UNIT-1

INTRODUCTION and FUNDAMENTALS

1.1 INTRODUCTION

The digital image processing deals with developing a digital system that performs operations on a digital image.
An image is nothing more than a two dimensional signal. It is defined by the mathematical function f(x,y) where x and y are the two co-ordinates horizontally and vertically and the amplitude of f at any pair of coordinate (x, y) is called the intensity or gray level of the image at that point.
When x, y and the amplitude values of f are all finite discrete quantities, we call the image a digital image. The field of image digital image processing refers to the processing of digital image by means of a digital computer.
A digital image is composed of a finite number of elements, each of which has a particular location and values of these elements are referred to as picture elements, image elements, pels and pixels.

1.1.1 Motivation and Perspective

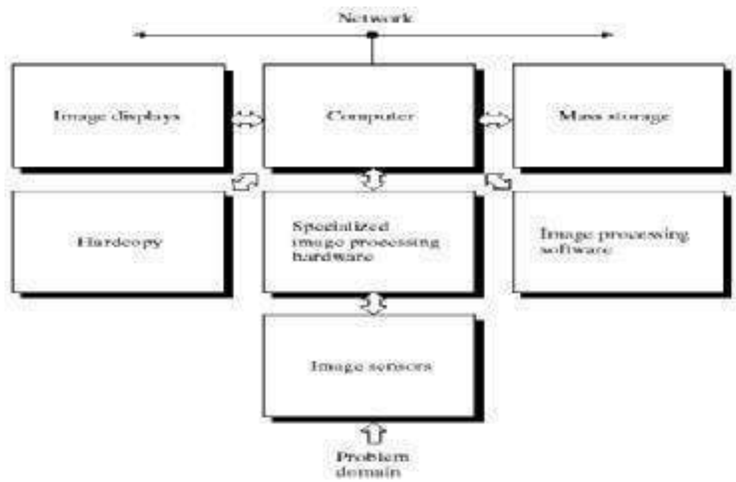
Digital image processing deals with manipulation of digital images through a digital computer. It is a subfield of signals and systems but focus particularly on images. DIP focuses on developing a computer system that is able to perform processing on an image. The input of that system is a digital image and the system process that image using efficient algorithms, and gives an image as an output. The most common example is Adobe Photoshop. It is one of the widely used application for processing digital images.

1.1.2 Applications

Some of the major fields in which digital image processing is widely used are mentioned below

(1) Gamma Ray Imaging- Nuclear medicine and astronomical observations.
(2) X-Ray imaging – X-rays of body.
(3) Ultraviolet Band –Lithography, industrial inspection, microscopy, lasers.
(4) Visual And Infrared Band – Remote sensing.
(5) Microwave Band – Radar imaging.

1.1.3 **Components of Image Processing System**



i) Image Sensors

With reference to sensing, two elements are required to acquire digital image. The first is a physical device that is sensitive to the energy radiated by the object we wish to image and second is specialized image processing hardware.

ii) Specialize image processing hardware –

It consists of the digitizer just mentioned, plus hardware that performs other primitive operations such as an arithmetic logic unit, which performs arithmetic such addition and subtraction and logical operations in parallel on images

iii) Computer

It is a general purpose computer and can range from a PC to a supercomputer depending on the application. In dedicated applications, sometimes specially designed computer are used to achieve a required level of performance

iv) Software

It consist of specialized modules that perform specific tasks a well designed package also includes capability for the user to write code, as a minimum, utilizes the specialized module. More sophisticated software packages allow the integration of these modules.

v) Mass storage –

This capability is a must in image processing applications. An image of size 1024 x1024 pixels ,in which the intensity of each pixel is an 8- bit quantity requires one megabytes of storage space if the image is not compressed .Image processing applications falls into three principal categories of storage

i) Short term storage for use during processing
ii) On line storage for relatively fast retrieval
iii) Archival storage such as magnetic tapes and disks

vi) <u>Image displays-</u>

Image displays in use today are mainly color TV monitors. These monitors are driven by the outputs of image and graphics displays cards that are an integral part of computer system

vii) <u>Hardcopy devices -</u>

The devices for recording image includes laser printers, film cameras, heat sensitive devices inkjet units and digital units such as optical and CD ROM disk. Films provide the highest possible resolution, but paper is the obvious medium of choice for written applications.

viii) <u>Networking</u>

It is almost a default function in any computer system in use today because of the large amount of data inherent in image processing applications. The key consideration in image transmission bandwidth.

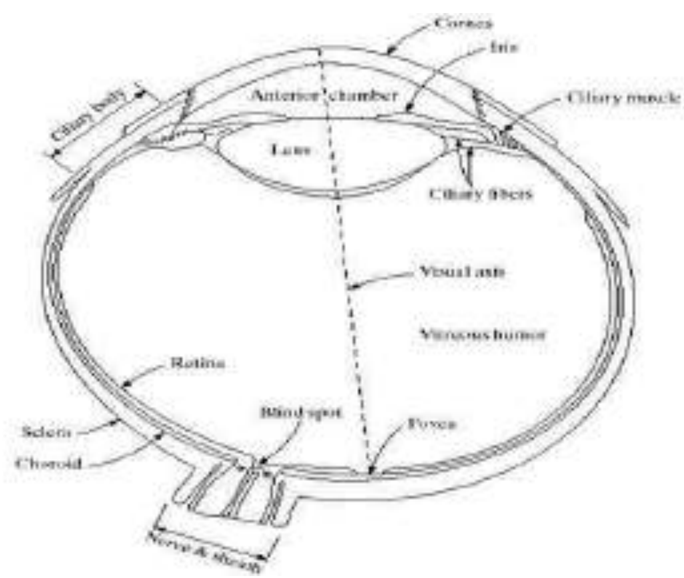
1.1.4 **<u>Elements of Visual Perception</u>**

1.1.4.1 <u>Structure of the human Eye</u>

The eye is nearly a sphere with average approximately 20 mm diameter. The eye is enclosed with three membranes

a) <u>The cornea and sclera</u> - it is a tough, transparent tissue that covers the anterior surface of the eye. Rest of the optic globe is covered by the sclera

b) <u>The choroid –</u>
It contains a network of blood vessels that serve as the major source of nutrition to the eyes. It helps to reduce extraneous light entering in the eye
It has two parts
(1) Iris Diaphragms- it contracts or expands to control the amount of light that enters the eyes
(2) Ciliary body



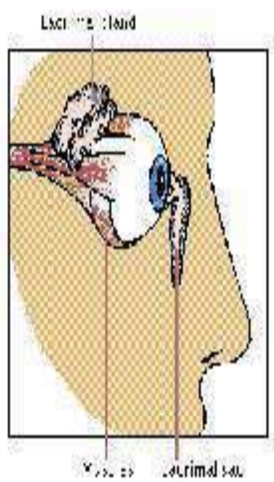
(c) Retina – it is innermost membrane of the eye. When the eye is properly focused, light from an object outside the eye is imaged on the retina.
There are various light receptors over the surface of the retina
The two major classes of the receptors are-

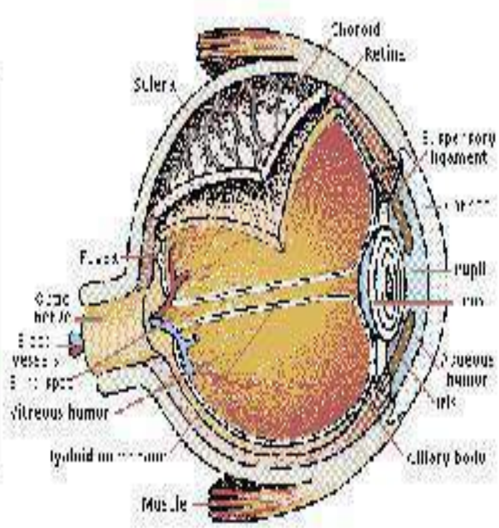
1) <u>cones</u>- it is in the number about 6 to 7 million. These are located in the central portion of the retina called the fovea. These are highly sensitive to

color. Human can resolve fine details with these cones because each one is connected to its own nerve end. Cone vision is called photopic or bright light vision

2) Rods – these are very much in number from 75 to 150 million and are distributed over the entire retinal surface. The large area of distribution and the fact that several roads are connected to a single nerve give a general overall picture of the field of view. They are not involved in the color vision and are sensitive to low level of illumination. Rod vision is called is scotopic or dim light vision.

The absent of reciprocators is called blind spot





1.1.4.2 Image Formation in the Eye

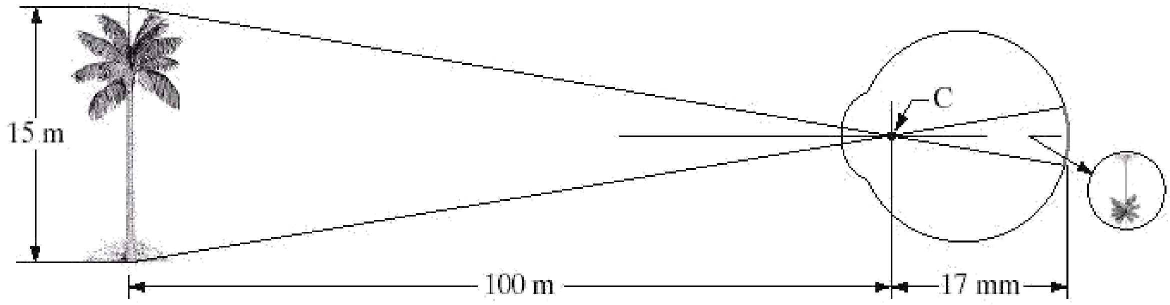
The major difference between the lens of the eye and an ordinary optical lens in that the former is flexible.

The shape of the lens of the eye is controlled by tension in the fiber of the ciliary body. To focus on the distant object the controlling muscles allow the lens to become thicker in order to focus on object near the eye it becomes relatively flattened.

The distance between the center of the lens and the retina is called the focal length and it varies from 17mm to 14mm as the refractive power of the lens increases from its minimum to its maximum**.**

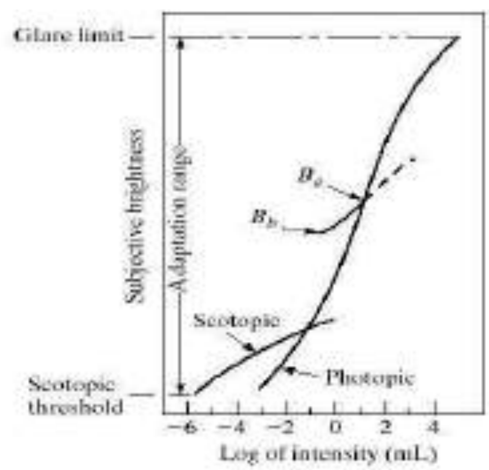
When the eye focuses on an object farther away than about 3m.the lens exhibits its lowest refractive power. When the eye focuses on a nearly object. The lens is most strongly refractive.

The retinal image is reflected primarily in the area of the fovea. Perception then takes place by the relative excitation of light receptors, which transform radiant energy into electrical impulses that are ultimately decoded by the brain.



1.1.4.3 <u>Brightness Adaption and Discrimination</u>

Digital image are displayed as a discrete set of intensities. The range of light intensity levels to which the human visual system can adopt is enormous- on the order of 10^{10}- from scotopic threshold to the glare limit. Experimental evidences indicate that subjective brightness is a logarithmic function of the light intensity incident on the eye.



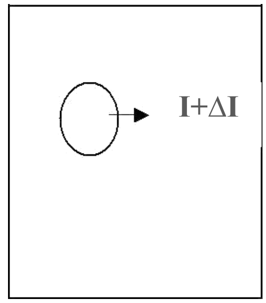
The curve represents the range of intensities to which the visual system can adopt. But the visual system cannot operate over such a dynamic range simultaneously. Rather, it is accomplished by change in its overcall sensitivity called brightness adaptation.

For any given set of conditions, the current sensitivity level to which of the visual system is called brightness adoption level , B_a in the curve. The small intersecting curve represents the range of subjective brightness that the eye can perceive when adapted to this level. It is restricted at level B_b , at and below which all stimuli are perceived as indistinguishable blacks. The upper portion of the curve is not actually restricted. whole simply raise the adaptation level higher than B_a .

The ability of the eye to discriminate between change in light intensity at any specific adaptation level is also of considerable interest.

Take a flat, uniformly illuminated area large enough to occupy the entire field of view of the subject. It may be a diffuser such as an opaque glass, that is illuminated from behind by a light source whose intensity, I can be varied. To this field is added an increment of illumination ΔI in the form of a short duration flash that appears as circle in the center of the uniformly illuminated field.

If ΔI is not bright enough, the subject cannot see any perceivable changes.



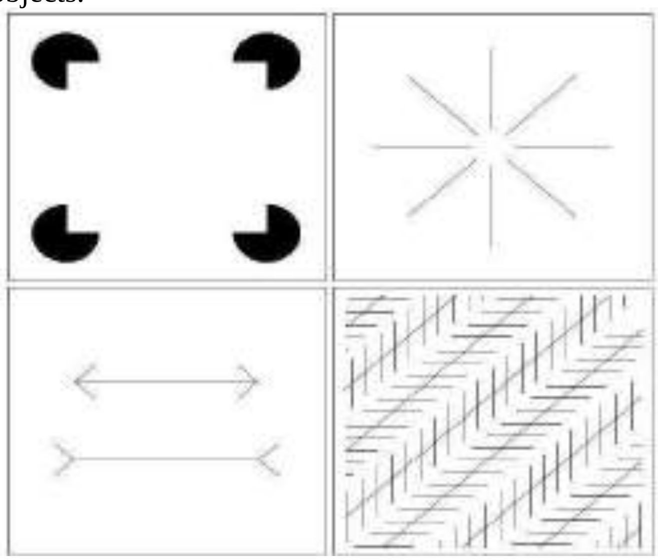
As ΔI gets stronger the subject may indicate of a perceived change. ΔI_c is the increment of illumination discernible 50% of the time with background illumination I. Now, $\Delta I_c / I$ is called the Weber ratio.

Small value means that small percentage change in intensity is discernible representing "good" brightness discrimination.

Large value of Weber ratio means large percentage change in intensity is required representing "poor brightness discrimination".

1.1.4.4 Optical illusion

In this the eye fills the non existing information or wrongly pervious geometrical properties of objects.



1.1.5 **Fundamental Steps in Digital Image Processing**

There are two categories of the steps involved in the image processing –

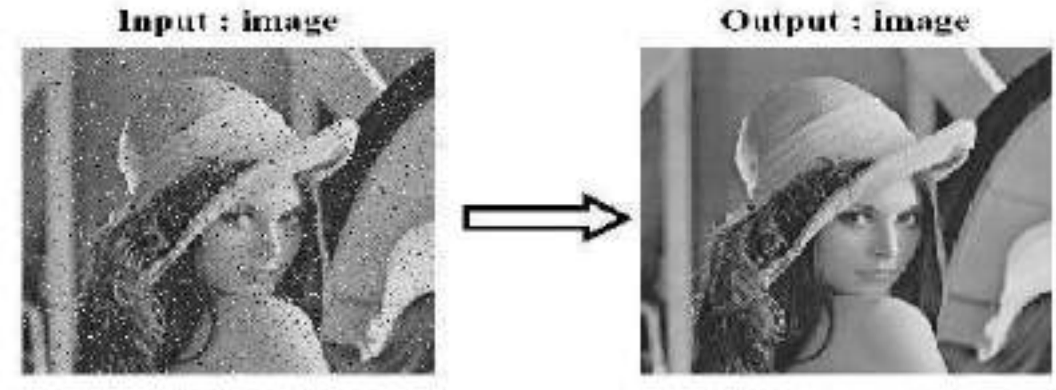
(1) Methods whose outputs are input are images.
(2) Methods whose outputs are attributes extracted from those images.

i) Image acquisition

It could be as simple as being given an image that is already in digital form. Generally the image acquisition stage involves processing such scaling.

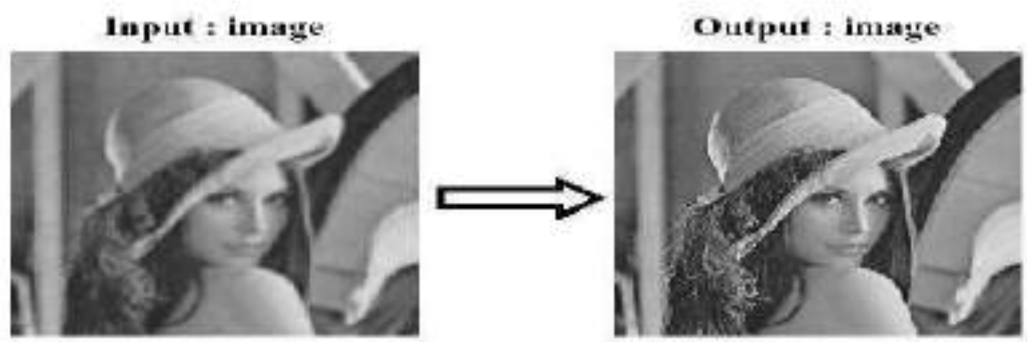
ii) Image Enhancement

It is among the simplest and most appealing areas of digital image processing. The idea behind this is to bring out details that are obscured or simply to highlight certain features of interest in image. Image enhancement is a very subjective area of image processing.



iii) <u>Image Restoration</u> –

It deals with improving the appearance of an image. It is an objective approach, in the sense that restoration techniques tend to be based on mathematical or probabilistic models of image processing. Enhancement, on the other hand is based on human subjective preferences regarding what constitutes a “good” enhancement result



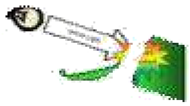
Color Image Processing	Wavelets & Multiresolution Processing	Image Compression	Morphological Image Processing
Image Restoration			Image Segmentation
Image Enhancement	Knowledge Base		Representation and description
Image Acquisition			Objects recognition

Fig: Fundamental Steps in DIP

.

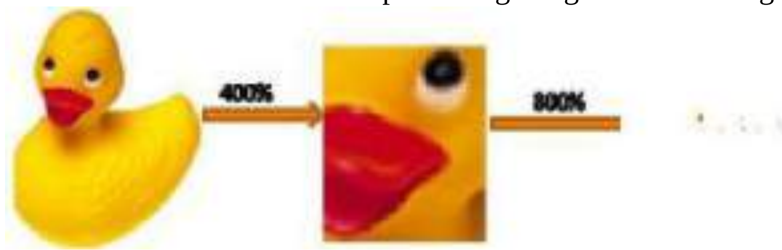
iv) <u>Color image processing</u> –

It is an area that is been gaining importance because of the use of digital images over the internet. Color image processing deals with basically color models and their implementation in image processing applications.



v) <u>Wavelets and Multiresolution Processing -</u>

These are the foundation for representing image in various degrees of resolution

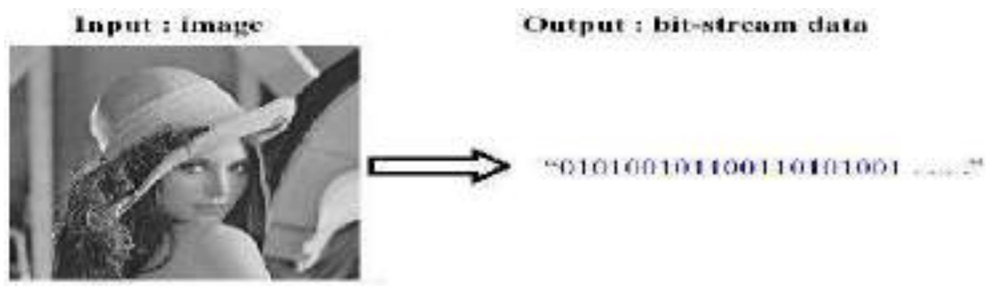


vi) <u>Compression -</u>

It deals with techniques reducing the storage required to save an image, or the bandwidth required to transmit it over the network. It has to major approaches

a) Lossless Compression

b) Lossy Compression



vii) <u>Morphological processing –</u>

It deals with tools for extracting image components that are useful in the representation and description of shape and boundary of objects. It is majorly used in automated inspection applications.

viii) <u>Representation and Description-</u>

It always follows the output of segmentation step that is, raw pixel data, constituting either the boundary of an image or points in the region itself. In either case converting the data to a form suitable for computer processing is necessary.

ix) <u>Recognition –</u>

It is the process that assigns label to an object based on its descriptors. It is the last step of image processing which use artificial intelligence of softwares.

<u>Knowledge base</u>

Knowledge about a problem domain is coded into an image processing system in the form of a knowledge base. This knowledge may be as simple as detailing regions of an image where the information of the interest in known to be located. Thus limiting search that has to be conducted in seeking the information. The knowledge base also can be quite complex such interrelated list of all major possible defects in a materials inspection problems or an image database containing high resolution satellite images of a region in connection with change detection application

1.1.6 A Simple Image Model

An image is denoted by a two dimensional function of the form f{x, y}. The value or amplitude of f at spatial coordinates {x,y} is a positive scalar quantity whose physical meaning is determined by the source of the image.
When an image is generated by a physical process, its values are proportional to energy radiated by a physical source. As a consequence, f(x,y) must be nonzero and finite; that is

$$o<f(x,y) <co$$

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

- The amount of the source illumination incident on the scene being viewed.
- The amount of the source illumination reflected back by the objects in the scene

These are called illumination and reflectance components and are denoted by i (x,y) an r (x,y) respectively.
The functions combine as a product to form f(x,y)
We call the intensity of a monochrome image at any coordinates (x,y) the gray level (l) of the image at that point

$$l= f (x, y.)$$

$$L\ min \leq l \leq Lmax$$

Lmin is to be positive and Lmax must be finite

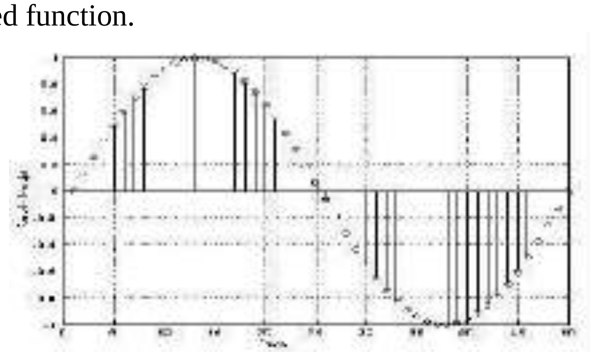
$$Lmin = imin\ rmin$$

$$Lamx = imax\ rmax$$

The interval [Lmin, Lmax] is called gray scale. Common practice is to shift this interval numerically to the interval [0, L-l] where l=0 is considered black and l= L-1 is considered white on the gray scale. All intermediate values are shades of gray of gray varying from black to white.

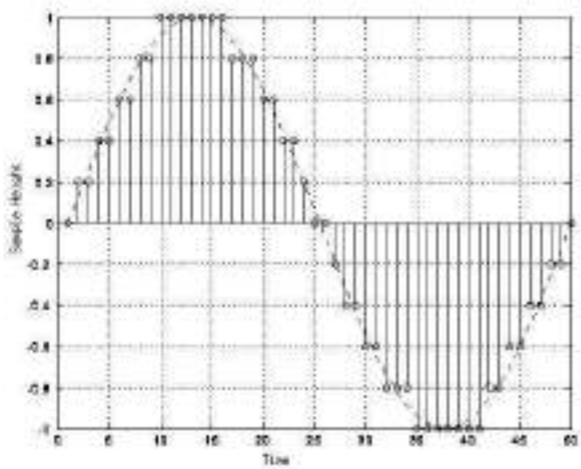
1.1.7 Image Sampling And Quantization

To create a digital image, we need to convert the continuous sensed data into digital from. This involves two processes – sampling and quantization. An image may be continuous with respect to the x and y coordinates and also in amplitude. To convert it into digital form we have to sample the function in both coordinates and in amplitudes.
Digitalizing the coordinate values is called sampling
Digitalizing the amplitude values is called quantization
There is a continuous the image along the line segment AB.
To simple this function, we take equally spaced samples along line AB. The location of each samples is given by a vertical tick back (mark) in the bottom part. The samples are shown as block squares superimposed on function the set of these discrete locations gives the sampled function.



In order to form a digital, the gray level values must also be converted (quantized) into discrete quantities. So we divide the gray level scale into eight discrete levels ranging from block to white. The vertical tick mark assign the specific value assigned to each of the eight level values.

The continuous gray levels are quantized simply by assigning one of the eight discrete gray levels to each sample. The assignment it made depending on the vertical proximity of a simple to a vertical tick mark.

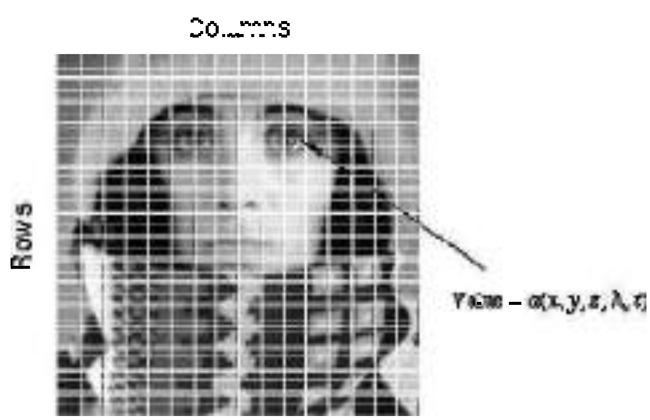


Starting at the top of the image and covering out this procedure line by line produces a two dimensional digital image.

1.1.8 **Digital Image Definition**

A digital image $f[m,n]$ described in a 2D discrete space is derived from an analog image $f(x,y)$ in a 2D continuous space through a sampling process that is frequently referred to as digitization. The mathematics of that sampling process will be described in subsequent Chapters. For now we will look at some basic definitions associated with the digital image. The effect of digitization is shown in figure 1.

The 2D continuous image $f(x,y)$ is divided into N rows and M columns. The intersection of a row and a column is termed a pixel. The value assigned to the integer coordinates $[m,n]$ with $\{m = 0,1,2,\ldots,M-1\}$ and $\{n = 0,1,2,\ldots,N-1\}$ is $f[m,n]$. In fact, in most cases $f(x,y)$, is actually a function of many variables including depth (z), color (λ) and time (t).



There are three types of computerized processes in the processing of image

1) Low level process -these involve primitive operations such as image processing to reduce noise, contrast enhancement and image sharpening. These kind of processes are characterized by fact the both inputs and output are images.

2) Mid level image processing - it involves tasks like segmentation, description of those objects to reduce them to a form suitable for computer processing, and classification of individual objects. The inputs to the process are generally images but outputs are attributes extracted from images.

3) High level processing – It involves "making sense" of an ensemble of recognized objects, as in image analysis, and performing the cognitive functions normally associated with vision.

1.1.9 **Representing Digital Images**

The result of sampling and quantization is matrix of real numbers. Assume that an image f(x,y) is sampled so that the resulting digital image has M rows and N Columns. The values of the coordinates (x,y) now become discrete quantities thus the value of the coordinates at orgin become 9X,y) =(o,o) The next Coordinates value along the first signify the iamge along the first row. it does not mean that these are the actual values of physical coordinates when the image was sampled.

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,M-1) \\ f(1,0) & f(1,1) & \cdots & f(1,M-1) \\ \vdots & & & \vdots \\ f(N-1,0) & f(N-1,1) & \cdots & f(N-1,M-1) \end{bmatrix}$$

Thus the right side of the matrix represents a digital element, pixel or pel. The matrix can be represented in the following form as well.

The sampling process may be viewed as partitioning the xy plane into a grid with the coordinates of the center of each grid being a pair of elements from the Cartesian products Z2 which is the set of all ordered pair of elements (Zi, Zj) with Zi and Zj being integers from Z.

Hence f(x,y) is a digital image if gray level (that is, a real number from the set of real number R) to each distinct pair of coordinates (x,y). This functional assignment is the quantization process. If the gray levels are also integers, Z replaces R, the and a digital image become a 2D function whose coordinates and she amplitude value are integers.
Due to processing storage and hardware consideration, the number gray levels typically is an integer power of 2.

$$L=2^K$$

Then, the number, b, of bites required to store a digital image

is $B=M * N * k$

When M=N

The equation become $b=N^2*k$

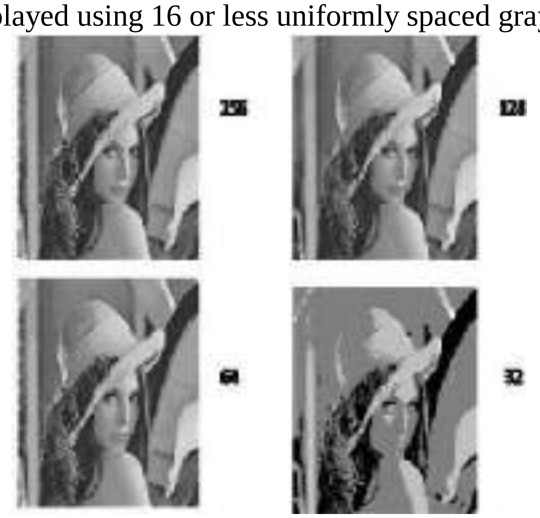
When an image can have 2^k gray levels, it is referred to as "k- bit" . An image with 256 possible gray levels is called an "8- bit image"($256=2^8$)

1.1.10 **Spatial and Gray Level Resolution**

Spatial resolution is the smallest discernible details are an image. Suppose a chart can be constructed with vertical lines of width w with the space between the also having width W, so a line pair consists of one such line and its adjacent space thus. The width of the line pair is 2w and there is 1/2w line pair per unit distance resolution is simply the smallest number of discernible line pair unit distance.

Gray levels resolution refers to smallest discernible change in gray levels

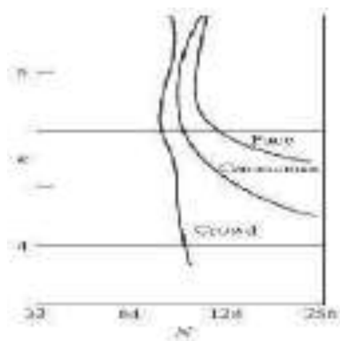
Measuring discernible change in gray levels is a highly subjective process reducing the number of bits R while repairing the spatial resolution constant creates the problem of false contouring .it is caused by the use of an insufficient number of gray levels on the smooth areas of the digital image . It is called so because the rides resemble top graphics contours in a map. It is generally quite visible in image displayed using 16 or less uniformly spaced gray levels.



1.1.11 **<u>Iso Preference Curves</u>**

To see the effect of varying N and R simultaneously. There picture are taken having littlie, mid level and high level of details.

Different image were generated by varying N and k and observers were then asked to rank the results according to their subjective quality. Results were summarized in the form of iso preference curve in the N-k plane.



The iospreference curve tends to shift right and upward but their shapes in each of the three image categories are shown in the figure. A shift up and right in the curve simply means large values for N and k which implies better picture quality

The result shows that iosreference curve tends to become more vertical as the detail in the image increases. The result suggests that for image with a large amount of details only a few gray levels may be needed. For a fixed value of N, the perceived quality for this type of image is nearly independent of the number of gray levels used.

1.1.12 **Zooming and Shrinking of Digital Images**

Zooming may be said oversampling and shirking may be called as under sampling these techniques are applied to a digital image.
These are two steps of zooming-

i) Creation of new pixel locations
ii) Assignment of gray level to those new locations.

- In order to perform gray –level assignment for any point in the overly, we look for the closet pixel in the original image and assign its gray level to the new pixel in the grid. This method rowan as nearest neighbor interpolation

- Pixel replication - Is a special case of nearest neighbor interpolation, It is applicable if we want to increase the size of an image an integer number of times.

- For eg.- to increase the size of image as double. We can duplicate each column. This doubles the size of the image horizontal direction. To increase assignment of each of each vertical direction we can duplicate each row. The gray level assignment of each pixel is determined by the fact that new location are exact duplicates of old locations.

Drawbacks

(i) Although nearest neighbor interpolation is fast ,it has the undesirable feature that it produces a check board that Is not desirable

- Bilinear interpolation-

Using the four nearest neighbor of a point .let (x,y) denote the coordinate of a point in the zoomed image and let v(x1,y1) denote the gray levels assigned to it .for bilinear interpolation .the assigned gray levels is given

by V(x1,y1)-ax1+by1+cx1y1+d

Where the four coefficient are determined from the four equation in four unknowns that can be writing using the four nearest neighbor of point (x1,y1)

Shrinking is done in the similar manner .the equivalent process of the pixel replication is row –column deletion .shrinking leads to the problem of aliasing.

1.1.13 **Pixel Relationships**

1.1.13.1 Neighbor of a pixel

A pixel p at coordinate (x,y) has four horizontal and vertical neighbor whose coordinate can be given by

(x+1, y) (X-1,y) (X ,y + 1) (X, y-1)

This set of pixel called the 4-neighbours
Of ,p is denoted by n4(p) ,Each pixel is a unit distance from (x,y) and some of the neighbors of P lie outside the digital image of (x,y) is on the border if the image .
The four diagonal neighbor of P have coordinated
(x+1,y+1),(x+1,y+1),(x-1,y+1),(x-1,y-1)

And are deported by nd (p) .these points, together with the 4-neighbours are called 8 – neighbors of P denoted by ns(p)

1.1.13.2 Adjacency

Let v be the set of gray –level values used to define adjacency ,in a binary image ,v={1} if we are reference to adjacency of pixel with value. Three types of adjacency

4- Adjacency – two pixel P and Q with value from V are 4 –adjacency if A is in the set n4(P)

8- Adjacency – two pixel P and Q with value from V are 8 –adjacency if A is in the set n8(P)

M-adjacency –two pixel P and Q with value from V are m – adjacency if

(i) Q is in n4 (p) or
(ii)Q is in nd (q) and the set N4(p) U N4(q) has no pixel whose values are from V

1.1.13.3 Distance measures

For pixel p,q and z with coordinate (x.y) ,(s,t) and (v,w) respectably D is a distance function or metric if
D [p.q] ≥ O {D[p.q] = O iff
p=q} D [p.q] = D [p.q] and
D [p.q] ≥ O {D[p.q]+D(q,z)

The Education Distance between p and is defined as
De (p,q) = Iy – t I

The D4 Education Distance between p and is defined as
De (p,q) = Iy – t I

1.2 IMAGE ENHANCEMENT IN FREQUENCY DOMAIN

1.2.1 Fourier Transform and the Frequency Domain

Any function that periodically reports itself can be expressed as a sum of sines and cosines of different frequencies each multiplied by a different coefficient, this sum is called Fourier series. Even the functions which are non periodic but whose area under the curve if finite can also be represented in such form; this is now called Fourier transform.

A function represented in either of these forms and can be completely reconstructed via an inverse process with no loss of information.

1.2.1.1 1-D Fourier Transformation and its Inverse

If there is a single variable, continuous function f(x) , then Fourier transformation F (u) may be given as

$$\mathcal{F}\{f(x)\} = F(u) = \int_{-\infty}^{\infty} f(x)\exp(-j2\pi ux)dx \qquad j=\sqrt{-1}$$

And the reverse process to recover f(x) from F(u) is

$$\mathcal{F}^{-1}\{F(u)\} = f(x) = \int_{-\infty}^{\infty} F(u)\exp[j2\pi ux]du$$

Equation (a) and (b) comprise of Fourier transformation pair.

Fourier transformation of a discrete function of one variable f(x), x=0, 1, 2, m-1 is given by

$$F(u) = \frac{1}{N}\sum_{x=0}^{N-1} f(x)\exp[-j2\pi ux/N] \quad \text{for } u=0,1,2,\ldots,N-1$$

to obtain f(x) from F(u)

$$f(x) = \sum_{u=0}^{N-1} F(u)\exp[j2\pi ux/N] \quad \text{for } x=0,1,2,\ldots,N-1$$

The above two equation (e) and (f) comprise of a discrete Fourier transformation pair. According to Euler's formula

$$e^{jx} = \cos x + j \sin x$$

Substituting these value to equation (e)

F(u)=**Σ**f(x)[cos 2**π**ux/N+jSin 2**π**ux/N] for u=0,1,2,……,N-1

Now each of the m terms of F(u) is called a frequency component of transformation

"The Fourier transformation separates a function into various components, based on frequency components. These components are complex quantities.

F(u) in polar coordinates

$$F(u) = R(u) + jI(u) \qquad F(u) = |F(U)|e^{j\phi(u)}$$

or

$$|F(u)| = [R^2(u) + I^2(u)]^{1/2} \qquad \phi(u) = \tan^{-1}\left[\frac{I(u)}{R(u)}\right]$$

1.2.1.2 2-D Fourier Transformation and its Inverse

The Fourier Transform of a two dimensional continuous function f(x,y) (an image) of size M * N is given by

$$\mathcal{F}\{f(x,y)\} = F(u,v) = \int\int_{-\infty} f(x,y)\exp[-j2\pi(ux+vy)]dxdy$$

Inverse Fourier transformation is given by equation

$$\mathcal{F}^{-1}\{F(u,v)\} = f(x,y) = \int_{-\infty}^{\infty}\int F(u,v)\exp[j2\pi(ux+vy)]dudv$$

Where (u,v) are frequency variables.
Preprocessing is done to shift the origin of F(u,v) to frequency coordinate (m/2,n/2) which is the center of the M*N area occupied by the 2D-FT. It is known as frequency rectangle.

It extends form u =0 to M-1 and v=0 to N-1. For this, we multiply the input image by $(-1)^{x+y}$ prior to compute the transformation

$$\mathcal{F}\{f(x,y)\,(-1)^{x+y}\} = F(u-M/2,\ v-N/2)$$

$\mathcal{F}(.)$ denotes the Fourier transformation of the argument Value of transformation at (u,v)=(0,0) is

$$F(0,0)=1/MN\Sigma\Sigma f(x,y)$$

1.2.1.3 Discrete Fourier Transform

$\{f(x_0), f(x_0+\Delta x), \ldots \ldots f(x_0+[N-1]\Delta x)\}$

$\Rightarrow f(x)=f(x_0+x\,\Delta x)$

f(0),f(1),f(2)……f(N-1) denotes any N uniformly spaced samples.

$$\text{DFT } F(u)=\frac{1}{N}\sum_{x=0}^{N-1} f(x)\exp\left[-j2\pi ux/N\right] \text{ for } u=0,1,2,\ldots,N-1$$

Extending it to two variables

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

for u=0,1,2,….,M-1 v=0,1,2,…N-1

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

for x=0,1,…,M-1 y=0,1,…..,N-1

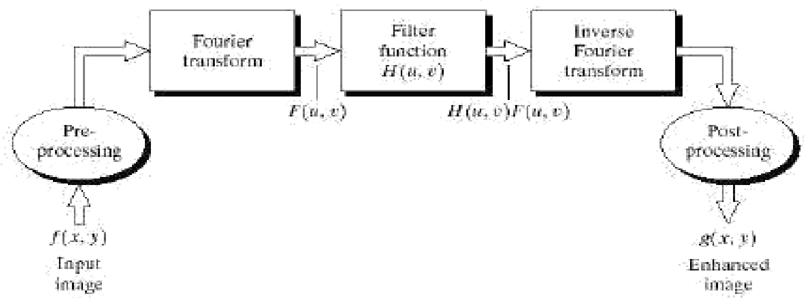
$$\Delta u=\frac{1}{M\Delta x} \qquad \Delta v=\frac{1}{N\Delta y}$$

1.2.2 **Basis of Filtering in Frequency Domain**

Basic steps of filtering in frequency Domain

i) Multiply the input image by $(-1)^{X+Y}$ to centre the transform
ii) Compute F(u,v), Fourier Transform of the image
iii) Multiply f(u,v) by a filter function H(u,v)
iv) Compute the inverse DFT of Result of (iii)
v) Obtain the real part of result of (iv)
vi) Multiply the result in (v) by $(-1)^{x=y}$

Frequency domain filtering operation



H(u,v) called a filter because it suppresses certain frequencies from the image while leaving others unchanged.

1.2.3 **Filters**

1.2.3.1 Smoothing Frequency Domain Filters

Edges and other sharp transition of the gray levels of an image contribute significantly to the high frequency contents of its Fourier transformation. Hence smoothing is achieved in the frequency domain by attenuation a specified range of high frequency components in the transform of a given image.
Basic model of filtering in the frequency domain is

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

F(u,v) - Fourier transform of the image to be smoothed

Objective is to find out a filter function H (u,v) that yields G (u,v) by attenuating the high frequency component of F (u,v)
There are three types of low pass filters

1. Ideal
2. Butterworth
3. Gaussian

1.2.3.1.1 IDEAL LOW PASS FILTER

It is the simplest of all the three filters
It cuts of all high frequency component of the Fourier transform that are at a distance greater that a specified distance D_0 form the origin of the transform.
it is called a two – dimensional ideal low pass filter (ILPF) and has the transfer function

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

Where D_0 is a specified nonnegative quantity and D(u,v) is the distance from point (u,v) to the center of frequency rectangle
If the size of image is M*N , filter will also be of the same size so center of the frequency rectangle (u,v) = (M/2, N/2) because of center transform

$$D(u,v) = (u^2 + v^2)^{1/2}$$

Because it is ideal case. So all frequency inside the circle are passed without any attenuation where as all frequency outside the circle are completely attenuated

For an ideal low pass filter cross section, the point of transition between H (u,v) =1 and H (u,v)=0 is called of the " cut of frequency

One way to establish a set of standard cut of frequency locus is to compute circle that include specified amount of total image Power Pt

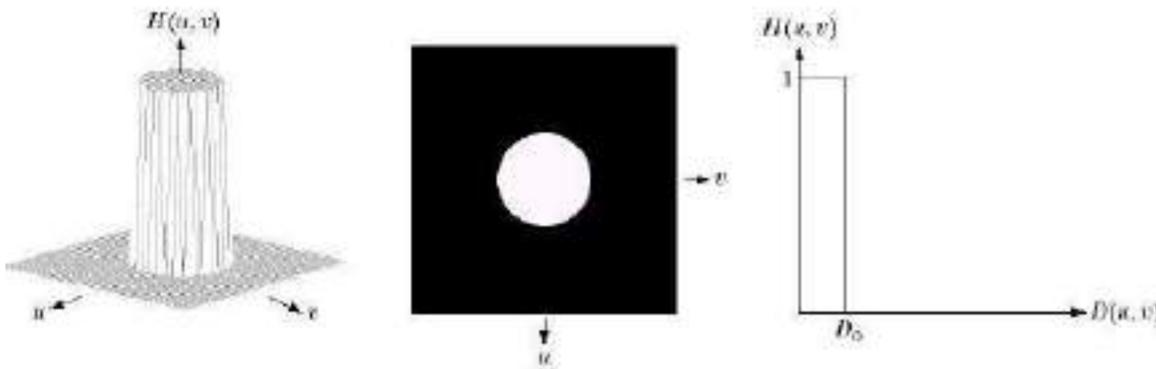
$$|100[\sum_{u}\sum_{v} P(u,v)/P_T$$

It can be obtained by summing the components of the power spectrum at each point (u,v) for u=0,1,2,3,4,..,,,,,,,,,,,,,,,, N-1.

If transform has been centered a circle of radius r with origin at the center of the frequency rectangle encloses ∞ percent of the power

For R = 5 ∞ = 92 % most blurred image because all sharp details are removed
R = 15 ∞ = 94.6 %
R = 30 ∞ = 96.4 %
R = 80 ∞ = 98 % maximum ringing only 2 % power is removed
R = 230 ∞ = 99.5 % very slight blurring only 0.5 % power is removed

ILPF is not suitable for practical usage. But they can be implemented in any computer system

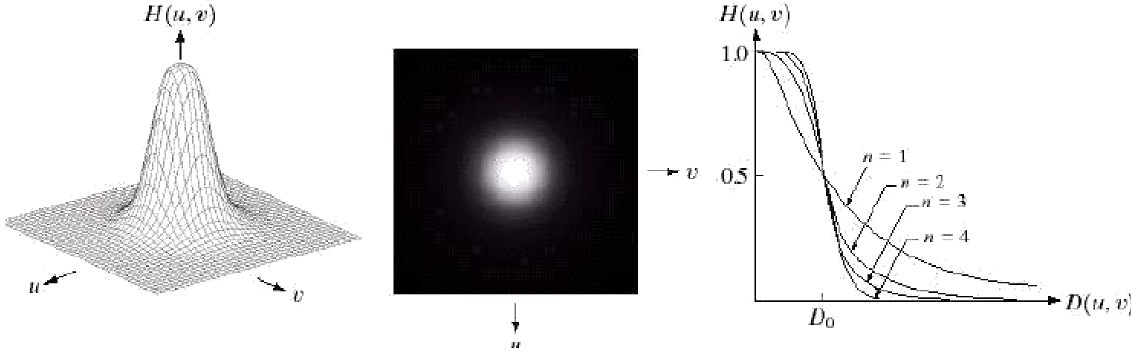


1.2.3.1.2 <u>BUTTERWORTH LOW PASS FILTER</u>

It has a parameter called the filter order.
For high values of filter order it approaches the form of the ideal filter whereas for low filter order values it reach Gaussian filter. It may be viewed as a transition between two extremes.
The transfer function of a Butterworth low pass filter (BLPF) of order n with cut off frequency at distance Do from the origin is defined as

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

Most appropriate value of n is 2.
It does not have sharp discontinuity unlike ILPF that establishes a clear cutoff between passed and filtered frequencies.
Defining a cutoff frequency is a main concern in these filters. This filter gives a smooth transition in blurring as a function of increasing cutoff frequency. A Butterworth filter of order 1 has no ringing. Ringing increases as a function of filter order. (Higher order leads to negative values)



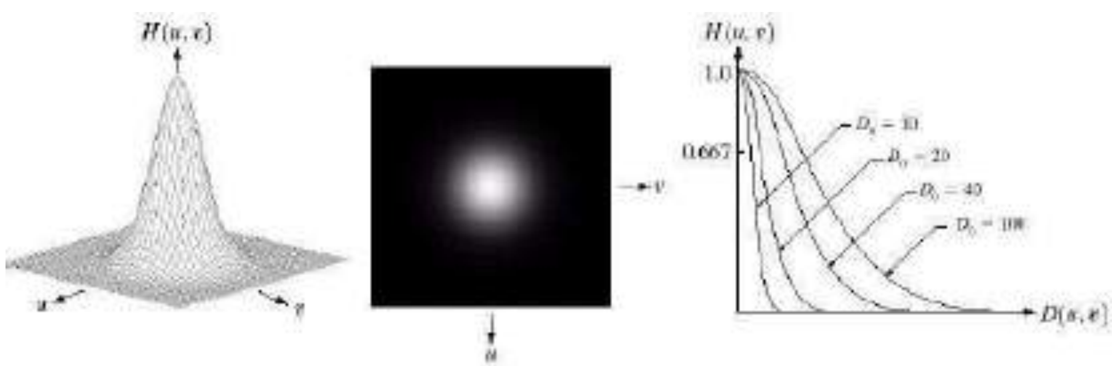
1.2.3.1.3 GAUSSIAN LOW PASS FILTER

The transfer function of a Gaussian low pass filter is

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

Where:
D(u,v)- the distance of point (u,v) from the center of the
transform **σ** = D_0- specified cut off frequency

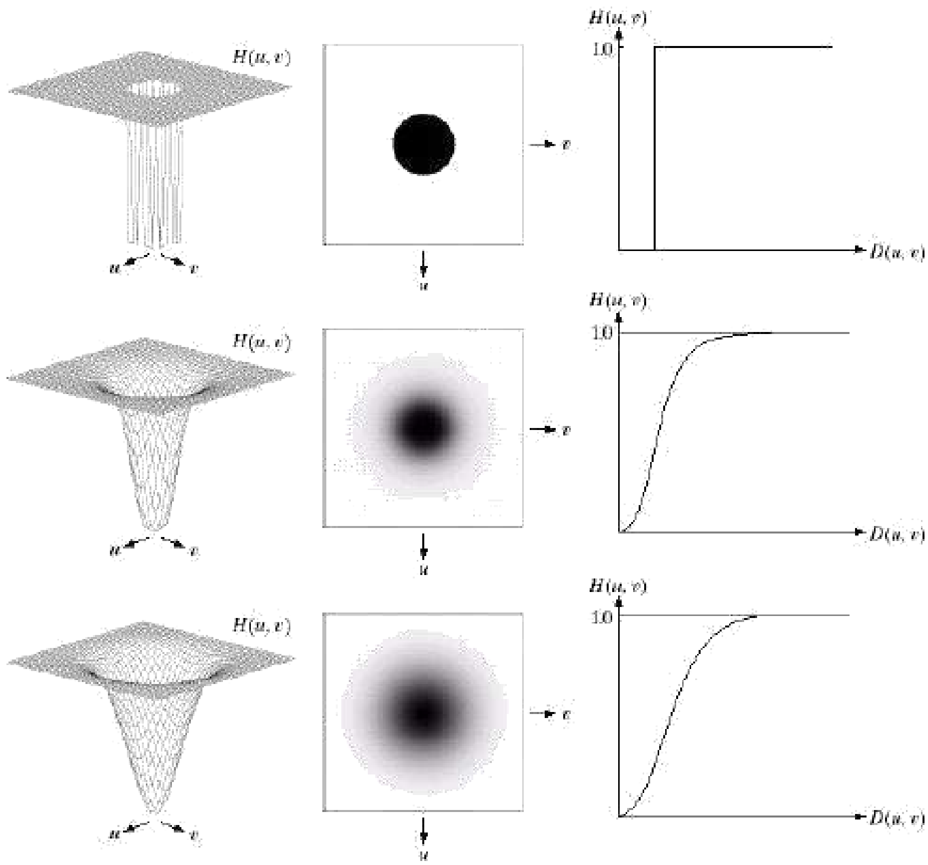
The filter has an important characteristic that the inverse of it is also Gaussain.



1.2.3.2 SHARPENING FREQUENCY DOMAIN FILTERS

Image sharpening can be achieved by a high pass filtering process, which attenuates the low-frequency components without disturbing high-frequency information. These are radially symmetric and completely specified by a cross section.
If we have the transfer function of a low pass filter the corresponding high pass filter can be obtained using the equation

$$Hhp\,(u,v) = 1 - Hlp\,(u,v)$$



1.2.3.2.1 IDEAL HIGH PASS FILTER

This filter is opposite of the Ideal Low Pass filter and has the transfer function of the form

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

1.2.3.2.2 BUTTERWORTH HIGH PASS FILTER

The transfer function of Butterworth High Pass filter of order n is given by the equation

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

1.2.3.2.3 GAUSSIAN HIGH PASS FILTER

The transfer function of a Gaussain High Pass Filter is given by the equation

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

1.2.4 **Homomorphic Filtering**

Homomorphic filters are widely used in image processing for compensating the effect of no uniform illumination in an image. Pixel intensities in an image represent the light reflected from the corresponding points in the objects. As per as image model, image f(z,y) may be characterized by two components: (1) the amount of source light incident on the scene being viewed, and (2) the amount of light reflected by the objects in the scene. These portions of light are called the illumination and reflectance components, and are denoted i (x , y) and r (x , y) respectively. The functions i (x , y) and r (x , y) combine multiplicatively to give the image function f (x , y):

$$f(x, y) = i(x, y).r(x, y) \quad (1)$$

where 0 < i (x , y) < a and 0 < r(x , y) < 1. Homomorphic filters are used in such situations where the image is subjected to the multiplicative interference or noise as depicted in Eq. 1. We cannot easily use the above product to operate separately on the frequency components of illumination and reflection because the Fourier transform of f (x , y) is not separable; that is

F[f(x,y)) not equal to F[i(x, y)].F[r(x, y)].

We can separate the two components by taking the logarithm of the two sides

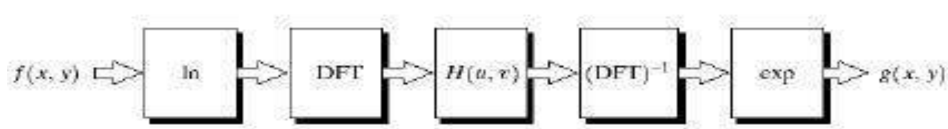
$$\ln f(x,y) = \ln i(x, y) + \ln r(x, y).$$

Taking Fourier transforms on both sides we get,

F[ln f(x,y)} = F[ln i(x, y)} + F[ln r(x, y)].

that is, F(x,y) = I(x,y) + R(x,y), where F, I and R are the Fourier transforms ln f(x,y),ln i(x, y) , and ln r(x, y). respectively. The function F represents the Fourier transform of the sum of two images: a low-frequency illumination image and a high-frequency reflectance image. If we now apply a filter with a transfer function that suppresses low- frequency components and enhances high-frequency components, then we can suppress the illumination component and enhance the reflectance component. Taking the inverse transform of F (x , y) and then anti-logarithm, we get

$$f'(x, y) = i'(x, y) + r'(x, y)$$



UNIT -2

IMAGE ENHENCEMENT IN SPATIAL DOMAIN

2.1 IMAGE ENHENCEMENT IN SPATIAL DOMAIN

2.1.1 Introduction

The principal objective of enhancement is to process an image so that the result is more suitable than the original image for a specific application. Image enhancement approaches fall into two board categories

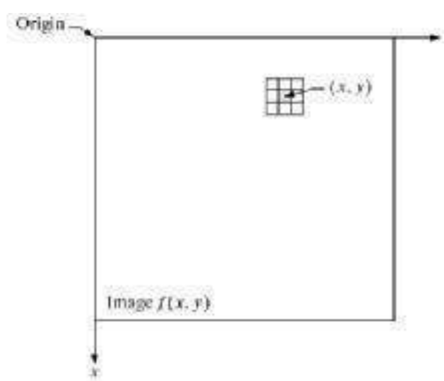
- ⇨ Spatial domain methods
- ⇨ Frequency domain methods

The term spatial domain refers to the image plane itself and approaches in this categories are based on direct manipulation of pixel in an image.
Spatial domain process are denoted by the expression

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

$f(x,y)$- input image T- operator on f, defined over some neighborhood of $f(x,y)$
$g(x,y)$-processed image

The neighborhood of a point (x,y) can be explain by using as square or rectangular sub image area centered at (x,y).



The center of sub image is moved from pixel to pixel starting at the top left corner. The operator T is applied to each location (x,y) to find the output g at that location . The process utilizes only the pixel in the area of the image spanned by the neighborhood.

2.1.2 Basic Gray Level Transformation Functions

It is the simplest form of the transformations when the neighborhood is of size IXI. In this case g depends only on the value of f at (x,y) and T becomes a gray level transformation function of the forms

$$S=T(r)$$

r- Denotes the gray level of $f(x,y)$

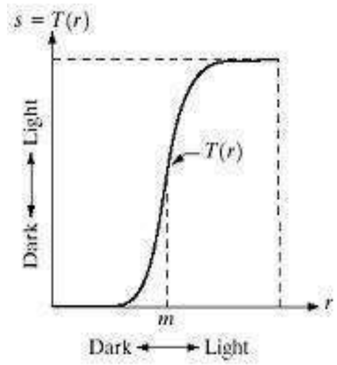
s- Denotes the gray level of g(x,y) at any point (x,y)

Because enhancement at any point in an image deepens only on the gray level at that point, technique in this category are referred to as point processing.
There are basically three kinds of functions in gray level transformation –

2.1.2.1 Point Processing

2.1.2.1.1 Contract stretching -

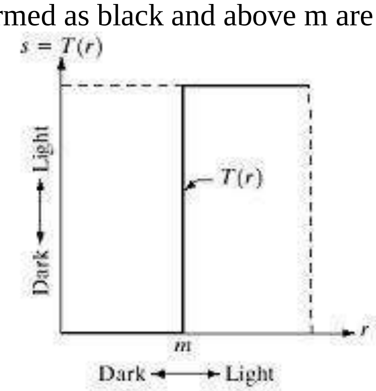
It produces an image of higher contrast than the original one.
The operation is performed by darkening the levels below m and brightening the levels above m in the original image.



In this technique the value of r below m are compressed by the transformation function into a narrow range of s towards black .The opposite effect takes place for the values of r above m.

2.1.2.1.2 Thresholding function -

It is a limiting case where T(r) produces a two levels binary image.
The values below m are transformed as black and above m are transformed as white.



2.1.2.2 Basic Gray Level Transformation

These are the simplest image enhancement techniques.

2.1.2.2.1 Image Negative –

The negative of in image with gray level in the range [0, l-1] is obtained by using the negative transformation.
The expression of the transformation is

$$s= L-1-r$$

Reverting the intensity levels of an image in this manner produces the equivalent of a photographic negative. This type of processing is practically suited for enhancing white or gray details embedded in dark regions of an image especially when the black areas are dominant in size.

2.1.2.2.2 Log transformations

The general form of the log transformation is

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

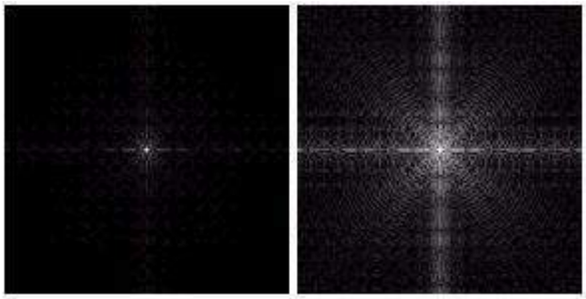
Where c- constant

$R \geq 0$

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

The log transformation function has an important characteristic that it compresses the dynamic range of images with large variations in pixel values.

Eg- Fourier spectrum



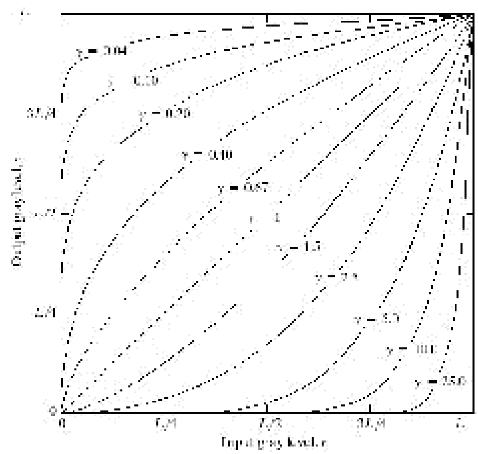
2.1.2.2.3 Power Law Transformation

Power law transformations has the basic form

$$S=cr^{y}$$

Where c and y are positive constants.

Power law curves with fractional values of y map a narrow range of dark input values into a wider range of output values, with the opposite being true for higher values of input gray levels. We may get various curves by varying values of y.



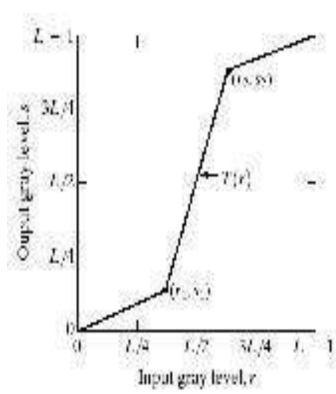
A variety of devices used for image capture, printing and display respond according to a power law. The process used to correct this power law response phenomenon is called gamma correction.
For eg-CRT devices have intensity to voltage response that is a power function.
Gamma correction is important if displaying an image accurately on a computer screen is of concern. Images that are not corrected properly can look either bleached out or too dark.
Color phenomenon also uses this concept of gamma correction. It is becoming more popular due to use of images over the internet.
It is important in general purpose contract manipulation. To make an image black we use y>1 and y<1 for white image.

2.1.2.3 Piece wise Linear transformation functions

The principal advantage of piecewise linear functions is that these functions can be arbitrarily complex. But their specification requires considerably more user input.

2.1.2.3.1 Contrast Stretching

It is the simplest piecewise linear transformation function.
We may have various low contrast images and that might result due to various reasons such as lack of illumination, problem in imaging sensor or wrong setting of lens aperture during image acquisition.
The idea behind contrast stretching is to increase the dynamic range of gray levels in the image being processed.



The location of points (r1,s1) and (r_2,s_2) control the shape of the curve

a) If $r_1=r_2$ and $s_1=s_2$, the transformation is a linear function that deduces no change in gray levels.

b) If $r1=s1$, $s1=0$, and $s2=L-1$, then the transformation become a thresholding function that creates a binary image

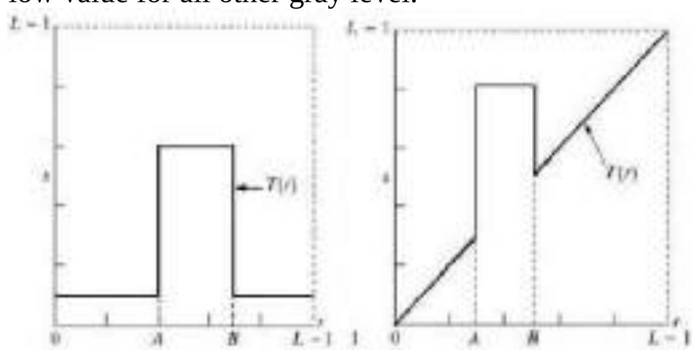
c) Intermediate values of $(r1, s1)$ and $(r2, s2)$ produce various degrees of spread in the gray value of the output image thus effecting its contract.

Generally $r1\leq r2$ and $s1 \leq s2$ so that the function is single valued and monotonically increasing

2.1.2.3.2 Gray Level Slicing

Highlighting a specific range of gray levels in an image is often desirable
For example when enhancing features such as masses of water in satellite image and enhancing flaws in x- ray images.
There are two ways of doing this-

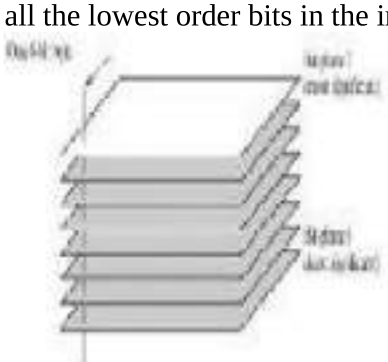
(1) One method is to display a high value for all gray level in the range. Of interest and a low value for all other gray level.

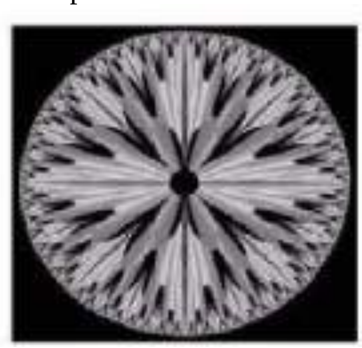


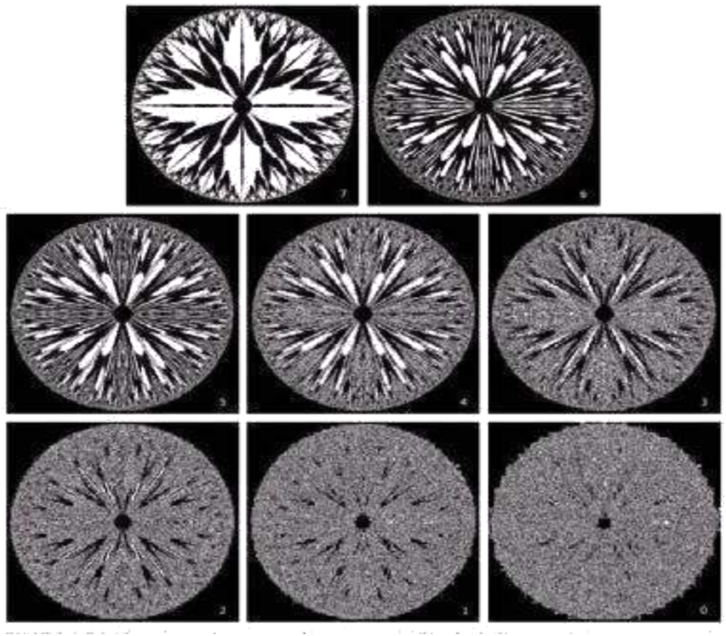
(2) Second method is to brighten the desired ranges of gray levels but preserve the background and gray level tonalities in the image.

2.1.2.3.3 Bit Plane Slicing

Sometimes it is important to highlight the contribution made to the total image appearance by specific bits. Suppose that each pixel is represented by 8 bits.
Imagine that an image is composed of eight 1-bit planes ranging from bit plane 0 for the least significant bit to bit plane 7 for the most significant bit. In terms of 8-bit bytes, plane 0 contains all the lowest order bits in the image and plane 7 contains all the high order bits.







.
High order bits contain the majority of visually significant data and contribute to more subtle details in the image.

Separating a digital image into its bits planes is useful for analyzing the relative importance played by each bit of the image.

It helps in determining the adequacy of the number of bits used to quantize each pixel. It is also useful for image compression.

2.1.3 **Histogram Processing**

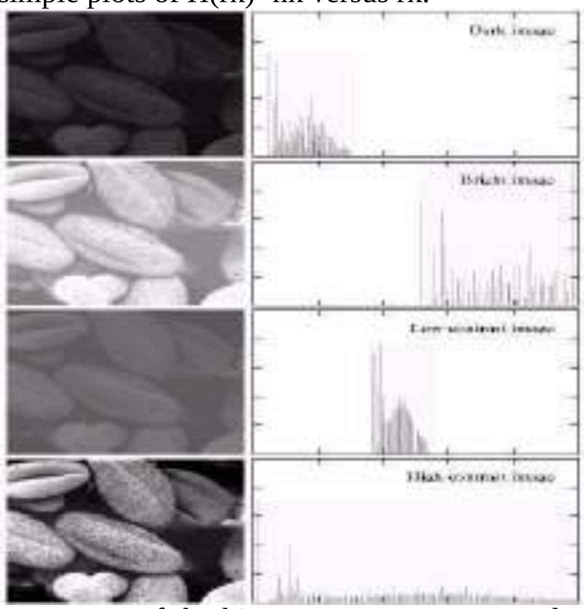
The histogram of a digital image with gray levels in the range [0, L-1] is a discrete function of the form

$$H(r_k)=n_k$$

where r_k is the kth gray level and n_k is the number of pixels in the image having the level r_k.. A normalized histogram is given by the equation

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

$P(r_k)$ gives the estimate of the probability of occurrence of gray level r_k.
The sum of all components of a normalized histogram is equal to 1.
The histogram plots are simple plots of $H(r_k)=n_k$ versus r_k.



In the dark image the components of the histogram are concentrated on the low (dark) side of the gray scale. In case of bright image the histogram components are baised towards the high side of the gray scale.
The histogram of a low contrast image will be narrow and will be centered towards the middle of the gray scale.
The components of the histogram in the high contrast image cover a broad range of the gray scale. The net effect of this will be an image that shows a great deal of gray levels details and has high dynamic range.

2.1.3.1 Histogram Equalization

Histogram equalization is a common technique for enhancing the appearance of images. Suppose we have an image which is predominantly dark. Then its histogram would be skewed towards the lower end of the grey scale and all the image detail are compressed into the dark end of the histogram. If we could 'stretch out' the grey levels at the dark end to produce a more uniformly distributed histogram then the image would become much clearer.
Let there be a continuous function with r being gray levels of the image to be enhanced.
The range of r is [0, 1] with r=0 repressing black and r=1 representing white.
The transformation function is of the form

$S=T(r)$ where $0<r<1$ It produces a level s for every pixel value r in the original image. The transformation function is assumed to fulfill two condition T(r) is single valued and monotonically increasing in the internal

$$0<T(r)<1 \text{ for } 0<r,1$$

The transformation function should be single valued so that the inverse transformations should exist. Monotonically increasing condition preserves the increasing order from black to white in the output image.The second conditions guarantee that the output gray levels will be in the same range as the input levels.

The gray levels of the image may be viewed as random variables in the interval [0.1]

The most fundamental descriptor of a random variable is its probability density function (PDF)

Pr(r) and Ps(s) denote the probability density functions of random variables r and s respectively.

Basic results from an elementary probability theory states that if Pr(r) and Tr are known and T-1 (s) satisfies conditions (a), then the probability density function Ps(s) of the transformed variable s is given by the formula-

$$P_s(s) = P_r(r)\frac{dr}{ds},$$

Thus the PDF of the transformed variable s is the determined by the gray levels PDF of the input image and by the chosen transformations function.

A transformation function of a particular importance in image processing

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

This is the cumulative distribution function of r.

Using this definition of T we see that the derivative of s with respect to r is

$$\frac{ds}{dr} = P_r(r)$$

Substituting it back in the expression for Ps we may get

$$P_s(s) = P_r(r)\frac{1}{P_r(r)} = 1$$

An important point here is that Tr depends on Pr(r) but the resulting Ps(s) always is uniform, and independent of the form of P(r).

For discrete values we deal with probability and summations instead of probability density functions and integrals.

The probability of occurrence of gray levels rk in an image as approximated

$$Pr(r) = nk/N$$

N is the total number of the pixels in an image.

nk is the number of the pixels that have gray level rk.

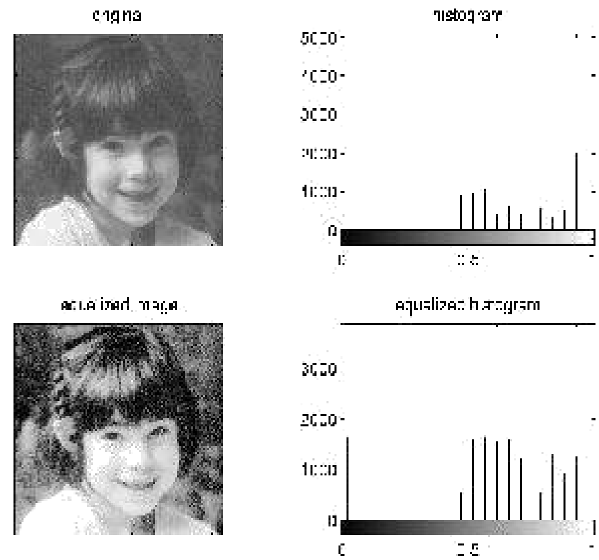
L is the total number of possible gray levels in the image. The discrete transformation function is given by

$$s_k = T(r_k) = \sum_{i=0}^{k} \frac{n_i}{N} = \sum_{i=0}^{k} P_r(r_i).$$

Thus a processed image is obtained by mapping each pixel with levels rk in the input image into a corresponding pixel with level sk in the output image.

A plot of Pr (rk) versus rk is called a histogram. The transformation function given by the above equation is the called histogram equalization or linearization.

Given an image the process of histogram equalization consists simple of implementing the transformation function which is based information that can be extracted directly from the given image, without the need for further parameter specification.



Equalization automatically determines a transformation function that seeks to produce an output image that has a uniform histogram. It is a good approach when automatic enhancement is needed

2.1.3.2 Histogram Matching (Specification)

In some cases it may be desirable to specify the shape of the histogram that we wish the processed image to have.
Histogram equalization does not allow interactive image enhancement and generates only one result: an approximation to a uniform histogram. Sometimes we need to be able to specify particular histogram shapes capable of highlighting certain gray-level ranges. The method use to generate a processed image that has a specified histogram is called histogram matching or histogram specification.

Algorithm

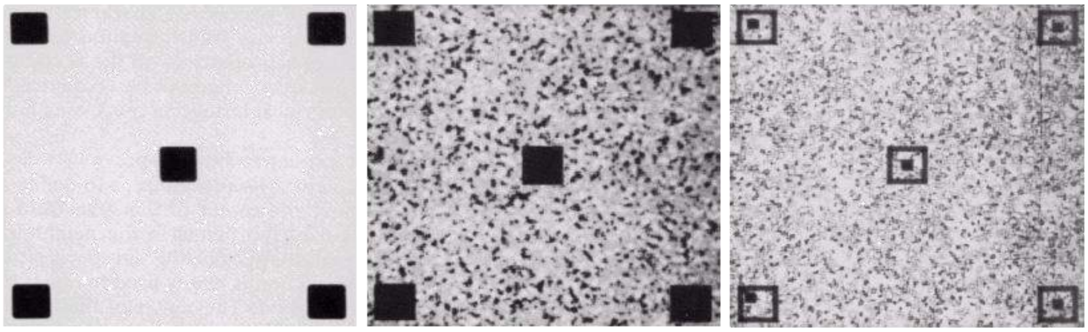
1. Compute sk=Pf (k), k = 0, …, L-1, the cumulative normalized histogram of f .
2. Compute G(k), k = 0, …, L-1, the transformation function, from the given histogram hz 3. Compute G-1(sk) for each k = 0, …, L-1 using an iterative method (iterate on z), or in effect, directly compute G-1(Pf (k))
4. Transform f using G-1(Pf (k)) .

2.1.4 **Local Enhancement**

In earlier methods pixels were modified by a transformation function based on the gray level of an entire image. It is not suitable when enhancement is to be done in some small areas of the image. This problem can be solved by local enhancement where a transformation function is applied only in the neighborhood of pixels in the interested region.

Define square or rectangular neighborhood (mask) and move the center from pixel to pixel. For each neighborhood

1) Calculate histogram of the points in the neighborhood
2) Obtain histogram equalization/specification function
3) Map gray level of pixel centered in neighborhood
4) The center of the neighborhood region is then moved to an adjacent pixel location and the procedure is repeated.

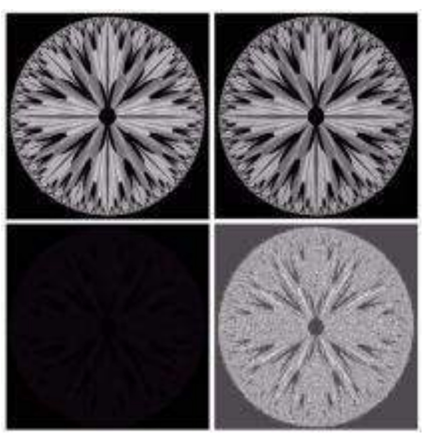


2.1.5 **Enhancement Using Arithmetic/Logic Operations**

These operations are performed on a pixel by basis between two or more images excluding not operation which is performed on a single image. It depends on the hardware and/or software that the actual mechanism of implementation should be sequential, parallel or simultaneous.
Logic operations are also generally operated on a pixel by pixel basis.
Only AND, OR and NOT logical operators are functionally complete. Because all other operators can be implemented by using these operators.
While applying the operations on gray scale images, pixel values are processed as strings of binary numbers.
The NOT logic operation performs the same function as the negative transformation.
Image Masking is also referred to as region of Interest (Ro1) processing. This is done to highlight a particular area and to differentiate it from the rest of the image.
Out of the four arithmetic operations, subtraction and addition are the most useful for image enhancement.

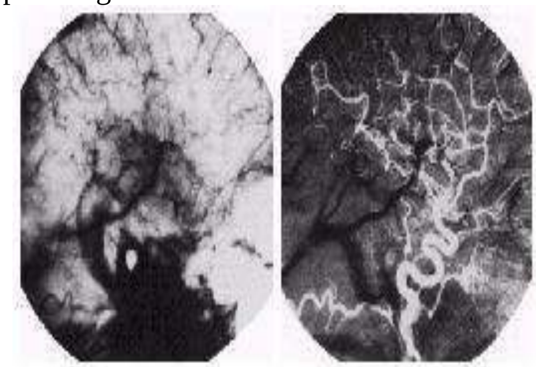
2.1.5.1 Image Subtraction

The difference between two images f(x,y) and h(x,y) is expressed
as g(x,y)= f(x,y)-h(x,y)
It is obtained by computing the difference between all pairs of corresponding pixels from f and h. The key usefulness of subtraction is the enhancement of difference between images.
This concept is used in another gray scale transformation for enhancement known as bit plane slicing The higher order bit planes of an image carry a significant amount of visually relevant detail while the lower planes contribute to fine details.
It we subtract the four least significant bit planes from the image The result will be nearly identical but there will be a slight drop in the overall contrast due to less variability in the gray level values of image .



The use of image subtraction is seen in medical imaging area named as mask mode radiography. The mask h (x,y) is an X-ray image of a region of a patient’s body this image is captured by using as intensified TV camera located opposite to the x-ray machine then a consistent medium is

injected into the patient's blood storm and then a series of image are taken of the region same as h(x,y).
The mask is then subtracted from the series of incoming image. This subtraction will give the area which will be the difference between f(x,y) and h(x,y) this difference will be given as enhanced detail in the output image.



This procure produces a move shoving now the contrast medium propagates through various arteries of the area being viewed.
Most of the image in use today is 8- bit image so the values of the image lie in the range 0 to 255. The value in the difference image can lie from -255 to 255. For these reasons we have to do some sort of scaling to display the results
There are two methods to scale an image

(i) Add 255 to every pixel and then divide at by 2.

This gives the surety that pixel values will be in the range 0 to 255 but it is not guaranteed whether it will cover the entire 8 – bit range or not.
It is a simple method and fast to implement but will not utilize the entire gray scale range to display the results.

(ii) Another approach is

(a) Obtain the value of minimum difference
(b) Add the negative of minimum value to the pixels in the difference image(this will give a modified image whose minimum value will be 0)
(c) Perform scaling on the difference image by multiplying each pixel by the quantity 255/max. This approach is complicated and difficult to implement.

Image subtraction is used in segmentation application also

2.1.5.2 <u>Image Averaging</u>

Consider a noisy image g(x,y) formed by the addition of noise n(x,y) to the original image f(x,y) g(x,y) = f(x,y) + n(x,y)
Assuming that at every point of coordinate (x,y) the noise is uncorrelated and has zero average value
The objective of image averaging is to reduce the noise content by adding a set of noise images, {gi(x,y)}
If in image formed by image averaging K different noisy images

$$\bar{g}(x, y) = \frac{1}{K}\sum_{i=1}^{K} g_i(x, y)$$

$$E\{\bar{g}(x, y)\}=f(x,y)$$

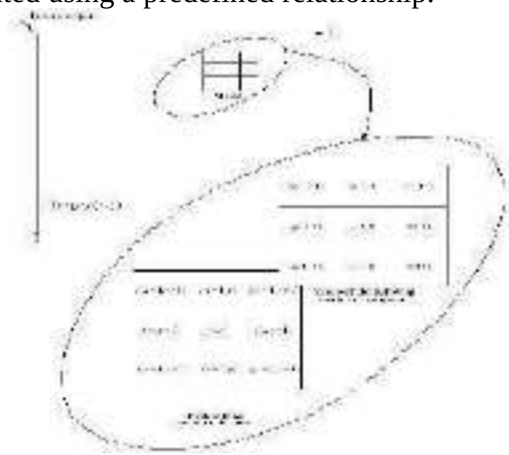
As k increases the variability (noise) of the pixel value at each location (x,y) decreases
E{g(x,y)} = f(x,y) means that g(x,y) approaches f(x,y) as the number of noisy image used in the averaging processes increases

Image averaging is important in various applications such as in the field of astronomy where the images are low light levels

2.1.6 **Basic of Spatial Filtering**

Spatial filtering is an example of neighborhood operations, in this the operations are done on the values of the image pixels in the neighborhood and the corresponding value of a sub image that has the same dimensions as of the neighborhood

This sub image is called a filter, mask, kernel, template or window; the values in the filter sub image are referred to as coefficients rather than pixel. Spatial filtering operations are performed directly on the pixel values (amplitude/gray scale) of the image
The process consists of moving the filter mask from point to point in the image. At each point (x,y) the response is calculated using a predefined relationship.



For linaer spatial filtering the response is given by a sum of products of the filter coefficient and the corresponding image pixels in the area spanned by the filter mask.
The results R of liner filtering with the filter mask at point (x,y) in the image is

$$R = w(-1,-1)f(x-1,y-1) + w(-1,0)f(x-1,y) + \cdots + w(0,0)f(x,y) + \cdots + w(1,0)f(x+1,y) + w(1,1)f(x+1,y+1)$$

The sum of products of the mask coefficient with the corresponding pixel directly under the mask. The coefficient w (0,0) coincides with image value f(x,y) indicating that mask it centered at (x,y) when the computation of sum of products takes place

For a mask of size MxN we assume m=2a+1 and n=2b+1, where a and b are nonnegative integers. It shows that all the masks are of add size.

In the general liner filtering of an image of size f of size M*N with a filter mask of size m*m is given by the expression

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

Where a= (m-1)/2 and b = (n-1)/2

To generate a complete filtered image this equation must be applied for x=0, 1, 2, -----M-1 and y=0,1,2---,N-1. Thus the mask processes all the pixels in the image.

The process of linear filtering is similar to frequency domain concept called convolution. For this reason, linear spatial filtering often is referred to as convolving a mask with an image. Filter mask are sometimes called convolution mask.

$$R= W,Z,+W_2, Z_2 +\ldots+ W_{mn} Z_{mn}$$

Where w's are mask coefficients and
z's are the values of the image gray levels corresponding to those coefficients. mn is the total number of coefficients in the mask.

An important point in implementing neighborhood operations for spatial filtering is the issue of what happens when the center of the filter approaches the border of the image.
There are several ways to handle this situation.

i) To limit the excursion of the center of the mask to be at distance of less than (n-1) /2 pixels form the border. The resulting filtered image will be smaller than the original but all the pixels will be processed with the full mask.

ii) Filter all pixels only with the section of the mask that is fully contained in the image. It will create bands of pixels near the border that will be processed with a partial mask.

iii) Padding the image by adding rows and columns of o's & or padding by replicating rows and columns. The padding is removed at the end of the process.

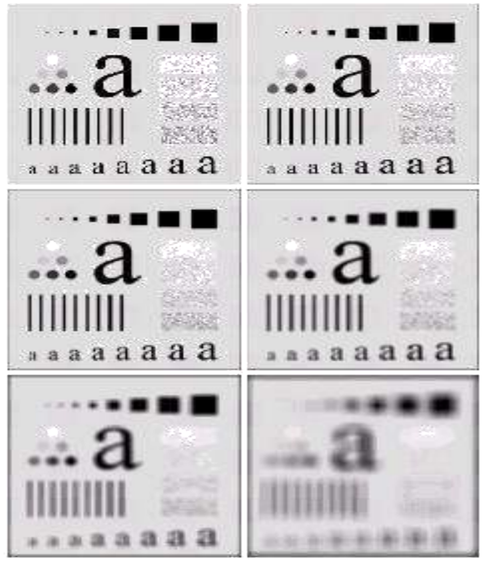
2.1.6.1 Smoothing Spatial Filters

These filters are used for blurring and noise reduction blurring is used in preprocessing steps such as removal of small details from an image prior to object extraction and bridging of small gaps in lines or curves.

2.1.6.1.1 Smoothing Linear Filters

The output of a smoothing liner spatial filter is simply the average of the pixel contained in the neighborhood of the filter mask. These filters are also called averaging filters or low pass filters.

The operation is performed by replacing the value of every pixel in the image by the average of the gray levels in the neighborhood defined by the filter mask. This process reduces sharp transitions in gray levels in the image.



A major application of smoothing is noise reduction but because edge are also provided using sharp transitions so smoothing filters have the undesirable side effect that they blur edges . It also removes an effect named as false contouring which is caused by using insufficient number of gray levels in the image.
Irrelevant details can also be removed by these kinds of filters, irrelevant means which are not of our interest.

A spatial averaging filter in which all coefficients are equal is sometimes referred to as a "box filter"

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

A weighted average filter is the one in which pixel are multiplied by different coefficients.

2.1.6.1.2 Order Statistics Filter

These are nonlinear spatial filter whose response is based on ordering of the pixels contained in the image area compressed by the filter and the replacing the value of the center pixel with value determined by the ranking result.

The best example of this category is median filter. In this filter the values of the center pixel is replaced by median of gray levels in the neighborhood of that pixel. Median filters are quite popular because, for certain types of random noise, they provide excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filters.

These filters are particularly effective in the case of impulse or salt and pepper noise. It is called so because of its appearance as white and black dots superimposed on an image.

The median £ of a set of values is such that half the values in the set less than or equal to £ and half are greater than or equal to this. In order to perform median filtering at a point in an image, we first sort the values of the pixel in the question and its neighbors, determine their median and assign this value to that pixel.

We introduce some additional order-statistics filters. Order-statistics filters are spatial filters whose response is based on ordering (ranking) the pixels contained in the image area encompassed by the filter. The response of the filter at any point is determined by the ranking result

2.1.6.1.2.1 Median filter

The best-known order-statistics filter is the median 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:

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

The original value of the pixel is included in the computation of the median. Median filters are quite popular because, for certain types of random noise, they provide excellent noise-reduction capabilities, with considerably less blurring than linear smoothing filters of similar size. Median filters are particularly effective in the presence of both bipolar and unipolar impulse noise. In fact, the median filter yields excellent results for images corrupted by this type of noise.

2.1.6.1.2.2 Max and min filters

Although the median filter is by far the order-statistics filter most used 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 statis¬tics that ranking lends itself to many other possibilities. For example, using the 100th perccntile results in the so-called max filter given by:

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

This filter is useful for finding the brightest points in an image. Also, because pepper noise has very low values, it is reduced by this filter as a result of the max selection process in the subimage area S. The 0th percentile filter is the Min filter.

2.1.6.2 Sharpening Spatial Filters

The principal objective of sharpening is to highlight fine details in an image or to enhance details that have been blurred either in error or as a natural effect of particular method for image acquisition.

The applications of image sharpening range from electronic printing and medical imaging to industrial inspection and autonomous guidance in military systems.

As smoothing can be achieved by integration, sharpening can be achieved by spatial differentiation. The strength of response of derivative operator is proportional to the degree of discontinuity of the image at that point at which the operator is applied. Thus image differentiation enhances edges and other discontinuities and deemphasizes the areas with slow varying grey levels.

It is a common practice to approximate the magnitude of the gradient by using absolute values instead of square and square roots.

A basic definition of a 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 can define a second order derivative as the difference

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

2.1.6.2.1 The LAPLACIAN

The second order derivative is calculated using Laplacian. It is simplest isotropic filter. Isotropic filters are the ones whose response is independent of the direction of the image to which the operator is applied.

The Laplacian for a two dimensional function f(x,y) is defined as

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

Partial second order directive 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

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

The digital implementation of a two-dimensional Laplacian obtained by summing the two components

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

The equation can be represented using any one of the following masks

0	1	0	1	1	1
1	−4	1	1	−8	1
0	1	0	1	1	1
0	−1	0	−1	−1	−1
−1	4	−1	−1	8	−1
0	−1	0	−1	−1	−1

Laplacian highlights gray-level discontinuities in an image and deemphasize the regions of slow varying gray levels. This makes the background a black image. The background texture can be recovered by adding the original and Laplacian images.

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

$$\begin{aligned} g(x, y) &= f(x, y) - [f(x+1, y) + f(x-1, y) \\ &\quad + f(x, y+1) + f(x, y-1)] + 4f(x, y) \\ &= 5f(x, y) - [f(x+1, y) + f(x-1, y) \\ &\quad + f(x, y+1) + f(x, y-1)]. \end{aligned}$$

For example:

.

The strength of the response of a derivative operator is propositional to the degree of discontinuity of the image at that point at which the operator is applied. Thus image differentiation enhances eddies and other discontinuities and deemphasizes areas with slowly varying gray levels.

The derivative of a digital function is defined in terms of differences. Any first derivative definition

(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
(3) Must be nonzero along ramps.

Any second derivative definition

(1) Must be zero in flat areas
(2) Must be nonzero at the onset and end of a gray level step or ramp
(3) Must be zero along ramps of constant slope .

It is common practice to approximate the magnitude of the gradient by using also lute values instead or squares and square roots:

<u>Roberts Goss gradient operators</u>

For digitally implementing the gradient operators
Let center point, 5z denote f(x,y), Z1 denotes f(x-1,y) and so on

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

−1	0
0	1

0	−1
1	0

But it different implement even sized mask. So the smallest filter mask is size 3x3 mask is

−1	−2	−1
0	0	0
1	2	1

−1	0	1
−2	0	2
−1	0	1

The difference between third and first row a 3x3 mask approximates the derivate in the x-direction and difference between the third and first column approximates the derivative in y-direction. These masks are called sobel operators.

2.1.7 Unsharp Masking and High Boost Filtering

Unsharp masking means subtracting a blurred version of an image form the image itself. Where f(x,y) denotes the sharpened image obtained by unsharp masking and f(x,y) is a blurred version of (x,y)

$$f_s(x, y) = f(x, y) - \overline{f}(x, y)$$

A slight further generalization of unsharp masking is called high boost filtering. A high boost filtered image is defined at any point (x,y) as

$$f_{hb}(x, y) = Af(x, y) - \overline{f}(x, y)$$

UNIT-3

IMAGE RESTORATION

3.1 IMAGE RESTORATION

Restoration improves image in some predefined sense. It is an objective process. Restoration attempts to reconstruct an image that has been degraded by using a priori knowledge of the degradation phenomenon. These techniques are oriented toward modeling the degradation and then applying the inverse process in order to recover the original image.

Image Restoration refers to a class of methods that aim to remove or reduce the degradations that have occurred while the digital image was being obtained.
All natural images when displayed have gone through some sort of degradation:

a) During display mode
b) Acquisition mode, or
c) Processing mode

The degradations may be due to

a) Sensor noise
b) Blur due to camera mis focus
c) Relative object-camera motion
d) Random atmospheric turbulence
e) Others

3.1.1 A Model of Image Restoration Process

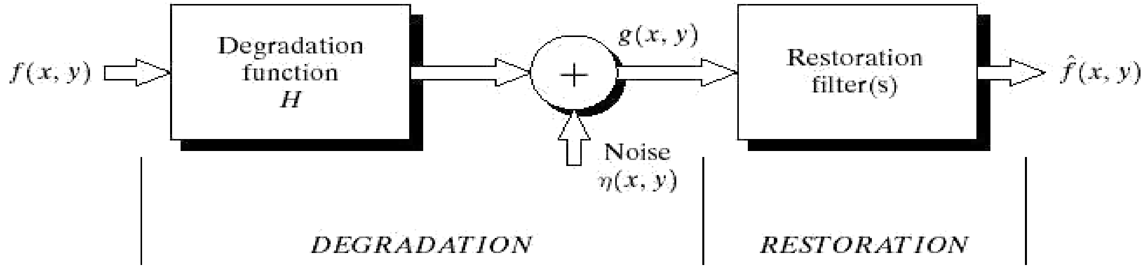
Degradation process operates on a degradation function that operates on an input image with an additive noise term.
Input image is represented by using the notation f(x,y), noise term can be represented as **η**(x,y).These two terms when combined gives the result as g(x,y).
If we are given g(x,y), some knowledge about the degradation function H or J and some knowledge about the additive noise teem **η**(x,y), the objective of restoration is to obtain an estimate *f'(x,y)* of the original image. We want the estimate to be as close as possible to the original image. The more we know about h and **η** , the closer f(x,y) will be to f'(x,y).
If it is a linear position invariant process, then degraded image is given in the spatial domain by

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

h(x,y) is spatial representation of degradation function and symbol * represents convolution. In frequency domain we may write this equation as

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

The terms in the capital letters are the Fourier Transform of the corresponding terms in the spatial domain.



The image restoration process can be achieved by inversing the image degradation process, i.e.,

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

where $1/\hat{H}(u,v)$ is the inverse filter, and $\hat{F}(u,v)$ is the recovered image. Although the concept is relatively simple, the actual implementation is difficult to achieve, as one requires prior knowledge or identifications of the unknown degradation function $H(u,v)$ and the unknown noise source $n(x,y)$.

In the following sections, common noise models and method of estimating the degradation function are presented.

3.1.2 **<u>Noise Models</u>**

The principal source of noise in digital images arises during image acquisition and /or transmission. The performance of imaging sensors is affected by a variety of factors, such as environmental conditions during image acquisition and by the quality of the sensing elements themselves. Images are corrupted during transmission principally due to interference in the channels used for transmission. Since main sources of noise presented in digital images are resulted from atmospheric disturbance and image sensor circuitry, following assumptions can be made:

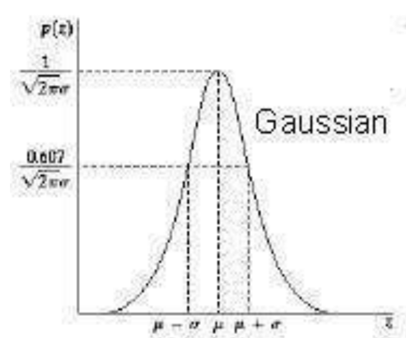
- The noise model is spatial invariant, i.e., independent of spatial location.
- The noise model is uncorrelated with the object function.

I. <u>Gaussian Noise</u>

These noise models are used frequently in practices because of its tractability in both spatial and frequency domain.
The PDF of Gaussian random variable, z is given by

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

z= gray level
μ= mean of average value of
z **σ**= standard deviation



II. Rayleigh Noise

Unlike Gaussian distribution, the Rayleigh distribution is no symmetric. It is given by the formula.

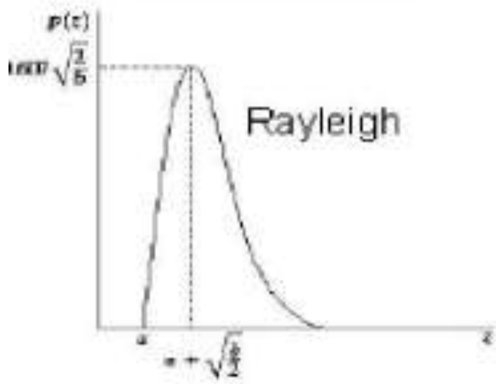
$$p(z) = \frac{2}{b}(z-a)e^{-\frac{(z-a)^2}{b}}, \quad \text{for } z \geq a$$

The mean and variance of this density

Mean/variance

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

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



It is displaced from the origin and skewed towards the right.

III. Erlang (gamma) Noise

The PDF of Erlang noise is given by

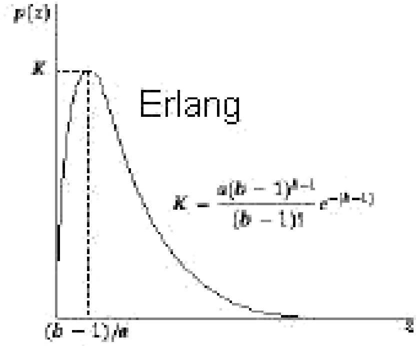
$$p(z) = \frac{a^b z^{b-1}}{(b-1)!} e^{-az}, \quad \text{for } z \geq 0$$

The mean and variance of this noise is

Mean/variance

$$\mu = \frac{b}{a}$$

$$\sigma^2 = \frac{b}{a^2}$$



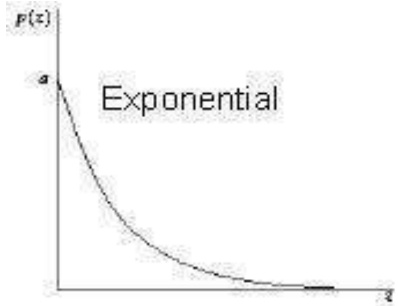
Its shape is similar to Rayleigh disruption.
This equation is referred to as gamma density it is correct only when the denominator is the gamma function.

IV. <u>Exponential Noise</u>

Exponential distribution has an exponential shape.
The PDF of exponential noise is given as

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

Where a>0
It is a special case of Erlang with b=1



V. <u>Uniform Noise</u>

The PDF of uniform noise is given by

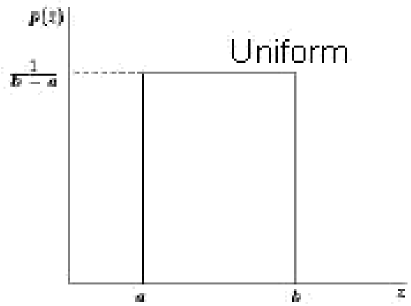
$$p(z) = \begin{cases} \frac{1}{(b-a)} & a \leq z \leq b \\ 0 & \text{otherwise} \end{cases}$$

The mean of this density function is given by

Mean/variance

$$\mu = \frac{a+b}{2}$$

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

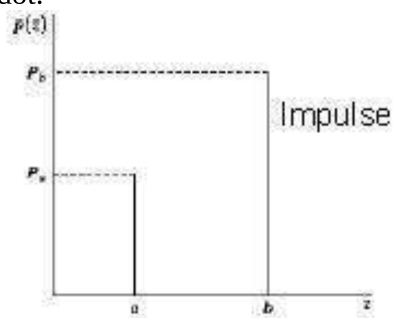


VI. Impulse (Salt and Pepper)Noise

In this case, the noise is signal dependent, and is multiplied to the image.
The PDF of bipolar (impulse) noise is given by

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

If b>a, gray level b will appear as a light dot in image.
Level a will appear like a dark dot.



3.1.3 Restoration In the Presence of Noise Only-Spatial Filtering

When the only degradation present in an image is noise, i.e.

$$g(x,y)= f(x,y)+ \eta(x,y) \text{ or}$$
$$G(u,v)= F(u,v)+ N(u,v)$$

The noise terms are unknown so subtracting them from g(x,y) or G(u,v) is not a realistic approach. In the case of periodic noise it is possible to estimate N(u,v) from the spectrum G(u,v). So N(u,v) can be subtracted from G(u,v) to obtain an estimate of original image. Spatial filtering can be done when only additive noise is present.
The following techniques can be used to reduce the noise effect:

3.1.3.1 Mean Filter

3.1.3.1.1 Arithmetic Mean Filter

It is the simplest mean filter. Let S_{xy} represents the set of coordinates in the sub image of size m*n centered at point (x,y). The arithmetic mean filter computes the average value of the corrupted image g(x,y) in the area defined by S_{xy}. The value of the restored image f at any point (x,y) is the arithmetic mean computed using the pixels in the region defined by S_{xy}.

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

This operation can be using a convolution mask in which all coefficients have value 1/mn
A mean filter smoothes local variations in image Noise is reduced as a result of blurring. For every pixel in the image, the pixel value is replaced by the mean value of its neighboring pixels ([illegible]) with a weight [illegible]. This will resulted in a smoothing effect in the image.

3.1.3.1.2 Geometric mean filter

An image restored using a geometric mean filter is given by the expression

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

Here, each restored pixel is given by the product of the pixel in the subimage window, raised to the power 1/mn. A geometric mean filters but it to loose image details in the process.

3.1.3.1.3 Harmonic mean filter

The harmonic mean filtering operation is given by the expression

$$\hat{f}(x,y) = \sum_{(s,t)\in S_{xy}} g(s,t)^{Q+1} \Big/ \sum_{(s,t)\in S_{xy}} g(s,t)^{Q}$$

The harmonic mean filter works well for salt noise but fails for pepper noise. It does well with Gaussian noise also.

3.1.3.1.4 Order statistics filter

Order statistics filters are spatial filters whose response is based on ordering the pixel contained in the image area encompassed by the filter.
The response of the filter at any point is determined by the ranking result.

Median filter

It is the best order statistic filter; it replaces the value of a pixel by the median of gray levels in the Neighborhood of the pixel.

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

The original of the pixel is included in the computation of the median of the filter are quite possible because for certain types of random noise, the provide excellent noise reduction capabilities with considerably less blurring then smoothing filters of similar size. These are effective for bipolar and unipolor impulse noise.

Max and Min Filters

Using the l00th percentile of ranked set of numbers is called the max filter and is given by the equation

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

It is used for finding the brightest point in an image. Pepper noise in the image has very low values, it is reduced by max filter using the max selection process in the sublimated area sky.

The 0th percentile filter is min filter

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

This filter is useful for flinging the darkest point in image. Also, it reduces salt noise of the min operation.

a. Midpoint Filter

The midpoint filter simply computes the midpoint between the maximum and minimum values in the area encompassed by the filter

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

It comeliness the order statistics and averaging .This filter works best for randomly distributed noise like Gaussian or uniform noise.

3.1.4 **Periodic Noise By Frequency Domain Filtering**

These types of filters are used for this purpose-

3.1.4.1 Band Reject Filters

It removes a band of frequencies about the origin of the Fourier transformer.

3.1.4.1.1 Ideal Band reject Filter

An ideal band reject filter is given by the expression

$$H(u,v) = \begin{cases} 1 & \text{if} \quad D(u,v) < D_0 - W/2 \\ 0 & \text{if} \quad D_0 - W/2 \le D(u,v) \le D_0 + W/2 \\ 1 & \text{if} \quad D(u,v) > D_0 + W/2 \end{cases}$$

D(u,v)- the distance from the origin of the centered frequency rectangle.
W- the width of the band
Do- the radial center of the frequency rectangle.

3.1.4.1.2 Butterworth Band reject Filter

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

3.1.4.1.3 Gaussian Band reject Filter

$$H(u,v) = 1 - \exp\left[-\frac{1}{2} \left(\frac{D^2(u,v) - D_0^2}{D(u,v)W} \right)^2 \right]$$

These filters are mostly used when the location of noise component in the frequency domain is known. Sinusoidal noise can be easily removed by using these kinds of filters because it shows two impulses that are mirror images of each other about the origin. Of the frequency transform.

3.1.4.2 Band Pass Filters

The function of a band pass filter is opposite to that of a band reject filter It allows a specific frequency band of the image to be passed and blocks the rest of frequencies.
The transfer function of a band pass filter can be obtained from a corresponding band reject filter with transfer function $H_{br}(u,v)$ by using the equation-

$$H_{BP}(u,v) = 1 - H_{BR}(u,v)$$

These filters cannot be applied directly on an image because it may remove too much details of an image but these are effective in isolating the effect of an image of selected frequency bands.

3.1.5 **Notch Filters**

This type of filters rejects frequencies I predefined in neighborhood above a centre frequency These filters are symmetric about origin in the Fourier transform the transfer function of ideal notch reject filter of radius do with centre at (…) and by symmetry at (……) is

$$H(u,v) = \begin{cases} 0 & \textbf{if} \quad D_1(u,v) \le D_0 \textbf{ or } D_2(u,v) \le D_0 \\ 1 & \textbf{otherwise} \end{cases}$$

Where

$$D_1(u,v) = \sqrt{(u - M/2 - u_0)^2 + (v - N/2 - v_0)^2}$$
$$D_2(u,v) = \sqrt{(u - M/2 + u_0)^2 + (v - N/2 + v_0)^2}$$

Butterworth notch reject filter of order n is given by

$$H(u,v) = 1 - \exp\left[-\frac{1}{2}\left(\frac{D_1(u,v)D_2(u,v)}{D_0^2}\right)\right]$$

A Gaussian notch reject filter has the fauna

$$H(u,v) = 1 \Bigg/ \left[1 + \left(\frac{D_0^2}{D_1(u,v)D_2(u,v)}\right)^n\right]$$

These filter become high pass rather than suppress. The frequencies contained in the notch areas. These filters will perform exactly the opposite function as the notch reject filter.
The transfer function of this filter may be given as

$$Hnp(u,v) = 1 - Hnr(u,v)$$

$H_{np}(u,v)$- transfer function of the pass filter
$H_{nr}(u,v)$- transfer function of a notch reject filter

3.1.6 **Minimum Mean Square Error (Wiener) Filtering**

This filter incorporates both degradation function and statistical behavior of noise into the restoration process.
The main concept behind this approach is that the images and noise are considered as random variables and the objective is to find an estimate f of the uncorrupted image f such that the mean sequence error between then is minimized.

$$\hat{f}(x) = \sum_{s=-\infty}^{\infty} h_W(x-s)g(s),$$

This error measure is given by

$$e^2 = E\{[f(x) - \hat{f}(x)]^2\} = \min$$

Where e( ) is the expected value of the argument
Assuming that the noise and the image are uncorrelated (means zero average value) one or other has zero mean values
The minimum error function of the above expression is given in the frequency …….. is given by the expression.

$$H_W(u,v) = \frac{H^*(u,v)S_{ff}(u,v)}{|H(u,v)|^2 S_{ff}(u,v) + S_{nn}(u,v)} = \frac{1}{H(u,v)} \left[\frac{|H(u,v)|^2}{|H(u,v)|^2 + S_{nn}(u,v)/S_{ff}(u,v)}\right]$$

Product of a complex quantity with its conjugate is equal to the magnitude of …… complex quantity squared. This result is known as wiener Filter The filter was named so because of the name of its inventor N Wiener. The term in the bracket is known as minimum mean square error filter or least square error filter.
H*(u,v)-degradation function .
H*(u,v)-complex conjugate of
H(u,v) H(u,v) H(u,v)
Sn(u,v)=IN(u,v)I^2- power spectrum of the noise
Sf(u,v)=IF(u,v)^2- power spectrum of the underrated image
H(u,v)=Fourier transformer of the degraded function
G(u,v)=Fourier transformer of the degraded image
The restored image in the spatial domain is given by the inverse Fourier transformed of the frequency domain estimate F(u,v).
Mean square error in statistical form can be approveiment by the function

$$H_W(u,v) = \frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + K}$$

3.1.7 **Inverse Filtering**

It is a process of restoring an image degraded by a degradation function H. This function can be obtained by any method.
The simplest approach to restoration is direct, inverse filtering.
Inverse filtering provides an estimate F(u,v) of the transform of the original image simply by during the transform of the degraded image G(u,v) by the degradation function.

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

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

It shows an interesting result that even if we know the depredation function we cannot recover the underrated image exactly because N(u,v) is not known .
If the degradation value has zero or very small values then the ratio N(u,v)/H(u,v) could easily dominate the estimate F(u,v).

UNIT-4
MORPHOLOGICAL IMAGE PROCESSING

4.1 MORPHOLOGICAL IMAGE PROCESSING

4.1.1 Introduction

The word morphology refers to the scientific branch that deals the forms and structures of animals/plants. Morphology in image processing is a tool for extracting image components that are useful in the representation and description of region shape, such as boundaries and skeletons. Furthermore, the morphological operations can be used for filtering, thinning and pruning.
This is middle level of image processing technique in which the input is image but the output is attributes extracted meaning from an image. The language of the Morphology comes from the set theory, where image objects can be represented by sets. For example an image object containing black pixels can be considered a set of black pixels in 2D space of Z^2, where each elements of the set is a tuple (2-D vector) whose coordinates are the (x,y) coordinates are the coordinates of white pixel in an image.
Gary scale images can be represented as sets whose components are in Z^3 two components of each elements of the set refers to the coordinates of a pixel and the third correspond to the discrete intensity value.

4.1.2 Basics Of Set Theory

Let A be set in Z^2 and a= (a1, a2) then

a is an element of A : $a \in A$

If a is not an element of a then $a \notin A$

If every element of set A is also an element of set B, the A said be a subset of B Written as

$$A \subseteq B$$

The union of A and B is the collection of all elements that are in one both set. It is represented as

$$C = A \bigcup B$$

The intersection of the sets A and B is the set element belonging to both A and B is represented as

$$D = A \bigcap B$$

If these are no common elements in A and B, then the sets are called disjoint sets represented as

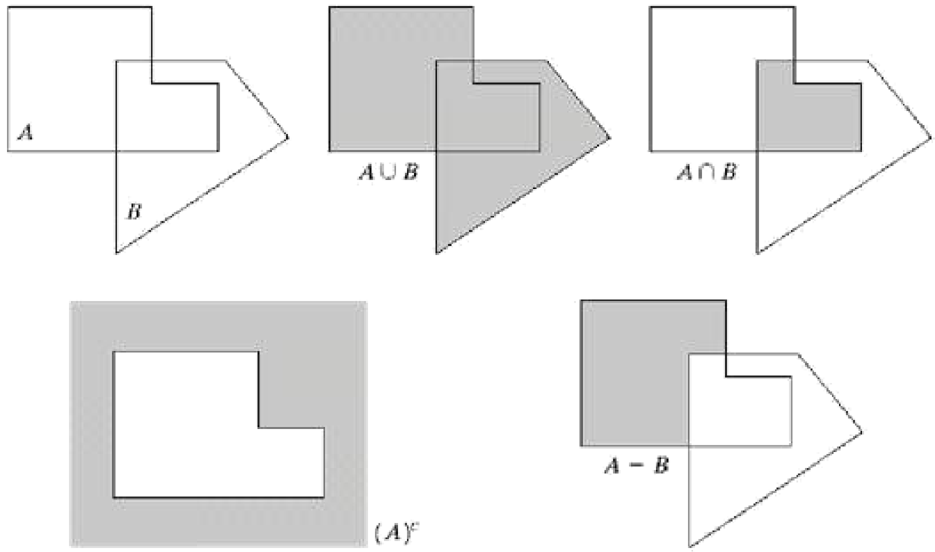
$$A \bigcap B = \varnothing$$

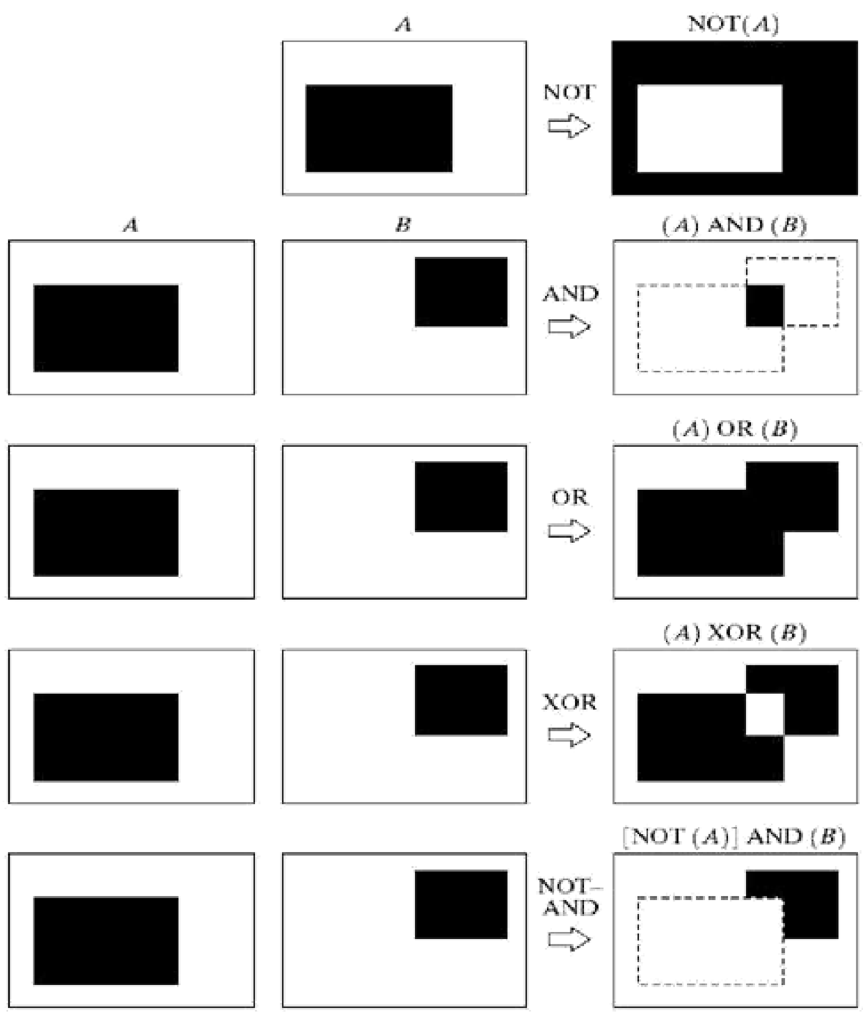
Ø is the name of the set with no members
The complements of a sets a is the set of elements in the image not contained A

$$A^c = \{\omega | \omega \notin A\}$$

The difference of two sets A and B is denoted by

$$A - B = \{\omega | \omega \in A, \omega \notin B\} = A \bigcap B^c$$


A
B
A ∪ B
A ∩ B
(A)^c
A − B


A
NOT(A)
NOT
A
B
(A) AND (B)
AND
(A) OR (B)
OR
(A) XOR (B)
XOR
[NOT (A)] AND (B)
NOT–AND

The reflection of two set B, denoted by B is defined as

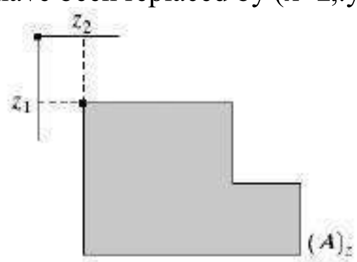
$$\hat{B} = \{\omega \mid \omega = -b, \ for\, b \in B\}$$

If B is the set of pixel representing an object in an image. Then A is simply the set of points in A whose (x,y) coordinates have been replaced by (z)

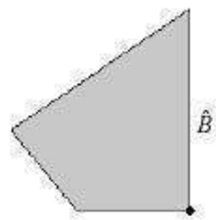
The translation of a set B by a point z= (z1,z2) denoted (B)z is defined as

$$(A)_z = \{\omega \mid \omega = a + z, \ for\, a \in A\}$$

If B is the set of set of pixel representing as object in an image Then (B)z is the set of points in B whose (x,y) coordinates have been replaced by (x+z,.y+z…)



Translation of A by z.



Reflection of B

4.1.3 **<u>Erosion & Dilation</u>**

Dilation and erosion are the two fundamental operations used in morphological image processing. Almost all morphological algorithms depend on these two operations:

4.1.3.1 <u>Dilation</u>

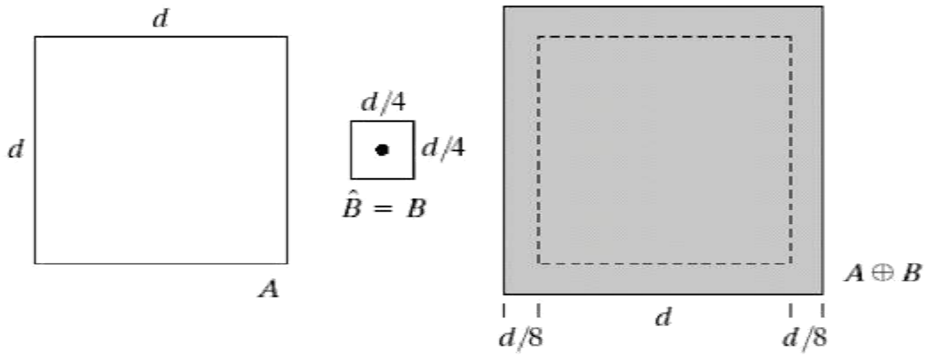
With A and B as set inZ^2 the dilation of A by B is defined as

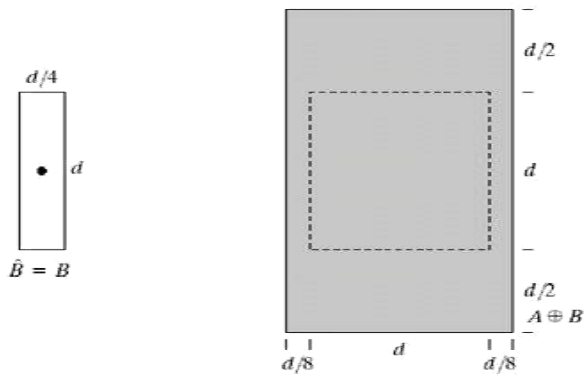
$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \varnothing\}$$

Obtaining the reflection of B about the origin and shifting this reflection by z. The dilation is then the set of all displacements Z such that B and A overlap atheist by elements.
The equation may be rewritten as

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

The set B is referred to as the structuring element in the dilation. This structuring element may be through of as a convolution mask.
Because the basic operation of flipping B about its origin and then successively displacing it so that it slides over the image A is analogue to the convolution process.





The structuring element and its refection are equal because it is symmetric with respect to the origin. The dashed line shows the boundary constitute beyond which any further displacement by z would cause the intersection of B and A to be empty.
Therefore all the points inside this boundary constitute the dilation of A to B. dilation has an advantage over low pass filtering that morphological method results directly in a binary image and convert it into a gray scale image which would require a pass with a thresh holding function to convert it back to binary form.

4.1.3.2 Erosion

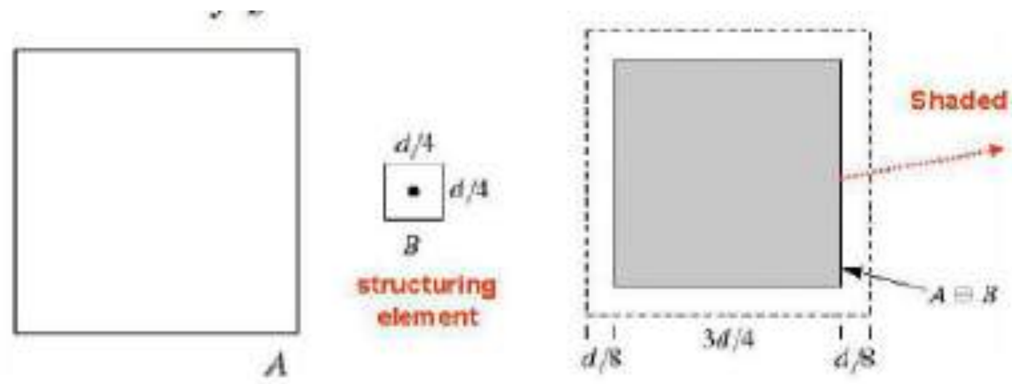
Erosion shrinks an image object. The basic effect of erosion is to erode away the boundaries of for ground pixel thus area of foreground pixel shrinks to size and holes within those areas become larger.
Mathematically, erosion of sets A by sets B is a set of all points x such that B translated by x is still contained in A.

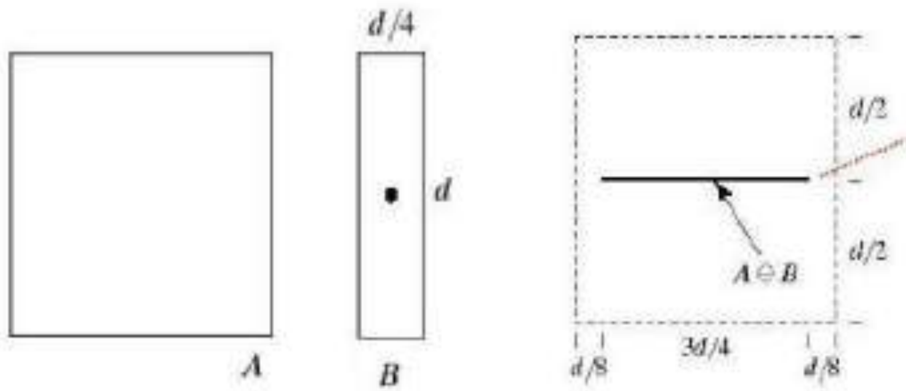
$$A \ominus B = \{ z \mid (B)_z \subseteq A \}$$

Characteristics

- ⇨ It generally decreases the size of objects and removes small anomalies by subtracting objects with a radius smaller than the structuring element.
- ⇨ With gray scale images erosion reduces the brightness of bright objects on a dark back ground by taking the neighborhood minimum when passing the structuring element over the image.
- ⇨ With erosion binary images it completely removes objects smaller than the structuring element and removes perimeter pixels from larger image objects. For sets A and B in Z^2 the erosion of A by B denoted AøB is defined as

Erosion of A by B is the set of all points z such that B translated by z is contained in A
The boundary of the shaded region shows the limit beyond which further displacement of the origin of B would cause this set to cease being completed contained in A.





The boundary of shaded region shows the limit beyond which further displacement of the origin of B would cause this set to cease being completely contained in A.
Dilation and erosion are duals of each other with respect to set complementation and reflection,

4.1.4 Structuring Elements

These are also called kernel. It consists of a pattern specified as the coordinates of a number of discrete points suitable to some origin. All the techniques probe an image with this small shape or temples. It generally consists of a matrix of o’s and 1’s. Typically it is much smaller than the image being processed. The center pixel of the structuring elements is called the origin and it identifies the pixel of the interest of the pixel being processed. The pixels in the structuring elements containing 1’s define the neighborhood of the structuring element.

It differs from the input image coordinates set in that it is normally much smaller. And its coordinate’s origin is often not in a corner so that some coordinate element will have negative value.

The structuring element is positioned at all positions in the image and it is the compared with the corresponding neighborhood of pixels. Two main characteristics that are directly related to structuring elements.

(i) Shape

The element may be ball or line: convex a ring. By choosing particular structuring elements. One sets a way of differentially some objects from others according to their shape or spatial orientation.

(ii) Size

The structuring element can be a 3x3 or a 21x21 square.

4.1.5 Opening & Closing

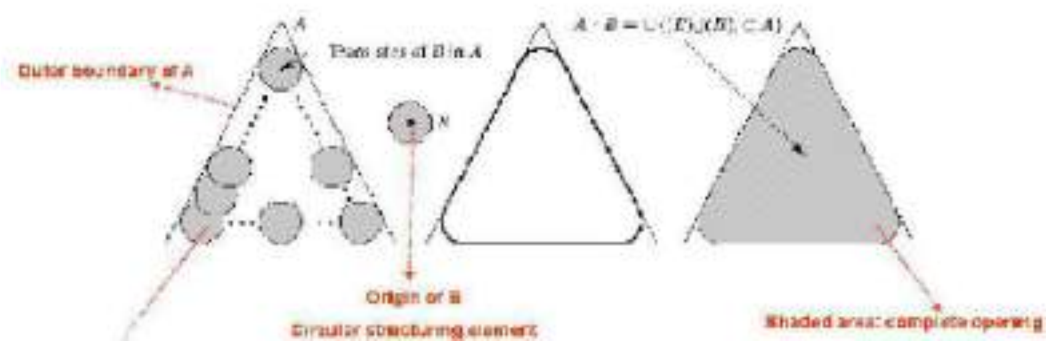
4.1.5.1 Opening

The process of erosion followed by dilation is called opening. It has the effect of eliminating small and thin objects, breaking the objects at thin points and smoothing the boundaries/contours of the objects.
Given set A and the structuring element B. opening of a set A by structuring element B is defined as

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

The opening of A by the structuring element B is obtained by taking the union of all translates of B that fit into A.
The opening operation can also be expressed by the following formula:

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



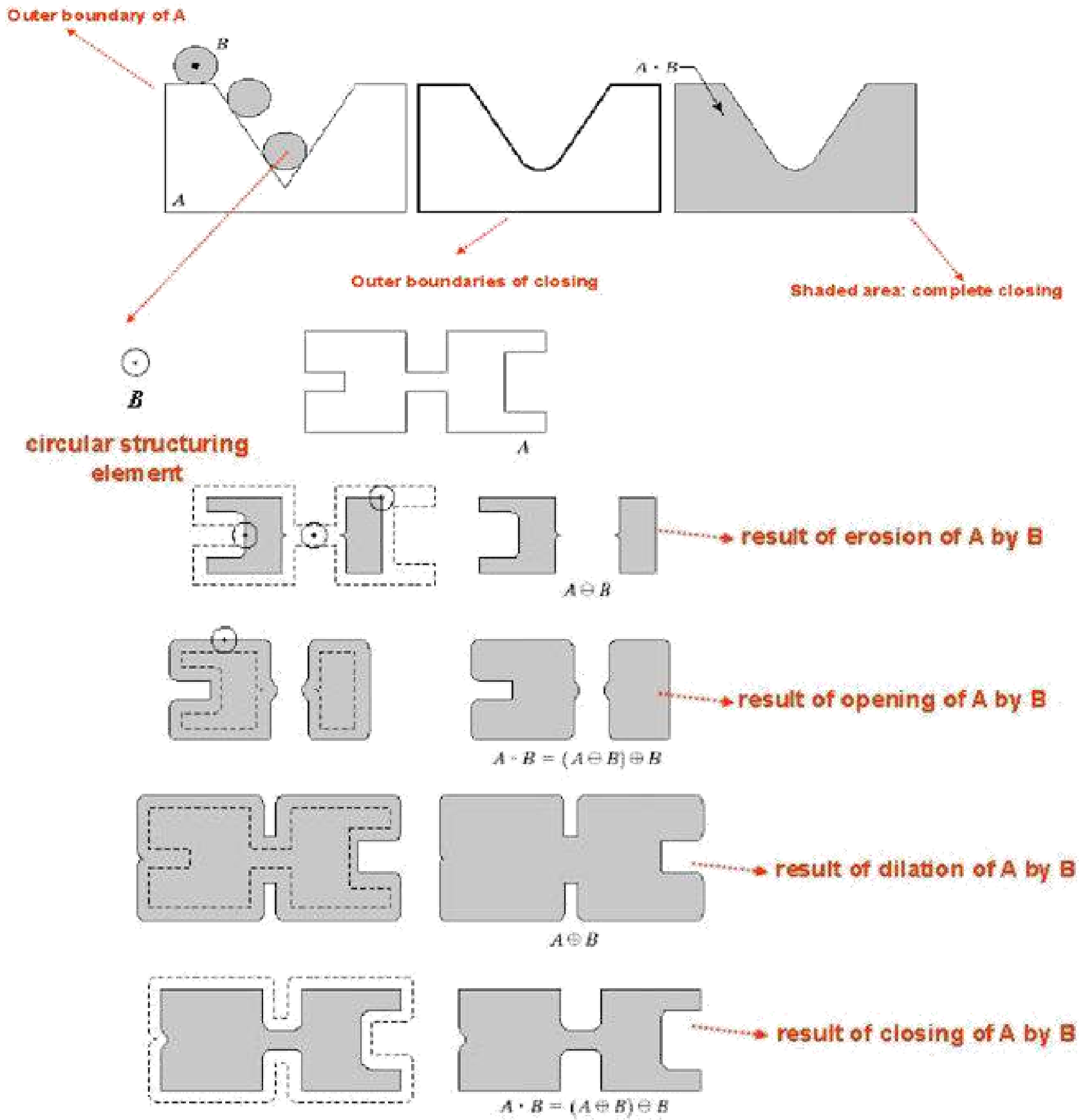
4.1.5.2 Closing

The process of dilation followed by erosion is called closing. It has the effect of filling small and thin holes, connecting nearby objects and smoothing the boundaries/contours of the objects. Given set A and the structuring element B. Closing of A by structuring element B is defined by:

$$A \bullet B = (A \oplus B) \ominus B$$

The closing has a similar geometric interpretation except that we roll B on the outside of the boundary.
The opening operation can also be expressed by the following formula:

$$A \bullet B = \bigcup \{ (B_z) \mid (B_z) \cap A \neq \emptyset \}$$

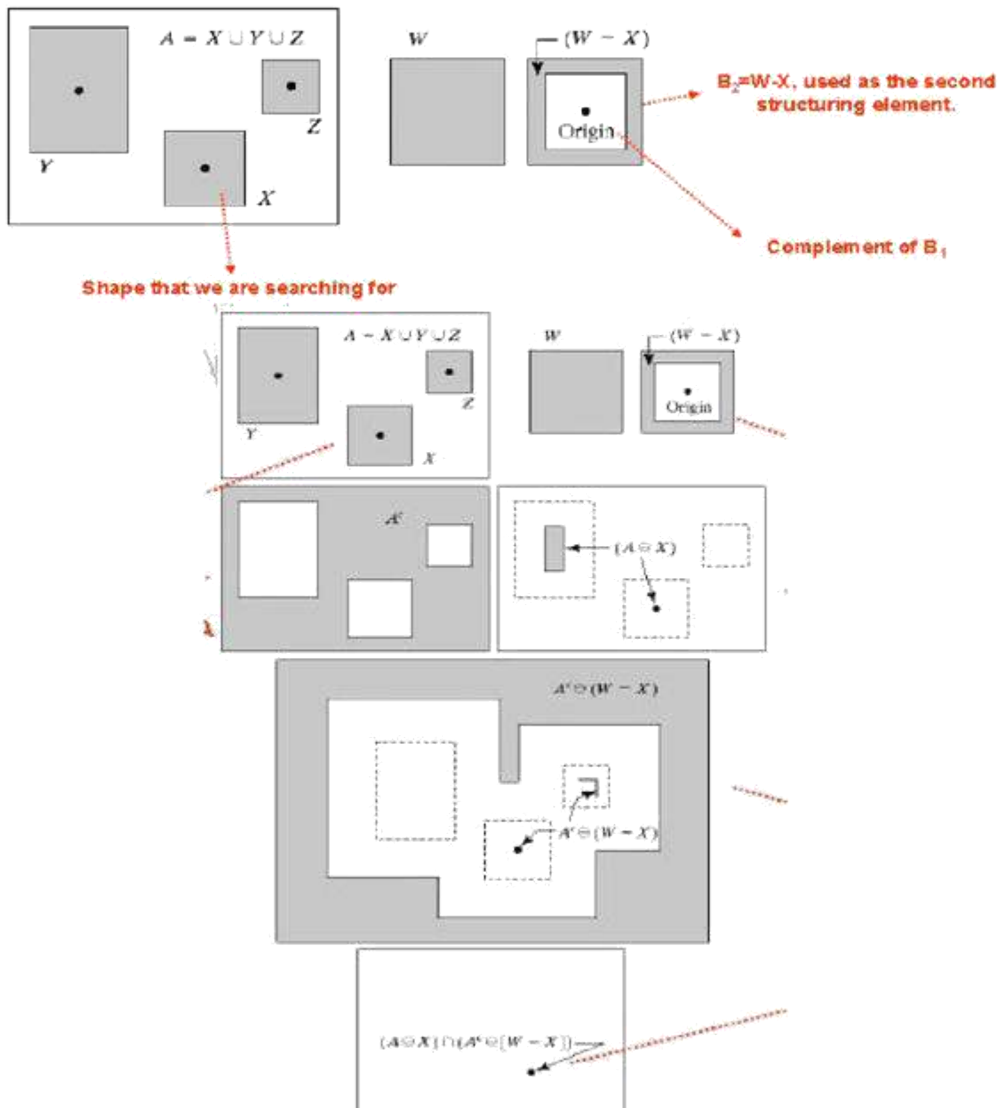


4.1.6 Hit or Miss Transformation (Template Matching)

Hit-or-miss transform can be used for shape detection/ Template matching.
Given the shape as the structuring element B1 the Hit-or-miss transform is defined by:

$$A \circledast B = (A \ominus B_1) \cap (A^c \ominus B_2)$$

Where B2 =W-X and B1=X. W is the window enclosing B1. Windowing is used to isolate the structuring element/object.

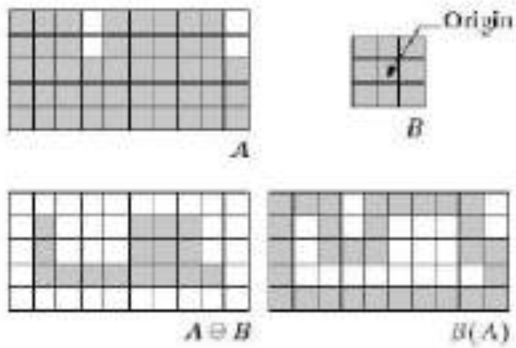

A = X ∪ Y ∪ Z
Y
Z
X
W
(W − X)
Origin
B2=W-X, used as the second structuring element.
Complement of B1
Shape that we are searching for
A^c
(A ⊖ X)
A^c ⊖ (W − X)
(A⊖X) ∩ (A^c⊖(W − X))

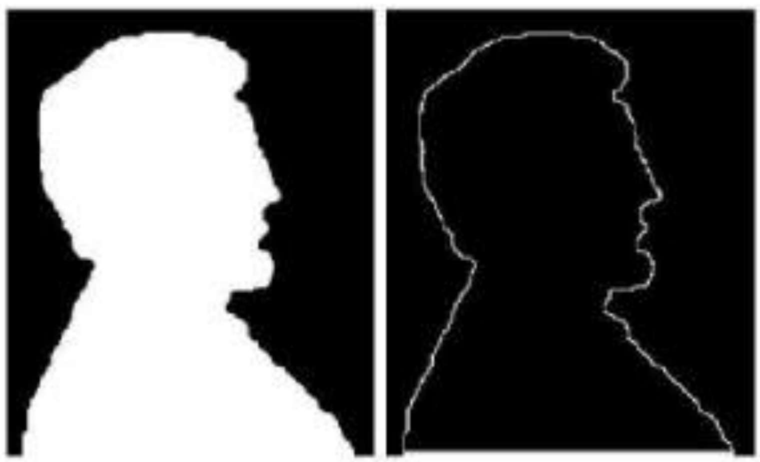
4.1.7 **Morphological Algorithms**

4.1.7.1 Boundary Extraction

The boundaries/edges of a region/shape can be extracted by first applying erosion on A by B and subtracting the eroded A from A.

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





4.1.7.2 Region Filling

Region filling can be performed by using the following definition. Given a symmetric structuring element B, one of the non-boundary pixels (Xk) is consecutively diluted and its intersection with the complement of A is taken as follows:

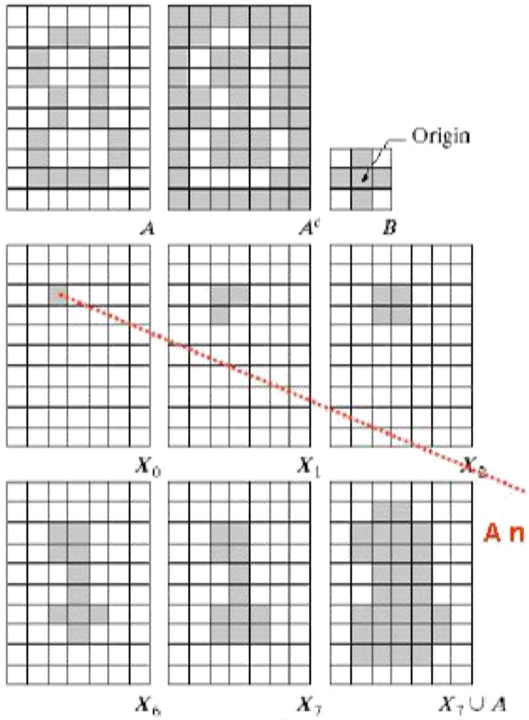
$$X_k = (X_{k-1} \oplus B) \cap A^c \qquad k = 1,2,3,\ldots$$

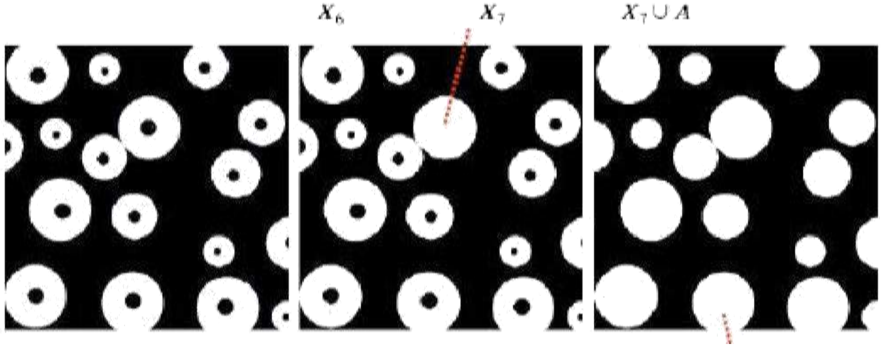
$$\text{terminates when } X_k = X_{k-1}$$

$$X_0 = 1 \ (\text{inner pixel})$$

Following consecutive dilations and their intersection with the complement of A, finally resulting set is the filled inner boundary region and its union with A gives the filled region F(A).

$$F(A) = X_k \cup A$$





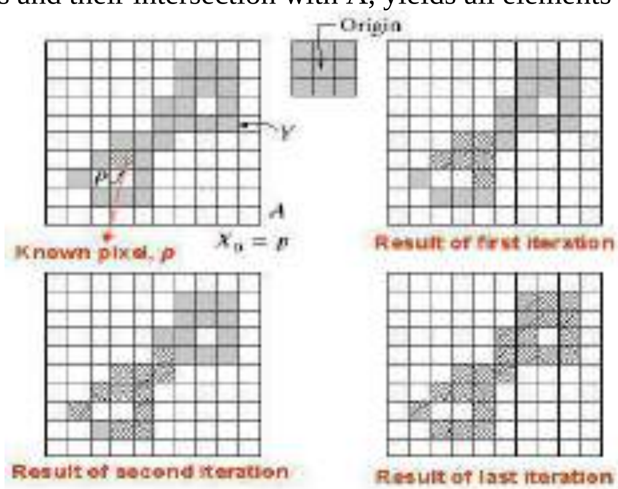
4.1.7.3 Connected Component Extraction

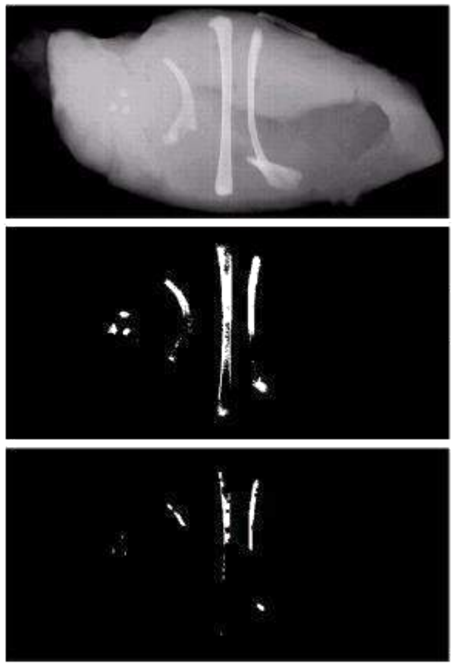
The following iterative expression can be used to determine all the pixels in component Y which is in A.

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

X0=1 corresponds to one of the pixels on the component Y. Note that one of the pixel locations on the component must be known.

Consecutive dilations and their intersection with A, yields all elements of component Y.





4.1.7.4 <u>Thinning & Thickening</u>

4.1.7.4.1 <u>Thinning</u>

Thinning of A by the structuring element B is defined by:

$$A \otimes B = A - (A \circledast B)$$

hit-or-miss transform/template matching

Note that we are only interested in pattern matching of B in A, so no background operation is required of the hit-miss-transform.

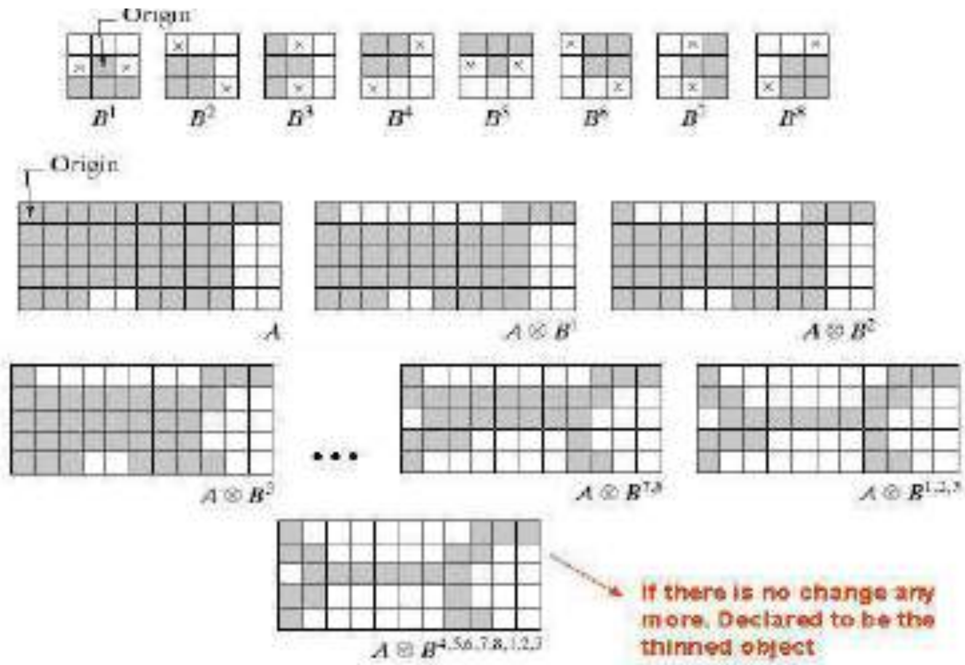
$$\{B\} = \{B^1, B^2, B^3, \ldots, B^n\}$$

The structuring element B consists of a sequence of structuring elements, where Bi is the rotated version of Bi-1. Each structuring elements helps thinning in one direction. If there are 4 structuring elements thinning is performed from 4 directions separated by 90o. If 8 structuring elements are used the thinning is performed in 8 directions separated by 45o.

The process is to thin A by one pass with B1, then the result with one pass of B2, and continue until A is thinned with one pass of Bn.

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

The following sets of structuring elements are used for thinning operation**.**



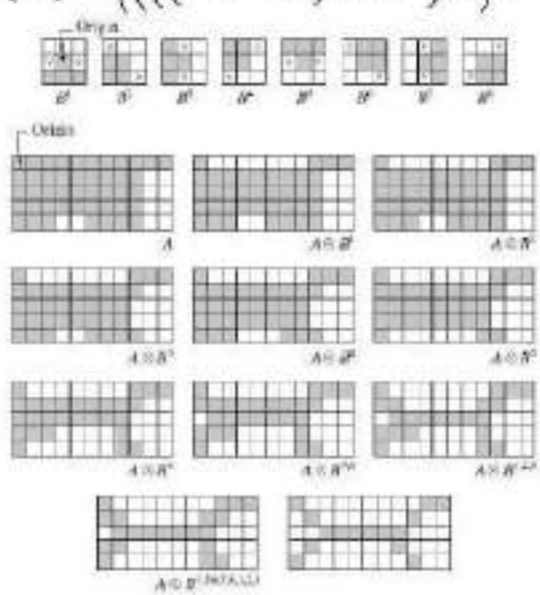
4.1.7.4.2 <u>Thickening</u> Thinning and

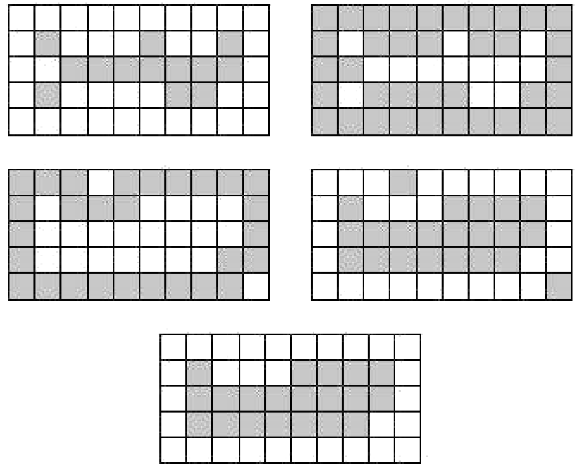
is defined by the expression

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

As in thinning thickening can be defined as a sequential operation;

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





4.1.7.5 <u>Skeletons</u>

$$S(A) = \bigcup_{k=0}^{k} S_k(A)$$

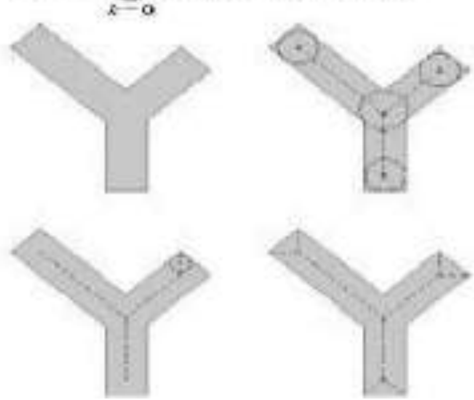
$$S_k(A) = (A \Theta kB) - (A \Theta kB) \circ B$$

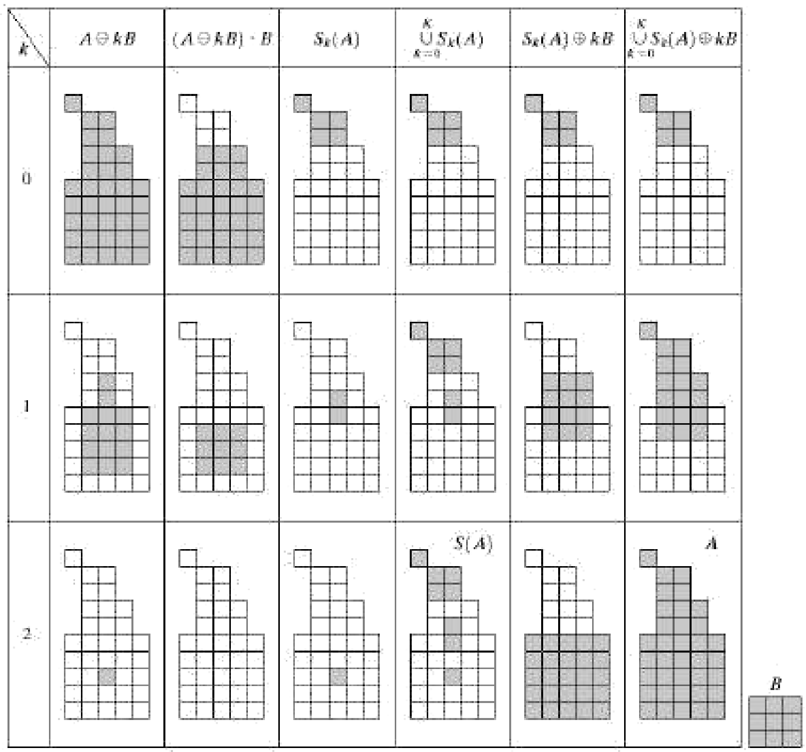
$$(A \Theta kB) = (\ldots\ldots(A \Theta B) \Theta B) \Theta \ldots\ldots) \Theta B$$

$$K = \max\left\{k \,\middle|\, (A \Theta kB) \neq \varnothing\right\}$$

A can be reconstructed from these subsets by using the equation

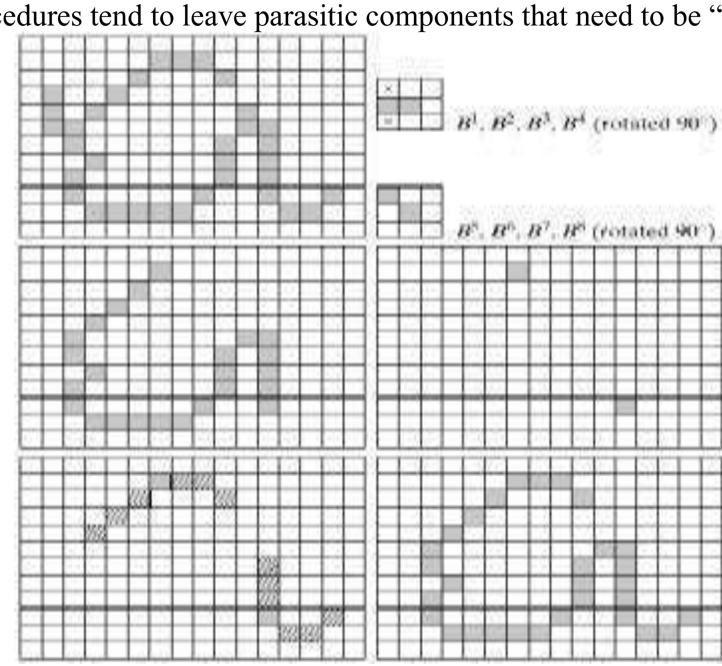
$$A = \bigcup_{k=0}^{k} (S_k(A) \oplus kB)$$





4.1.7.6 Pruning

Pruning methods are an essential complement to thinning and skeleton zing algorithms because these procedures tend to leave parasitic components that need to be "cleaned up".



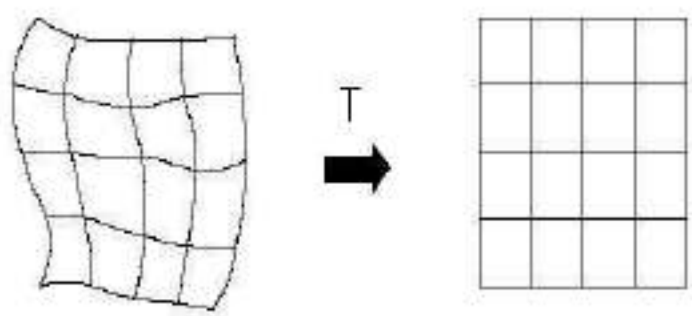
UNIT-5

IMAGE REGISTRATION & SEGMENTATION

5.1 **REGISTRATION**

5.1.1 **Geometric Transformation**

Geometric transforms permit the elimination of geometric distortion that occurs when an image is captured. An example is an attempt to match remotely sensed images of the same area taken after one year, when the more recent image was probably not taken from precisely the same position. To inspect changes over the year, it is necessary first to execute a geometric transformation, and then subtract one image from the other.



A geometric transform is a vector function T that maps the pixel (x,y) to a new position (x',y').

$$x' = T_x(x, y), \qquad y' = T_y(x, y)$$

The transformation equations are either known in advance or can be determined from known original and transformed images. Several pixels in both images with known correspondence are used to derive the unknown transformation.

5.1.1.1 Plane to Plane Transformation

General case of finding the co-ordinates of a point in the output image after a geometric transform. It is usually approximated by a polynomial equation

$$x' = \sum_{r=0}^{m} \sum_{k=0}^{m-r} a_{rk}\, x^r\, y^k \qquad y' = \sum_{r=0}^{m} \sum_{k=0}^{m-r} b_{rk}\, x^r\, y^k$$

This transform is linear with respect to the coefficients a_{rk}, b_{rk}. If pairs of corresponding points (x,y), (x',y') in both images are known, it is possible to determine a_{rk}, b_{rk} by solving a set of linear equations. More points than coefficients are usually used to get robustness. If the geometric transform does not change rapidly depending on position in the image, low order approximating polynomials, m=2 or m=3, are used, needing at least 6 or 10 pairs of corresponding points. The corresponding points should be distributed in the image in a way that can express the geometric transformation - usually they are spread uniformly. The higher the degree of the approximating polynomial, the more sensitive to the distribution of the pairs of corresponding points the geometric transform.

In practice, the geometric transform is often approximated by the bilinear transformation. 4 pairs of corresponding points are sufficient to find transformation coefficients

$$x' = a_0 + a_1 x + a_2 y + a_3 xy$$
$$y' = b_0 + b_1 x + b_2 y + b_3 xy$$

Even simpler is the affine transformation for which three pairs of corresponding points are sufficient to find the coefficients

$$x' = a_0 + a_1 x + a_2 y$$
$$y' = b_0 + b_1 x + b_2 y$$

The affine transformation includes typical geometric transformations such as rotation, translation, scaling and skewing.

Rotation - by the angle phi about the origin

$$\begin{aligned} x' &= x \cos\phi + y \sin\phi \\ y' &= -x \sin\phi + y \cos\phi \end{aligned}$$

Change of scale - a in the x axis and b in the y axis

$$\begin{aligned} x' &= ax \\ y' &= bx \end{aligned}$$

Skewing by the angle phi

$$\begin{aligned} x' &= x + y \tan\phi \\ y' &= y \end{aligned}$$

5.2 <u>**Segmentation**</u>

5.2.1 <u>**Introduction**</u>

If an image has been preprocessed appropriately to remove noise and artifacts, segmentation is often the key step in interpreting the image. Image segmentation is a process in which regions or features sharing similar characteristics are identified and grouped together. Image segmentation may use statistical classification, thresholding, edge detection, region detection, or any combination of these techniques. The output of the segmentation step is usually a set of classified elements, Most segmentation techniques are either region-based or edge based.

- Region-based techniques rely on common patterns in intensity values within a cluster of neighboring pixels. The cluster is referred to as the region, and the goal of the segmentation algorithm is to group regions according to their anatomical or functional roles.
- Edge-based techniques rely on discontinuities in image values between distinct regions, and the goal of the segmentation algorithm is to accurately demarcate the boundary separating these regions.

Segmentation is a process of extracting and representing information from an image is to group pixels together into regions of similarity.

Region-based segmentation methods attempt to partition or group regions according to common image properties. These image properties consist of :

i. Intensity values from original images, or computed values based on an image operator
ii. Textures or patterns that are unique to each type of region
iii. Spectral profiles that provide multidimensional image data

Elaborate systems may use a combination of these properties to segment images, while simpler systems may be restricted to a minimal set on properties depending of the type of data available.

<u>Categories of Image Segmentation Methods</u>

- Clustering Methods
- Histogram-Based Methods
- Edge Detection Methods
- Region Growing Methods
- Level Set Methods
- Graph portioning methods
- Watershed Transformation
- Neural Networks Segmentation

- Model based Segmentation/knowledge-based segmentation - involve active shape and appearance models, active contours and deformable templates.
- Semi-automatic Segmentation - Techniques like Livewire or Intelligent Scissors are used in this kind of segmentation.

5.2.2 **<u>Pixel-Based Approach</u>**

Gray level thresholding is the simplest segmentation process. Many objects or image regions are characterized by constant reflectivity or light absorption of their surface. Thresholding is computationally inexpensive and fast. Thresholding can easily be done in real time using specialized hardware. Complete segmentation can result from thresholding in simple scenes.

$$R = \bigcup_{i=1}^{S} R_i \qquad R_i \cap R_j = \emptyset \qquad i \neq j$$

Search all the pixels f(i,j) of the image f. An image element g(i,j) of the segmented image is an object pixel if f(i,j) >= T, and is a background pixel otherwise.

$$g(i,j) \begin{array}{ll} = 1 & \text{for } f(i,j) \geq T \\ = 0 & \text{for } f(i,j) < T \end{array}$$

Correct threshold selection is crucial for successful threshold segmentation. Threshold selection can be interactive or can be the result of some threshold detection method

5.2.2.1 Multi-Level Thresholding

The resulting image is no longer binary

$$g(i,j) \begin{array}{ll} = 1 & \text{for } f(i,j) \in D_1 \\ = 2 & \text{for } f(i,j) \in D_2 \\ = 3 & \text{for } f(i,j) \in D_3 \\ = 4 & \text{for } f(i,j) \in D_4 \\ \ldots & \\ = n & \text{for } f(i,j) \in D_n \\ = 0 & \text{otherwise} \end{array}$$

5.2.2.2 Local Thresholding

It is successful only under very unusual circumstances. Gray level variations are likely due to non-uniform lighting, non-uniform input device parameters or a number of other factors.

T=T(f)

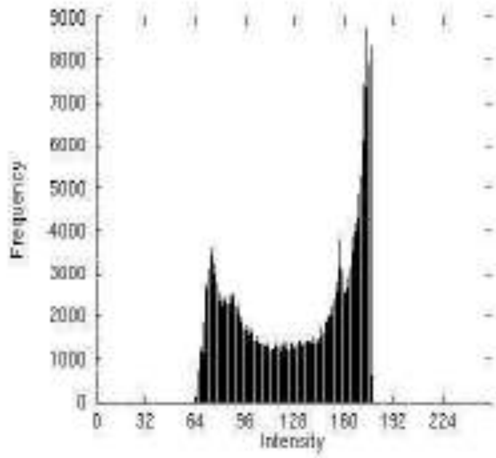
5.2.2.3 Threshold Detection Method

If some property of an image after segmentation is known a priori, the task of threshold selection is simplified, since the threshold is chosen to ensure this property is satisfied. A printed text sheet may be an example if we know that characters of the text cover 1/p of the sheet area.

- P-tile-thresholding
  - choose a threshold T (based on the image histogram) such that 1/p of the image area has gray values less than T and the rest has gray values larger than T
  - in text segmentation, prior information about the ratio between the sheet area and character area can be used
  - if such a priori information is not available - another property, for example the average width of lines in drawings, etc. can be used - the threshold can be determined to provide the required line width in the segmented image

More complex methods of threshold detection

- based on histogram shape analysis
- bimodal histogram - if objects have approximately the same gray level that differs from the gray level of the background



- **Mode method** - find the highest local maxima first and detect the threshold as a minimum between them. To avoid detection of two local maxima belonging to the same global maximum, a minimum distance in gray levels between these maxima is usually required or techniques to smooth histograms are applied.

5.2.3 Region-Based Approach

5.2.3.1 Region-Growing Based segmentation

Homogeneity of regions is used as the main segmentation criterion in region growing. The criteria for homogeneity:

- gray level
- color
- texture
- shape
- model

The basic purpose of region growing is to segment an entire image R into smaller sub-images, Ri, i=1,2,….,N. which satisfy the following conditions:

$$R = \bigcup_{i=1}^{N} R_i ; R_1 \bigcap R_j = \Phi, i \neq j$$

$$H(R_i) = True, i = 1,2,\ldots, N;$$

$$H(R_i \bigcup R_j) = False, i \neq j;$$

5.2.3.2 Region Splitting

The basic idea of region splitting is to break the image into a set of disjoint regions, which are coherent within themselves:

- Initially take the image as a whole to be the area of interest.
- Look at the area of interest and decide if all pixels contained in the region satisfy some *similarity constraint.*
- If TRUE then the area of interest corresponds to an entire region in the image.
- If FALSE split the area of interest (usually into four equal subareas) and consider each of the sub-areas as the area of interest in turn.
- This process continues until no further splitting occurs. In the worst case this happens when the areas are just one pixel in size.

If only a splitting schedule is used then the final segmentation would probably contain many neighboring regions that have identical or similar properties. We need to merge these regions.

5.2.3.3 Region Merging

The result of region merging usually depends on the order in which regions are merged. The simplest methods begin merging by starting the segmentation using regions of 2x2, 4x4 or 8x8 pixels. Region descriptions are then based on their statistical gray level properties. A region description is compared with the description of an adjacent region; if they match, they are merged into a larger region and a new region description is computed. Otherwise regions are marked as non-matching.

Merging of adjacent regions continues between all neighbors, including newly formed ones. If a region cannot be merged with any of its neighbors, it is marked `final' and the merging process stops when all image regions are so marked.

Merging Heuristics:

- Two adjacent regions are merged if a significant part of their common boundary consists of weak edges
- Two adjacent regions are also merged if a significant part of their common boundary consists of weak edges, but in this case not considering the total length of the region borders.

Of the two given heuristics, the first is more general and the second cannot be used alone because it does not consider the influence of different region sizes.

Region merging process could start by considering

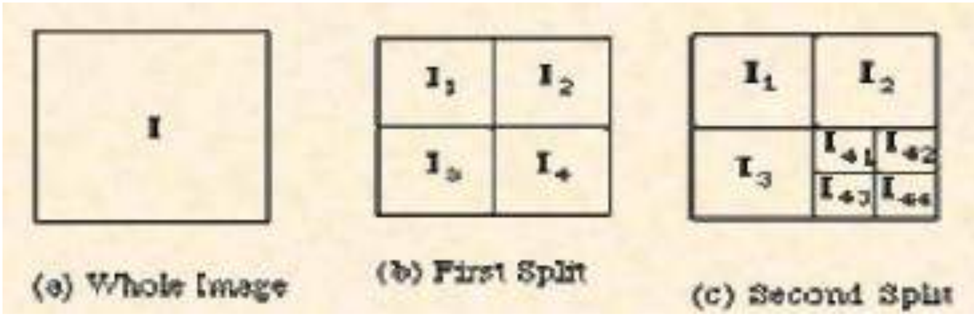
- small segments (2*2,…,8*8) selected a priori from the image
- segments generated by thresholding
- regions generated by a region splitting module

The last case is called as “Split and Merge” method. Region merging methods generally use similar criteria of homogeneity as region splitting methods, and only differ in the direction of their application.

5.2.3.4 <u>Split & Merge</u>

To illustrate the basic principle of split and merge methods, let us consider an imaginary image.

- Let I denote the whole image shown in Fig. (a)
- Not all the pixels in Fig (a) are similar. So the region is split as in Fig. (b).
- Assume that all pixels within each of the regions I1, I2 and I3 are similar, but those in I4 are not.
- Therefore I4 is split next, as shown in Fig. (c).
- Now assume that all pixels within each region are similar with respect to that region, and that after comparing the split regions, regions I43 and I44 are found to be identical.
- These pair of regions is thus merged together, as in shown in Fig. (d).



A combination of splitting and merging may result in a method with the advantages of both the approaches. Split-and-merge approaches work using pyramid image representations. Regions are square-shaped and correspond to elements of the appropriate pyramid level.

If any region in any pyramid level is not homogeneous (excluding the lowest level), it is split into four sub-regions -- these are elements of higher resolution at the level below. If four regions exist at any pyramid level with approximately the same value of homogeneity measure, they are merged into a single region in an upper pyramid level.

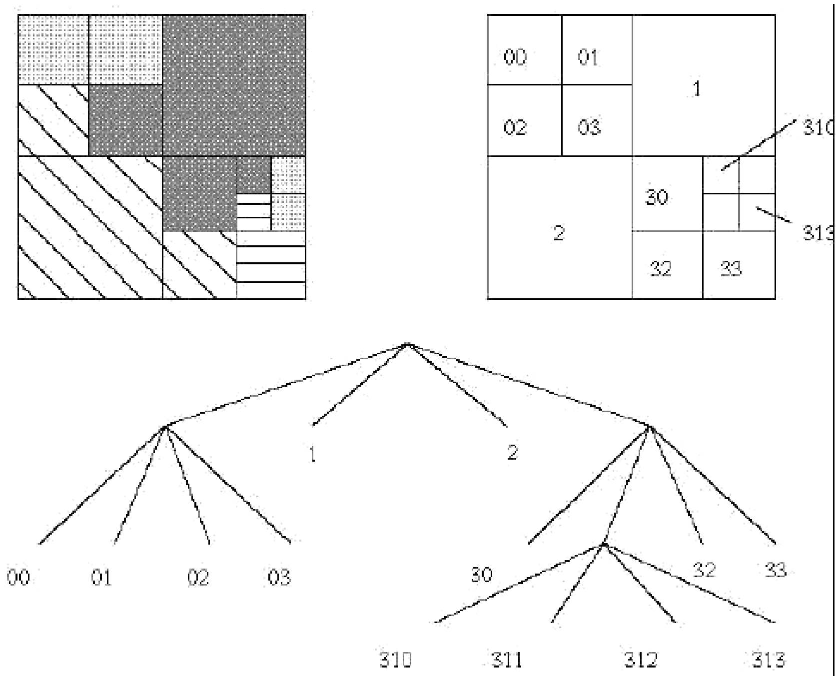
We can also describe the splitting of the image using a tree structure, called a modified *quadtree*. Each non-terminal node in the tree has at most four descendants, although it may have less due to merging.

Quadtree decomposition is an operation that subdivides an image into blocks that contain "similar" pixels. Usually the blocks are square, although sometimes they may be rectangular. For the purpose of this demo, pixels in a block are said to be "similar" if the range of pixel values in the block are not greater than some threshold.

Quadtree decomposition is used in variety of image analysis and compression applications.

An unpleasant drawback of segmentation quadtrees, is the square region shape assumption. It is not possible to merge regions which are not part of the same branch of the segmentation tree. Because both split-and-merge processing options are available, the starting segmentation does not have to satisfy any of the homogeneity conditions.

The segmentation process can be understood as the construction of a segmentation quadtree where each leaf node represents a homogeneous region. Splitting and merging corresponds to removing or building parts of the segmentation quadtree.

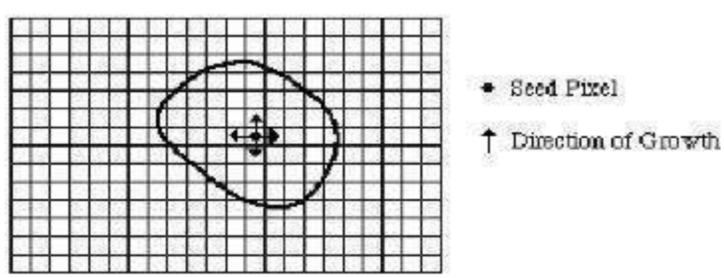


5.2.3.5 <u>Region Growing</u>

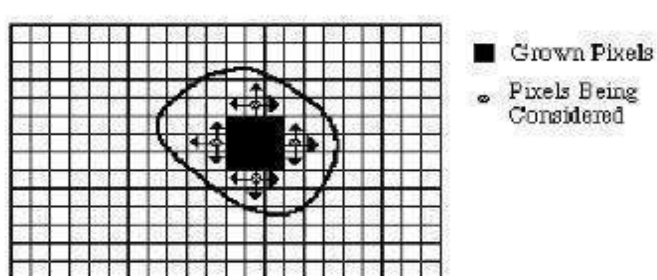
Region growing approach is the opposite of the split and merges approach:

- An initial set of small areas is iteratively merged according to similarity constraints.
- Start by choosing an arbitrary *seed pixel* and compare it with neighboring pixels
- Region is *grown* from the seed pixel by adding in neighboring pixels that are similar, increasing the size of the region.
- When the growth of one region stops we simply choose another seed pixel which does not yet belong to any region and start again.
- This whole process is continued until all pixels belong to some region.
- A *bottom up* method.

Region growing methods often give very good segmentations that correspond well to the observed edges.



(a) Start of Growing a Region



However starting with a particular seed pixel and letting this region grow completely before trying other seeds biases the segmentation in favour of the regions which are segmented first.

This can have several undesirable effects:

- Current region dominates the growth process -- ambiguities around edges of adjacent regions may not be resolved correctly.
- Different choices of seeds may give different segmentation results.
- Problems can occur if the (arbitrarily chosen) seed point lies on an edge.

To counter the above problems, *simultaneous region growing* techniques have been developed.

- Similarities of neighboring regions are taken into account in the growing process.
- No single region is allowed to completely dominate the proceedings.
- A number of regions are allowed to grow at the same time.
- Similar regions will gradually coalesce into expanding regions.
- Control of these methods may be quite complicated but efficient methods have been developed.
- Easy and efficient to implement on parallel computers.

5.2.4 **Edge and Line Detection**

5.2.4.1 Edge Detection

Edges are places in the image with strong intensity contrast. Since edges often occur at image locations representing object boundaries, edge detection is extensively used in image segmentation when we want to divide the image into areas corresponding to different objects. Representing an image by its edges has the further advantage that the amount of data is reduced significantly while retaining most of the image information.

Canny edge Detection

It is optimal for step edges corrupted by white noise. Optimality related to three criteria

- detection criterion ... important edges should not be missed, there should be no spurious responses
- localization criterion ... distance between the actual and located position of the edge should be minimal
- one response criterion ... minimizes multiple responses to a single edge (also partly covered by the first criterion since when there are two responses to a single edge one of them should be considered as false)

Canny's edge detector is based on several ideas:

1) The edge detector was expressed for a 1D signal and the first two optimality criteria. A closed form solution was found using the calculus of variations.

2) If the third criterion (multiple responses) is added, the best solution may be found by numerical optimization. The resulting filter can be approximated effectively with error less than 20% by the first derivative of a Gaussian smoothing filter with standard deviation σ; the reason for doing this is the existence of an effective implementation.

3) The detector is then generalized to two dimensions. A step edge is given by its position, orientation, and possibly magnitude (strength).

- Suppose G is a 2D Gaussian and assume we wish to convolute the image with an operator G_n which is a first derivative of G in the direction n.

$$G_n = \frac{\partial G}{\partial \mathbf{n}} = \mathbf{n} \cdot \nabla G$$

- The direction **n** should be oriented perpendicular to the edge
  - this direction is not known in advance
  - however, a robust estimate of it based on the smoothed gradient direction is available
  - if g is the image, the normal to the edge is estimated as

$$\mathbf{n} = \frac{\nabla(G * g)}{|\nabla(G * g)|}$$

- The edge location is then at the local maximum in the direction **n** of the operator G_n convoluted with the image g

$$\frac{\partial}{\partial \mathbf{n}} G_n * g = 0$$

- Substituting in equation for G_n from equation, we get

$$\frac{\partial^2}{\partial \mathbf{n}^2} G * g = 0$$

- This equation shows how to find local maxima in the direction perpendicular to the edge; this operation is often referred to as **non-maximum suppression**.

- As the convolution and derivative are associative operations in equation
  - first convolute an image g with a symmetric Gaussian G
  - then compute the directional second derivative using an estimate of the direction **n**
  - strength of the edge (magnitude of the gradient of the image intensity function g) is measured as

$$|G_n * g| = |\nabla(G * g)|$$

4) Spurious responses to the single edge caused by noise usually create a so called **'streaking'** problem that is very common in edge detection in general.

- Output of an edge detector is usually thresholded to decide which edges are significant.
- Streaking means breaking up of the edge contour caused by the operator fluctuating above and below the threshold.
- Streaking can be eliminated by **thresholding with hysteresis**.
  - If any edge response is above a **high threshold**, those pixels constitute definite output of the edge detector for a particular scale.
  - Individual weak responses usually correspond to noise, but if these points are connected to any of the pixels with strong responses they are more likely to be actual edges in the image.
  - Such connected pixels are treated as edge pixels if their response is above a **low threshold**.
  - The low and high thresholds are set according to an estimated signal to noise ratio.

5) The correct scale for the operator depends on the objects contained in the image.

- The solution to this unknown is to use multiple scales and aggregate information from them.
- Different scale for the Canny detector is represented by different standard deviations σ of the Gaussians.
- There may be several scales of operators that give significant responses to edges (i.e., signal to noise ratio above the threshold); in this case the operator with the smallest scale is chosen as it gives the best localization of the edge.
- Feature synthesis approach.
  - All significant edges from the operator with the smallest scale are marked first.
  - Edges of a hypothetical operator with larger σ are synthesized from them (i.e., a prediction is made of how the large σ should perform on the evidence gleaned from the smaller σ).
  - Then the synthesized edge response is compared with the actual edge response for larger σ.
  - Additional edges are marked only if they have significantly stronger response than that predicted from synthetic output.
- This procedure may be repeated for a sequence of scales, a cumulative edge map is built by adding those edges that were not identified at smaller scales.

Algorithm: Canny edge detector

1. Repeat steps (2) till (6) for ascending values of the standard deviation σ.
2. Convolve an image g with a Gaussian of scale σ.
3. Estimate local edge normal directions **n** for each pixel in the image.
4. Find the location of the edges (non-maximal suppression).
5. Compute the magnitude of the edge
6. Threshold edges in the image with hysteresis to eliminate spurious responses.

7. Aggregate the final information about edges at multiple scale using the `feature synthesis' approach.

5.2.4.2 **Edge Operators**

Since edges consist of mainly high frequencies, we can, in theory, detect edges by applying a high pass frequency filter in the Fourier domain or by convolving the image with an appropriate kernel in the spatial domain. In practice, edge detection is performed in the spatial domain, because it is computationally less expensive and often yields better results.

`Since edges correspond to strong illumination gradients, we can highlight them by calculating the derivatives of the image.We can see that the position of the edge can be estimated with the maximum of the 1st derivative or with the zero-crossing of the 2nd derivative. Therefore we want to find a technique to calculate the derivative of a two-dimensional image. For a discrete one-dimensional function *f(i)*, the first derivative can be approximated by

$$\frac{d\,f(i)}{d(i)} = f(i+1) - f(i)$$

Calculating this formula is equivalent to convolving the function with [-1 1]. Similarly the 2nd derivative can be estimated by convolving *f(i)* with [1 -2 1].

Different edge detection kernels which are based on the above formula enable us to calculate either the 1st or the 2nd derivative of a two-dimensional image. There are two common approaches to estimate the 1st derivative in a two-dimensional image, Prewitt compass edge detection and *gradient edge detection*.

Prewitt compass edge detection involves convolving the image with a set of (usually 8) kernels, each of which is sensitive to a different edge orientation. The kernel producing the maximum response at a pixel location determines the edge magnitude and orientation. Different sets of kernels might be used: examples include Prewitt, Sobel, Kirsch and Robinson kernels.

Gradient edge detection is the second and more widely used technique. Here, the image is convolved with only two kernels, one estimating the gradient in the *x*-direction, *Gx*, the other the gradient in the *y*-direction, *Gy*. The absolute gradient magnitude is then given by

$$|G| = \sqrt{Gx^2 + Gy^2}$$

and is often approximated with

$$|G| = |Gx| + |Gy|$$

In many implementations, the gradient magnitude is the only output of a gradient edge detector, however the edge orientation might be calculated with

The most common kernels used for the gradient edge detector are the Sobel, Roberts Cross and Prewitt operators.

After having calculated the magnitude of the 1st derivative, we now have to identify those pixels corresponding to an edge. The easiest way is to threshold the gradient image, assuming that all pixels having a local gradient above the threshold must represent an edge. An alternative technique is to look for local maxima in the gradient image, thus producing one pixel wide edges. A more sophisticated technique is used by the Canny edge detector. It first applies a gradient edge detector to the image and then finds the edge pixels using *non-maximal suppression* and *hysteresis tracking*.

An operator based on the 2nd derivative of an image is the Marr edge detector, also known as *zero crossing detector*. Here, the 2nd derivative is calculated using a Laplacian of Gaussian (LoG) filter. The Laplacian has the advantage that it is an isotropic measure of the 2nd derivative of an image, *i.e.* the edge magnitude is obtained independently from the edge orientation by convolving the image with only one kernel. The edge positions are then given by the zero-crossings in the LoG image. The scale of the edges which are to be detected can be controlled by changing the variance of the Gaussian.

A general problem for edge detection is its sensitivity to noise, the reason being that calculating the derivative in the spatial domain corresponds to accentuating high frequencies and hence magnifying noise. This problem is addressed in the Canny and Marr operators by convolving the image with a smoothing operator (Gaussian) before calculating the derivative.

5.2.5 <u>**Line Detection**</u>

While edges (**I.E.** boundaries between regions with relatively distinct graylevels) are by far the most common type of discontinuity in an image, instances of thin lines in an image occur frequently enough that it is useful to have a separate mechanism for detecting them. A convolution based technique can be used which produces an image description of the thin lines in an input image. Note that the Hough transform can be used to detect lines; however, in that case, the output is a **PARAMETRIC** description of the lines in an image.

The line detection operator consists of a convolution kernel tuned to detect the presence of lines of a particular width *n*, at a particular orientation **θ**. Figure below shows a collection of four such kernels, which each respond to lines of single pixel width at the particular orientation shown.

a)

–1	–1	–1
2	2	2
–1	–1	–1

b)

–1	2	–1
–1	2	–1
–1	2	–1

c)

–1	–1	2
–1	2	–1
2	–1	–1

d)

2	–1	–1
–1	2	–1
–1	–1	2

Four line detection kernels which respond maximally to horizontal, vertical and oblique (+45 and - 45 degree) single pixel wide lines.

These masks above are tuned for light lines against a dark background, and would give a big negative response to dark lines against a light background. If we are only interested in detecting dark lines against a light background, then we should negate the mask values. Alternatively, we might be interested in either kind of line, in which case, we could take the absolute value of the convolution output.

If R_i denotes the response of kernel I, we can apply each of these kernels across an image, and for any particular point, if $R_i > R_j$ for all $j \neq i$ that point is more likely to contain a line whose orientation (and width) corresponds to that of kernel I. One usually thresholds R_i to eliminate weak lines corresponding to edges and other features with intensity gradients which have a different scale than the desired line width. In order to find complete lines, one must join together line fragments, E.G ., with an EDGE TRACKING operator.

5.2.6 **Corner Detection**

Input to the corner detector is the gray-level image. Output is an image in which values are proportional to the likelihood that the pixel is a corner. The Moravec detector is maximal in pixels with high contrast. These points are on corners and sharp edges.

$$MO(i,j) = \frac{1}{8} \sum_{k=i-1}^{i+1} \sum_{l=j-1}^{j+1} |g(k,l) - g(i,j)|$$

Using ZH Operator

The image function f is approximated in the neighborhood of the pixel (i,j) by a cubic polynomial with coefficients c_k. This is a cubic facet model. The ZH operator estimates the corner strength based on the coefficients of the cubic facet model.

$$g(i,j) = c_1 + c_2x + c_3y + c_4x^2 + c_5xy + c_6y^2 + c_7x^3 + c_8x^2y + c_9xy^2 + c_{10}y^3$$

$$ZH(i,j) = \frac{-2\,(c_2^2c_6 - c_2c_3c_5 - c_3^2c_4)}{(c_2^2 + c_3^2)^{\frac{3}{2}}}$$