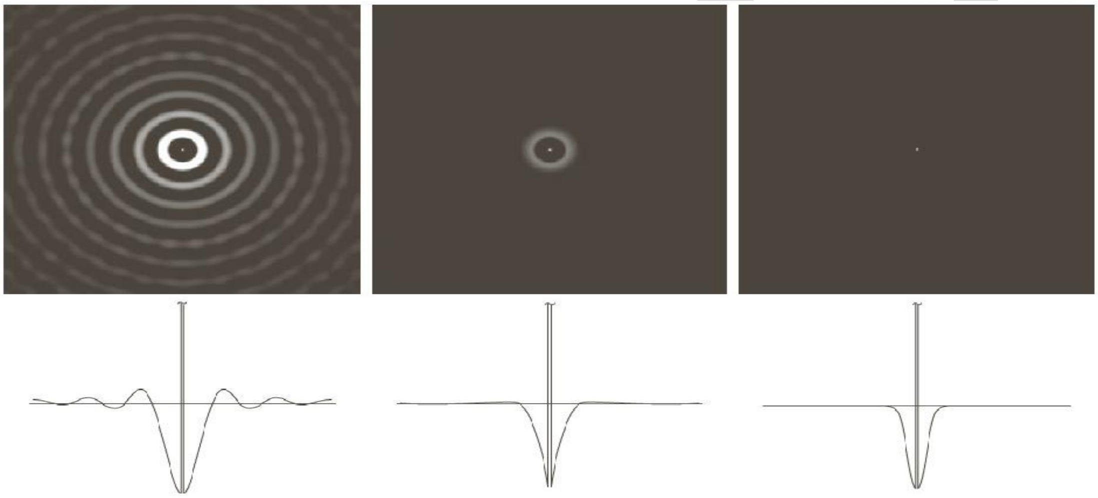


Fig: Special Representation of a)Ideal b)Butterworth c)Gaussian Frequency Domain of Highpass Filters

Gaussian Lowpass Filters

The form of these filters in two dimensions is given by

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

D(u, v) is the distance form of the origin of the Fourier transform.

is a measure of the spread of the Gaussian curve. By letting = D_0.

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

where D_0 is the cutoff frequency. When D(u, v) = D_0, the filter is down to 0.607 of its maximum value.

The inverse Fourier transform of the Gaussian lowpass filter also is Gaussian.

UNIT III

IMAGE RESTORATION AND SEGMENTATION

Noise models – Mean Filters – Order Statistics – Adaptive filters – Band reject Filters – Band pass Filters – Notch Filters – Optimum Notch Filtering – Inverse Filtering – Wiener filtering Segmentation: Detection of Discontinuities–Edge Linking and Boundary detection – Region based segmentation- Morphological processing- erosion and dilation.

PART A 2 MARKS

1. What do you mean by Restoration, Degradation and Distortion?

Restoration techniques involve modeling of degradation and applying the inverse process in order to recover the image

Degradation -gray value altered

Distortion- pixel shifted

2. What is meant by Image Restoration? [AUC NOV/DEC 2013]

Restoration attempts to reconstruct or recover an image that has been degraded by using a clear knowledge of the degrading phenomenon.

3. Explain additivity property in Linear Operator?

H[f1(x,y)+f2(x,y)]=H[f1(x,y)]+H[f2(x,y)]

The additive property says that if H is the linear operator,the response to a sum of two is equal to the sum of the two responses.

4. What are the two methods of algebraic approach? [APR/MAY 2012]

- Unconstraint restoration approach
- Constraint restoration approach

5. What is meant by Noise probability density function?

The spatial noise descriptor is the statistical behavior of gray level values in the noise component of the model.

6. Why the restoration is called as unconstrained restoration? [APR MAY 2017]

In the absence of any knowledge about the noise 'n', a meaningful criterion function is to seek an f^ such that H f^ approximates of in a least square sense by assuming the noise term is as small as possible.

Where H = system operator.

f^ = estimated input image.

g = degraded image.

7. What are the types of noise models? [AUC APR/MAY 2010]

- Guassian noise
- Rayleigh noise
- Erlang noise
- Exponential noise
- uniform noise
- Impulse noise

8. What is meant by least mean square filter or wiener filter? (Dec'12)

The limitation of inverse and pseudo inverse filter is very sensitive noise. The wiener filtering is a method of restoring images in the presence of blur as well as noise.

9. What is pseudo inverse filter? (Dec'13)

It is the stabilized version of the inverse filter. For a linear shift invariant systemwith frequency response H(u, v) the pseudo inverse filter is defined as

$H^{-}(u,v)=1/(H(u,v)$ H#0

0 H=0

10. Compare constrained and unconstrained restoration(May'14)

Constrained Restoration	**Unconstrained Restoration**
In the absence of any knowledge about the noise 'n', based on Lagrange multiplier and linear operator, a meaningful criterion function is to seek an $\hat{f}$ such that $H\hat{f}$ approximates of in a least square sense by assuming the noise term is as small as possible. Where H =system operator. $\hat{f}$ =estimated input image. g=degraded image.	In the absence of any knowledge about the noise 'n', a meaningful criterion function is to seek an $\hat{f}$ such that $H\hat{f}$ approximates of in a least square sense by assuming the noise term is as small as possible. Where H =system operator. $\hat{f}$ = estimated input image. g=degraded image.

11. What is edge? (Dec'13)

An edge is a set of connected pixels that lie on the boundary between two regions edges are more closely modelled as having a ramp like profile. The slope of the ramp is inversely proportional to the degree of blurring in the edge.

12. How edges are linked through hough transform?(Dec'14)

The edges are linked through hough transform by using intersecting of 2 lines equations. The straight line equation is $y=mx+b$. In polar coordinates $\rho=x\cos\theta+y\sin\theta$ where ρ & θ are the coordinates of parameter space. The hough transform of a straight line in the x,y space is a single point in ρ,θ space.

13. State the problems in region splitting and merging based image segmentation. (Dec'14)

- Initial seed points – different set of initial seed point cause different segmented result.
- Time consuming problem
- This method is not suitable for color images and produce fault colors sometime.

Region growth may stop at any time when no more pixel satisfy the criteria.

PART B 16 Marks

1. Write about Noise Probability Density Functions. [APR/MAY 2012]

or

Describe the filters used for noise distribution removal from images. [MAY/JUNE 2016]

Classification of noise is based upon:

- The shape of probability density function (analog case of noise)
- Histogram (discrete case of noise)

Uncorrelated noise is defined as the variations within an image that have no spatial dependences from image to image

The term noise has the following meanings:

1. An undesired disturbance within the frequency band of interest; the summation of unwanted or disturbing energy introduced into a communications system from man-made and natural sources.
2. A disturbance that affects a signal and that may distort the information carried by the signal.
3. Random variations of one or more characteristics of any entity such as voltage, current, or data.
4. A random signal of known statistical properties of amplitude, distribution, and spectral density.

Noise has long been studied. People analyze its property, type, influence and what can be done about it. Most of the research is done in mathematics and close related to Probability Theory

Types of mixing noise with signal

In many applications it is assumed that noise is additive and statistically independent of the signal

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

For thermal noise often, noise is signal-dependent. Examples: Speckle photon noise… Many noise sources can be modeled by a multiplicative model:

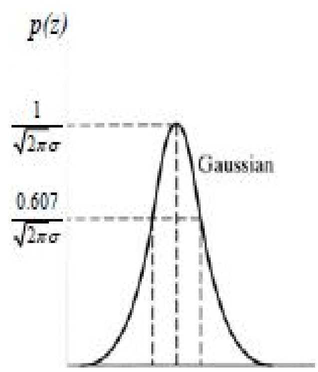
$$g(x, y) = f(x, y)\,\eta(x, y)$$

In CMOS sensors there is a fixed-pattern noise and mixture of additive and multiplicative noise

There are different noise models.

- For good strategy in removing noise and restoring image quality one needs to determine noise distribution.
  (Check the histogram in a reasonable size smooth region with visibly small variation in values)
- Once the noise model is estimated use an appropriate filter.
  Histogram in the same region indicates level of success.
- Denoising (i.e. removing noise) often introduce other side effects.
- Advanced de-noising filters are based on adaptive strategy, i.e. the procedure tries to adapt the application of the filter as it progresses
- Frequency domain filters provide powerful de-noising methods.
- Noise in Colour images may have different characteristics in different colour channels, but removing noise uses the same strategy

1. Gaussian (normal) noise is very attractive from a mathematical point of view since its DFT is another Gaussian process.



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

Here *z* represents intensity, *ž* is the mean (average) value of *z* and σ is its standard

z, deviation. σ^2 is the variance of *z*.

Electronic circuit noise, sensor noise due to low illumination or high temperature.

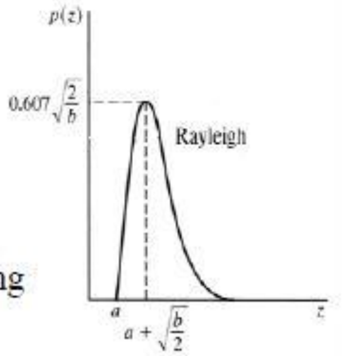
2. Rayleigh noise is specified as

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

The mean and variance are given by

$$\bar{z}=a+\sqrt{\pi b/4}$$

$$\sigma^2=\frac{b(4-\pi)}{4}$$

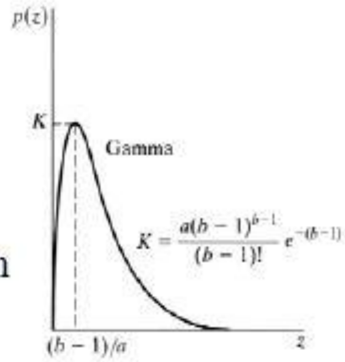


The Rayleigh density is useful for approximating skewed histograms. Used in range imaging.

Radar range and velocity images typically contain noise that can be modeled by the Rayleigh distribution.

3. Erlang noise is specified as

$$p(z)=\begin{cases}\frac{a^b z^{b-1}}{(b-1)!}e^{-az} & z\ge 0\\ 0 & z<0\end{cases}$$



Here $a>0$ and b is a positive integer. The mean and variance are given by

$$\bar{z}=b/a$$

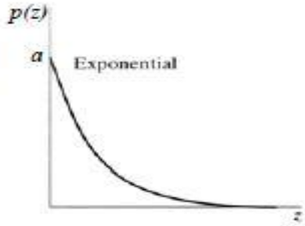
$$\sigma^2=b/a^2$$

When the denominator is the gamma function, the pdf describes the gamma distribution.

4. Exponential noise is specified as

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

Here $a > 0$. The mean and variance are given by

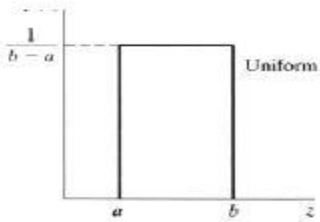


$$\bar{z} = 1/a$$
$$\sigma^2 = 1/a^2$$

Exponential pdf is a special case of Erlang pdf with b =1.
Used in laser imaging.

5. Uniform noise is specified as

$$\text{Histogram Uniform} = \begin{cases} \frac{1}{b-a} & a \leq z \leq b \\ 0 & otherwise \end{cases}$$



The mean and variance are given by

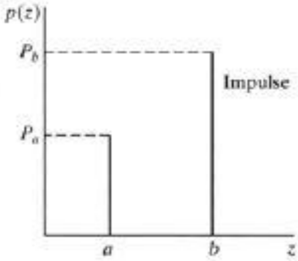
$$\bar{z} = \frac{a+b}{2}$$
$$\sigma^2 = \frac{(b-a)^2}{12}$$

The gray level values of the noise are evenly distributed across a specific range

• Quantization noise has an approximately uniform distribution

6. Impulse (salt-and-pepper) noise (bipolar) is specified as

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



If b>a, intensity b will appear as a light dot on the image and a appears as a dark dot. If either P_a or P_b is zero, the noise is called *unipolar*. Frequently, a and b are ***saturated*** values, resulting in positive impulses being white and negative impulses being black. This noise shows up when quick transitions – such as faulty switching – take place.

2. Write a short note on Inverse filtering and Wiener filter in detail.

[AUC APR/MAY 2010,]

or

Discuss the concept of inverse filters for image restoration.

[MAY/JUNE 2016]

or

Describe inverse filtering for removal of blur caused by any motion and describe how it restore the image. [APRIL/MAY 2015]

1) Inverse Filtering: (un constrained)

- In most images, adjacent pixels are highly correlated, while the gray levels of widely separated pixels are only loosely correlated.
- Therefore, the autocorrelation function of typical images generally decreases away from the origin.
- Power spectrum of an image is the Fourier transform of its autocorrelation function, therefore, we can argue that the power spectrum of an image generally decreases with frequency
- Typical noise sources have either a flat power spectrum or one that decreases with frequency more slowly than typical image power spectra.
- Therefore, the expected situation is for the signal to dominate the spectrum at low frequencies while the noise dominates at high frequencies.

Until now our focus was the calculation of degradation function H(u,v).

Having H(u,v) calculated/ estimated the next step is the restoration of the degraded image. There are different types of filtering techniques for obtaining or for restoring the original image from a degraded image. The simplest kind of Approach to restoration is direct inverse filtering technique.

The simplest way of image restoration is by using Inverse filtering:

Now, the concept of inverse filtering is very simple. Our expression is that G (u, v) that is the Fourier transform of the degraded image is given by H (u, v) into F' (u, v) where H (u, v) is the degradation function in the frequency domain and estimate F' (u, v) is the Fourier transform of the original image, G (u, v) is the Fourier transform of the degraded image. The division is an array operation.

$$\hat{F}(u,v) = \frac{G(u,v)}{H(u,v)}$$

, $\hat{F}(u,v)$ is the Fourier transform of the restored image

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

Unknown random function

Must not be very small. Otherwise the noise dominates

Noise is enhanced when H(u,v) is small.

To avoid the side effect of enhancing noise, we can apply this formulation to freq. component (u,v) within a radius D_0 from the center of H(u,v).

So, this expression says that even if H (u, v) is known exactly, the perfect reconstruction may not be possible because N (u,v) is not known.

Again if H(u,v) is near zero, N(u,v)/H(u,v) will dominate the F'(u,v) estimate.

Now, because this H (u, v) into F (u, v), this is a point by point multiplication. That is for every value u and v, the corresponding F component and the corresponding H component will be multiplied together to give you the final matrix which is again in the frequency domain. This problem could be reduced by limiting the analysis to frequencies near the origin.

The solution is again to carry out the restoration process in a limited neighborhood about the origin where H(u,v) is not very small. This procedure is called pseudo inverse filtering.

In Inverse filtering, we simply take H(u,v) such that the noise does not dominate the result. This is achieved by including only the low frequency components of H(u,v) around the origin. Note that, the origin, H(M/2,N/2), corresponds to the highest amplitude component.

Consider the degradation function of the atmospheric turbulence for the origin of the frequency spectrum,

$$H(u,v) = e^{-k\left[(u-M/2)^2+(v-N/2)^2\right]^{5/6}}$$

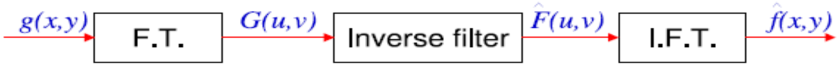
If we consider a Butterworth Lowpass filter of H(u,v) around the origin we will only pass the low frequencies (high amplitudes of H(u,v)).

•As we increase the cutoff frequency of the LPF more smaller amplitudes will be included. Therefore, instead of the degradation function the noise will be dominating.

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

Must not be very small. Otherwise the noise dominates

Restoration with an inverse filter



3. Wiener filter (constrained) Direct Method (Stochastic Regularization)

or

Explain the use of wiener filtering in image restoration. [MAY/JUNE 2016, APR/MAY 2017]

or

How wiener filter is helpful to reduce the mean square error when image is corrupted by motion blur and additive noise? [APRIL/MAY 2015]

Inverse Filter considers degradation function only and does not consider the noise part.

In case of Wiener filtering approach, the Wiener filtering tries to reconstruct the degraded image by minimizing an error function. So, it is something like this

Restoration: *Wiener* filter

Degradation model:

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

Wiener filter: a statistical approach to seek an estimate $\hat{f}$ that minimizes the statistical function (mean square error):

$$E^2 = E\{(f - \hat{f})^2\}$$

*Assumpions:

Image and noise are uncorrelated

Image and/or noise has zero mean

Gray levels in the estimate are linear function of the levels in the degraded image

Restoration with a Wiener filter

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

$\hat{F}(u,v) = W(u,v)\,G(u,v)$

$$g(x,y) \rightarrow \boxed{\text{F.T.}} \xrightarrow{G(u,v)} \boxed{\text{Wiener filter}} \xrightarrow{\hat{F}(u,v)} \boxed{\text{I.F.T.}} \xrightarrow{\hat{f}(x,y)}$$

- In frequency domain:

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

$$= \left[\frac{H^*(u,v)}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)}\right]G(u,v)$$

$$= \left[\frac{1}{H(u,v)} \cdot \frac{|H(u,v)|^2}{|H(u,v)|^2 + S_\eta(u,v)/S_f(u,v)}\right]G(u,v)$$

* $H(u, v)$: degradation function
* $|H(u, v)|^2 = H^*(u, v)H(u, v)$ → $H^*(u, v)$: complex conjugate

The Wiener filter

$\hat{F}(u,v) = W(u,v)\,G(u,v)$

$$W(u,v) = \frac{H^*(u,v)}{|H(u,v)|^2 + K(u,v)}$$

where

$K(u,v) = S_\eta(u,v)/S_f(u,v)$

$S_f(u,v) = |F(u,v)|^2$ power spectral density of $f(x,y)$

$S_\eta(u,v) = |N(u,v)|^2$ power spectral density of $\eta(x,y)$

* $S\eta(u, v) = |N(u, v)|^2$: the power spectrum of the noise G(u,v) transform of the degraded image

R is constant that is added to all terms of modulus $H(u,v)^2$

Now, in this case, you might notice that if the image does not contain any noise; then obviously, S(u,v) which is the power spectrum of the noise will be equal to 0 and in that case, this wiener filter becomes identical with the inverse filter. But if the degraded image also contains additive noise in addition to the blurring; in that case, the wiener filter and the inverse filter is different.

4. Explain the Adaptive Filters. [AUC APR/MAY 2010]

Adaptive filters are filters whose behavior changes based on statistical characteristics of the image inside the filter region defined by the m X n rectangular window S_{xy}.

Adaptive, local noise reduction filter:

The simplest statistical measures of a random variable are its mean and variance. These are reasonable parameters on which to base an adaptive filler because they are quantities closely related to the appearance of an image. The mean gives a measure of average gray level in the region over which the mean is computed, and the variance gives a measure of average contrast in that region.

This filter is to operate on a local region, S_{xy}. The response of the filter at any point (x, y) on which the region is centered is to be based on four quantities: (a) g(x, y), the value of the noisy image at (x, y); (b) a2, the variance of the noise corrupting /(x, y) to form g(x, y); (c) ray, the local mean of the pixels in S_{xy}; and (d) $\sigma^2{}_L$, the local variance of the pixels in S_{xy}.

The behavior of the filter to be as follows:

This is the trivial, zero-noisecase in which g (x, y) is equal to f (x, y).

2. If the local variance is high relative to $\sigma^2{}_\eta$ the filter should return a value close to g (x, y).

3. If the two variances are equal, we want the filter to return the arithmetic mean value of the pixels in S_{xy}. This condition occurs when the local area has the same properties as the overall image, and local noise is to be reduced simply by averaging. Adaptive local noise filter is

given by

$$f(x,y) = g(x,y) - \sigma^2\eta / \sigma^2 L[\, g(x,y) - mL \,]$$

The only quantity that needs to be known or estimated is the variance of the overall noise, a2. The other parameters are computed from the pixels in S_{xy} at each location (x, y) on which the filter window is centered.

Adaptive median filter:

The median filter performs well as long as the spatial density of the impulse noise is not large (as a rule of thumb, P_a and P_b less than 0.2). The adaptive median filtering can handle impulse noise with probabilities even larger than these. An additional benefit of the adaptive median filter is that it seeks to preserve detail while smoothing non impulse noise, something that the "traditional" median filter does not do. The adaptive median filter also works in a rectangular window area S_{xy}. Unlike those filters, however, the adaptive median filter changes (increases) the size of S_{xy} during filter operation, depending on certain conditions. The output of the filter is a single value used to replace the value of the pixel at (x, y), the particular point on which the window S_{xy} is centered at a given time.

Consider the following notation:

z_{min} = minimum gray level value in S_{xy}

z_{max} = maximum gray level value in S_{xy}

z_{mcd} = median of gray levels in S_{xy}

z_{xy} = gray level at coordinates (x, y)

S_{max} = maximum allowed size of S_{xy}.

The adaptive median filtering algorithm works in two levels, denoted level A and level B, as follows:

Level A: $A1 = z_{med} - z_{min}$

A2 =

zmed - zmax If A1 > 0 AND

A2 < 0, Go to level B Else

increase the window size

If window size ≤ S_{max}

repeat level A Else output

z_{xy}

Level B: $B1 = z_{xy} - z_{min}$

$B2 = z_{xy} - z_{max}$

If B1> 0 AND B2 < 0, output zxy

Else output z_{med}

5.Discuss in detail about Edge Linking and Boundary Detection.

Edge Linking and Boundary Detection

Ideally, the methods discussed in the previous section should yield pixels lying only on edges. In practice, this set of pixels seldom characterizes an edge completely because of noise, breaks in the edge from non uniform illumination, and other effects that introduce spurious intensity discontinuities. Thus edge detection algorithms typically are followed by linking procedures to assemble edge pixels into meaningful edges. Several basic approaches are suited to this purpose.

Local Processing

One of the simplest approaches for linking edge points is to analyze the characteristics of pixels in a small neighborhood (say, 3 x 3 or 5 x 5) about every point (x, y) in an image that has been labeled an edge point. All points that are similar according to a set of predefined criteria are linked, forming an edge of pixels that share those criteria.

The two principal properties used for stabling similarity of edge pixels in this kind of analysis are

(1) the strength of the response of the gradient operator used to produce the edge pixel; and (2) the direction of the gradient vector. The first property is given by the value of Δf, Thus an edge pixel with coordinates (x_0, y_0) in a predefined neighborhood of (x, y), is similar in magnitude to the pixel at (x, y) if $|\Delta f(x,y) - \Delta f(xo, yo)| \leq E$ where *E* is a nonnegative threshold.

An edge pixel at (x_0, y_0) in the predefined neighborhood of (x, y) has an angle similar to the pixel at (x, y) if $|\alpha(x,y) - \alpha(xo, yo)| < A$ where A is a nonnegative angle threshold

Global Processing

Given n points in an image, suppose that we want to find subsets of these points that lie on straight lines. One possible solution is to first find all lines determined by every pair of points and then find all subsets of points that are close to particular lines. The problem with this procedure is that it involves finding $n (n – 1)/ n^2$ lines and then performing $(n) (n(n – 1))/2\ n^3$ comparisons of every point to all lines. This approach is computationally prohibitive in all but the most trivial applications

Global Processing via Graph-Theoretic Techniques

Global approach for edge detection and linking based on representing edge segments in the form of a graph and searching the graph for low-cost paths that correspond to significant edges. This representation provides a rugged approach that performs well in the presence of noise.

6. Explain about Region based segmentation [APR/MAY 2017, NOV DEC 2016]

Region based segmentation

The objective of segmentation is to partition an image into regions.

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.

Another problem in region growing is the formulation of a stopping rule. 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.

Region Splitting and Merging

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.

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 divided 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*.

If only splitting were used, the final partition likely would contain adjacent region with identical properties. This drawback may be remedied by allowing merging, as well as splitting. 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 U R_k) = TRUE.

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 U R_k) = TRUE
3. Stop when no further merging or splitting is possible.

7. Discuss about Morphological Processing

Morphological Processing

Morphological image processing is a collection of non-linear operations related to the shape or morphology of features in an image. Binary images may contain numerous imperfections. In particular, the binary regions produced by simple thresholding are distorted by noise and texture. Morphological image processing pursues the goals of removing these imperfections by accounting for the form and structure of the image. These techniques can be extended to greyscale images.

Morphological operations rely only on the relative ordering of pixel values, not on their numerical values, and therefore are especially suited to the processing of binary images. Morphological operations can also be applied to greyscale images such that their light transfer functions are unknown and therefore their absolute pixel values are of no or minor interest.

Morphological techniques probe an image with a small shape or template called a **structuring element**. The structuring element is positioned at all possible locations in the image and it is compared with the corresponding neighbourhood of pixels. Some operations test whether the element "fits" within the neighbourhood, while others test whether it "hits" or intersects the neighbourhood:

A morphological operation on a binary image creates a new binary image in which the pixel has a non-zero value only if the test is successful at that location in the input image.

The **structuring element** is a small binary image, i.e. a small matrix of pixels, each with a value of zero or one:

- The matrix dimensions specify the *size* of the structuring element.
- The pattern of ones and zeros specifies the *shape* of the structuring element.
- An *origin* of the structuring element is usually one of its pixels, although generally the origin can be outside the structuring element.

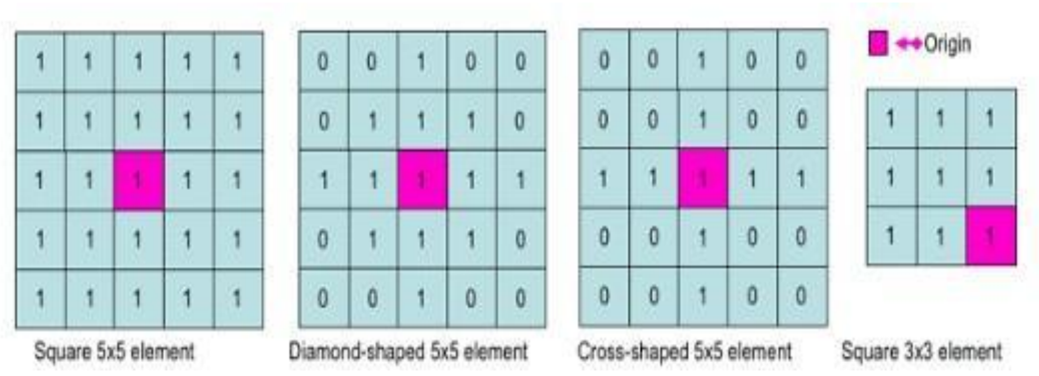


Fig: Example of Simple Structuring Elements

A common practice is to have odd dimensions of the structuring matrix and the origin defined as the centre of the matrix. Stucturing elements play in moprphological image processing the same role as convolution kernels in linear image filtering.

When a structuring element is placed in a binary image, each of its pixels is associated with the corresponding pixel of the neighbourhood under the structuring element. The structuring element is said to **fit** the image if, for each of its pixels set to 1, the corresponding image pixel is also 1. Similarly, a structuring element is said to **hit**, or intersect, an image if, at least for one of its pixels set to 1 the corresponding image pixel is also 1.

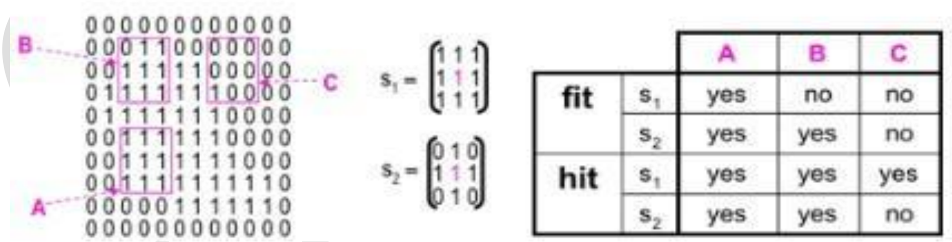


Fig: Fitting and Hitting of Binary images

Zero-valued pixels of the structuring element are ignored, i.e. indicate points where the corresponding image value is irrelevant.

Fundamental operations

Erosion and Dilation

Compound operations

Many morphological operations are represented as combinations of erosion, dilation, and simple set-theoretic operations such as the **complement** of a binary image:

$f^c(x,y) = 1$ if $f(x,y) = 0$, and $f^c(x,y) = 0$ if $f(x,y) = 1$, the **intersection** $h = f \cap g$ of two binary images f and g: $h(x,y) = 1$ if $f(x,y) = 1$ and $g(x,y) = 1$, and $h(x,y) = 0$ otherwise, and the **union** $h = f \cup g$ of two binary images f and g:
$h(x,y) = 1$ if $f(x,y) = 1$ or $g(x,y) = 1$, and $h(x,y) = 0$ otherwise:

Segmentation by Morphological Watersheds

The concept of watersheds is based on visualizing an image in three dimensions: two spatial coordinates versus gray levels. In such a "topographic" interpretation, we consider three types of points: (a) points belonging to a regional minimum; (b) points at which a drop of water, if placed at the location of any of those points, would fall with certainly to a single minimum; and (c) points at which water would be equally likely to fall to more than one such minimum. For a particular regional minimum, the set of points satisfying condition (b) is called the *catchment basin or watershed* of that minimum. The points satisfying condition (c) form crest lines on the topographic surface and are termed *divide lines or watershed lines*.

The principal objective of segmentation algorithms based on these concepts is to find the watershed lines. The basic idea is simple: Suppose that a hole is punched in each regional minimum and that the entire topography is flooded from below by letting water rise through the holes at a uniform rate. When the rising water in distinct catchment basins is about to merge, a dam is built to prevent the merging. The flooding will eventually reach a stage when only the tops of the dams are visible above the water line. These dam boundaries correspond to the divide lines of the watersheds. Therefore, they are the (continuous) boundaries extracted by a watershed segmentation algorithm.

Erosion and Dilation

Two very common morphology operators: Dilation and Erosion. For this purpose, we will use the following OpenCV functions: ◦erode ◦dilate

Morphological Operations:

- In short: A set of operations that process images based on shapes. Morphological operations apply a structuring element to an input image and generate an output image.

- The most basic morphological operations are two: Erosion and Dilation. They have a wide array of uses, i.e. :
- Removing noise
  - Isolation of individual elements and joining disparate elements in an image.
  - Finding of intensity bumps or holes in an image

1.Dilation:

- This operations consists of convoluting an image A with some kernel (B), which can have any shape or size, usually a square or circle.
- The kernel B has a defined anchor point, usually being the center of the kernel.
- As the kernel B is scanned over the image, we compute the maximal pixel value overlapped by B and replace the image pixel in the anchor point position with that maximal value. As you can deduce, this maximizing operation causes bright regions within an image to "grow" (therefore the name dilation). Take as an example the image above. Applying dilation we can get:

The background (bright) dilates around the black regions of the letter.

Erosion:

- This operation is the sister of dilation. What this does is to compute a local minimum over the area of the kernel.
- As the kernel B is scanned over the image, we compute the minimal pixel value overlapped by B and replace the image pixel under the anchor point with that minimal value.
- Analagously to the example for dilation, we can apply the erosion operator to the original image . In the result below that the bright areas of the image (the background, apparently), get thinner, whereas the dark zones (the "writing") gets bigger.

UNIT IV

WAVELETS AND IMAGE COMPRESSION 9

Wavelets – Sub band coding - Multi resolution expansions - Compression: Fundamentals – Image Compression models – Error Free Compression – Variable Length Coding – Bit-Plane Coding – Lossless Predictive Coding – Lossy Compression – Lossy Predictive Coding – Compression Standards.

PART A

1. What is image, Data Compression and its type?

Image compression

Image compression refers to the process of redundancy amount of data required to represent the given quantity of information for digital image. The basis of reduction process is removal of redundant data.

Data Compression

Data compression requires the identification and extraction of source redundancy. In other words, data compression seeks to reduce the number of bits used to store or transmit information.

Types

- ✓ Lossless compression
- ✓ Lossy compression

2. What is the need for Compression? (May’14) (May’13)

In terms of storage, the capacity of a storage device can be effectively increased with methods that compress a body of data on its way to a storage device and decompress it when it is retrieved.

- In terms of communications, the bandwidth of a digital communication link can be effectively increased by compressing data at the sending end and decompressing data at the receiving end.
- At any given time, the ability of the Internet to transfer data is fixed. Thus, if data can effectively be compressed wherever possible, significant improvements of data throughput can be achieved. Many files can be

combined into one compressed document making sending easier.

3. Define is coding redundancy and inter pixel redundancy?

coding redundancy

If the gray level of an image is coded in a way that uses more code words than necessary to represent each gray level, then the resulting image is said to contain coding redundancy.

Inter pixel redundancy

The value of any given pixel can be predicted from the values of its neighbours. The information carried by is small. Therefore the visual contribution of a single pixel to an image is redundant. Otherwise called as spatial redundant geometric redundant or inter pixel redundant. Eg: Run length coding.

4. What is run length coding?(May'14, APR MAY 2017)

Run-length Encoding, or RLE is a technique used to reduce the size of a repeating string of characters. This repeating string is called a *run*; typically RLE encodes a run of symbols into two bytes, a count and a symbol. RLE can compress any type of data regardless of its information content, but the content of data to be compressed affects the compression ratio. Compression is normally measured with the compression ratio:

5. Define compression ratio. (June'12) (Dec'14)

Compression Ratio =original size/ compressed size

6. Define encoder and source encoder?

Encoder

Source encoder is responsible for removing the coding and inter pixel redundancy and psycho visual redundancy. There are two components A)Source Encoder B)Channel Encoder

source encoder

Source encoder performs three operations

- ✓ Mapper - this transforms the input data into non-visual format. It reduces the inter pixel redundancy.
- ✓ Quantizer - It reduces the psycho visual redundancy of the input images. This step is omitted if the system is error free.

✓ Symbol encoder - This reduces the coding redundancy. This is the final stage of encoding process.

7. Define channel encoder and types of decoder?

Channel encoder

The channel encoder reduces the impact of the channel noise by inserting redundant bits into the source encoded data. Eg: Hamming code

types of decoder

Sourced decoder has two components

a)Symbol decoder - This performs inverse operation of symbol encoder.

b)Inverse mapping - This performs inverse operation of map per. Channel decoder-this is omitted if the system is error free.

8. What are the operations performed by error free compression and Variable Length Coding? [APR/MAY 2017]

error free compression

✓ Devising an alternative representation of the image in which its inter pixel redundant are reduced.

✓ Coding the representation to eliminate coding redundancy

Variable Length Coding

Variable Length Coding is the simplest approach to error free compression. It reduces only the coding redundancy. It assigns the shortest possible codeword to the most probable gray levels.

9. Define Huffman coding and mention its limitation (June'12& (Dec'13))

1. Huffman coding is a popular technique for removing coding redundancy.

2. When coding the symbols of an information source the Huffman code yields the smallest possible number of code words, code symbols per source symbol.

Limitation: For equi probable symbols, Huffman coding produces variable code words.

10. Define Block code, instantaneous code and B2 code?

Block code

Each source symbol is mapped into fixed sequence of code symbols or code words. So it is called as block code.

Instantaneous code

A codeword that is not a prefix of any other codeword is called instantaneous or prefix codeword.

B2 code

Each codeword is made up of continuation bit can information bit which are binary numbers. This is called B2 code or B code. This is called B2 code because two information bits are used for continuation bits

11. Define the procedure for Huffman shift coding (Dec'12) (May'13)

List all the source symbols along with its probabilities in descending order. Divide the total number of symbols into block of equal size. Sum the probabilities of all the source symbols outside the reference block. Now apply the procedure for reference block, including the prefix source symbol. The code words for the remaining symbols can be constructed by means of one or more prefix code followed by the reference block as in the case of binary shift code.

11. What is bit plane Decomposition? (Dec'13)

An effective technique for reducing an image's inter pixel redundancies is to process the image's bit plane individually. This technique is based on the concept of decomposing multilevel images into a series of binary images and compressing each binary image via one of several well-known binary compression methods.

12. What are the coding systems in JPEG? (Dec'12)

- ✓ A lossy baseline coding system, which is based on the DCT and is adequate for most compression application.
- ✓ An extended coding system for greater compression, higher precision or progressive reconstruction applications.
- ✓ A lossless independent coding system for reversible compression.

13. What is JPEG and basic steps in JPEG?

JPEG

The acronym is expanded as "Joint Photographic Expert Group". It is an international standard in1992. It perfectly Works with color and gray scale images, Many applications e.g., satellite, medical.

basic steps in JPEG

The Major Steps in JPEG Coding involve:

- ✓ DCT (Discrete Cosine Transformation)
- ✓ Quantization
- ✓ Zigzag Scan
- ✓ DPCM on DC component
- ✓ RLE on AC Components
- ✓ Entropy Coding

14. What is shift code?(Dec'14)

The two variable length codes (Binary shift, Huffman Shift) are referred to as shift codes.

A shift code is generated by

- ✓ Arranging probabilities of the source symbols are monotonically decreasing.
- ✓ Dividing the total number of symbols into symbol blocks of equal size.
- ✓ Coding the individual elements within all blocks identically.
- ✓ Adding special shift up/down symbols to identify each block.

PART B

1. What is image compression? Explain any four variable length coding compression schemes. (Dec'13, APR/MAY 2017, NOV DEC 2016

OR

Explain the schematics of image compression standard JPEG. (May'14)

Image File Formats:

Image file formats are standardized means of organizing and storing digital images. Image files are composed of digital data in one of these formats that can be for use on a computer display or printer. An image file format may store data in uncompressed, compressed, or vector formats.

1. Image file sizes:

In raster images, **Image file size** is positively correlated to the number of pixels in an image and the color depth, or bits per pixel, of the image. Images can be compressed in various ways, however. Compression uses an algorithm that stores an exact representation or an approximation of the original image in a smaller number of bytes that can be expanded back to its uncompressed form with a corresponding decompression algorithm. Considering different compressions, it is common for two images of the same number of pixels and color depth to have a very different compressed file size. Considering exactly the same compression, number of pixels, and color depth for two images, different graphical complexity of the original images may also result in very different file sizes after compression due to the nature of compression algorithms. With some compression formats, images that are less complex may result in smaller compressed file sizes. This characteristic sometimes results in a smaller file size for some lossless formats than lossy formats. For example, graphically simple images (i.e images with large continuous regions like line art or animation sequences) may be losslessly compressed into a GIF or PNG format and result in a smaller file size than a lossy JPEG format. Vector images, unlike raster images, can be any dimension independent of file size. File size increases only with the addition of more vectors.

2. Image file compression

There are two types of **image file compression** algorithms: lossless and lossy.

a. Lossless compression algorithms reduce file size while preserving a perfect copy of the original uncompressed image. Lossless compression generally, but not exclusively, results in larger files than lossy compression. Lossless compression should be used to avoid accumulating stages of re-compression when editing images.

b. Lossy compression algorithms preserve a representation of the original uncompressed image that may appear to be a perfect copy, but it is not a perfect copy. Often lossy compression is able to achieve smaller file sizes than lossless compression. Most lossy compression algorithms allow for variable compression that trades image quality for file size.

Major graphic file formats

The two main families of graphics Raster and Vector.

Raster formats

1.JPEG/JFIF

JPEG (Joint Photographic Experts Group) is a compression method; JPEG-compressed images are usually stored in the **JFIF** (JPEG File Interchange Format) file format. JPEG compression is (in most cases) lossy compression. The JPEG/JFIF filename extension is **JPG** or **JPEG**. Nearly every digital camera can save images in the JPEG/JFIF format, which supports 8-bit gray scale images and 24-bit color images (8 bits each for red, green, and blue). JPEG applies lossy compression to images, which can result in a significant reduction of the file size.

2. JPEG 2000

JPEG 2000 is a compression standard enabling both lossless and lossy storage. The compression methods used are different from the ones in standard JFIF/JPEG; they improve quality and compression ratios, but also require more computational power to process. JPEG 2000 also adds features that are missing in JPEG. It is not nearly as common as JPEG, but it is used currently in professional movie editing and distribution (some digital cinemas, for example, use JPEG 2000 for individual movie frames).

3. EXIF

The **EXIF** (Exchangeable image file format) format is a file standard similar to the JFIF format with TIFF extensions; it is incorporated in the JPEG-writing software used in most cameras. Its purpose is to record and to standardize the exchange of images with image metadata between digital cameras and editing and viewing software. The metadata are recorded for individual images and include such things as camera settings, time and date, shutter speed, exposure, image size, compression, name of camera, color information. When images are viewed or edited by image editing software, all of this image information can be displayed.

4. TIFF

The **TIFF** (Tagged Image File Format) format is a flexible format that normally saves 8 bits or 16 bits per color (red, green, blue) for 24-bit and 48-bit totals, respectively, usually using either the **TIFF** or **TIF** filename extension. TIFFs can be lossy and lossless; some offer relatively good lossless compression for bi-level (black & white) images. Some digital cameras can save in TIFF format, using the LZW compression algorithm for lossless storage. TIFF image format is not widely supported by web browsers. TIFF remains widely accepted as a photograph file standard in the printing business. TIFF can handle device-specific color spaces, such as the CMYK defined by a particular set of printing press inks. OCR (Optical Character Recognition) software packages commonly generate some (often monochromatic) form of TIFF image for scanned text pages.

5. RAW

RAW refers to raw image formats that are available on some digital cameras, rather than to a specific format. These formats usually use a lossless or nearly lossless compression, and produce file sizes smaller than the TIFF formats. Although there is a standard raw image format, (ISO 12234-2, TIFF/EP), the raw formats used by most cameras are not standardized or documented, and differ among camera manufacturers.

Most camera manufacturers have their own software for decoding or developing their raw file format, but there are also many third-party raw file converter applications available that accept raw files from most digital cameras. Some graphic programs and image editors may not accept some or all raw file formats, and some older ones have been effectively orphaned already.

6. GIF

GIF (Graphics Interchange Format) is limited to an 8-bit palette or 256 colors. This makes the GIF format suitable for storing graphics with relatively few colors such as simple diagrams, shapes, logos and cartoon style images. The GIF format supports animation and is still widely used to provide image animation effects. It also uses a lossless compression that is more effective when large areas have a single color, and ineffective for detailed images or dithered images.

7. BMP

The **BMP file format** (Windows bitmap) handles graphics files within the Microsoft Windows OS. Typically, BMP files are uncompressed, hence they are large; the advantage is their simplicity and wide acceptance in Windows programs.

8. PNG

The **PNG** (Portable Network Graphics) file format was created as the free, open-source successor to GIF. The PNG file format supports 8 bit palette images (with optional transparency for all palette colors) and 24 bit true color (16 million colors) or 48 bit true color with and without alpha channel - while GIF supports only 256 colors and a single transparent color. Compared to JPEG, PNG excels when the image has large, uniformly colored areas. Thus lossless PNG format is best suited for pictures still under edition - and the lossy formats, like JPEG, are best for the final distribution of photographic images, because in this case JPG files are usually smaller than PNG files. The Adam7-interlacing allows an early preview, even when only a small percentage of the image data has been transmitted.

9. HDR Raster formats

Most typical raster formats cannot store HDR data (32 bit floating point values per pixel component), which is why some relatively old or complex formats are still predominant here, and worth mentioning separately. Newer alternatives are showing up, though.

10. Other image file formats of the raster type

Other image file formats of raster type include:

- ✓ JPEG XR (New JPEG standard based on Microsoft HD Photo)
- ✓ TGA (TARGA)
- ✓ ILBM (IFF-style format for up to 32 bit in planar representation, plus optional 64 bit extensions
- ✓ DEEP (IFF-style format used by TV Paint

- ✓ IMG (Graphical Environment Manager image file; planar, run-length encoded)
- ✓ PCX (Personal Computer eXchange)
- ✓ ECW (Enhanced Compression Wavelet)

2. Define Compression and Explain the general compression system model?

OR

Explain the parts of JPEG compression block diagram. MAY/JUNE 2016

OR

How an image is compressed using JPEG Image compression with an image matrix? MAY/JUNE 2015

A compression system consists of two distinct structural blocks: an encoder and a decoder.' An input image f (x, y) is fed into the encoder, which creates a set of symbols from the input data. After transmission over the channel, the encoded representation is fed to the decoder, where a reconstructed output image f(x, y) is generated. In general, f'(x, y) may or may not be an exact replica of f(x, y). If it is, the system is error free or information preserving; if not, some level of distortion is present in the reconstructed image.

Both the encoder and decoder shown in Fig. 4.1 consist of two relatively in- dependent functions or sub blocks. The encoder is made up of a source encoder, which removes input redundancies and a channel encoder, which increases the noise immunity of the source encoder's output. As would be expected, the decoder includes a channel decoder followed by a source decoder. If the channel between the encoder and decoder is noise free (not prone to error), the channel encoder and decoder are omitted, and the general encoder and decoder be- come the source encoder and decoder, respectively.

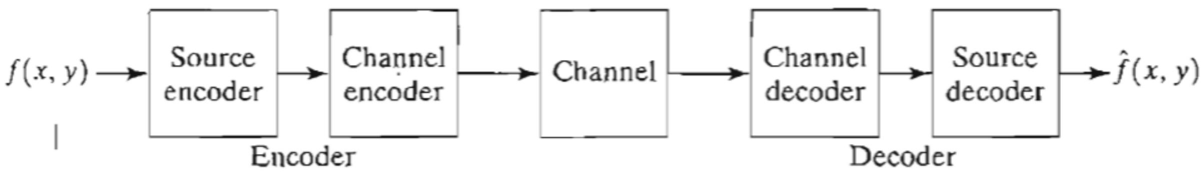
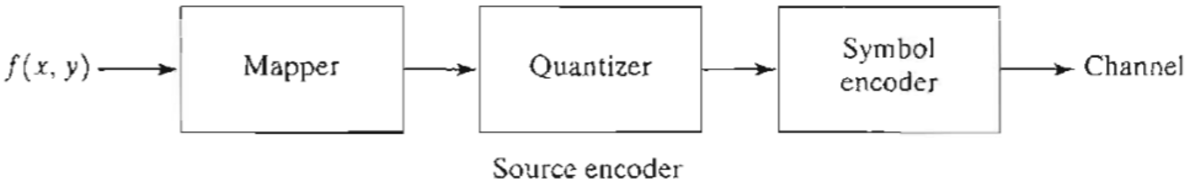


Figure. 4.1 A general compression system model



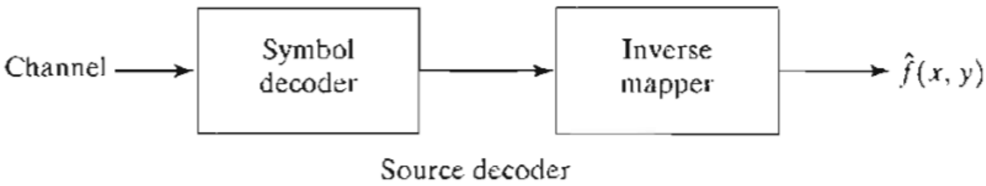


Figure. 4.2 (a) Source encoder and (b) source decoder model.

The Source Encoder and Decoder

The source encoder is responsible for reducing or eliminating any coding, inter pixel, or psycho visual redundancies in the input image. The specific application and associated fidelity requirements dictate the best encoding approach to use in any given situation. Normally, the approach can be modelled by a series of three independent operations. A Fig. 4.2(a) shows, each operation is designed to reduce one of the three redundancies and Figure 4.2(b) depicts the corresponding source decoder.

In the first stage of the source encoding process, the mapper transforms the input data into a (usually non visual) format designed to reduce inter pixel redundancies in the input image. This operation generally is reversible and may or may not reduce directly the amount of data required to represent the image.

The second stage, or quantizer block in fig. 4.2(a), reduces the accuracy of the mapper's output in accordance with some pre-established fidelity criterion. This stage reduces the psycho visual redundancies of the input image.

In the third and final stage of the source encoding process, the symbol coder creates a fixed- or variable-length code to represent the quantizer output and maps the output in accordance with the code. The term symbol coder distinguishes this coding operation from the overall source encoding process. In most cases: a variable-length code is used to represent the mapped and quantized data set. It assigns the shortest code words to the most frequently occurring output values and thus reduces coding redundant.

Figure 4.2(a) shows the source encoding process as three successive operations, but all three operations are not necessarily included in every compression system. Recall, for example, that the quantizer must be omitted when error-free compression is desired.

The source decoder shown in Fig. 4.2(b) contains only two components; a symbol decoder and an inverse mapper. These blocks perform, in reverse order, the inverse operations of the source encoder's symbol encoder and mapper blocks. Because quantization results in irreversible information loss, an inverse quantizer block is not included in the general source decoder model shown in Fig. 4.2(b).

The Channel Encoder and Decoder

The channel encoder and decoder play an important role in the overall encoding-decoding process when the channel of Fig. 4.1 is noisy or prone to error. They are designed to reduce the impact of channel noise by inserting a con- trolled form of redundancy into the source encoded data. As the output of the source encoder contains little redundancy, it would be highly sensitive to trans- mission noise without the addition of this "controlled redundancy."

One of the most useful channel encoding techniques was devised by R. W. Hamming (Hamming [1950]). It is based on appending enough bits to the data being encoded to ensure that some minimum number of bits must change between valid code words. Hamming showed, for example, that if 3 bits of redundancy are added to a 4-bit word, so that the distanced between any two valid code words is 3, all single-bit errors can be detected and corrected. (By ap- pending additional bits of redundancy, multiple-bit errors can be detected and corrected). The 7-bit Hamming (7,4) code word h_1, h_2... . h_5, h_6, h_7 associated with a 4-bit binary number b_3, b_2, b_1, b_0, is

$$\begin{aligned} h_1 &= b_3 \oplus b_2 \oplus b_0 & h_3 &= b_3 \\ h_2 &= b_3 \oplus b_1 \oplus b_0 & h_5 &= b_2 \\ h_4 &= b_2 \oplus b_1 \oplus b_0 & h_6 &= b_1 \\ & & h_7 &= b_0 \end{aligned}$$

where @ denotes the exclusive OR operation. Note that bits h1,h2 and h4, are even- parity bits for the bit fields $b_3b_2b_0$,$b_3b_1b_0$ and $b_2b_1b_0$ respectively. (Recall that a string of binary bits has even parity if the number of bits with a value of 1 is even).

3. Explain in full details about Error free Compression? APR 2017

OR

Describe run length encoding with examples. APRIL/MAY 2015

OR

With an example Huffman coding scheme results with image compression? NOV DEC 2016

The principal of error-free compression strategies are currently in used are discussed here. They normally provide compression ratios of 2 to 10. Moreover, they are equally applicable to both binary and gray-scale images. The error-free compression techniques generally are composed of two relatively independent operations:

- ✓ devising an alternative representation of the image in which its inter pixel redundancies are reduced.
- ✓ coding the representation to eliminate coding redundancies.

Variable-Length Coding

The simplest approach to error-free image compression is to reduce only coding redundancy. Coding redundancy normally is present in any natural binary encoding of the gray levels in an image. To do so requires construction of a variable-length code that assigns the shortest possible code words to the most probable gray levels. Here, we examine several optimal and near optimal techniques for constructing such a code. These techniques are formulated in the language of information theory. In practice, the source symbols may be either the gray levels of an image or the output of a gray-level mapping operation (pixel differences, run lengths, and so on).

Huffman coding

The most popular technique for removing coding redundancy is due to Huffman (Huffman [1952]). When coding the symbols of an information source individually, Huffman coding yields the smallest possible number of code symbols per source symbol. In terms of the noiseless coding theorem (see Section 8.3.3), the resulting code is optimal for a fixed value of n, subject to the constraint that the source symbols be coded one at a time.

The first step in Huffman's approach is to create a series of source reductions by ordering the probabilities of the symbols under consideration and combining the lowest probability symbols into a single symbol that replaces them in the next source reduction. Figure 4.3 illustrates this process for binary coding (K-ary Huffman codes can also be constructed). At the far left, a hypothetical set of source symbols and their probabilities are ordered from top to bottom in terms of decreasing probability values. To form the first source reduction, the bottom two probabilities, 0.06 and 0.04, are combined to form a "compound symbol" with probability 0.1. This compound symbol and its associated probability are placed in the first source reduction column so that the probabilities of the reduced source are also ordered from the most to the least probable. This process is then repeated until a reduced source with two symbols (at the far right) is reached.

The second step in Huffman's procedure is to code each reduced source, starting with the smallest source and working back to the original source. The minimal length binary code for a two-symbol source, of course, is the symbols 0 and 1. As Fig. 4.4 shows, these symbols are assigned to the two symbols on the right (the assignment is arbitrary; reversing the order of the 0 and 1 would work just as well). As the reduced source symbol with probability 0.6 was generated by combining two symbols in the reduced source to its left, the 0 used to code it is now assigned to both of these symbols, and a 0 and 1 are arbitrarily appended to each to distinguish them from each other. This operation is then repeated for each reduced source until the original source is reached. The final code appears at the far left in Fig. 4.4. The average length of this code is and the entropy of the source is 2.14 bits/symbol. and the resulting Huffman code efficiency is 0.973.

Original source		Source reduction			
Symbol	Probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	0.6
a_6	0.3	0.3	0.3	0.3	0.4
a_1	0.1	0.1	0.2	0.3	
a_4	0.1	0.1	0.1		
a_3	0.06	0.1			
a_5	0.04				

Figure. 4.3 Huffman source reductions.

Original source			Source reduction							
Sym.	Prob.	Code	1		2		3		4	
a_2	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
a_6	0.3	00	0.3	00	0.3	00	0.3	00	0.4	1
a_1	0.1	011	0.1	011	0.2	010	0.3	01		
a_4	0.1	0100	0.1	0100	0.1	011				
a_3	0.06	01010	0.1	0101						
a_5	0.04	01011								

Figure. 4.4 Huffman code assignment procedure.

Arithmetic coding

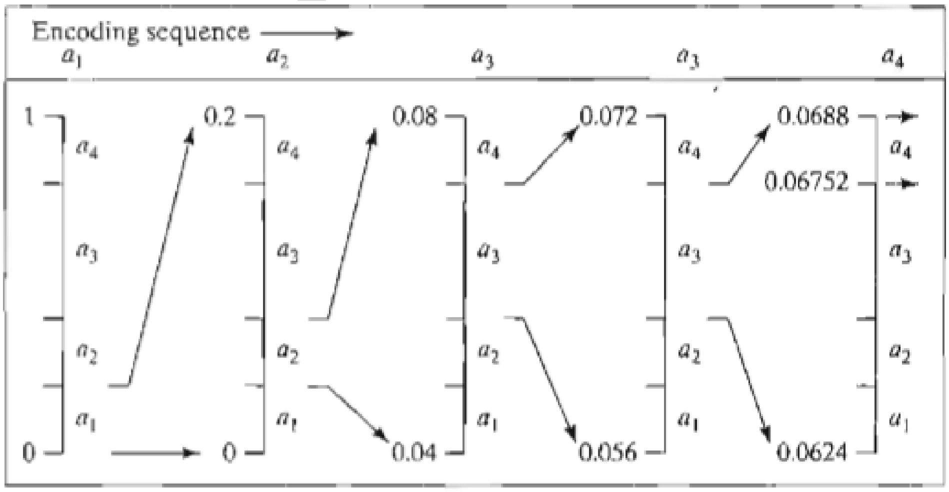


Figure. 4.5 Arithmetic coding Procedure

Figure 4.5 illustrates the basic arithmetic coding process. Here, a five-symbol sequence or message, a_1, a_2, a_3, a_3, a_5 from a four-symbol source is coded. At the start of the coding process, the message is assumed to occupy the entire half- open interval [0, 1]. As Table 4.1 shows, this interval is initially subdivided into four regions based on the probabilities of each source symbol. Symbol a_1 for example, is associated with subinterval [0,0.2). Because it is the first symbol of the message being coded, the message interval is initially narrowed to [0, 0.2).Thus in Fig. 4.5 [0, 0.2) is expanded to the full height of the figure and its end points labelled by the values of the narrowed range. The narrowed range is then subdivided in accordance with the original source symbol probabilities and the process continues with the next message symbol. In this manner, symbol n, narrows the subinterval to [0.04, 0.08), a_3 further narrows it to [0.056,0.072), and so on. The final message symbol, which must be reserved as a special end-of- message indicator, narrows the range to [0.06752,0.0688). Of course, any number within this subinterval-for example, 0.068-can be used to represent the message.

Source Symbol	Probability	Initial Subinterval
a_1	0.2	[0.0, 0.2)
a_2	0.2	[0.2, 0.4)
a_3	0.4	[0.4, 0.8)
a_4	0.2	[0.8, 1.0)

Table 4.1 Arithmetic coding example.

LZW Coding

LZW coding is conceptually very simple (Welch [1984]). At the onset of the coding process, a codebook or "dictionary" containing the source symbols to be coded is constructed. For 8-bit monochrome images, the first 256 words of the dictionary are assigned to the gray values 0, 1, 2,. . . , 255. As the encoder sequentially examines the image's pixels, gray-level sequences that are not in the dictionary are placed in algorithmically determined (e.g., the next unused) locations. If the first two pixels of the image are white, for instance, sequence "255-255" might be assigned to location 256, the address following the locations reserved for gray levels 0 through 255.The next time that two consecutive white pixels are encountered, code word 256, the address of the location containing sequence 255-255, is used to represent them. If a 9-bit, 512-word dictionary is employed in the coding process, the original (8 + 8) bits that were used to represent the row pixels are replaced by a single 9-bit code word. Cleary, the size of the dictionary is an important system parameter. If it is too small, the detection of matching gray-level sequences will be less likely; if it is too large, the size of the code words will adversely affect compression performance.

Bit-Plane Coding

Another effective technique for reducing an image's inter pixel redundancies is to process the image's bit planes individually. The technique, called bit-plane coding, is based on the concept of decomposing a multilevel (monochrome or color) image into a series of binary images and compressing each binary image via one of several well-known binary compression methods. In this section, we describe the most popular decomposition approaches and review several of the more commonly used compression methods.

Bit-plane decomposition

The gray levels of an m-bit gray-scale image can be represented in the form of the base 2 polynomial

$$a_{m-1}2^{m-1} + a_{m-2}2^{m-2} + \cdots + a_1 2^1 + a_0 2^0.$$

Based on this property, a simple method of decomposing the image into a collection of binary images is to separate the m coefficients of the polynomial into m 1-bit planes.

4. Explain in full details about Lossless Predictive Coding?

The approach, commonly referred to as lossless predictive coding, is based on eliminating the inter pixel 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 4.6 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 f'_n and used to form the difference or prediction error

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

which is coded using a variable-length code (by the symbol encoder) to generate the next element of the compressed data stream. The decoder of Fig. 4.6(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 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. In 1-D linear predictive coding, for example we can be written

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

where each subscripted variable is now expressed explicitly as a function of spatial coordinates x and y. The 1-D linear prediction j(x, y) is a function of the previous pixels on the current line alone. In 2-D predictive coding, the prediction is a function of the previous pixels in a left-to- right, top-to-bottom scan of an image. In the 3-D case, it is based on these pixels and the previous pixels of preceding frames.

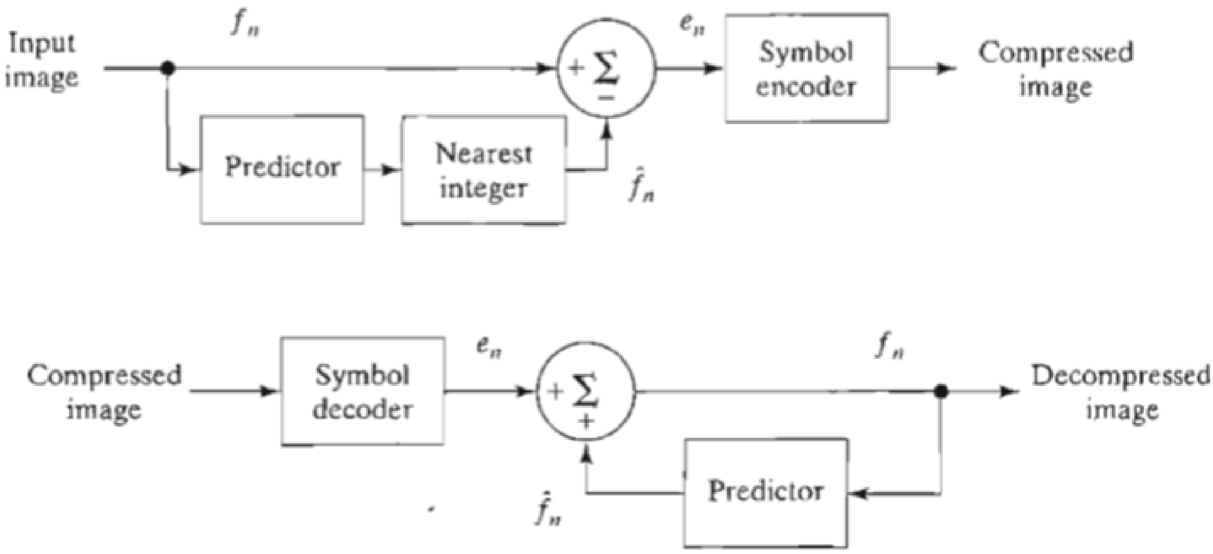


Figure. 4.6 A lossless predictive coding model: (a) encoder; (b) decoder.

5. Explain in full details about Lossy Compression and Lossy Predictive Coding?

Lossy Compression

The lossy encoding is based on the concept of compromising the accuracy of the reconstructed image in exchange for increased compression. If the resulting distortion (which may or may not be visually apparent) can be tolerated, the increase in compression can be significant. In fact, many lossy encoding techniques are capable of reproducing recognizable monochrome images from data that have been compressed by more than 100: 1 and images that are virtually indistinguishable from the originals at 10: 1 to 50: 1. Error-free encoding of monochrome images, however, seldom results in more than a 3: 1 reduction in data.

Lossy Predictive Coding

As Fig. 4.7 shows, the quantizer which absorbs the nearest integer function of the error-free encoder, is inserted between the symbol encoder and the point at which the prediction error is formed. It maps the prediction error into a limited range of outputs, denoted e'_n, which establish the amount of compression and distortion associated with lossy predictive coding.

In order to accommodate the insertion of the quantization step, the error-free encoder of Fig. 4.6(a) must be altered so that the predictions generated by the encoder and decoder are equivalent.

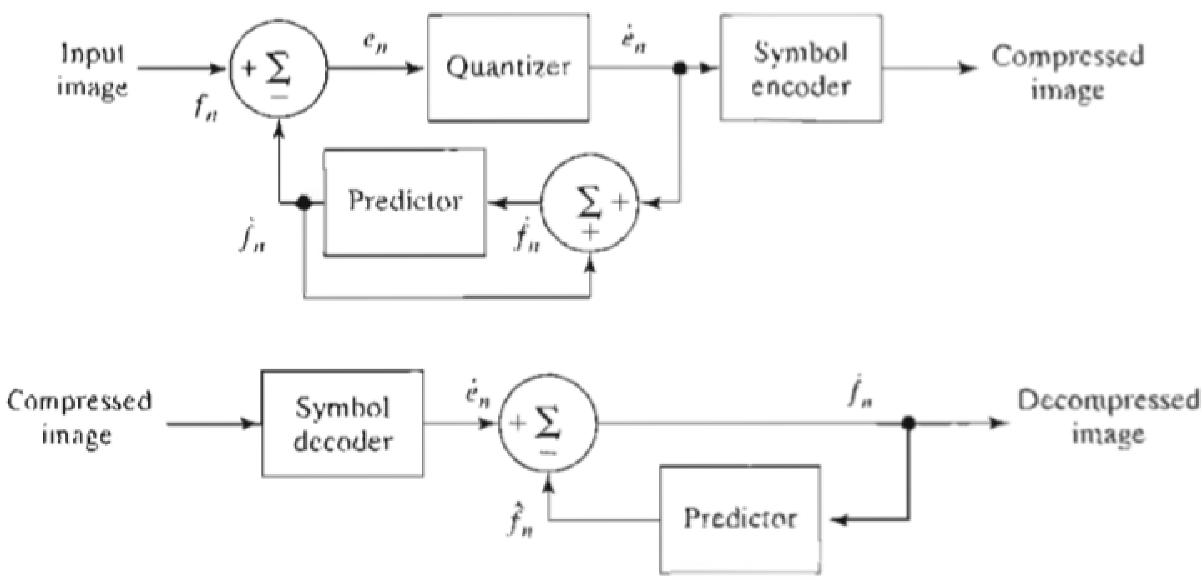


Figure. 4.7 A lossy predictive coding model: (a) encoder and (b) decoder.

As Fig. 4.7(a) shows, this is accomplished by placing the lossy encoder's predictor within a feedback loop, where its input, denoted f'_n, is generated as a function of past predictions and the corresponding quantized errors. That is,

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

where f'_n, is closed loop configuration at the decoder's output.

Delta modulation (DM)

Delta modulation (DM) is a simple but well-known form of lossy predictive coding in which the predictor and quantizer are defined as

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

and

$$\dot{e}_n = \begin{cases} +\zeta & \text{for } e_n > 0 \\ -\zeta & \text{otherwise} \end{cases}$$

where α is a prediction coefficient (normally less than 1) and 5 is a positive constant. The output of the quantizer, e_n, can be represented by a single bit (Fig. 4.8 a), so the symbol encoder of Fig. 4.7(a) can utilize a I-bit fixed-length code. The resulting DM code rate is 1 bit/pixel.

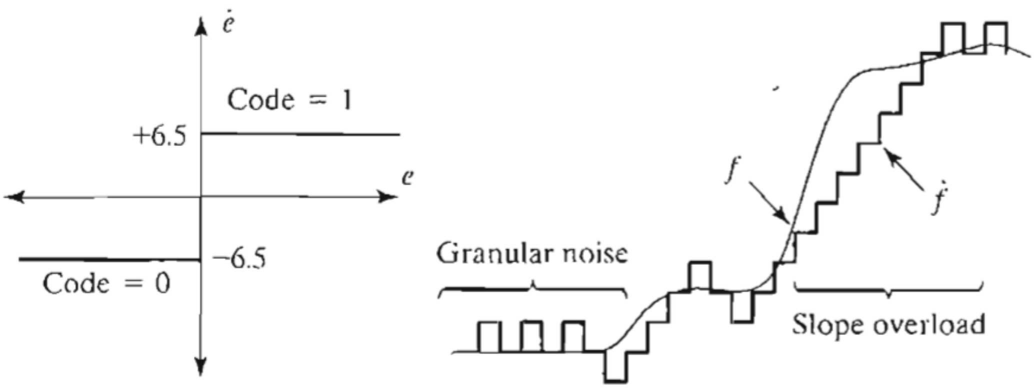


Figure. 4.8 Delta modulation (DM)

Transform Coding

In transform coding, a reversible, linear transform (such as the Fourier transform) is used to map the image into a set of transform coefficients, which are then quantized and coded. For most natural images, a significant number of the coefficients have small magnitudes and can be coarsely quantized (or discarded entirely) with little image distortion.

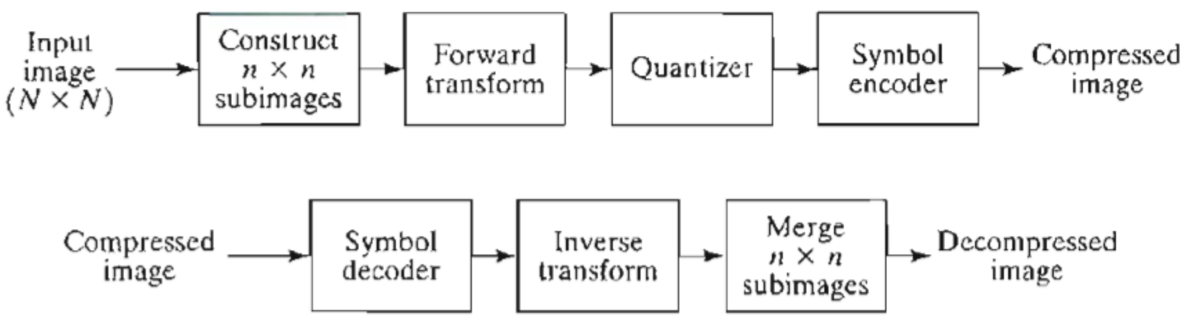


Figure. 4.9 A transform coding system: (a) encoder; (b) decoder.

Figure 4.9 shows a typical transform coding system. The decoder implements the inverse sequence of steps (with the exception of the quantization function) of the encoder, which performs four relatively straightforward operations: sub image decomposition, transformation, quantization, and coding. An N x N input image first is subdivided into sub images of size n x n, which are then transformed to generate $(N/n)^2$ sub image transform arrays, each of size n X n. The goal of the transformation process is to decor relate the pixels of each sub image, or to pack as much information as possible into the smallest number of transform coefficients. The quantization stage then selectively eliminates or more coarsely quantizes the coefficients that carry the least information. These coefficients have the smallest impact on reconstructed sub image quality. The encoding process terminates by coding (normally using a variable- length code) the quantized coefficients. 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.

Wavelet Coding

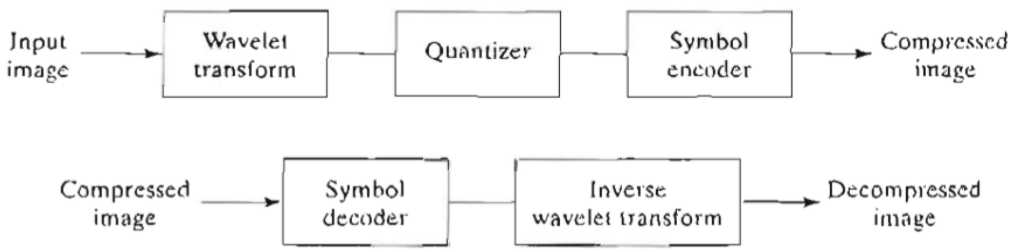


Figure. 4.10 wavelet coding system (a) Encoder, (b) Decoder

Figure 4.10 shows a typical wavelet coding system. To encode a 2^J X 2^J image, an analyzing wavelet, ψ and minimum decomposition level, J - P, are selected and used to compute the image's discrete wavelet transform. Since many of the computed coefficients carry little visual information, they can be quantized and coded to minimize inter coefficient and coding redundancy. Moreover, the quantization can be adapted to exploit any positional correlation across the P decomposition levels.

The principal difference between the wavelet-based system of Fig. 4.10 and the transform coding system of Fig. 4.9 is the omission of the transform coder's sub image processing stages.

6. Explain in full details about Compression Standards? NOV DEC 2016

Many of the lossy and error-free compression methods described so far play important roles in popular image compression standards. In this section we examine a few of these standards and use them to demonstrate the methods presented earlier. Most of the standards discussed are sanctioned by the International Standardization Organization (ISO) and the Consultative Committee of the International Telephone and Tclcgraph (CCITT).

Binary Image Compression Standards

Two of the most widely used image compression standards are the CCITT Group 3 and 4 standards for binary image compression. Although they are currently utilized in a wide variety of computer applications, they were originally designed as facsimile (FAX) coding methods for transmitting documents over telephone networks. The Group 3 standard applies a non adaptive, 1-D run- length coding technique in which the last K - 1 lines of each group of K lines (For K = 2 or 4) are optionally code in a 2-D manner. The Group 4 standard is a simplified or streamlined version of the Group 3 standard in which only 2-D coding is allowed. Both standards use the same non adaptive 2-D coding approach.

During the development of the CCTTT standards, eight representative "test" documents were selected and used as a baseline for evaluating various binary compression alternatives. The existing Group 3 and 4 standards coin press these documents, which include both typed and handwritten text (in several languages) as well as a few line drawings, by about 15 : 1. Because the Group 3 and 4 standards are based on non adaptive techniques, however, they sometimes result in data expansion (e.g., with half-tone images). To overcome this and related problems, the Joint Bilevel Imaging Group (JBIG)-a joint committee of the CCJT and ISO- has adopted and/or proposed several other binary compression standards. These include JBIG1, an adaptive arithmetic compression technique that provides both the best average and worst-case binary compression variable currently and JBIG2 (now a final committee draft), which achieves compressions that are typically 2 to 4 times greater than JBIG1.These standards can be used to compress both binary and gray-scale images of up to 6 gray-coded bits/pixel (on a bit plane basis.)

Continuous Tone Still Image Compression Standards

The CCITT and IS0 have defined several continuous tone (as opposed to binary) image compression standards. These standards, which are in various phases of the adoption process, address both monochrome and color image compression. To develop the standards, CCTTT and IS0 committees solicited algorithm recommendations from a large number of companies, universities, and research 1aboratories.The best of those submitted were selected on the basis of image quality and compression performance. The resulting standards, which include the original DCT-based PEG standard, the recently proposed wavelet-based JPEG 2000 standard, and the JPEG-LS standard, a lossless to near lossless adaptive prediction scheme that includes a mechanism for flat region detection and run-length coding (ISO/IEC [1999), represent the state of the art in continuous tone image compression.

JPEG

One of the most popular and comprehensive continuous tone, still frame compression standards is the JPEG standard. It defines three different coding systems: (1) a lossy baseline coding system, which is based on the DCT and is adequate for most compression applications; (2) an mended coding system for greater compression, higher precision, or progressive reconstruction applications; and (3) a lossless independent coding system for reversible compression. To be JPEG compatible, a product or system must include support for the baseline system. No particular file format, spatial resolution, or color space model is specified.

In the baseline system, often called the sequential baseline system, the input and output data precision is limited to 8 bits, whereas the quantized DCT values are restricted to 11 bits. The compression itself is performed in three sequential steps: DCT computation, quantization, and variable-length code assignment. The image is first subdivided into pixel blocks of size 8 X 8, which are processed left to right, top to bottom. As each 8 x 8 block or sub image is encountered, its 64 pixels are level shifted by subtracting the quantity 2^{n-1}, where 2^n is the maximum number of gray levels.

JPEG 2000

Although not yet formally adopted, JPEG 2000 extends the initial JPEG standard to provide increased flexibility in both the compression of continuous tone still images and access to the compressed data. For example, portions of a JPEG 2000 compressed image can be extracted for retransmission, storage, display, and/or editing. Coefficient quantization is adapted to individual scales and sub bands and the quantized coefficients are arithmetically coded on a bit-plane basis. Using the notation of the standard, an image is encoded as follows (ISO/IEC [2000]).

Video Compression Standards

Video compression standards extend the transform-based, still image compression techniques of the previous section to include methods for reducing temporal or frame-to-frame redundancies. Although there are a variety of video coding standards in use today, most rely on similar video compression techniques. Depending on the intended application, the standards can be grouped into two broad categories: (1) video teleconferencing standards and (2) multi- media standards.

A number of video teleconferencing standards, including H.261 (also referred to as PX64), H.262, H.263, & H.320, have been defined by the International Telecommunications Union (ITU), the successor to the CCITT. H.261 is intended for operation at affordable telecom bit rates and to support full motion video transmission over T1 lines with delays of less than 150 ms. Delays exceeding 150 ms do not provide viewers the "feeling" of direct visual feedback. H.263, on the other hand, is designed for very low bit rate video, in the range of 10 to 30 kbit/s, and H.320, a superset of H.261, is constructed for Integrated Services Digital Network' (TSDN) bandwidths. Each standard uses a motion- compensated, DCT-based coding scheme. Since motion estimation is difficult to perform in the transform domain, blocks of pixels, called macro blocks, are compared to neigh boring blocks of the previous frame and used to compute a motion compensated prediction error. The prediction error is then discrete cosine transformed in 8 X 8 pixel blocks, quantized, and coded for transmission or storage.

Multimedia video compression standards for video on demand, digital HDTV broadcasting, and image/video database services use similar motion estimation and coding techniques. The principal standards-MPEG-1, MPEG-2, and MPEG-4 were developed under the auspices of the Motion Picture Experts Group of the CCTT and ISO. MPEG-1 is an "entertainment quality" coding standard for the storage and retrieval of video on digital media like compact disk read-only memories (CD-ROMs).

Figure 4.11 shows a typical MPEG encoder. It exploits redundancies within and between adjacent video frames, motion uniformity between frames, and the psycho visual properties of the human visual system. The input of the encoder is an 8 x 8 array of pixels, called an image block. The standards define a macro block as a 2 x 2 array of image blocks (i.e., a 16 X I6 array of image elements) and a slice as a row of non overlapping macro blocks.

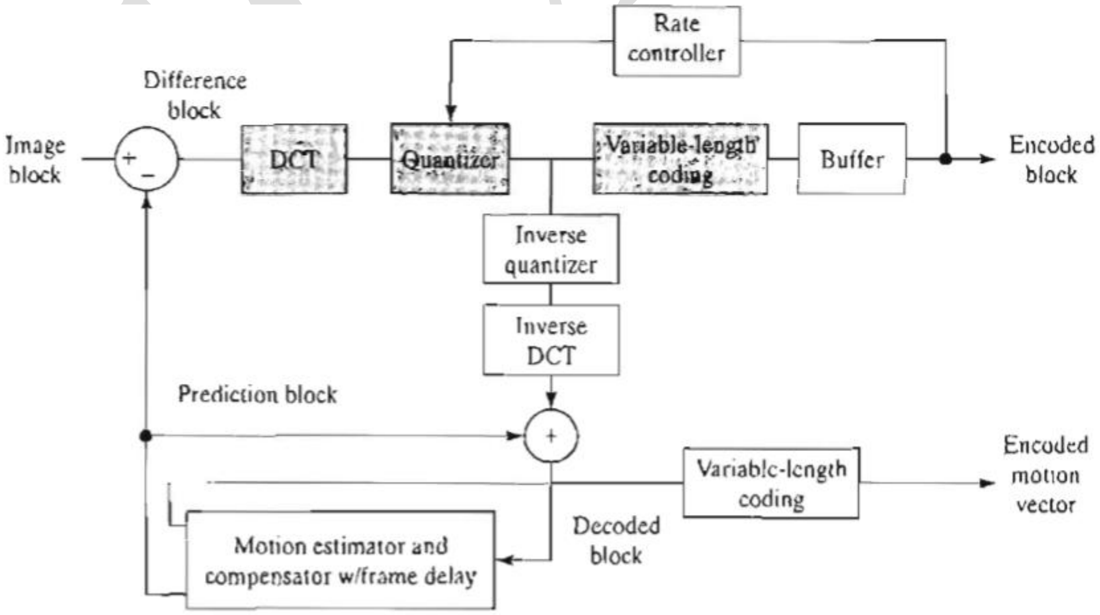


Figure. 4.11A basic DPCM/ DCT encoder for motion compensated video compression

For color video, a macro block, is composed of four luminance blocks, denoted Y, through Y4, and two chrominance blocks, C, and C,. Recall that color difference signal C, is blue minus luminance and C, is red minus luminance. Because the eye has far less spatial acuity for color than for luminance, these two components are often sampled at half the horizontal and vertical resolution of the luminance signa1,resulting in a 4: 1 : 1 sampling ratio between Y' : C_b: C_r.

UNIT V

IMAGE REPRESENTATION AND RECOGNITION

Boundary representation – Chain Code – Polygonal approximation, signature, boundary segments – Boundary description – Shape number – Fourier Descriptor, moments- Regional Descriptors – Topological feature, Texture - Patterns and Pattern classes - Recognition based on matching.

1. Define chain codes?

Chain codes are used to represent a boundary by a connected sequence of straight line segment of specified length and direction. Typically this representation is based on 4 or 8 connectivity of the segments. The direction of each segment is coded by using a numbering scheme.

2. What are the demerits of chain code?

The demerits of chain code are:

- The resulting chain code tends to be quite long
- Any small disturbance along the boundary due to noise causes changes in the code that may not be related to the shape of the boundary.

3. What is thinning or skeletonising algorithm? NOV DEC 2016

An important approach to represent the structural shape of a plane region is to reduce it to a graph. This reduction may be accomplished by obtaining the skeletonising algorithm. It play a central role in a broad range of problems in image processing, ranging from automated inspection of printed circuit boards to counting of asbestos fibres in air filter.

4. What is polygonal approximation method?

Polygonal approximation is a image representation approach in which a digital boundary can be approximated with arbitrary accuracy by a polygon. For a closed curve the approximation is exact when the number of segments in polygon is equal to the number of points in the boundary so that each pair of adjacent points defines a segment in the polygon.

5. Define Signature?

A signature is a 1-D representation of a boundary (which is a 2-D thing): it should be easier to describe. E.g.: distance from the centroid vs angle.

6. Describe Fourier descriptors?

This is a way of using the Fourier transform to analyze the shape of a boundary. The x-y coordinates of the boundary are treated as the real and imaginary parts of a complex number. Then the list of coordinates is Fourier transformed using the DFT . The Fourier coefficients are called the Fourier descriptors.

$$a(u) = \frac{1}{K}\sum_{k=0}^{K-1} s(k) e^{-j\frac{2\pi uk}{K}} \quad ,u = 0,1,2,\ldots,K-1$$

The complex coefficients a(u) are called Fourier descriptor of a boundary.

The inverse Fourier descriptor is given by:

$$s(k) \cong \hat{s}(k) = \sum_{u=0}^{P-1} a(u) e^{+j\frac{2\pi uk}{K}}$$

7. Define Texture and list the approaches to describe texture of a region. NOV DEC 2016

Texture is one of the regional descriptors. It provides the measure of properties such as smoothness, coarseness and regularity.

The approaches to describe the texture of a region of are:

i) Statistical approach.

ii) Structural approach

iii) Spectural approach

8. What are the features of Fourier spectrum?

- Peaks give principal directions of the patterns
- Location of the peaks gives the fundamental period(s)
- Periodic components can be removed via filtering; the remaining non-periodic image can be analyzed using statistical techniques

9. Define Pattern recognition?

Pattern is defined as an arrangement of descriptors. Pattern class is a family of patterns that share some common properties. Pattern recognition is used to assign patterns to their respective classes.

10. Specify the various Polygonal approximation methods.

The various Polygonal approximation methods are:

I. Minimum perimeter polygons.
II. Merging techniques.
III. Splitting techniques.

11. **Define shape numbers.**

It is defined as the first difference of smallest magnitude. The order “n” of a shape number is the number of digits in its representation.

12. Name a few measures used as simple descriptors in regional descriptors.

i) Area
ii) Perimeter
iii) Mean and median gray levels.
iv) Minimum and maximum of gray levels
v) Number of pixels with values above and below mean.

PART B

1. Define Boundary representation with an algorithm and also briefly explain about the Chain codes. NOV DEC 2016

The result of segmentation is a set of regions. Regions have then to be represented and described.

Two main ways of *representing* a region:

- external characteristics (its boundary): focus on shape
- internal characteristics (its internal pixels): focus on color, textures…

The next step: *description*

E.g.: a region may be *represented* by its boundary, and its boundary *described* by some features such as length, regularity…Features should be insensitive to translation, rotation, and scaling. Both *boundary* and *regional* descriptors are often used together.

In order to represent a boundary, it is useful to compact the raw data (list of boundary pixels) Image regions (including segments) can be represented by either the border or the pixels of the region. These can be viewed as external or internal characteristics, respectively.

In Boundary following algorithm whose output is an ordered sequence of points. Assume 1) In image, object and background points are labeled 1 & 0 respectively and 2) that images are padded with border of 0s to eliminate the possibility of an object merging with the image border.

Moore Boundary Tracking Algorithm:

1. Let the starting point be the leftmost point (bo) in the image. Denote c0 the west neighbour of bo. Start at c0 and proceed in clockwise direction.Let b1 is the first neighbour and c1 be the back ground point in the sequence. Store the locations of b0 and b1.

2. Let b = b1 and c = c1

3. Let the 8-neighbours of b, start at c and proceed in clockwise direction. (n1,n2...nk)

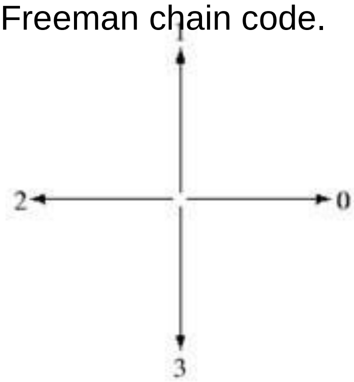
4. Let b = nk and c = nk-1

5. Repeat steps 3 & 4 until b=bo and next boundary point is b1.

Chain codes.

These are used to represent a boundary of a connected region. Also denoted by, list of segments with defined length and direction. They are based on 4-directional chain codes and 8-directional chain codes

A boundary code formed as a sequence of such directional numbers is referred as Freeman chain code.



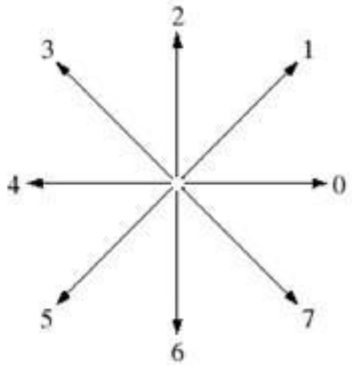


Fig. Direction numbers for a) 4-directional chain code b) 8-directional chain code

It may be useful to downsample the data before computing the chain code, to reduce the code dimension and to remove small detail along the boundary.

The chain code of a boundary depends on the starting point. The code can be normalized with respect to a starting point.

• To remove the dependence from the starting point:

The code is a circular sequence, the new starting point is the one who gives a sequence of numbers giving the smallest integer. Assuming the first difference code represent a closed path, rotation normalization can be achieved by circularly shifting the number of the code so that the list of numbers forms the smallest possible integer.

• To normalize wrt rotation:

The first difference of the chain code:

This difference is obtained by counting the number of direction changes (in a counter clockwise direction).

For example, the first difference of the 4-direction chain code 10103322 is 3133030. Size normalization can be achieved by adjusting the size of the resampling grid.

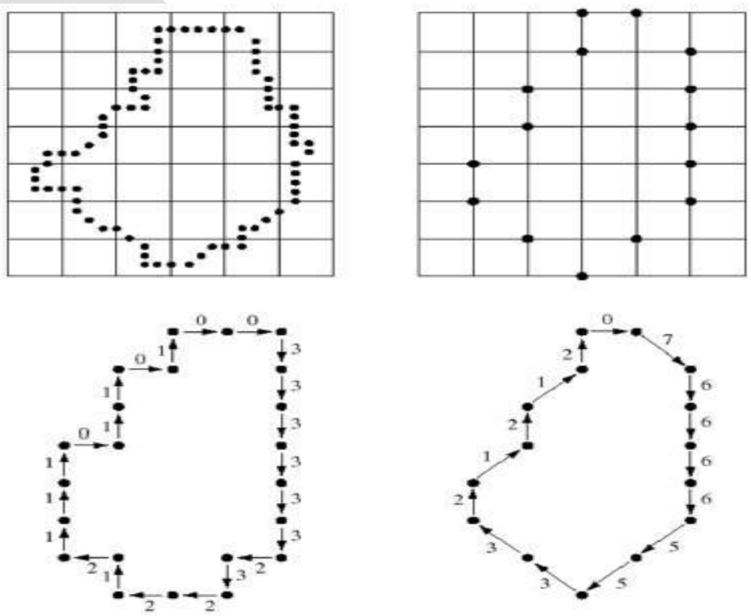


Fig. Digital boundary with resampling grid . b) Result of resampling c) 8-directional chain code boundary

2. **Explain in detail about the Polygonal approximation and some of its techniques.**

These are used to represent a boundary by straight line segments, and a closed path becomes a polygon. The number of straight line segments used determines the accuracy of the approximation. Only the minimum required number of sides necessary to preserve the needed shape information should be used (Minimum perimeter polygons). A larger number of sides will only add noise to the model.

Minimum perimeter polygons: (Merging and splitting)

Merging and splitting are often used together to ensure that vertices appear where they would naturally in the boundary. A least squares criterion to a straight line is used to stop the processing.

Cellular complex ≡ set of cells enclosing digital boundary.

Imagine the boundary as a "rubber band" and allow it to shrink. The maximum error per grid cell is √2d, where d is the dimension of a grid cell.

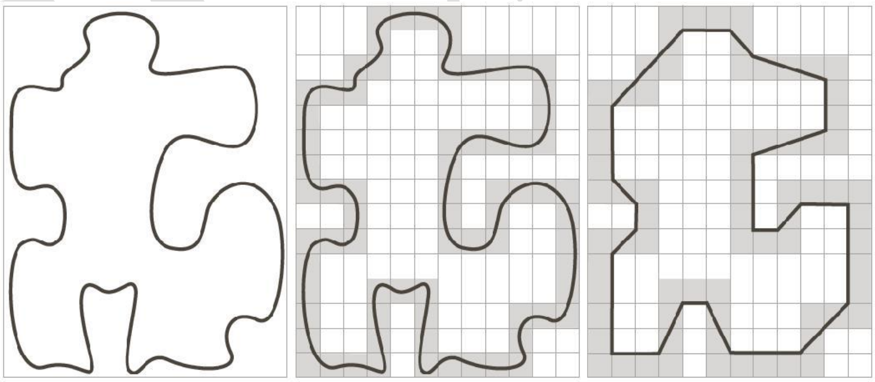


Fig. a) An object boundary b) Boundary enclosed by cells c) MPP obtained by allowing the boundary to shrink

MPP Observations:

✓ The MPP bounded by a simply connected cellular complex is not self-intersecting.

✓ Every convex vertex of the MPP is a W vertex, but not every W vertex of a boundary is a vertex of the MPP.

✓ Every mirrored concave vertex of the MPP is a B vertex, but not every B vertex of a boundary is a vertex of the MPP.

✓ All B vertices are on or outside the MPP, and all W vertices are on or inside the MPP.

✓ The uppermost, leftmost vertex in a sequence of vertices contained in a cellular complex is always a W

Let a=(x1,y1), b=(x2,y2), and c=(x3,y3)

$$A = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{bmatrix}$$

$$\operatorname{sgn}(a,b,c) = \det(A) = \begin{cases} > 0 & \text{if } (a,b,c) \text{ is a counterclockwise sequence} \\ = 0 & \text{if } (a,b,c) \text{ are collinear} \\ < 0 & \text{if } (a,b,c) \text{ is a clockwise sequence} \end{cases}$$

Form a list whose rows are the coordinates of each vertex and whether that vertex is W or B. The concave verttices must be mirrored, the vertices must be in sequential order, and the first uppermost, leftmost vertex VO is a W vertex. There is a white crawler (WC) and a black crawler (BC). The Wc crawls along the convex W vertices, and the BC crawls along the mirrored concave B vertices.

MPP Algorithm:

1. Set WC=BC=VO
2. (a) VK is on the positive side of the line (VL,WC) [sgn(VL,WC,VK)>0

   (b) VK is on the negative side of the line (VL,WC) or is collinear with it [sgn(VL,WC,VK)≤0; VK is on the positive side of the line (VL,BC) or is collinear with it [sgn(VL,BC,VK)≥0

   (c) VK is on the negative side of the line (VL,BC) [sgn(VL,BC,VK)<0

   If condition (a) holds the next MPP vertex is WC and VL=WC; set WC=BC=VL and continue with the next vertex.

   If condition (b) holds VK becomes a candidate MPP vertex. Set WC=VK if VK is convex otherwise set BC=VK. Continue with next vertex.

   If condition (c) holds the next vertex is BC and VL=BC. Re-initialize the algorithm by setting WC=BC=VL and continue with the next vertex after VL.

3. Continue until the first vertex is reached again.

The fundamental concept is to move the crawlers along the perimeter, calculate the curvatures, and determine if the vertex is a vertex of the MPP.

Other polygonal approximation approaches

Merging techniques

(1) Consider an arbitrary point on the boundary

(2) Consider the next point and fit a line through these two points: E = 0 (least squares error is zero)

(3) Now consider the next point as well, and fit a line through all three these points using a least squares approximation. Calculate E

(4) Repeat until E > T

(5) Store a and b of y = ax + b, and set E = 0

(6) Find the following line and repeat until all the edge pixels were considered.

(7) Calculate the vertices of the polygon, that is where the lines intersect.

Splitting techniques

- Joint the two furthest points on the boundary −→ line ab
- Obtain a point on the upper segment, that is c and a point on the lower segment, that is d,such that the perpendicular distance from these points to ab is as large as possible
- Now obtain a polygon by joining c and d with a and b
- Repeat until the perpendicular distance is less than some predefined fraction of ab

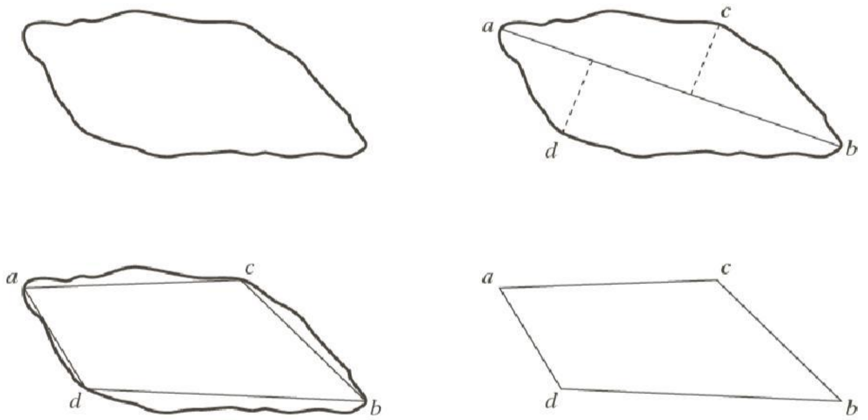
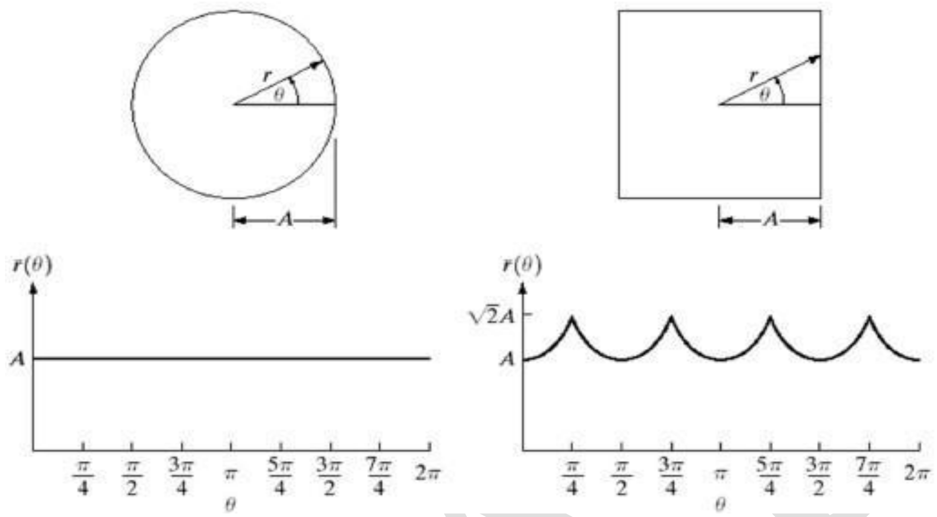


Fig.a) Original boundary b) Boundary divided into segments based on extreme points. c)Joining of vertices. d) Resulting polygon.

3. **Explain in detail about Signature and also explain Boundary segments. APR MAY 2017**

A *signature* is a 1-D representation of a boundary (which is a 2-D thing): it should be easier to describe. E.g.: distance from the centroid vs angle.



• Signatures are invariant to translation

• Invariance to rotation: depends on the starting point

- the starting point could e.g. be the one farthest from the centroid

• Scaling varies the amplitude of the signature

- invariance can be obtained by normalizing between 0 and 1, or by dividing by the variance of the signature.

Normalization for rotation:

(1) Choose the starting point as the furthest point from the centroid OR

(2) Choose the starting point as the point on the major axis that is the furthest from the centroid.

Normalization for scale Note: ↑ scale ⇒↑ amplitude of signature

(1) Scale signature between 0 and 1 Problem: sensitive to noise

(2) Divide each sample by its variance - assuming it is not zero

Alternative approach: plot Φ(θ)

• Φ: angle between the line tangent to the boundary and a reference line

• θ: angle with the positive x-axis

$\Phi(\theta)$ carry information about basic shape characteristics

Alternative approach: use the so-called slope density function as a signature, that is a histogram of the tangent-angle values

• Respond strongly to sections of the boundary with constant tangent angles (straight or nearly straight segments)

• Deep valleys in sections producing rapidly varying angles (corners or other sharp inflections).

Boundary segments.

* Decompose a boundary into segments.

* Use of the convex hull of the region enclosed by the boundary is a powerful tool for robust decomposition of the boundary.

Boundary segments are usually easier to describe than the boundary as a whole. We need a robust decomposition: convex hull

A convex set (region) is a set (region) in which any two elements (points) A and B in the set (region) can be joined by a line AB, so that each point on AB is part of the set (region). The convex hull H of an arbitrary set (region) S is the smallest convex set (region) containing S

Convex deficiency: $D = H - S$

The region boundary is partitioned by following the contour of S and marking the points at which a transition is made into or out of a component of the convex deficiency .The partitioning of irregular boundaries (that results from the digitization process or noise) usually leads to small meaningless components.

We therefore smooth the boundary prior to partitioning:

(1) Use averaging mask on coordinates of boundary pixels OR

(2) Polygonal approximation prior to computation of convex deficiency.

Skeleton

One way to represent a shape is to reduce it to a graph, by obtaining its *skeleton* via thinning (*skeletonization*). Skeletons are used to produce a one pixel wide graph that has the same basic shape of the region, like a stick figure of a human. It can be used to analyze the geometric structure of a region which has bumps and “arms”.

MAT (medial axis transformation) algorithm

Median Axis Transformation (MAT) of a region R with border B (guarantees connectivity of the skeleton)

1. For each point p in R find its closest* neighbor on B
2. If p has more than one "closest"* neighbor it belongs to the medial axis (skeleton) of B * Closest is defined using Euclidian distance

MAT is composed by all the points which have more than one closest boundary points ("prairie fire concept")

Before a thinning algorithm:

A contour point is any pixel with value 1 and having at least one 8-neighbor valued 0. Let N(p1) = P2 + P3 +....P9

T(p1) is the no. of 0-1 transitions in the ordered sequence P2,P3....P9, P2

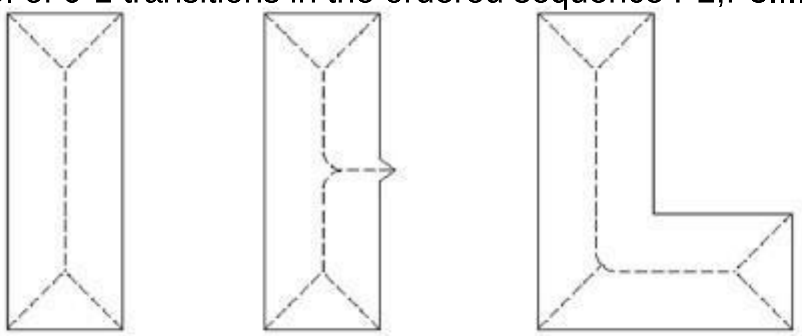


Fig. Medial axes of three simple regions

p_9	p_2	p_3
p_8	p_1	p_4
p_7	p_6	p_5

Fig. Neighbourhood arrangement used by the thinning algorithm.

Step 1: Flag a contour point p1 for deletion if the following conditions are satisfied a) 2≤ N(p1) ≤ 6 b) T(p1) = 1 c) p2 .p4.p6 = 0 d) p4.p6.p8 = 0

Step 2: Flag a contour point p1 for deletion again. However, conditions (a) and (b) remain the same, but conditions (c) and (d) are changed to p2.p6.p8 = 0, p2.p4.p8 = 0

A thinning algorithm:

✓ applying step 1 to flag border points for deletion

- ✓ deleting the flagged points
- ✓ applying step 2 to flag the remaining border points for deletion
- ✓ deleting the flagged points

This procedure is applied iteratively until no further points are deleted. One application of skeletonization is for character recognition. A letter or character is determined by the center-line of its strokes, and is unrelated to the width of the stroke lines.

4. Explain the following terms:

(i)Fourier descriptors

(ii) Statistical moments

(i)Fourier descriptors

This is a way of using the Fourier transform to analyze the shape of a boundary. The x-y coordinates of the boundary are treated as the real and imaginary parts of a complex number. Then the list of coordinates is Fourier transformed using the DFT . The Fourier coefficients are called the Fourier descriptors.

The basic shape of the region is determined by the first several coefficients, which represent lower frequencies. Higher frequency terms provide information on the fine detail of the boundary.

– It becomes a 1-D descriptor

– Fourier descriptors are not insensitive to translation..., but effects on the transform coefficients are known. Suppose that a boundary is represented by K coordinate pairs in the xy plane, (x0, y0), (x1, y1), (x2, y2), . . . , (xK−1, yK−1)

When we traverse this boundary in an anti-clockwise direction the boundary can be represented as the sequence of coordinates s(k) = [x(k), y(k)] for k = 0, 1, 2, . ,K − 1

1. Represent each point on a digital boundary as s(k) = x(k) + jy(x)
2. Compute the DFT of the set of boundary points

$$a(u)=\frac{1}{K}\sum_{k=0}^{K-1}s(k)e^{-j\frac{2\pi uk}{K}} \quad ,u=0,1,2,\ldots,K-1$$

3. The coefficients a(u) are the Fourier descriptors of the boundary Since K can be large we usually approximate the boundary by a smaller set of points, i.e., P, so that

$$s(k) \cong \hat{s}(k) = \sum_{u=0}^{P-1} a(u) e^{+j\frac{2\pi uk}{K}}$$

*Table:*Basic Properties of fourier descriptors

Transformation	Boundary	Fourier Descriptor
Identity	$s(k)$	$a(u)$
Rotation	$s_r(k) = s(k)e^{j\theta}$	$a_r(u) = a(u)e^{j\theta}$
Translation	$s_t(k) = s(k) + \Delta_{xy}$	$a_t(u) = a(u) + \Delta_{xy}\delta(u)$
Scaling	$s_s(k) = \alpha s(k)$	$a_s(u) = \alpha a(u)$
Starting point	$s_p(k) = s(k - k_0)$	$a_p(u) = a(u)e^{-j2\pi k_0 u/K}$

Rotation, scale, and translation of a boundary have simple effects on the Fourier description of that boundary.

(ii) Statistical moments

Statistical moments can be used to describe shape of boundary segment; A boundary segment can be represented by a 1-D discrete function g(r) ... The amplitude of g can now be treated as a discrete random variable v so that a histogram p(vi), i = 0, 1, . . . ,A − 1 is formed, where A is the number of amplitude increments

Moments are statistical measures of data. They come in integer orders.
Order 0 is just the number of points in the data.
Order 1 is the sum and is used to find the average.
Order 2 is related to the variance, and
Order 3 to the skew of the data.
Higher orders can also be used, but don't have simple meanings.

Once a boundary is described as a 1-D function, *statistical moments (mean, variance, and a few higher-order* central moments) can be used to describe it:

$$\mu_n(v) = \Sigma_i (v_i - m)^n p(v_i)$$
with
$$m = \Sigma_i v_i p(v_i)$$

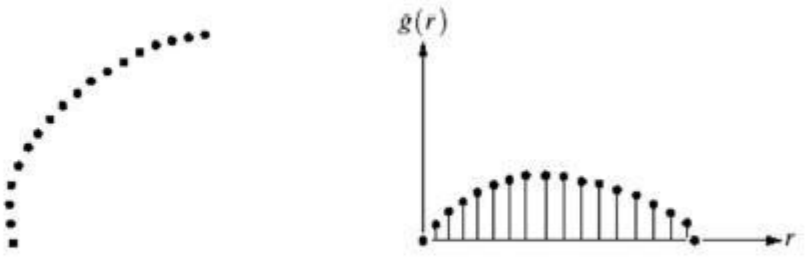


Fig. a) Boundary segment b) Representation as a 1-D function.

Generally, only the first few moments are required to differentiate between signatures of clearly distinct shapes.

Alternatively, treat $g(r_i)$ as the probability of value reoccurring, so that the moments are

$$\mu_n(r) = \sum_{i=0}^{K-1} (r_i - m)^n \, g(r_i),$$

$$m = \sum_{i=0}^{K-1} r_i \, g(r_i)$$

Here K is the number of points on the boundary, and μn(r) is directly related to the shape of g(r):

Spread of the curve: μ2(r)

Symmetry with reference to the mean: μ3(r)

5. Explain in detail about the Patterns and pattern classes.

Pattern : an arrangement of descriptors.

Pattern classes: a pattern class is a family of patterns that share some common properties.

Pattern recognition: to assign patterns to their respective classes.

Three common pattern arrangements used in practices are

* Vectors – quantitative description
* Strings – structural description
* Trees – structural description

Pattern vectors are represented by bold lowercase letters and are represented by columns (that is, n x 1 matrices). The nature of the components of a pattern vector **x** depends on the approach used to describe the physical pattern itself. Fig. Shows Three types of iris flowers described by two measurements widths and lengths of their petals.

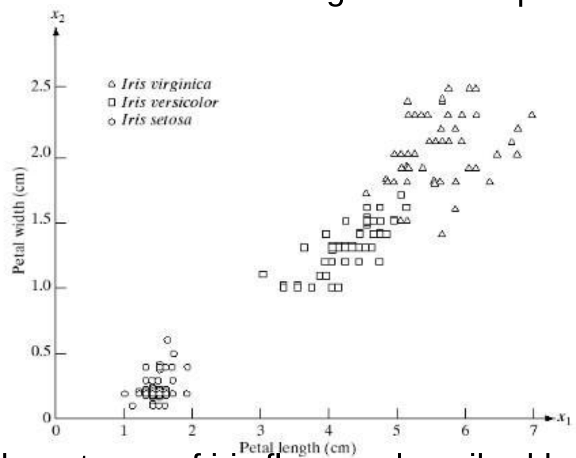


Fig. Three types of iris flowers described by two measurements.

Here is another example of pattern vector generation.

In this case, we are interested in different types of noisy shapes. Sample the signatures at some specified interval values of θ.

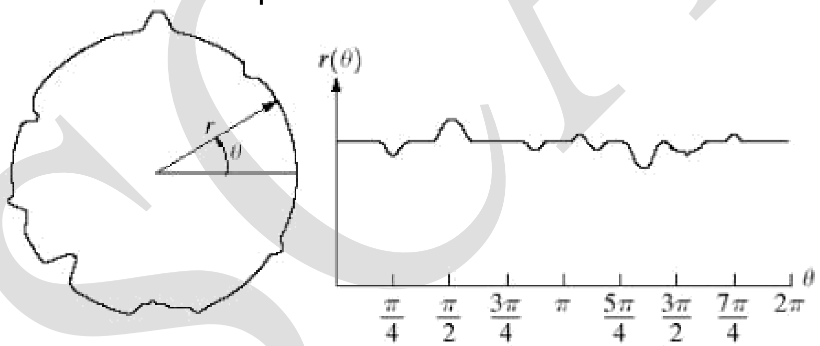


Fig. A noisy object and its corresponding signature

String descriptions adequately generate patterns of objects and other entities whose structure is based on relatively simple connectivity of primitives, usually associated with boundary shape.

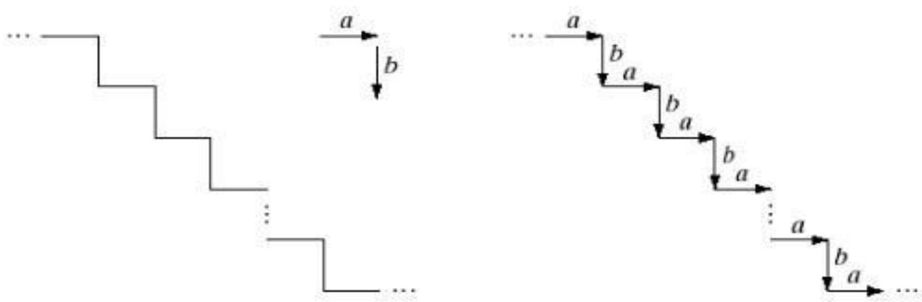


Fig. Staircase structure which is coded in terms of primitives a and b.

Tree descriptions is more powerful than string ones.

Most hierarchical ordering schemes lead to tree structure.

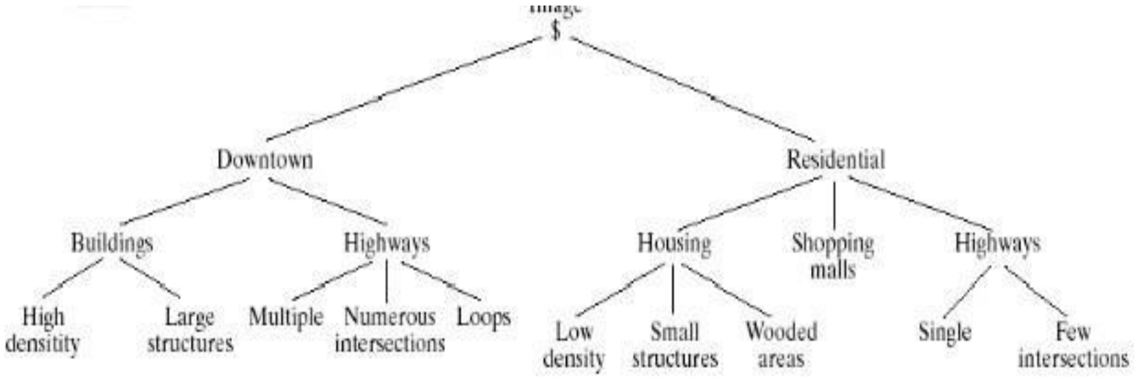


Fig. A tree description of the image

6. Define matching. How the matching is performed based on recognition.

Decision-theoretic approaches to recognition are based on the use decision functions.

Let represent an *n*-dimensional pattern vector. For *W* pattern classes , we want to find *W* decision functions with the property that, if a pattern **x** belongs to class , then

$$di(x) > dj(x) \quad j = 1,2,... W$$

The decision boundary separating class ωi and ωj is given by di(x) = dj(x)

Matching

Recognition techniques based on matching represent each class by a prototype pattern vector. An unknown pattern is assigned to the class to which it is closest in terms of predefined metric. The simplest approach is the minimum distance classifier, computes the Euclidean distance between the unknown and each of the prototype vectors. It chooses the smallest distance to make a decision.

Minimum distance classifier

Mean vector of the pattern of the class ωj

The minimum distance classifier works well when the distance between means is large compared to the spread or randomness of each class with respect to its mean.

$$m_j = \frac{1}{N_j} \sum_{x \in w_i} X \quad , \ j=1,2....W$$

distance measure

$$D_j = \|X - m_j\| \quad , \ j=1,2....W$$

the minimum distance

$$d_j(x) = X^T m_j - \frac{1}{2} m_j^T m_j \quad , \ j=1,2....W$$

decision boundary

$$d_{ij}(x) = X^T(m_i - m_j) - \frac{1}{2}(m_i - m_j)^T(m_i - m_j) = 0$$

Matching by correlation

The correlation of the mask w(x, y) of size m x n, with an image f (x, y) may be expressed in the form

$$c(x, y) = \Sigma\Sigma\, w(s, t) f(x + s, y+t)$$

where the limits of summation are taken over the region shared by ω and fSpatial correlation is related to the transforms of the functions via correlation theorem:

$$f(x, y) * w(x, y) = F^*(u, v)\, W(u, v)$$

Fig. Shows a template of size m x n whose center is at an arbitrary location(x, y). The correlation at this point is obtained by applying normalized correlation coefficient. Then the center of the teplate is incremented to an adjascent location and the procedure is repeated. The complete correlation coefficient is obtained by moving the center of the template so center of ω visits every pixel in f. At the end, maximum in γ(x, y) to find where the best match occured. It is possible to have multiple locations in γ(x, y) with the maximum value, indicating several matches between ω and f.

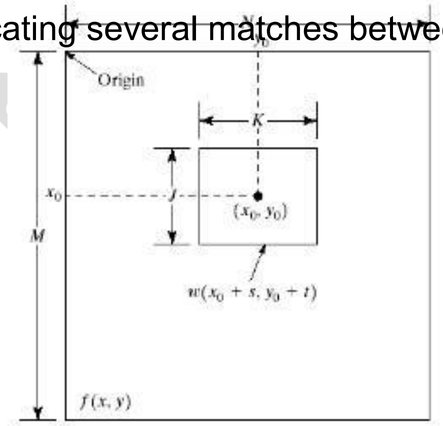


Fig. The mechanics of template matching.

B.E/B. Tech. DEGREE EXAMINATION, APRIL/MAY 2017

Sixth/Seventh Semester

Electronics and Communication Engineering

IT 6005 – DIGITAL IMAGE PROCESSING

Regulations (2013)

(Common to Biomedical Engineering, Computer Science Engineering, Electronics and Communication Engineering, Instrumentation and Control Engineering, Information Technology, Medical Electronics, Mechatronics Engineering, Electronics and Instrumentation Engineering)

Time: Three Hours **Maximum: 100 Marks**

Answer ALL questions.

PART – A (10 x 2=20 Marks)

1. When is fine sampling and coarse sampling used?
2. What is the function of an image sensor? page no. 12
3. Differentiate between image enhancement and restoration? page no. 11
4. If all the pixels in an image are shuffled, will there be any change in the histogram Justify your answer?
5. Why the restoration is called as unconstrained restoration?
6. Define region growing? page no. 80
7. What is run length coding?
8. What are the operations performed by error free compression?.
9. Does the use of chain code compress the description information of an object contour?
10. What is meant by pattern classes? page no. 126

PART B (5 x 16 = 80 Marks)

11. a. Explain the components of image processing systems? **(16)**

(OR)

(b) (i) Discuss the effects of non uniform sampling and quantization? **(8)**

(ii) Explain how color images are representaed using HSI color space model? **(8)**

12. (a) Explain the various enhancement techniques performed in spatial domain? **(16)**

(OR)

(b) If a low pass filter is formed the average the 4 neighbours of a point (x, y) but excludes point (x, y) itself. Find the equivalent filter function H(u, v) in the frequency domain. Show that it is a low pass filter **(16)**

13. (a) (i) Draw the block diagram for image degradation model and explain**.** **(8)**

(ii) Explain the use of wiener filtering in image restoration. **(8)**

(OR)

(b) Drive a wiener filter for image restoration and specify its advantages over inverse filter. **(16)**

14. (a) Explain region splitting and merging technique for image segmentation with suitable examples? **(16)**

(OR)

(b) Encode the sentence 'I LOVE IMAGE PROCESSING' using arithmetic code procedure **(16)**

15. (a) Explain in detail about the object recognition techniques based on matching. **(16)**

(OR)

(b)Explain the various boundary descriptors in detail with a neat diagram? **(16)**

B.E/B. Tech. DEGREE EXAMINATION, NOVEMBER/DECEMBER 2016

Seventh Semester

Electronics and Communication Engineering

IT60005 – DIGITAL IMAGE PROCESSING

Regulations (2013)

Time: Three Hours **Maximum: 100 Marks**

Answer ALL questions.

PART – A (10 x 2=20 Marks)

1. What is match band effect?
2. Define Checker Board Effect?
3. What is meant by bit plane slicing?
4. What is unsharp masking?
5. State the causes of degradation in an image?
6. What do you understand by Mexican hat function?
7. What is an image pyramid?
8. State whether the given Huffman code 0, 10, 01, 011 for the symbols a1, a2, a3, 4 is uniquely decodable or not?.
9. What is Skeletonizing?
10. Define texture?

PART B (5 x 16 = 80 Marks)

11. a. (i) With necessary diagrams explain how an anlog image is converted in to digital image. **(8)**

(ii) What is meant by image sensing? Explain in detail the construction and operation of various image acuisition devices? **(8)**

(OR)

(b) (i) What is color model? What are the types? Explain RGB and HSi models with necessary diagrams?. **(8)**

(ii) Explain the various distance measures used for image analysis? **(8)**

12. (a)(i) Briefly discuss about histogram Equalization techniques? **(8)**

(ii) Perform histogram equalization of the image **(8)**

4	4	4	4	4
3	4	5	4	3

3	5	5	5	3
3	4	5	4	3
4	4	4	4	4

(OR)

(b)(i) Explain the detail the method for smoothening the image in frequency domain? **(10)**

(ii) Explain Gradient operators for image enhancement? **(6)**

13. (a) (i) Apply order statistics filters on the selected pixels in the image? **(8)**

(ii) Explain how wiener filter is used for image restoration? **(8)**

1	2	3
0	1	2
1	4	5

(OR)

(b) (i) Explain the process of edge linking using Hough transform? **(8)**

(ii) Explain region based segmentation techniques? **(8)**

14. (a) (i)Explain two dimensional Discrete Wavelet Tramsform? **(8)**

(ii) Encode the word a1 a2 a3 a4 using arithmetic code and generate the tag for the given symbol with probabilities.

a1→ 0.2, a2→0.2, a3→0.4, a4→0.2. **(8)**

(OR)

(b) What is the need for imge compression? Explain image compression standards in details? **(16)**

15. (a) Explain in detail any two boundary representation schemes and illustrate with examples? **(16)**

(OR)

(b)Explain image recognition based on matching? **(16)**

B.E/B. Tech. DEGREE EXAMINATION, MAY/JUNE 2016

Seventh Semester

Electronics and Communication Engineering

EC2029/EC 708/10144 ECE 41 – DIGITAL IMAGE PROCESSING

Regulations (2008/2010)

(Common to 10144 ECE 41 – Digital Image Processing for B.E (Part Time) Seventh Semester – ECE – Regulations 2010)

Time: Three Hours **Maximum: 100 Marks**

Answer ALL questions.

PART – A (10 x 2=20 Marks)

1. Distinguish between monochrome and gray scale image?
2. What is the goal of an image transform?
3. What is image filtering?
4. Specify the need of image enhancement.
5. When will a constrained least square filter (CLS) reduce to an inverse filter?
6. What are the advantages of homomorphic filtering?
7. Compare canny and Gaussian edge detector?
8. Give two applications of image segmentation.
9. Determine whether the code (0, 01, 11) is uniquely decoded or not?
10. Differentiate Scalar and Vector quantization.

PART B (5 x 16 = 80 Marks)

11. a. (i) Explain the fundamental blocks in digital image processing system. **(7)**

(ii) Compute the DCT for the sub image of size 5x5 and the image is given as **(9)**

20	30	40	50	40
20	35	45	45	40
30	70	70	70	40
60	65	60	65	40
20	25	49	45	40

(OR)

(b) (i) Describe the elements of visual perception with suitable diagram. **(8)**

(ii) Discuss the properties and applications of KL Transform **(8)**

12. (a)(i) Explain the histogram equalization method of image enhancement.

(ii) Compare the various filters available under frequency domain for image enhancement. **(6)**

(OR)

(b)(i) Describe the filters used for noise distribution removal from images. **(8)**

(ii) Discuss the techniques applicable for color image enhancement. **(8)**

13. (a) (i) Draw the block diagram for image degradation model and explain**.** **(8)**

(ii) Explain the use of wiener filtering in image restoration. **(8)**

(OR)

(b) (i) Discuss the concept of inverse and pseudo inverse filters for image restoration. **(8)**

(ii) What are the spatial transformation techniques used for image restoration? Explain them in detail. **(8)**

14. (a) (i)Explain the thresholding approach of segmenting an image. **(8)**

(ii) Discuss the use of morphological watershed segmentation process. (**8)**

(OR)

(b) (i) Discuss in details any two region based image segmentation techniques. **(8)**

(ii) With an algorithm explain watersheds segmentation process. **(8)**

15. (a) (i) With a block diagram explain shift coding approach for image compression **(8)**

(ii) Describe the stages in MPEG image compression standard. **(8)**

(OR)

(b)(i) With an example Huffman coding scheme results with image compression **(8)**

(ii) Explain the parts of JPEG compression block diagram. **(8)**

B.E/B. Tech. DEGREE EXAMINATION, APRIL/MAY 2015.

Seventh Semester

Electronics and Communication Engineering

EC2029/EC 708/10144 ECE 41 – DIGITAL IMAGE PROCESSING

Regulations (2008/2010)

(Common to 10144 ECE 41 – Digital Image Processing for B.E (Part Time) Seventh Semester – ECE – Regulations 2010)

Time: Three Hours **Maximum: 100 Marks**

Answer ALL questions.

PART A -- (10 x 2=20 Marks)

1. Define simultaneous contrast and mach band effect.
2. Define brightness and contrast.
3. Give the PDF of uniform noise and sketch it.
4. Define and give the transfer function of Mean and Geometric Mean filter.
5. Define image degradation model and sketch it.
6. Define Geometric transformation
7. Write the properties of first order and second order derivative.
8. Define local thresholding for edge detection.
9. State the need for data compression and compare lossy and lossless compression techniques.
10. List the advantages of transform coding.

PART B --- (5 x 16 = 80 Marks)

11. (a) (i) Describe how the image is digitized by sampling and quantization and explain about checker board effect and false contouring with neat sketch. **(8)**

(ii) Find Discrete Cosine Transform and its inverse for the following image data. [0255; 2550] [2x2] matrix. **(8)**

(OR)

(b) Obtain Discrete Fourier transform for the given vectors. Input image

matrix = [0 0; 255 255] [2 x 2] matrix. Also analyse how the Fourier transform is used if the image is rotated or translated. **(16)**

12. (a) Describe historigram equalization.Obtain Historigram equalization for the following 8 bit image segment of size 5 x5. Write the inference on image segment before and after equalization.

200	200	200	180	240
180	180	180	180	190
190	200	220	220	240
230	180	190	210	230

(5 x 5) matrix **(16)**

(OR)

(b) (i)Describe how homomorphic filtering is used to separate illumination and reflectance component. **(8)**

(ii) How color image is enhanced and compare it with grayscale processing?

13. (a) Describe inverse filtering for removal of blur caused by any motion and describe how it restore the image. **(16)**

(OR)

(b) How wiener filter is helpful to reduce the mean square error when image is corrupted by motion blur and additive noise? **(16)**

14. (a) (i) How do you link edge pixels, through Hough transform? **(8)**

(ii) Describe Watershed segmentation algorithm. **(8)**

(OR)

(b) (i) Explain region based segmentation and region growing with an example. **(8)**

(ii) Discuss how to construct dams using morphological operation

15. (a) (i) Describe vector quantization with neat sketch. **8)**

(ii) A source emits letters from an alphabet A = (a1, a2, a3, a4, a5) with probabilities P (a1) = 0.3, P (a2) = 0.4, P (a3) = 0.15, P (a4) = 0.05 and P (a5) = 0.1. Find a Huffman code for this source? Find the average length of the code and its redundancy? **(8)**

(OR)

(b) (i) Decribe run length encoding with examples. **(8)**

(ii) How an image is compressed using JPEG Image compression with an image matrix? **(8)**

B.E/B. Tech. DEGREE EXAMINATION, NOVEMBER/DECEMBER, 2014

Seventh Semester

Electronics and Communication Engineering

EC2029/EC 708/10144 ECE 41 – DIGITAL IMAGE PROCESSING

Regulations (2008/2010)

(Common to 10144 ECE 41 – Digital Image Processing for B.E (Part Time) Seventh Semester – ECE – Regulations 2010)

Time: Three Hours **Maximum: 100 Marks**

Answer ALL questions.

PART A -- (10 x 2=20 Marks)

1. Compare RGB and HSI color image models.
2. Write the kernel for 2D-DCT and how this lead to data compression.
3. What are the possible ways, for adding noise in images?
4. For the following image region, obtain the medium filtered output.

72	55	33	65	32	30	21	12
15	20	3	5	18	21	65	30
35	40	34	255	200	17	51	87
0	255	20	100	101	87	59	4
65	32	18	78	86	50	21	11
30	11	8	97	108	129	151	2
68	72	19	37	14	27	50	64
36	202	111	18	26	192	23	63

5. What is Lagrange multiplier? Where it is used?
6. Why blur is to be removed from images?
7. How edges are linked through Hough transform?
8. State the problems in “region splitting and merging” based image segmentation.
9. What is shift code? How this is used in image analysis?
10. Write the performance metrics for image compression.

PART B --- (5 x 16 = 80 marks)

11. (a) (i) Write the elements of an image processing system and its work8ing. Describe the working principle of operation of vidicon camera. **(8)**

(ii) How do you obtain the 2D-DFT for a digital image? Discuss about the time complexities. **(8)**

(b) (i) What is visual perception model and explain. How this is analogous to a DIP system. **(8)**

(ii) When do you prpefer non-uniform sampling quantization? Justify. **(8)**

12. (a) (i) Write the salient features of image histogram. What do you infer? **(8)**

(ii) Explain any two techniques for color image enhancement.**(8)**

(OR)

(b) (i) How do you perform directional smoothing, in images? Why it is required? **(8)**

(ii) What is geometric mean and harmonic mean with reference to an image? What purpose do they serve for image analysis? Discuss. **(8)**

13. (a) (i) Describe how image restoration can be performed for black and white binary images. **(8)**

(ii) Compare restoration with image enhancement.

(OR)

(b) (i) What is Weiner filtering approach? How is used for image restoration? Describe. **(8)**

(ii) What are the performance measures for ascertaining the adequacy of image restoration? **(8)**

14. (a) (i) How edge detection is performed in digital images using

(1) Laplaction operator **(2)**

(2) Sobel operator and **(2)**

(3) Prewitt operator and compare their outcomes. **(2+2)**

(ii) Write morphological concepts applicable for image processing **(8)**

(OR)

(b) (i) What is meant by optimal thersholding? How do you obtain the threshold for image processing tasks? **(8)**

(ii) Describe watershed segmentation algorithm and compare with region based approaches. **(8)**

15. (a) (i) Discuss the need for image compression. Perform Huffman algorithm for the following intensity distribution, for a 64 x 64 image. Obtain the coding efficiency and compare with that of uniform length code. **(8)**

R0 = 1008
R1 = 320
R2 = 456
R3 = 686
R4 = 105
R5 = 803
R6 = 417
R7 = 301

(ii) What is arithmetic coding Illustrate. **(8)**

(OR)

(b) (i) Explain the procedure for obtaining Run length Coding(RLC) What are the advantages if any? **(8)**

(ii) Write short notes on

(1) Vector Quantization

(2) JPEG Standard. **(8)**

B.E/B. Tech. DEGREE EXAMINATION, MAY/JUNE 2014

Seventh Semester

Electronics and Communication Engineering

EC2029/EC 708 – DIGITAL IMAGE PROCESSING

Regulations (2008)

(Common to 10144 ECE 41 – Digital Image Processing for B.E (Part Time) Seventh Semester – ECE – Regulations 2010)

Time: Three Hours **Maximum: 100 Marks**

Answer ALL questions.

PART – A (10 x 2=20 Marks)

1. Define Hue and Saturation.
2. What do you mean by mach band effect?
3. Define Spatial Averaging.
4. Define the operation of a Harmonic Mean Filter.
5. Compare constrained and unconstrained Restoration.
6. What is the principle of Inverse filtering?
7. State the conditions for Region Splitting Merging Processes.
8. What are factors affecting the accuracy of Region Growing?
9. What is the need for image compression?
10. What is Run Length Encoding?

PART B – (5 x16=80 marks)

11. (a) (i) Briefly discuss about the elements of a Digital Image Processing System. **(8)**

(ii) Explain about the sampling. **(4)**

(iii) Write the kernel matrix for SVD transform. **(4)**

(OR)

(b) Explain in detail about the Vidicon and Digital camera working principles. **(16)**

12. (a) Briefly discuss about the Histogram Equalization and specification Techniques. **(16)**

(OR)

(b) Explain in detail about the Homomorphic Filtering and Harmonic mean filtering. **(16)**

13. (a) Explain the image restoration technique to remove the blur caused by uniform linear motion. **(16)**

(OR)

(b) Discuss about the Inverse Filtering and Wiener Filtering **(16)**

14. (a) (i) Write short notes on Region Merging. **(4)**

(ii) Discuss about the Edge detection and Edge linking methods. **(12)**

(OR)

(b) Explain in details about the segmentation methods by Morphological Water shed. **(16)**

15. (a) (i) What is the need for Date Compression **(6)**

(ii) Explain in details about the arithmetic Coding. **(10)**

(OR)

(b) Write short notes on the following images Codings:

(i) JPEG standard **(4)**

(ii) MPEG. **(4)**

(iii) Transform Coding