SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

SCHOOL OF ELECTRICAL AND ELECTRONICS

DEPARTMENT OF ELECTRONICS AND COMMMUNICATION ENGINEERING

UNIT - I

ADVANCED DIGITAL IMAGE PROCESSING – SECA7022

UNIT 1 DIGITAL IMAGE FUNDAMENTALS

Elements of Visual Perception;

Image Sensing and Acquisition;

Image Sampling and Quantization;

Basic Relationships between Pixels;

Monochromatic Vision Models;

Colour Vision Models;

Colour Fundamentals;

Colour Models;

Conversion of Colour

Models; Colour

Transformations.

Elements of Visual Perception;

The eye is nearly a sphere, with an average diameter of approximately 20 mm. Three membranes enclose the eye: the *cornea* and *sclera* outer cover; the *choroid*; and the *retina*. The cornea is a tough, transparent tissue that covers the anterior surface of the eye. Continuous with the cornea, the sclera is an opaque mem-brane that encloses the remainder of the optic globe.

The choroid lies directly below the sclera. This membrane contains a net-work of blood vessels that serve as the major source of nutrition to the eye. Even superficial injury to the choroid, often not deemed serious, can lead to severe eye damage as a result of inflammation that restricts blood flow. The choroid coat is heavily pigmented and hence helps to reduce the amount of ex-traneous light entering the eye and the backscatter within the optical globe. At its anterior extreme, the choroid is divided into the *ciliary body* and the *iris diaphragm*. The latter contracts or expands to control the amount of light that enters the eye. The central opening of the iris (the *pupil*) varies in diameter from approximately 2 to 8 mm. The front of the iris contains the visible pigment of the eye, whereas the back contains a black pigment.

The *lens* is made up of concentric layers of fibrous cells and is suspended by fibers that attach to the ciliary body. It contains 60 to 70% water, about 6% fat, and more protein than any other tissue in the eye.The lens is colored by a slightly yellow pigmentation that increases with age. In extreme cases, excessive clouding of the lens, caused by the affliction commonly referred to as *cataracts*, can lead to poor color discrimination and loss of clear vision. The lens absorbs approximate-ly 8% of the visible light spectrum, with relatively higher absorption at shorter wavelengths. Both infrared and ultraviolet light are absorbed appreciably by pro-teins within the lens structure and, in excessive amounts, can damage the eye.

The innermost membrane of the eye is the retina, which lines the inside of the wall’s entire posterior portion. When the eye is properly focused, light from an object outside the eye is imaged on the retina.

Pattern vision is afforded by the distribution of discrete light receptors over the surface of the retina. There are two classes of receptors: *cones* and *rods*. The cones in each eye number between 6 and 7 million. They are located primarily in the central portion of the retina, called the *fovea*, and are highly sensitive to color. Humans can resolve fine de-tails with these cones largely because each one is connected to its own nerve end. Muscles controlling the eye rotate the eyeball until the image of an object of interest falls on the fovea. Cone vision is called *photopic* or bright-light vision.

The number of rods is much larger: Some 75 to 150 million are distributed over the retinal surface. The larger area of distribution and the fact that sever-al rods are connected to a single nerve end reduce the amount of detail discernible by these receptors. Rods serve to give a general, overall picture of the field of view. They are not involved in color vision and are sensitive to low levels of illumination. For example, objects that appear brightly colored in day-light when seen by moonlight appear as colorless forms because only the rods are stimulated. This phenomenon is known as *scotopic* or dim-light vision.

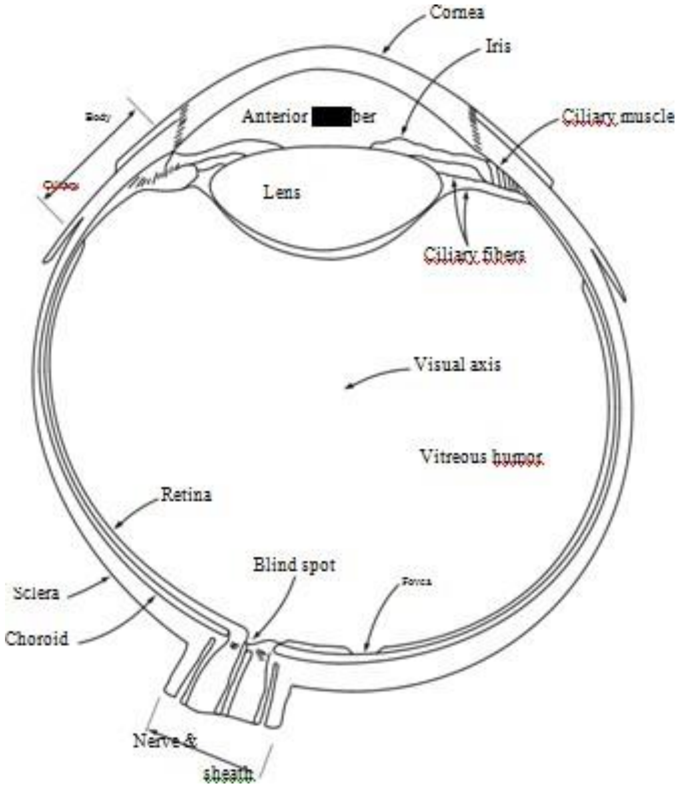
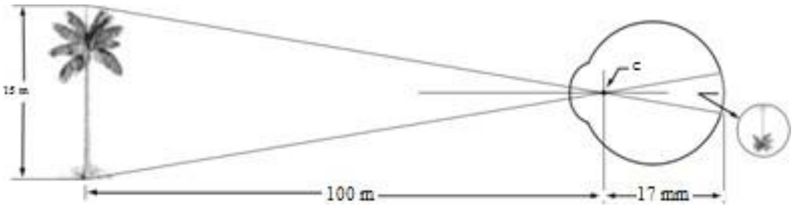


Image Formation in the Eye

The principal difference between the lens of the eye and an ordinary optical lens is that the former is flexible. As illustrated in Fig. 2.1, the radius of curvature of the anterior surface of the lens is greater than the radius of its posterior surface. The shape of the lens is controlled by tension in the fibers of the ciliary body. To focus on distant objects, the controlling muscles cause the lens to be relatively flattened. Similarly, these muscles allow the lens to become thicker in order to focus on objects near the eye.

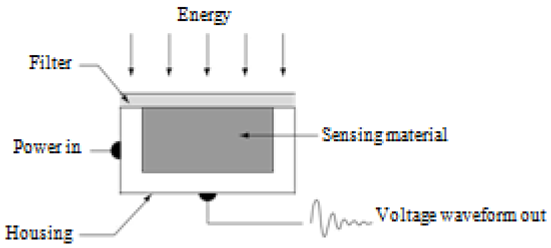


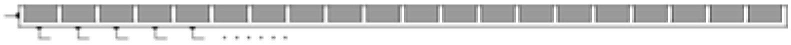
The distance between the center of the lens and the retina (called the *focal length*) varies from approximately 17 mm to about 14 mm, as the refractive power of the lens increases from its minimum to its maximum. When the eye focuses on an object farther away than about 3 m, the lens exhibits its lowest refractive power. When the eye focuses on a nearby object, the lens is most strong-ly refractive. This information makes it easy to calculate the size of the retinal image of any object. In Fig., for example, the observer is looking at a tree 15 m high at a distance of 100 m. If h is the height in mm of that object in the retinal image, the geometry of Fig. 2.3 yields 15/100=h/17 or h=2.55 mm. 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 ultimate-ly decoded by the brain.

Image Sensing and Acquisition;

The types of images in which we are interested are generated by the combination of an “illumination” source and the reflection or absorption of energy from that source by the elements of the “scene” being imaged. We enclose *illumination* and *scene* in quotes to emphasize the fact that they are considerably more general than the familiar situation in which a visible light source illuminates a common everyday 3-D (three-dimensional) scene. For example, the illumination may originate from a source of electromagnetic energy such as radar, infrared, or X-ray energy. But, as noted earlier, it could originate from less traditional sources, such as ultrasound or even a computer-generated illumination pattern. Similarly, the scene elements could be familiar objects, but they can just as easily be molecules, buried rock formations, or a human brain. We could even image a source, such as acquiring images of the sun. Depending on the nature of the source, illumination energy is reflected from, or transmitted through, objects. An example in the first category is light reflected from a planar surface. An example in the second category is when X-rays pass through a patient’s body for the purpose of generating a diagnostic X-ray film. In some applications, the reflected or transmitted energy is focused onto a photoconverter (e.g., a phosphor screen), which converts the energy into visible light. Electron microscopy and some applications of gamma imaging use this approach.

Figure shows the three principal sensor arrangements used to transform illumination energy into digital images. The idea is simple: Incoming energy is transformed into a voltage by the combination of input electrical power and sensor material that is responsive to the particular type of energy being detected. The output voltage waveform is the response of the sensor(s), and a dig-ital quantity is obtained from each sensor by digitizing its response.





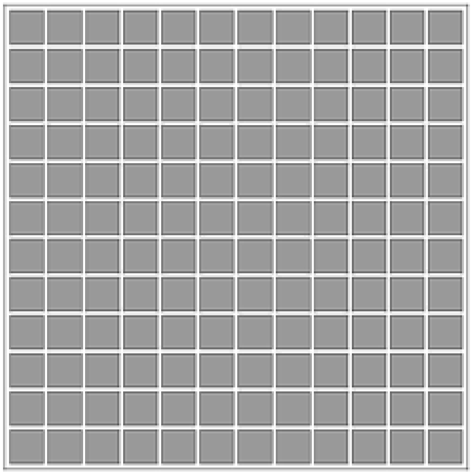


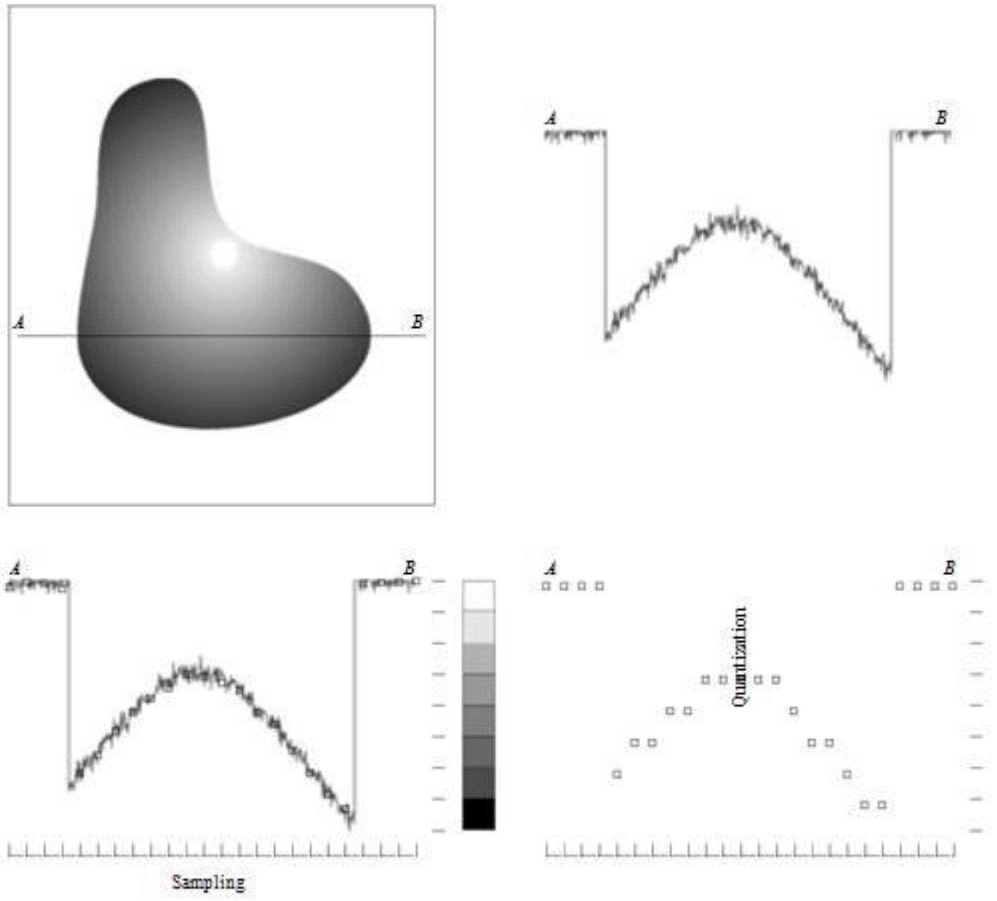
Image Sampling and Quantization;

The output of most sensors is a continuous voltage waveform whose amplitude and spatial behavior are related to the physical phenomenon being sensed. To create a digital image, we need to convert the continuous sensed data into digital form. 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 to digital form, we have to sample the function in both coordinates and in amplitude. Digitizing the coordinate values is called *sampling*. Digitizing the amplitude values is called *quantization*.

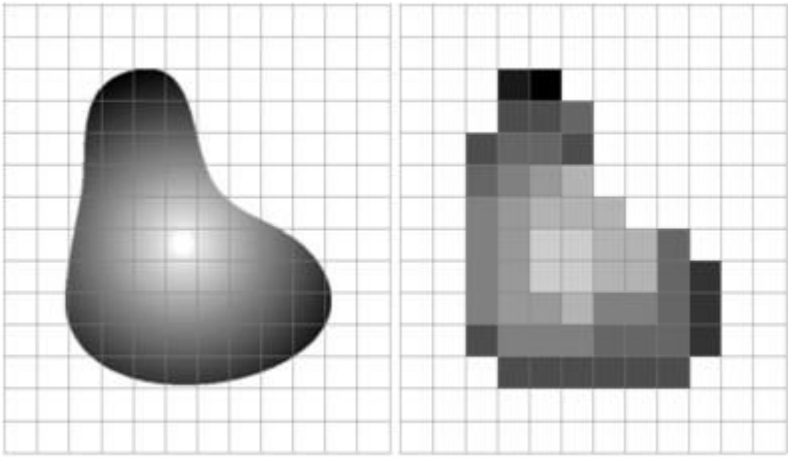
The one-dimensional function shown in Fig. is a plot of amplitude (gray level) values of the continuous image along the line segment AB in Fig. The random variations are due to image noise. To sample this function, we take equally spaced samples along line AB, as shown in Fig. The location of each sample is given by a vertical tick mark in the bottom part of the figure. The samples are shown as small white squares superimposed on the function. The set of these discrete locations gives the sampled function. However, the values of the samples still span (vertically) a continuous range of gray-level values. In order to form a digital function, the gray-level values also must be converted (*quantized*) into discrete quantities. The right side of Fig. shows the gray-level scale divided into eight discrete levels, ranging from black to white. The vertical tick marks indicate the specific value assigned to each of the eight gray levels. The continuous gray levels are quantized simply by assigning one of the eight discrete gray levels to each sample. The assignment is made depending on the vertical proximity of a sample to a vertical tick mark. The digital samples resulting from both sampling and quantization are shown in Fig. Starting at the top of the image and carrying out this procedure line by line produces a two- dimensional digital image.

Sampling in the manner just described assumes that we have a continuous image in both coordinate directions as well as in amplitude. In practice, the method of sampling is determined by the sensor arrangement used to generate the image. When an image is generated by a single sensing element combined with mechanical motion, as in Fig., the output of the sensor is quantized in the manner described above. However, sampling is accomplished by selecting the number of individual mechanical increments at which we activate the sensor to collect data. Mechanical motion can be made very exact so, in principle, there is almost no limit as to how fine we can sample an image. However, practical limits are established by imperfections in the optics used to focus on the sensor an illumination spot that is inconsistent with the fine resolution achiev-able with mechanical displacements.



When a sensing strip is used for image acquisition, the number of sensors in the strip establishes the sampling limitations in one image direction. Mechanical motion in the other direction can be controlled more accurately, but it makes little sense to try to achieve sampling density in one direction that exceeds the sampling limits established by the number of sensors in the other. Quantization of the sensor outputs completes the process of generating a digital image.

When a sensing array is used for image acquisition, there is no motion and the number of sensors in the array establishes the limits of sampling in both directions. Quantization of the sensor outputs is as before. Figure illustrates this concept. Figure shows a continuous image projected onto the plane of an array sensor. Figure shows the image after sampling and quantization. Clearly, the quality of a digital image is determined to a large degree by the number of samples and discrete gray levels used in sampling and quantization.



(a) Continuous image projected onto a sensor array. (b) Result of image sampling and quantization.

Basic Relationships between Pixels;

Neighbors of a Pixel

A pixel p at coordinates (x, y) has four *horizontal* and *vertical* neighbors whose coordinates are given by

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

This set of pixels, called the 4-*neighbors* of p, is denoted by N_4(p). Each pixel is a unit distance from (x, y), and some of the neighbors of p lie outside the digital image if (x, y) is on the border of the image.

The four *diagonal* neighbors of p have coordinates

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

and are denoted by N_D(p). These points, together with the 4-neighbors, are called the 8-*neighbors* of p, denoted by N_8(p). As before, some of the points in N_D(p) and N_8(p) fall outside the image if (x, y) is on the border of the image.

Adjacency, Connectivity, Regions, and Boundaries

Connectivity between pixels is a fundamental concept that simplifies the definition of numerous digital image concepts, such as regions and boundaries. To establish if two pixels are connected, it must be determined if they are neighbors and if their gray levels satisfy a specified criterion of similarity (say, if their gray levels are equal). For instance, in a binary image with values 0 and 1, two pixels may be 4- neighbors, but they are said to be connected only if they have the same value.

Let *V* be the set of gray-level values used to define adjacency. In a binary image, V={1} if we are referring to adjacency of pixels with value 1. In a gray-scale image, the idea is the same, but set *V* typically contains more elements. For example, in the adjacency of pixels with a range of possible gray- level values 0 to 255, set *V* could be any subset of these 256 values. We consider three types of adjacency:

(a) 4-*adjacency*. Two pixels p and q with values from *V* are 4-adjacent if q is in the set *N*4(p).

(b) 8-*adjacency*. Two pixels p and q with values from *V* are 8-adjacent if q is in the set *N*8(p).

(c) *m-adjacency* (mixed adjacency). Two pixels p and q with values from *V* are m- adjacent if

(i) q is in *N*4(p), *or*

(ii) q is in *N*D(p) *and* the set N4(p) ¨ N4(q) has no pixels whose values are from *V*.

Mixed adjacency is a modification of 8-adjacency. It is introduced to eliminate the ambiguities that often arise when 8-adjacency is used.

A (*digital*) *path* (or *curve*) from pixel p with coordinates (x, y) to pixel q with coordinates (s, t) is a sequence of distinct pixels with coordinates

A x0 , y0B , A x1 , y1B , ***p*** , A xn , ynB

where A x0 , y0B = (x, y), A xn , ynB = (s, t), and pixels A xi , yiB and A xi - 1 , yi - 1B are adjacent for 1 i n. In this case, n is the *length* of the path. If

A x0 , y0B = (xn , yn), the path is a *closed* path. We can define 4-, 8-, or m-paths de-pending on the type of adjacency specified

Let *S* represent a subset of pixels in an image. Two pixels p and q are said to be *connected* in *S* if there exists a path between them consisting entirely of pixels in *S*. For any pixel p in *S*, the *set* of pixels that are connected to it in *S* is called a *connected component* of *S*. If it only has one connected component, then set *S* is called a *connected set*.

Let *R* be a subset of pixels in an image. We call *R* a *region* of the image if *R* is a connected set. The *boundary* (also called *border* or *contour*) of a region *R* is the set of pixels in the region that have one or more neighbors that are not in *R*. If *R* happens to be an entire image (which we recall is a rectangular set of pixels), then its boundary is defined as the set of pixels in the first and last rows and columns of the image.

Distance Measures

For pixels p, q, and z, with coordinates (x, y), (s, t), and (v, w), respectively, *D* is a *distance function* or *metric* if

(a) D(p, q)=0 A D(p, q)=0 iff p=qB ,

(b) D(p, q)=D(q, p), and

(c) D(p, z) D(p, q)+D(q, z).

The *Euclidean distance* between p and q is defined as

$$(y - t)^{2} \; D \; ^{2}.$$

For this distance measure, the pixels having a distance less than or equal to some value *r* from (x, y) are the points contained in a disk of radius r centered at (x, y).

The *D*4 *distance* (also called *city-block distance*) between p and q is defined as

$$D4(p, q) = \sum x - s\sum + \sum y - t\sum .$$

In this case, the pixels having a *D*4 distance from (x, y) less than or equal to some value r form a diamond centered at (x, y). For example, the pixels with *D*4 distance 2 from (x, y) (the center point) form the following contours of constant distance:

```
      2
    2 1 2
  2 1 0 1 2
    2 1 2
      2
```

The pixels with *D*4=1 are the 4-neighbors of (x, y).

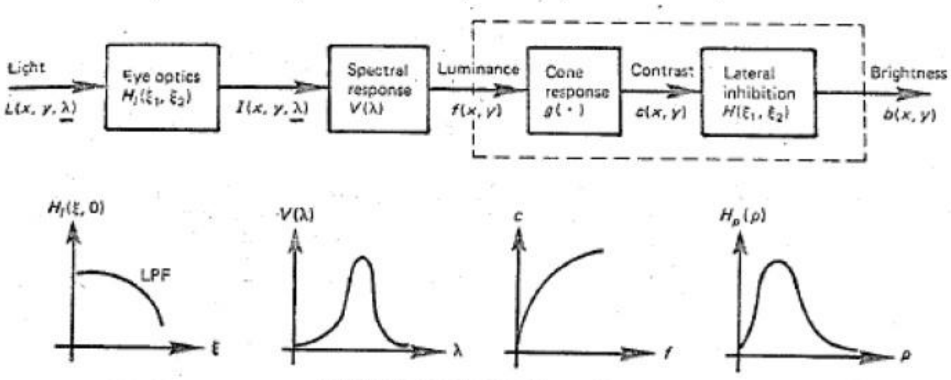
The *D*8 *distance* (also called *chessboard distance*) between p and q is defined

as $D8(p, q) = \max A \sum x - s\sum, \sum y - t\sum B$.

In this case, the pixels with *D*8 distance from (x, y) less than or equal to some value r form a square centered at (x, y).

Monochromatic Vision Models;

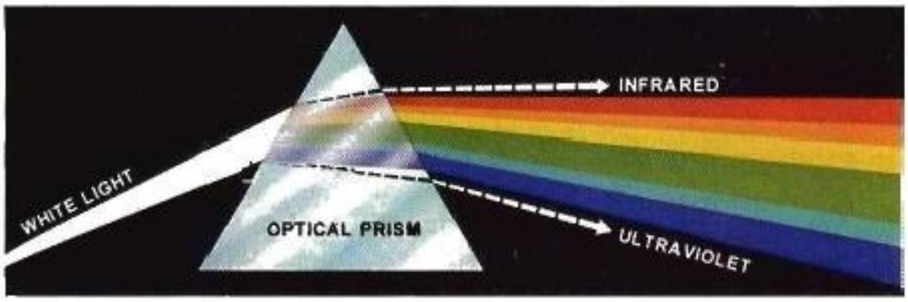
- The logarithmic/linear system eye model provides a reasonable prediction of visual response over a wide range of intensities.
- However, at high spatial frequencies and at very low or very high intensities, observed responses depart from responses predicted by the model.

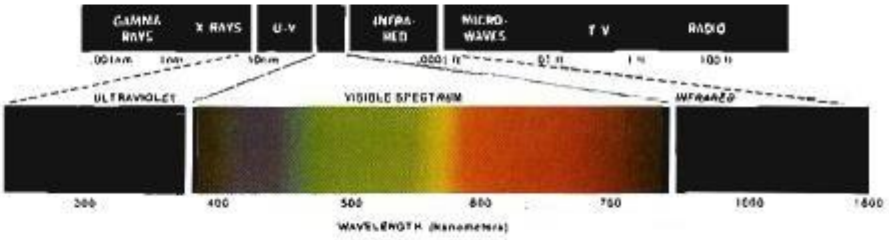


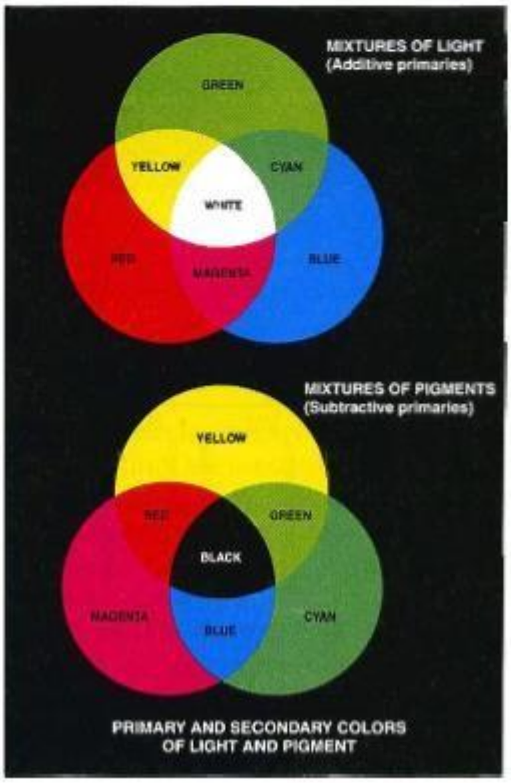
(a) Overall monochrome vision model.

Colour Vision Models;

Colour Fundamentals;







PRIMARY AND SECONDARY COLORS OF LIGHT AND PIGMENT

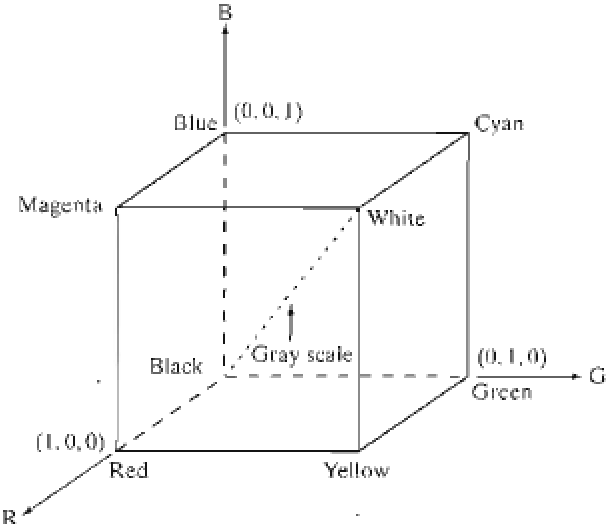
Chromatic light spans the electromagnetic spectrum from approximately 400 to 700 nrn. Three basic quantities are used to describe the quality of a chromatic light source: radiance, luminance and brightness. *Radiance* is the total amount of energy that flows from the light source, and it is usually measured in watts (W). Luminance measured in lumens (lm), gives a measure of the amount of energy an observer *perceived* from a light source. For example, light emitted from a source operating in the far infrared region of the spectrum could have significant energy (radiance), but an observer would hardly perceive it; its luminance would be almost zero. Finally *brightness* is a subjective descriptor that is practically impossible to measure. It embodies the achromatic notion of intensity and is one of thc key factors in describing color sensation.

Colour Models;

Most color models in use today are oriented either toward hardware (such as for color monitors and printers) or toward applications where color manipulation is a goal (such as in the creation of color graphics for animation). In terms of digital image processing, the hardware-oriented models most commonly used in practice are the KGB (red, green, blue) model far color monitors and a broad class of color video cameras; the CMY (cyan, magenta, yellow) and CMYK (cyan, magenta, yellow, black) models for color printing; and the HSI (hue, saturation, intensity) model, which corresponds closely with the way humans describe and interpret color. The HS1 model also has the advantage that it- decouples the color and gray-scale information in an image, making it suitable for many of the gray-scale techniques developed in this book. There are numerous color models in use today due to the fact that color science is a broad field that encompasses many areas of application.

The RGB Color Model:

In the RGB model, each color appears in its primary spectral components of red, green, and blue-This model is based on a Cartesian coordinate system. The color subspace of interest is the cube shown in Fig., in which RGB values are at three corners; cyan, magenta, and yellow are at three other corners; black *is* at the origin; and white is at the corner farthest from the origin. In this model, the gray scale (points of equal RGB values) extends from black to white along the line joining these two points. The different colors in this model are points on or inside the cube, and are defined by vectors extending from the origin.



The CMY and CMYK Color Models:

Cyan, magenta, and yellow are the secondary colors of light or, alternatively, the primary colors of pigments. For example, when a surface coated with cyan pigment is illuminated which white light, no red light is reflected from the surface. That is, cyan subtracts red light from reflected white light, which itself is composed of equal amounts of red, green, and blue light. Most devices that deposit colored pigments on paper, such as color printers and copiers, require CMY data input or perform an RGB to CMY conversion internally. This conversion is

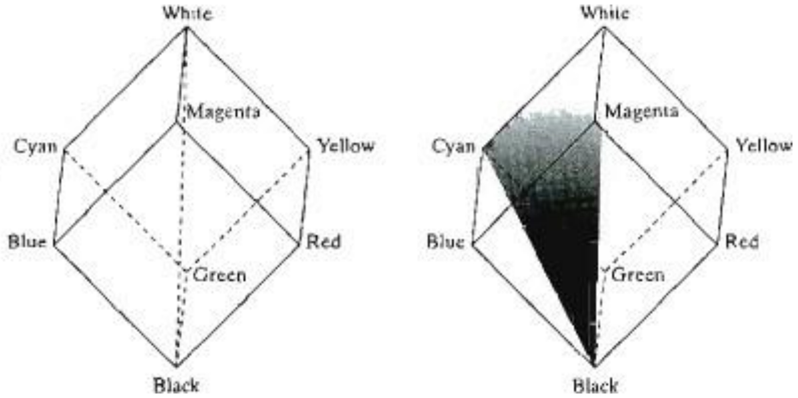
$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

performed using the simple operation where, again, the assumption is that all color values have been normalized to the range [0,1].

demonstrates that light reflected from a surface coated with pure cyan does not contain red (that is, C = 1 - R in the equation).

The HSI Color Model:

the *HSI* (hue, saturation, intensity) color model, decouples the intensity component from the color- carrying information (hue and saturation) in a color image, **As** a result, the HSI model is an idea1 tool for developing image processing algorithms based on color descriptions that are natural and intuitive to humans, who, after all, are the developers and users of these algorithms



Conversion of Colour Models (HSI-RGB):

$$B = I(1 - S)$$

$$R = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right]$$

$$G = 3I - (R + B).$$

$$H = H - 120^\circ.$$

Then the RGB components are

$$R = I(1 - S)$$

$$G = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right]$$

and

$$B = 3I - (R + G).$$

$$H = H - 240^\circ.$$

Then the RGB components are

$$G = I(1 - S)$$

$$B = I\left[1 + \frac{S\cos H}{\cos(60^\circ - H)}\right]$$

and

$$R = 3I - (G + B).$$

Colour Transformations.

Other types of transformations are more general and thus are capable of achieving a wider range of pseudocolor enhancement results than the simple slicing technique discussed in the preceding section. An approach that is particularly attractive is shown in Fig. 6.23. Basically, the idea underlying this approach is to perform three independent transformations on the gray level of any input pixel. The three results are then fed separately into the red, green, and blue channels of a color television monitor. This method produces a composite image whose color content is modulated by the nature of the transformation functions. Note that these are transformations on the gray-level values of an image and are not functions of position.

The method discussed in the previous section is a special case of the technique just described. There, piecewise linear functions of the gray levels (Fig. 6.19) are used to generate colors. The method discussed in this section, on the other hand, can be based on smooth, nonlinear functions, which, as might be expected, gives the technique considerable flexibility.

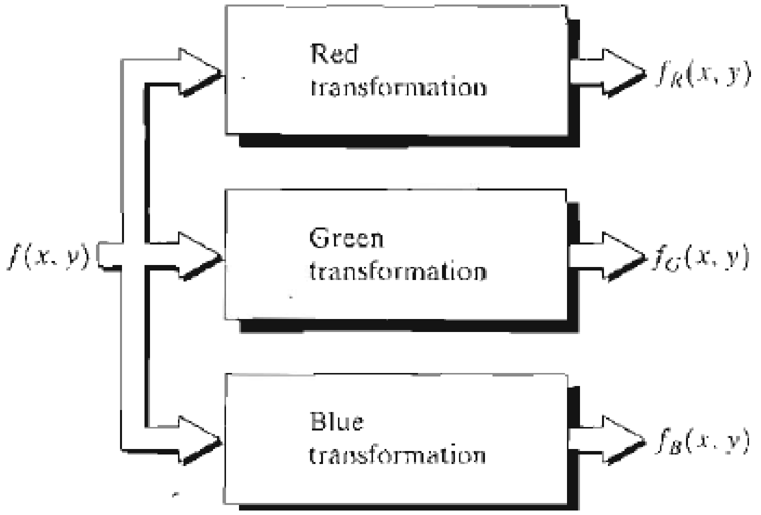


FIGURE 6.23 Functional block diagram for pseudocolor image processing. f_R, f_G, and f_B are fed into the corresponding red, green, and blue inputs of an RGB color monitor.

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

SCHOOL OF ELECTRICAL AND ELECTRONICS

DEPARTMENT OF ELECTRONICS AND COMMMUNICATION ENGINEERING

UNIT - II

ADVANCED DIGITAL IMAGE PROCESSING – SECA7022

UNIT 2 ENHANCEMENT & RESTORATION

Introduction;

Point Processing – Image Negatives, Log transformations, Power Law Transformations, Piecewise-Linear Transformation Functions; Arithmetic/Logic Operations – Image Subtraction, Image Averaging; Histogram Processing – Histogram Equalization, Histogram Matching; Spatial filtering – Smoothing, Sharpening; Smoothing Frequency Domain Filters – Ideal Low Pass, Butterworth Low Pass, Gaussian Low Pass; Sharpening Frequency Domain Filters – Ideal High Pass, Butterworth High Pass, Gaussian High Pass; Model of Image Degradation/Restoration Process; Noise Models; Inverse Filtering; Geometric Transformations.

IMAGE ENHANCEMENT

- Improving the interpretability or perception of information in images for human viewers
- The objective of enhancement technique is to process an image so that the result is more suitable than the original image for a particular application.
- Providing `better' input for other automated image processing techniques

o Spatial domain methods:

- operate directly on pixels

o Frequency domain methods:

- operate on the Fourier transform of an image

SPATIAL DOMAIN METHODS

* Suppose we have a digital image which can be represented by a two dimensional random field (x, y) .

* An image processing operator in the spatial domain may be expressed as a mathematical function T □□□ applied to the image (x, y) to produce a new image g(x, y) □T□ (x, y)□ as follows.

g(x, y) □T□ (x, y)□

The operator T applied on (x, y) may be defined over:

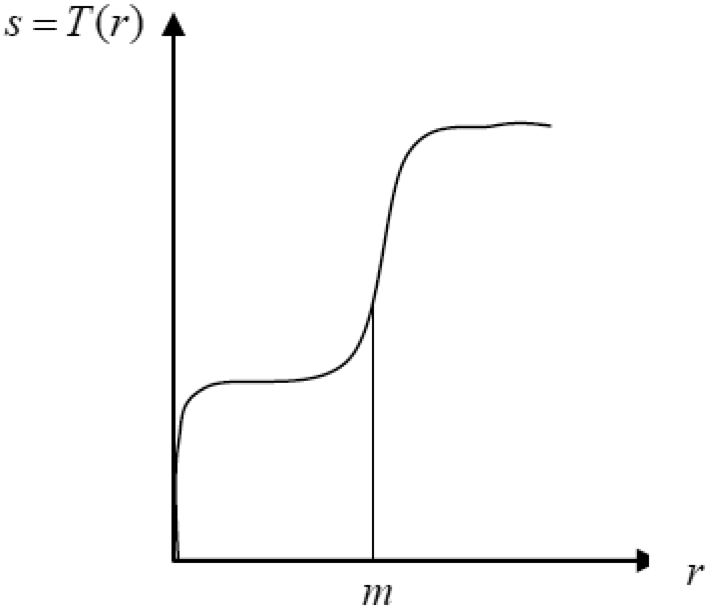
(i) A single pixel (x, y) . In this case T is a grey level transformation (or mapping) function.

(ii) Some neighbourhood of (x, y) .

(iii) T may operate to a set of input images instead of a single image

The result of the transformation shown in the figure below is to produce an image of higher contrast than the original, by darkening the levels below m and brightening the levels above m in the original image. This technique is known as contrast stretching.

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



BASIC GRAY LEVEL TRANSFORMATIONS

* We begin the study of image enhancement techniques by discussing gray-level transformation functions. These are among the simplest of all image enhancement techniques. * The values of pixels, before and after processing, will be denoted by r and s, respectively. As indicated in the previous section, these values are related by an expression of the form s=T(r), where T is a transformation that maps a pixel value r into a pixel value s. * Since we are dealing with digital quantities, values of the transformation function typically are stored in a one-dimensional array and the mappings from r to s are implemented via table lookups. For an 8-bit environment, a lookup table containing the values of T will have 256 entries. * As an introduction to gray-level transformations, consider Fig. which shows three basic types of functions used frequently for image enhancement: linear (negative and identity transformations),

logarithmic (log and inverse-log transformations), and power-law (nth power and nth root transformations). * The identity function is the trivial case in which output intensities are identical to input intensities. It is included in the graph only for completeness.

POINT PROCESSING

- The simplest kind of range transformations are these independent of position *x,y*:

  g = T(f)

- This is called point processing.
- **Important:** every pixel for himself – spatial information completely lost!

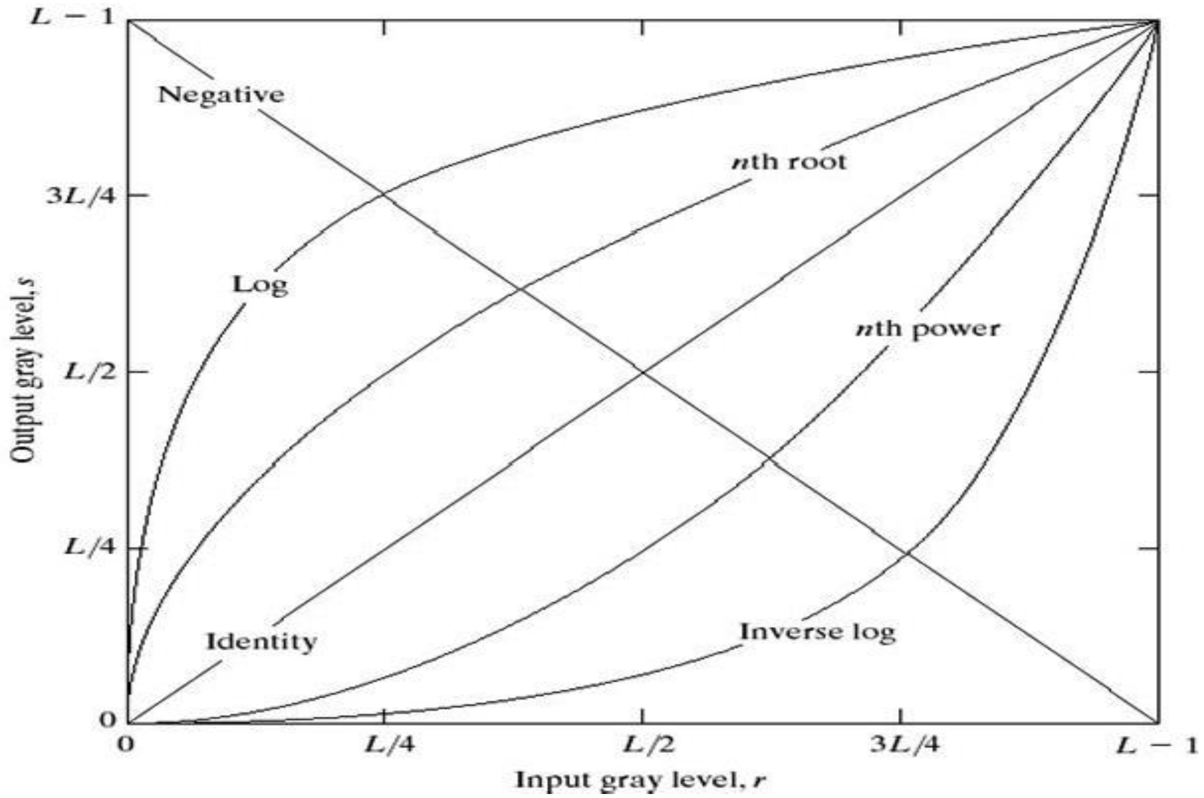
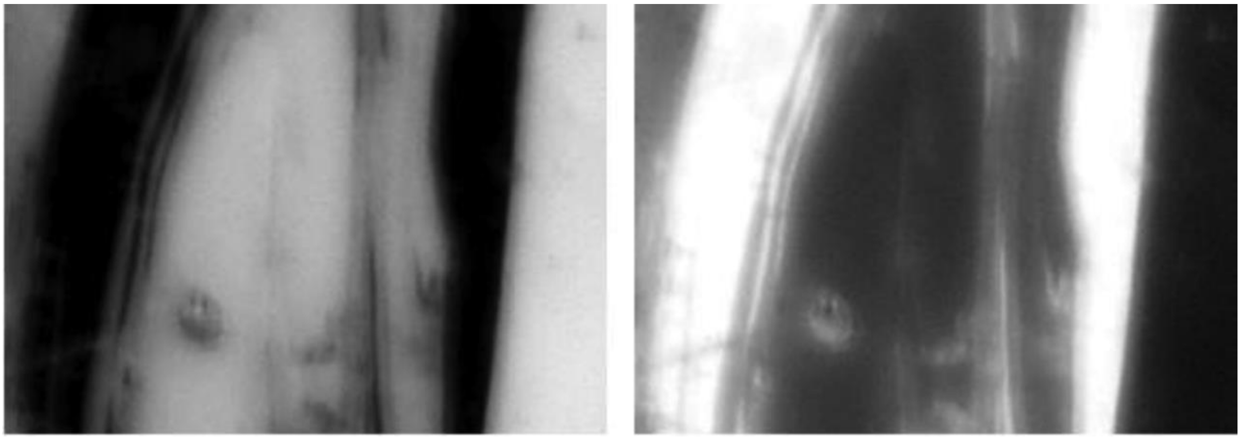


Image Negative

The negative of an image with gray levels in the range [0, L-1] is obtained by using the negative transformation shown in Fig. Reversing the intensity levels of an image in this manner produces the equivalent of a photographic negative. This type of processing is particularly suited for enhancing white or gray detail embedded in dark regions of an image, especially when the black areas are dominant in size.



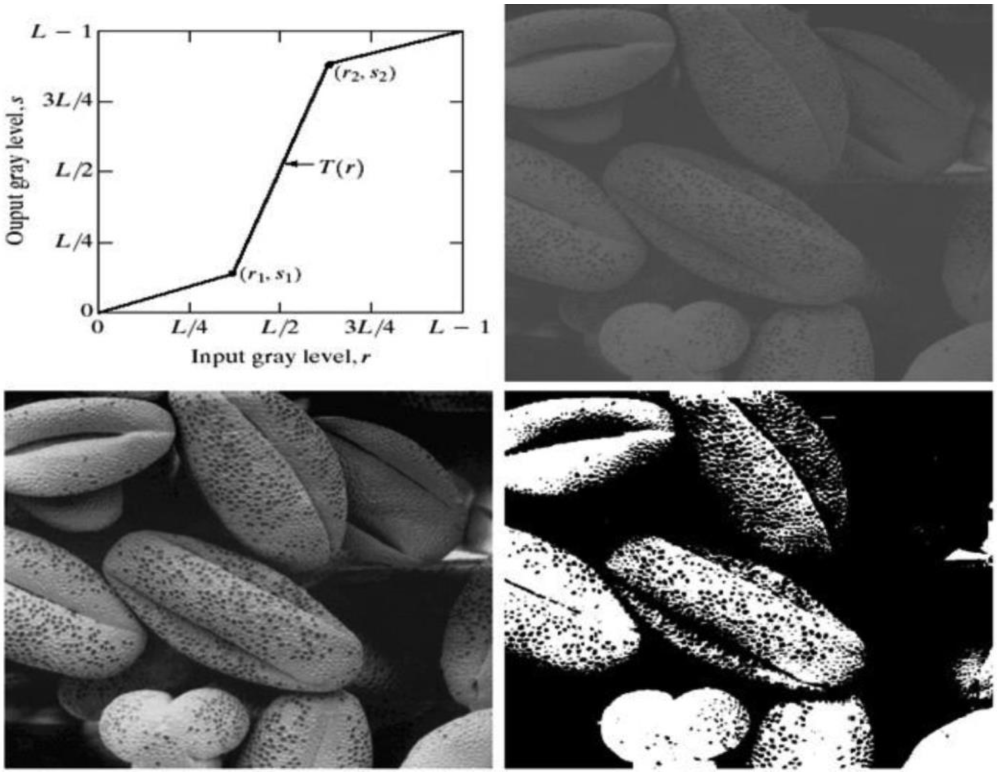
Log Transformations

The general form of the log transformation shown in Fig. 3.3 is s = c log (1 + r) where c is a constant, and it is assumed that r 0.

- The shape of the log curve in Fig. 3.3 shows that this transformation maps a narrow range of low gray-level values in the input image into a wider range of output levels.
- The opposite is true of higher values of input levels. We would use a transformation of this type to expand the values of dark pixels in an image while compressing the higher-level values. The opposite is true of the inverse log transformation

Contrast Stretching

Low contrast images occur often due to poor or non uniform lighting conditions, or due to nonlinearity, or small dynamic range of the imaging sensor. In the figure of Example 1 above you have seen a typical contrast stretching transformation.
$ontrast stretching reduces an image of higher contrast than the original by darkening the levels below m and brightening the levels above m in the image.



a b
c d

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

Histogram processing

- Measure frequency of occurrence of each grey/colour value

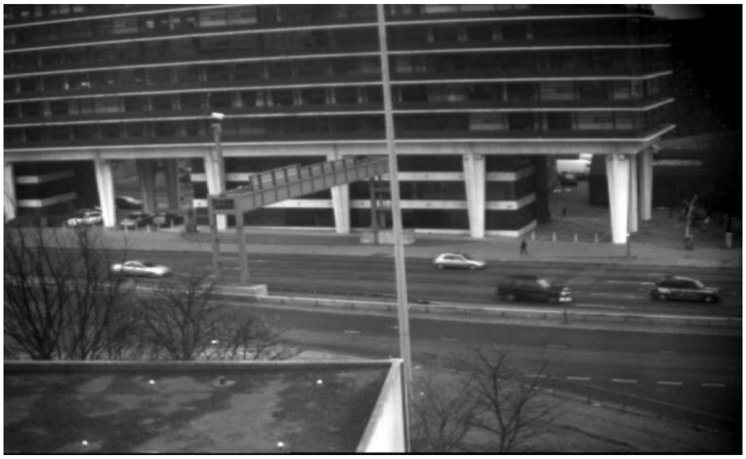
The histogram of a digital image with gray levels in the range [0, L-1] is a discrete function h(rk)=nk. rk-kth gray level nk-number of pixels in the image having gray level rk.

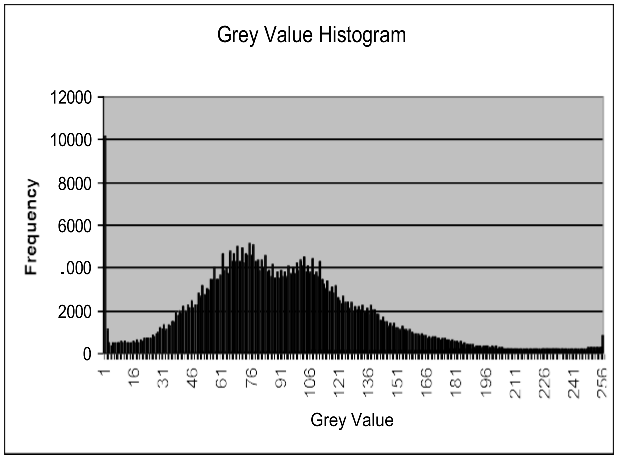
7y processing (modifying) the histogram of an image we can create a new image with specific desired properties. Suppose we have a digital image of size NXN with grey levels in the range [0,L-1] . The histogram of the image is defined as the following discrete function:

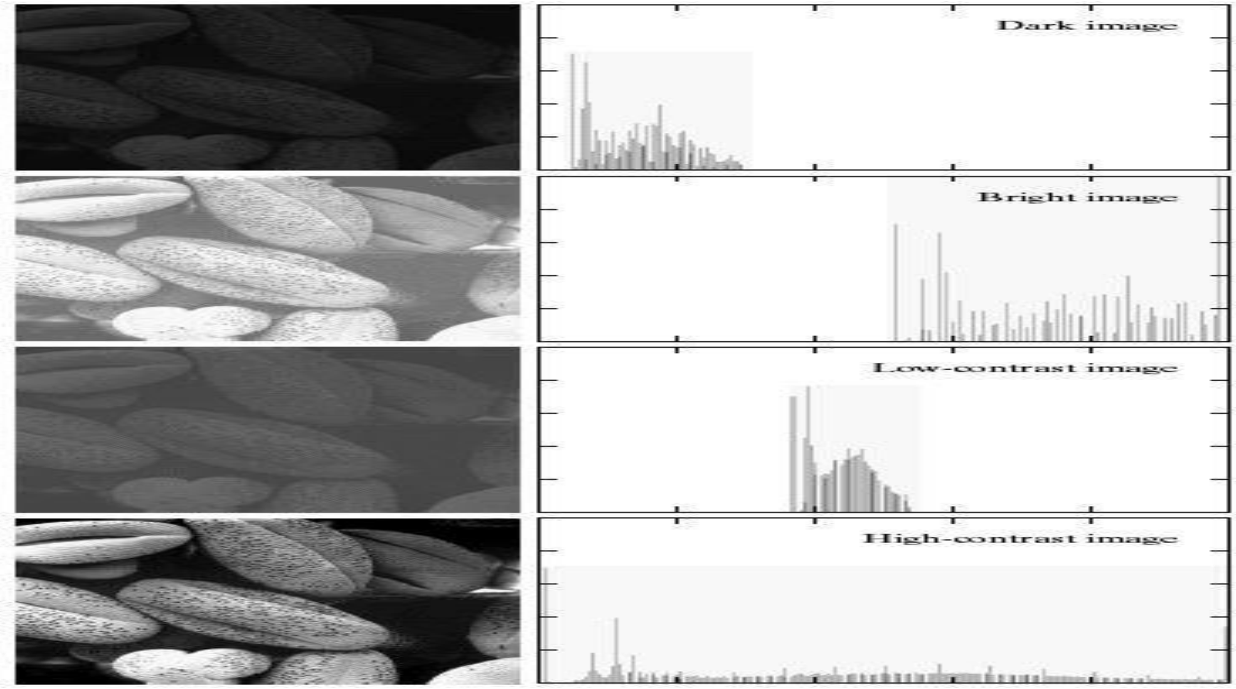
$$p(r_k) = \frac{n_k}{N^2}$$

r_k is the kth grey level, $k = 0,1,\ldots,L-1$
n_k is the number of pixels in the image with grey level r_k
N^2 is the total number of pixels in the image

The histogram represents the frequency of occurrence of the various grey levels in the image. A plot of this function for all values of provides a global description of the appearance of the image.







Directional smoothing

To protect the edges from blurring while smoothing, a directional averaging filter can be useful. Spatial averages $g(x, y : \theta)$ are calculated in several selected directions (for example could be horizontal, vertical, main diagonals

$$g(x, y : \theta) = \frac{1}{N_\theta} \sum_{(k,l)} \sum_{\in W_\theta} f(x-k, y-l)$$

and a direction θ^* is found such that $\left| f(x, y) - g(x, y : \theta^*) \right|$ is minimum. (Note that W_θ is the neighbourhood along the direction θ and N_θ is the number of pixels within this neighbourhood).

Then by replacing $g(x, y : \theta)$ with $g(x, y : \theta^*)$ we get the desired result.

Median filtering

The median m of a set of values is the value that possesses the property that half the values in the set are less than and half are greater than m . Median filtering is the operation that replaces each pixel by the median of the grey level in the neighborhood of that pixel. Median filters are non linear filters because for two sequences x(n) and y(n)

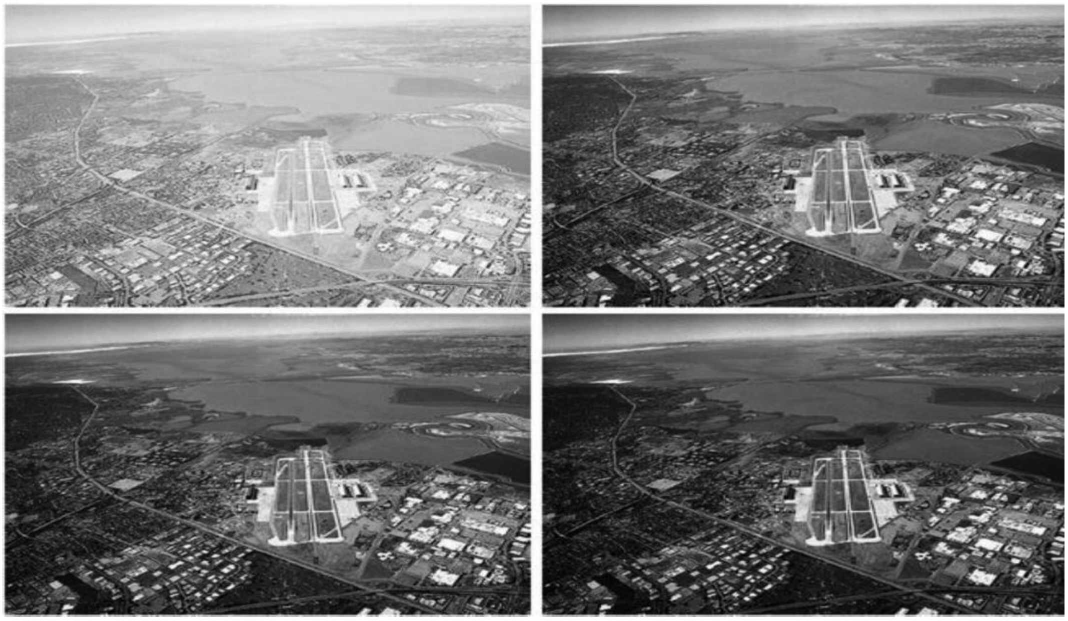
$$\text{median}\{x(n)+y(n)\} \neq \text{median}\{x(n)\}+\text{median}\{y(n)\}$$

Median filters are useful for removing isolated lines or points (pixels) while preserving spatial resolutions. They perform very well on images containing binary (salt and pepper) noise but perform poorly when the noise is Gaussian. Their performance is also poor when the number of noise pixels in the window is greater than or half the number of pixels in the window

$$\frac{1}{273}\times\begin{bmatrix}1 & 4 & 7 & 4 & 1\\ 4 & 16 & 26 & 16 & 4\\ 7 & 26 & 41 & 26 & 7\\ 4 & 16 & 26 & 16 & 4\\ 1 & 4 & 7 & 4 & 1\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 0\\ 0 & \overset{\text{Isolated}}{1} & 0\\ 0 & 0 & 0\end{bmatrix} \xrightarrow{\text{Median filtering}} \begin{bmatrix}0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}$$

Image Enhancement

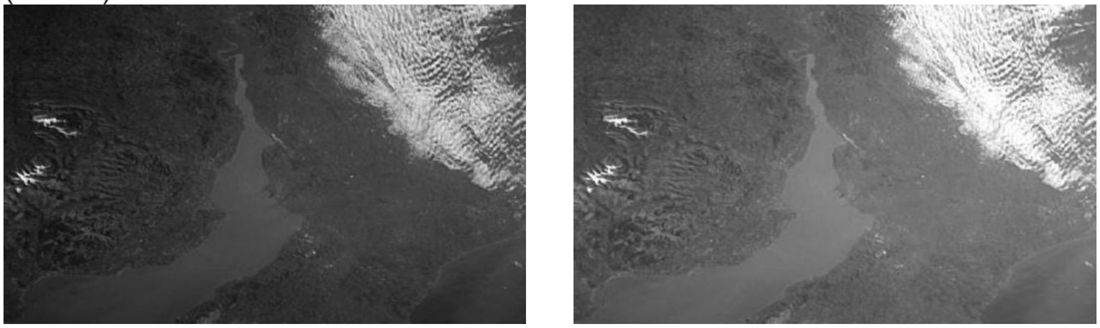


(a) Aerial image (b)-(d) results of applying the transformation with c=1and Y=3,4 and 5 respectively

Add a constant to all values

g’ = g + k

(k = 50)

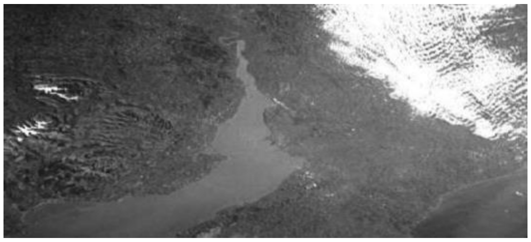


Contrast Adjustment

Scale all values by a constant

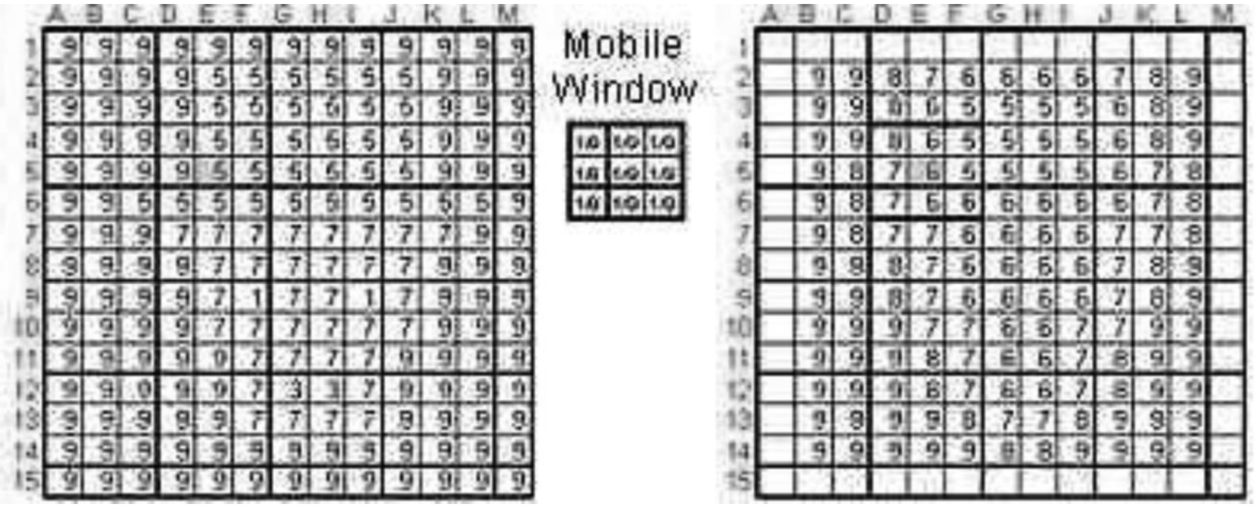
g’ = g*k

(k = 1.5)



Spatial Filtering

- Filters are commonly **used for** such things as edge **enhancement, noise removal, and the smoothing** of high frequency data
- The **principle** of the various filters is to **modify the numerical value of each pixel as a function of the neighboring pixels' values**.
- **For example**, if the value of each pixel is replaced by the average of its value and those of its eight neighbors the image is smoothed, that is to say, the finer details disappear and the image appears fuzzier.
- For example, the filtered value of the pixel located at E5 is (9*1/9) + (5*1/9) + (5*1/9) + (9*1/9) + (5*1/9) + (5*1/9) + (5*1/9) + (5*1/9) + (5*1/9) = 5.89, rounded up to 6.

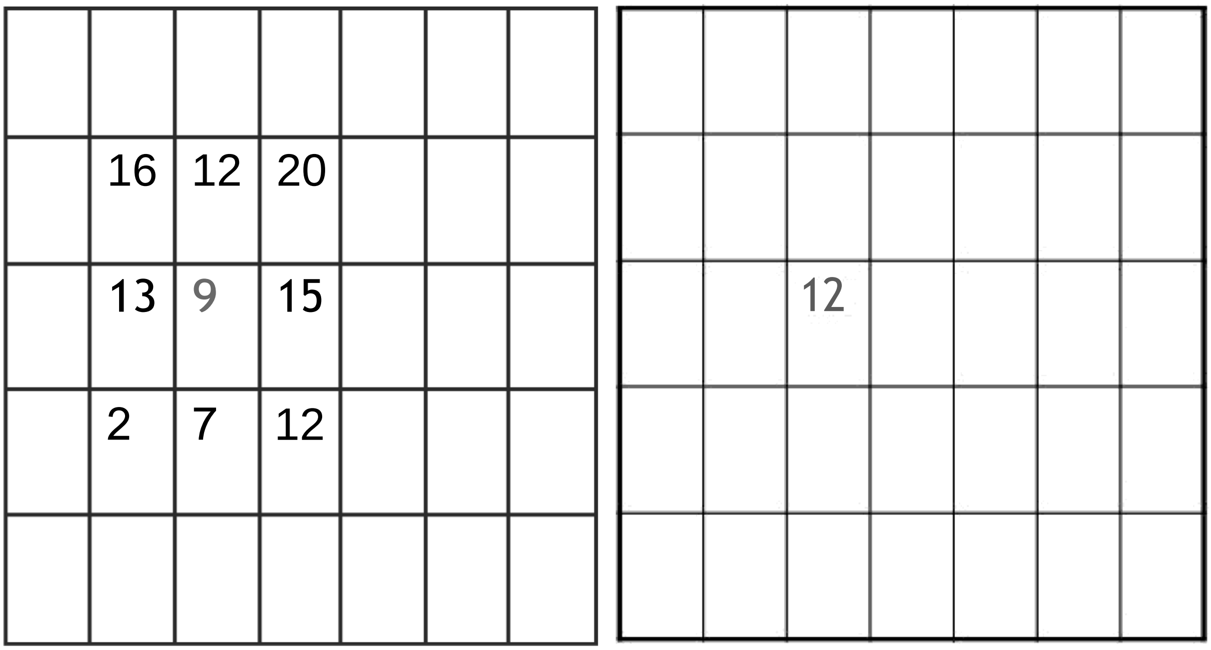


	A	B	C	D	E	F	G	H	I	J	K	L	M
1	9	9	9	9	9	9	9	9	9	9	9	9	9
2	9	9	9	9	5	5	5	5	5	5	9	9	9
3	9	9	9	9	5	6	5	6	5	6	9	9	9
4	9	9	9	9	5	5	5	6	5	5	9	9	9
5	9	9	9	9	5	5	5	6	5	5	9	9	9
6	9	9	5	5	5	5	5	6	5	5	5	5	9
7	9	9	9	7	7	7	7	7	7	7	7	9	9
8	9	9	9	9	7	7	7	7	7	7	9	9	9
9	9	9	9	9	7	1	7	7	1	7	9	9	9
10	9	9	9	9	7	7	7	7	7	7	9	9	9
11	9	9	9	9	9	7	7	7	7	9	9	9	9
12	9	9	9	9	9	7	3	3	7	9	9	9	9
13	9	9	9	9	9	7	7	7	7	9	9	9	9
14	9	9	9	9	9	9	9	9	9	9	9	9	9
15	9	9	9	9	9	9	9	9	9	9	9	9	9

Mobile Window

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

	A	B	C	D	E	F	G	H	I	J	K	L	M
1													
2		9	9	8	7	6	6	6	6	7	8	9	
3		9	9	8	6	5	5	5	5	6	8	9	
4		9	9	8	6	5	5	5	5	6	8	9	
5		9	8	7	6	5	5	5	5	6	7	8	
6		9	8	7	6	6	6	6	6	6	7	8	
7		9	8	7	7	6	6	6	6	7	7	8	
8		9	9	8	7	6	6	6	6	7	8	9	
9		9	9	8	7	6	6	6	6	7	8	9	
10		9	9	9	7	7	6	6	7	7	9	9	
11		9	9	9	8	7	6	6	7	8	9	9	
12		9	9	9	8	7	6	6	7	8	9	9	
13		9	9	9	9	8	7	7	8	9	9	9	
14		9	9	9	9	9	8	8	9	9	9	9	
15													



	16	12	20			
	13	9	15			
	2	7	12			

		12				

$$(16*1/9) + (12*1/9) + (20*1/9) + (13*1/9) + (9*1/9) + (15*1/9) + (2*1/9) + (7*1/9) + (12*1/9) = 12$$

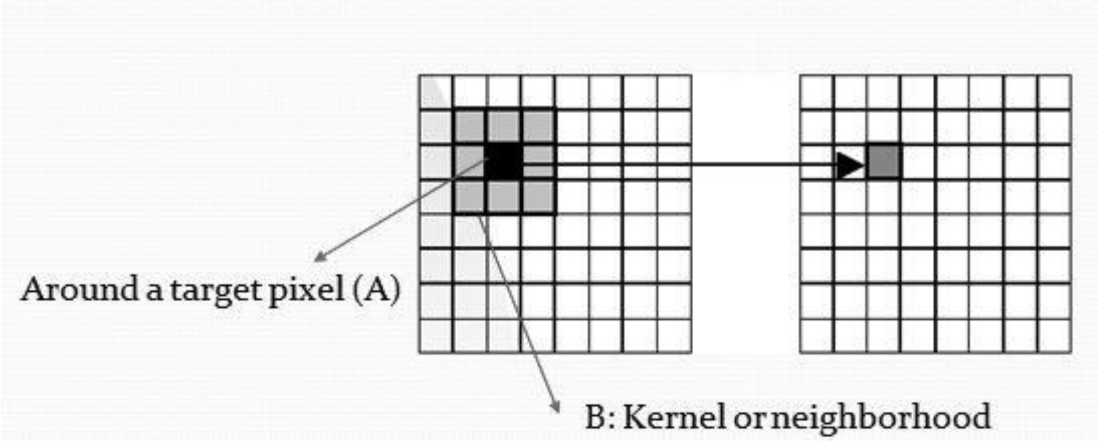
Spatial Feature Manipulation

- Spatial filters pass (emphasize) or suppress (de-emphasize) image data of various spatial frequencies
- Spatial frequency refers to the number of changes in brightness value, per unit distance, for any area within a scene
- Spatial frequency corresponds to image elements (both important details and noise) of certain size
- High spatial frequency 4rough areas
  - High frequency corresponds to image elements of smallest size
  - An area with high spatial frequency will have rapid change in digital values with distance (i.e. dense urban areas and street networks)

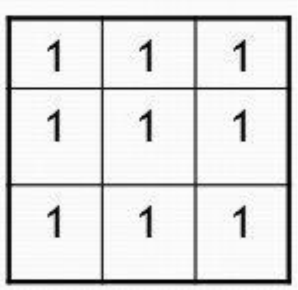
— Low spatial frequency 4 smooth areas

- Low frequency corresponds to image elements of (relatively) large size.
- An object with a low spatial frequency only changes slightly over many pixels and will have gradual transitions in digital values (i.e. a lake or a smooth water surface).

A **resampling technique** that **calculates the brightness value of a pixel** in a corrected image from the brightness value of the pixel nearest the location of the pixel in the input image

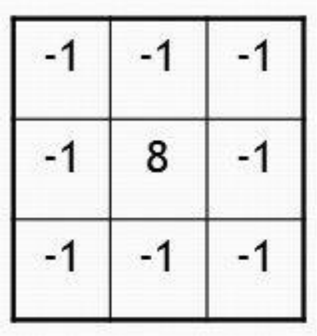


- Extract low frequency information (long wavelength)
- Suppress high frequency information (short wavelength)
- Low pass filter contains the same weights in each kernel element,
- Replacing the center pixel value with an average of the surrounding values
- Low pass filters are useful in smoothing an image, and reduce "salt and pepper" (speckle) noise from SAR images.



1	1	1
1	1	1
1	1	1

- Are used for removing , for example, stripe noise of low frequency (low energy, long short wavelengths)
- Filters that pass high frequencies (short wavelength)
- high pass filter uses a 3 x 3 kernel with a value of 8 center pixel and values of -1
- It can be used to enhance edges between different regions as well as to sharpen an image



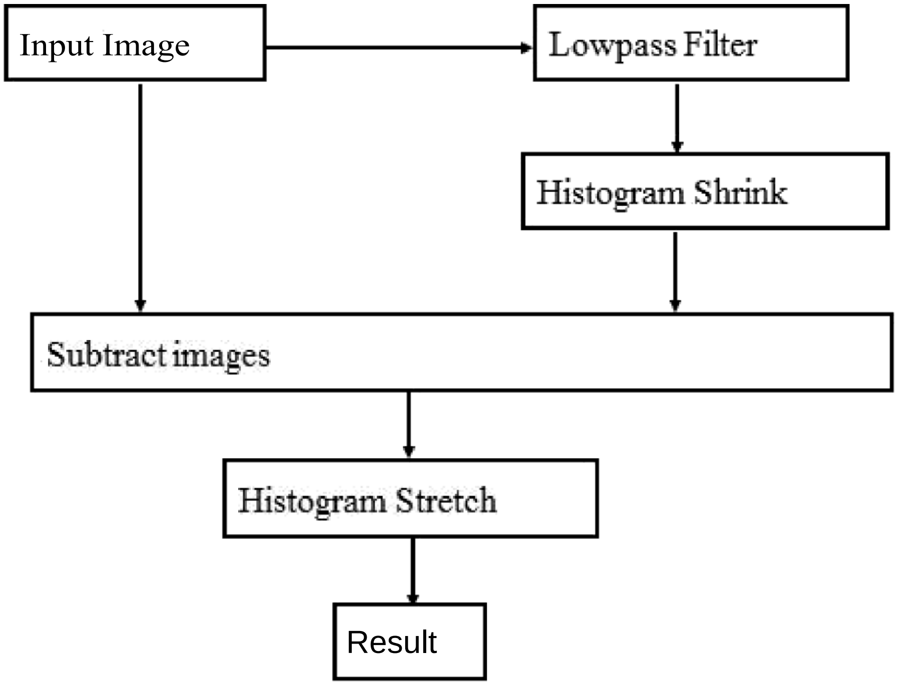
-1	-1	-1
-1	8	-1
-1	-1	-1

Maskinɡ

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

Unsharp Masking

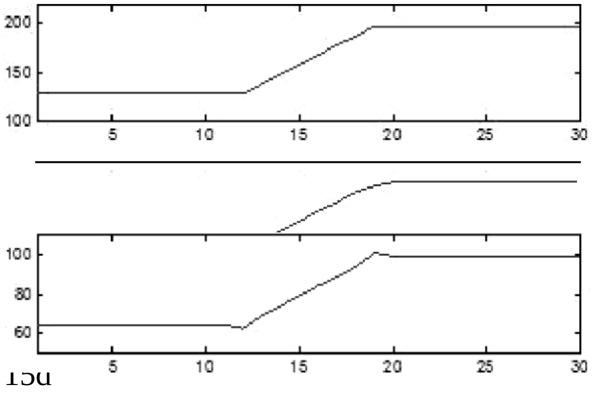
1. Originally a photographic sharpening technique
2. Superimpose a fraction of the blurred negative
3. Edge enhancement amplifies noise
4. Tradeoff between edge enhancement and noise enhancement
5. Equivalent to adding on a fraction of Laplacian



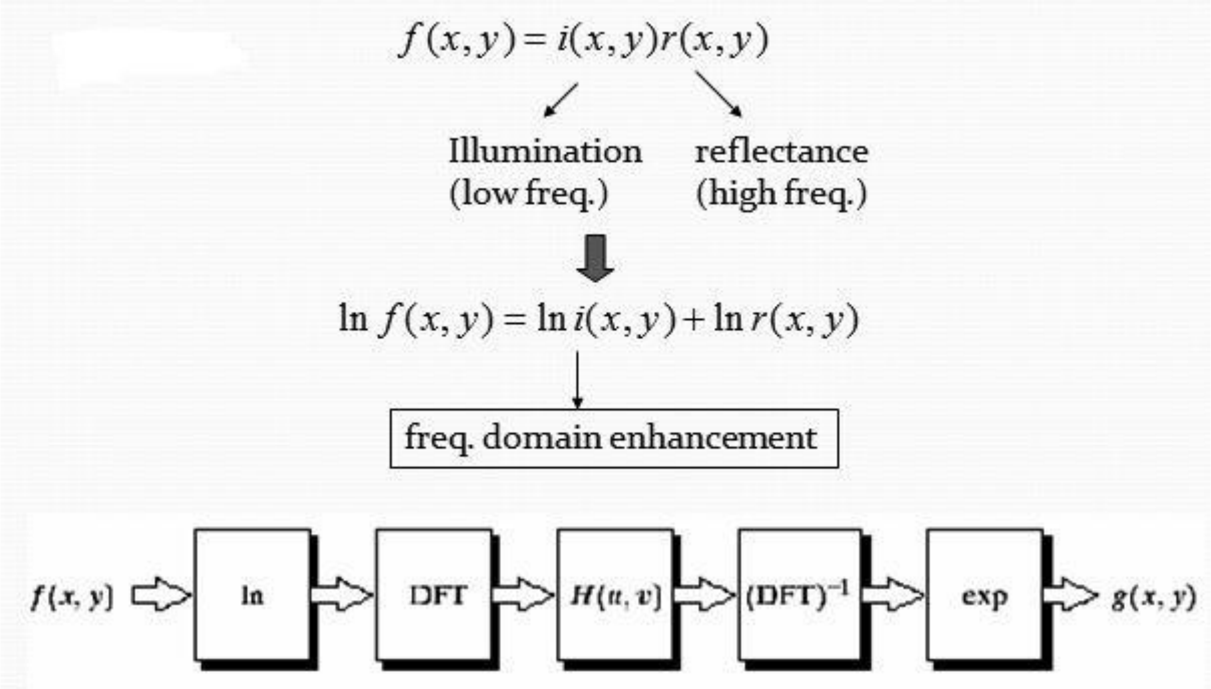
A(«)

$A(x)*B(x)$

(1+x)•A(x)—§xA(x)«@«)



Homomorphic filtering



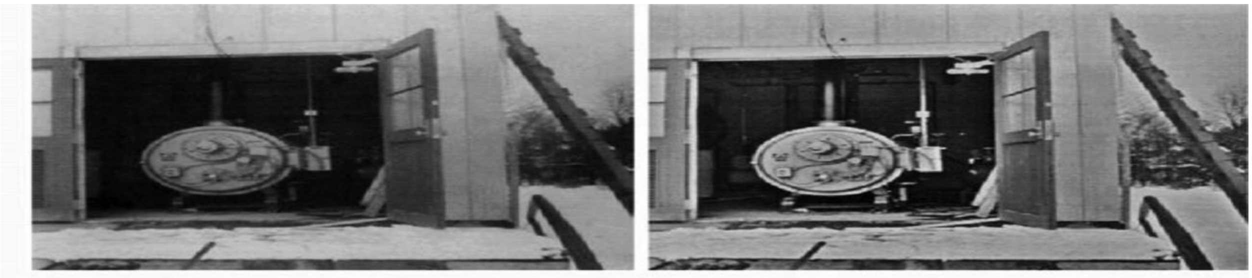

before
after

Image Restoration

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:

- ✓ during display mode during
- ✓ acquisition mode, or during
- ✓ processing mode

The degradations may be due to

- sensor noise
- blur due to camera misfocus
- relative object-camera motion
- random atmospheric turbulence
- others

In most of the existing image restoration methods we assume that the degradation process can be described using a mathematical model.

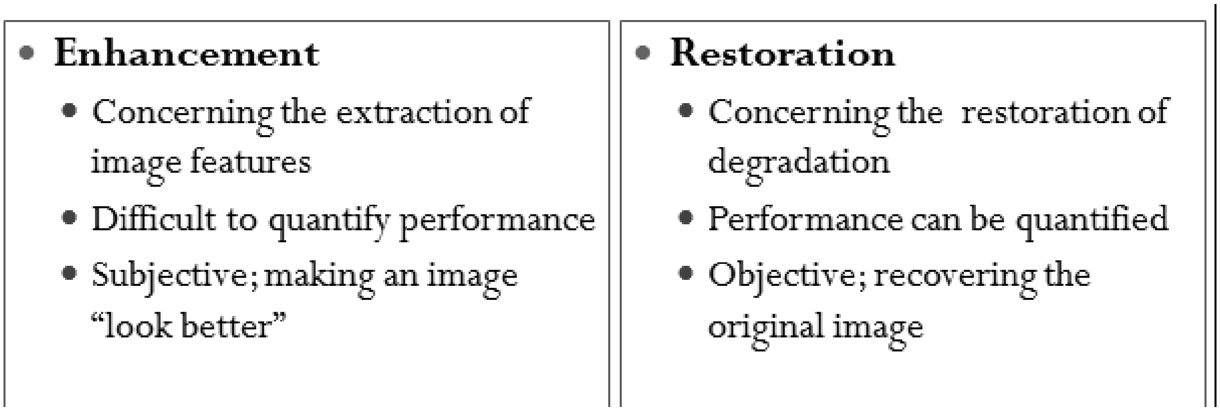
Objective of image restoration to recover a distorted image to the original form based on idealized models. The distortion is due to

- Image degradation in sensing environment e.g. random atmospheric turbulence
- Noisy degradation from sensor noise.
- Blurring degradation due to sensors e.g. camera motion or out-of-focus
- Geometric distortion e.g. earth photos taken by a camera in a satellite

Comparison of enhancement and restoration

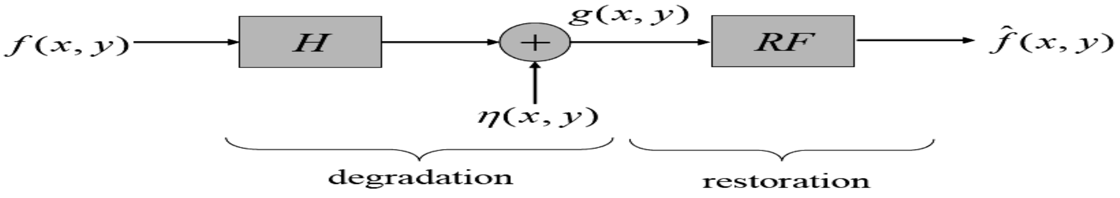
Image restoration differs from image enhancement in that the latter is concerned more with accentuation or extraction of image features rather than restoration of degradations.

Image restoration problems can be quantified precisely, whereas enhancement criteria are difficult to represent mathematically.



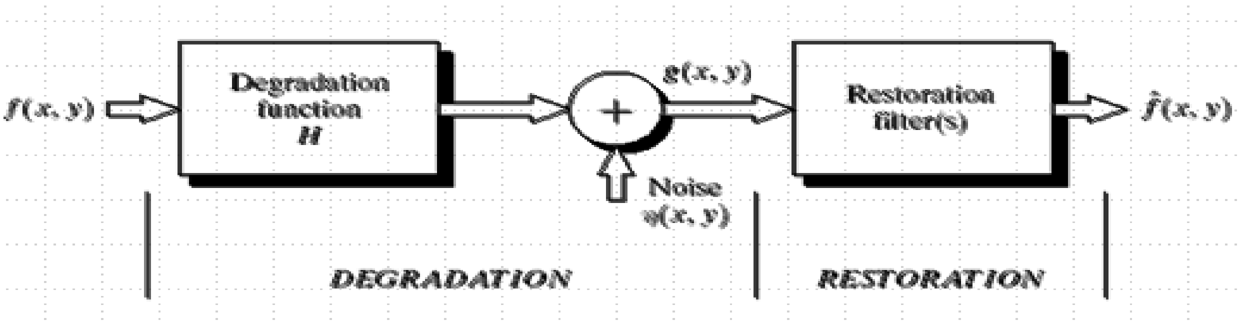
• Enhancement	• Restoration
• Concerning the extraction of image features	• Concerning the restoration of degradation
• Difficult to quantify performance	• Performance can be quantified
• Subjective; making an image "look better"	• Objective; recovering the original image

Image degradation / restoration model



- **When H is a LSI system**

$$g(x,y) = h(x,y) * f(x,y) + \eta(x,y)$$
$$G(u,v) = H(u,v)F(u,v) + N(u,v)$$



A general model of a simplified digital image degradation process

A simplified version for the image restoration process model is

y(i, j) = H[(i, j)]+ n(i, j)

where

y(i, j) the degraded image

(i, j) the original image

H an operator that represents the degradation process

n(i, j) the external noise which is assumed to be image-independent

Restoration methods could be classified as follows:

- deterministic' we work with sample by sample processing of the observed (degraded) image
- stochastic' we work with the statistics of the images involved in the process
- non-blind' the degradation process is known
- blind' the degradation process is unknown

- semi-blind' the degradation process could be considered partly known

the restoration is called as unconstrained restoration,

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

Where H = system operator. f^

= estimated input image.

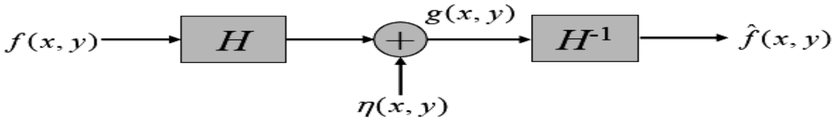
g = degraded image.

INVERSE FILTERING

- **When $H(u,v)$ is known, the simplest approach to restoration is direct inverse filtering**

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

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

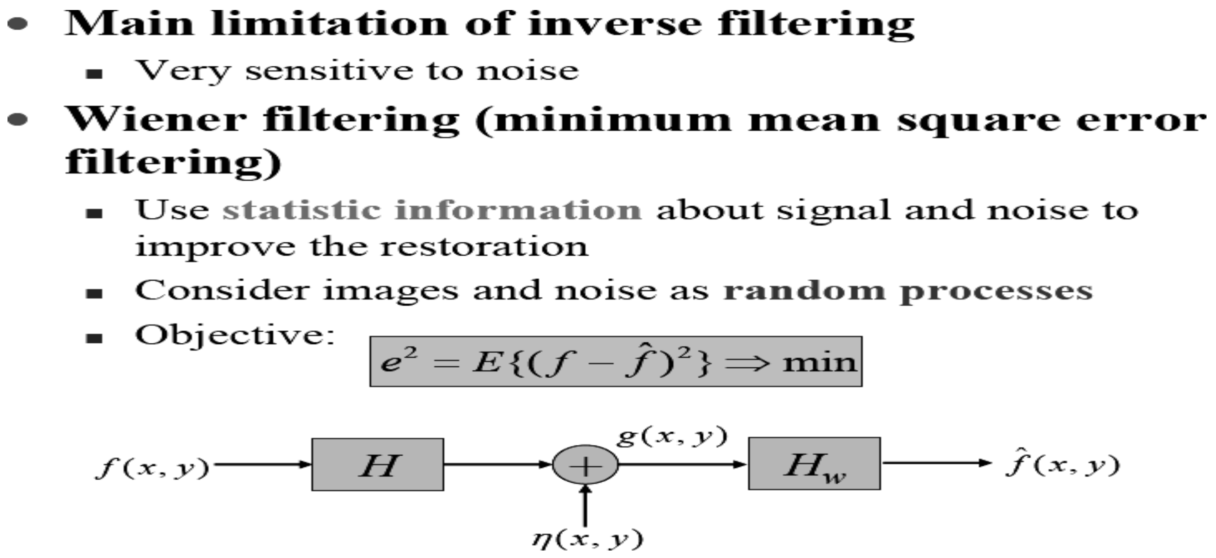


- **However, even if H is known completely, the undegraded image cannot be recovered exactly due to noise N**

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

- **Even worse when H has zero or very small values, N/H would dominate the estimated image**
- **One way to get around this problem is to limit the filter frequencies to values near the origin where H is large in general**

WIENER FILTERING



- **Main limitation of inverse filtering**
  - Very sensitive to noise
- **Wiener filtering (minimum mean square error filtering)**
  - Use **statistic information** about signal and noise to improve the restoration
  - Consider images and noise as **random processes**
  - Objective: $\boxed{e^2 = E\{(f-\hat{f})^2\} \Rightarrow \min}$

CONSTRAINED LEAST SQUARES (CLS) RESTORATION

It refers to a very large number of restoration algorithms. The problem can be formulated as follows. Minimize

$$J(\mathbf{f}) = \|\mathbf{n}(\mathbf{f})\|^2 = \|\mathbf{y} - \mathbf{H}\mathbf{f}\|^2$$

subject to

$$\|\mathbf{C}\,\mathbf{f}\|^2 < \varepsilon$$

where $f is a high pass filtered version of the image. The idea behind the above constraint is that the highpass version of the image contains a considerably large amount of noise! Algorithms of the above type can be handled using optimization techniques. $onstrained least squares ($LS) restoration can be formulated by choosing an f to minimize the Lagrangian

$$\min\left(\|\mathbf{y} - \mathbf{H}\mathbf{f}\|^2 + \alpha\|\mathbf{C}\mathbf{f}\|^2\right)$$

Typical choice for is the 2-; Laplacian operator given by $

$$\mathbf{C} = \begin{bmatrix} 0.00 & -0.25 & 0.00 \\ -0.25 & 1.00 & -0.25 \\ 0.00 & -0.25 & 0.00 \end{bmatrix}$$

α represents either a Lagrange multiplier or a fixed parameter known as regularisation parameter and it controls the relative contribution between the term $\|\mathbf{y} - \mathbf{H}\mathbf{f}\|^2$ and the term $\|\mathbf{C}\,\mathbf{f}\|^2$. The minimization of the above leads to the following estimate for the original image

$$\mathbf{f} = \left(\mathbf{H}^{\mathbf{T}}\mathbf{H} + \alpha\mathbf{C}^{\mathbf{T}}\mathbf{C}\right)^{-1}\mathbf{H}^{\mathbf{T}}\mathbf{y}$$

ITERATIVE DETERMINISTIC APPROACHES TO RESTORATION

They refer to a large class of methods that have been investigated extensively over the last decades. They possess the following advantages.

- There is no need to explicitly implement the inverse of an operator. The restoration process is monitored as it progresses. Termination of the algorithm may take place before convergence.
- The effects of noise can be controlled in each
- iteration. The algorithms used can be spatially adaptive.
- The problem specifications are very flexible with respect to the type of degradation.

Iterative techniques can be applied in cases of spatially varying or nonlinear degradations or in cases where the type of degradation is completely unknown (blind restoration). In general, iterative restoration refers to any technique that attempts to minimize a function of the form M(f) using an updating rule for the partially restored image.

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

SCHOOL OF ELECTRICAL AND ELECTRONICS

DEPARTMENT OF ELECTRONICS AND COMMMUNICATION ENGINEERING

UNIT - III

ADVANCED DIGITAL IMAGE PROCESSING – SECA7022

UNIT 3 IMAGE ANALYSIS AND REPRESENATION

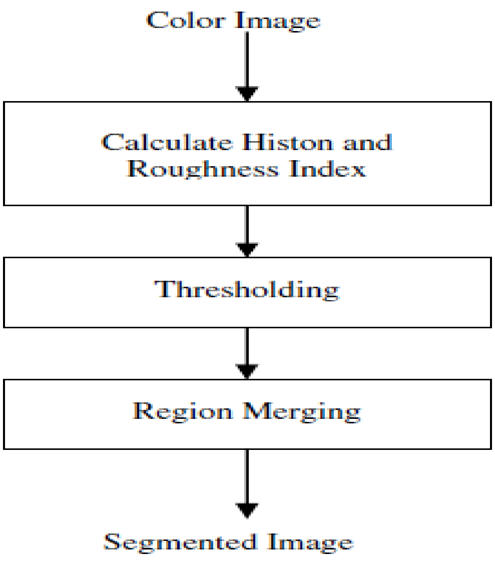
Introduction; Image Segmentation - Point, Line, Edge, Boundary Detection; Colour Image Segmentation;

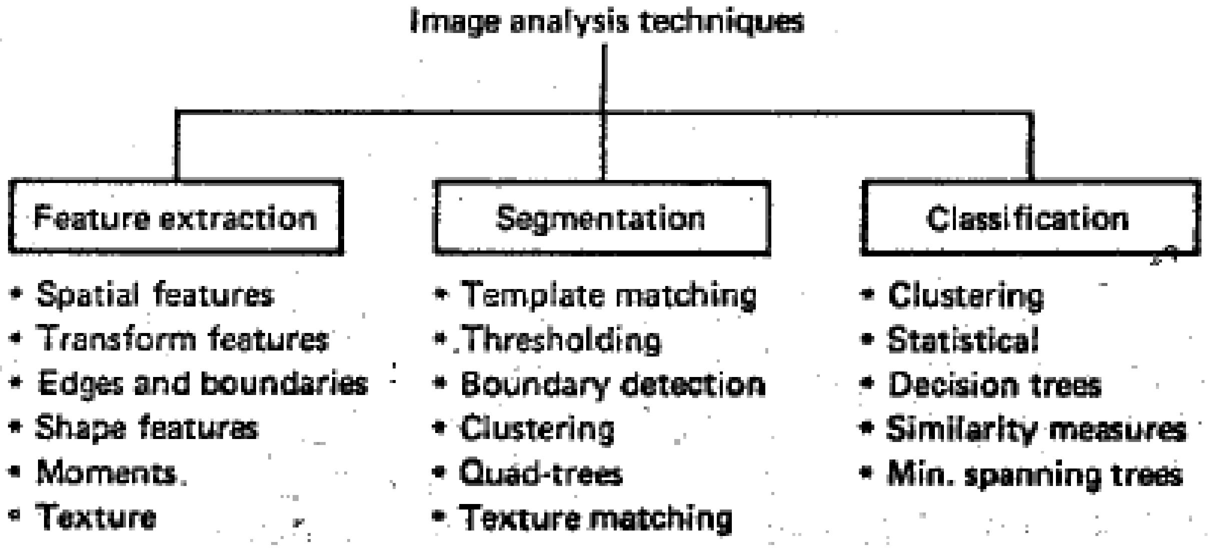
Thresholding- Basic Global Thresholding, Multiple Thresholding, Variable Thresholding; Region Based Segmentation;

Representation:
Chain codes, Signatures, Boundary segments, Skeletons, Description: Boundary Descriptors, Regional Descriptors.

Segmentation:

- In color image processing and its applications the great importance is attached to the techniques used for image segmentation.
- The quality of segmentation results have a big impact on the next steps of image processing, for example on the object recognition and tracking, the retrieval in image databases etc.
- The goal of image segmentation is partitioning of the image into homogeneous and connected regions without using an additional knowledge about objects in the image.
- The homogeneity of regions in color image segmentation involves in natural way colors and sometimes color textures [9]. In the segmented image the regions have, in contrast to single pixels, many interesting features like shape, texture etc.
- Color image segmentation: Rough-set theoretic approach
- Color-Based Image Salient Region Segmentation Using Novel Region Merging Strategy





Spatial features of an object may be characterized by its gray levels, their joint probability distributions, spatial distribution, and the like.

Amplitude Features

The simplest and perhaps the most useful features of an object are the amplitudes of its physical properties, such as reflectivity, transmissivity, tristimulus values (color), or multispectral response. For example, in medical X-ray images, the gray-level amplitude represents the absorption characteristics of the body masses and enables discrimination of bones from tissue or healthy tissue from diseased tissue. In infrared (IR) images amplitude represents temperature, which facilitates the segmentation of clouds from terrain .. In radar images, amplitude represents the *radar cross section,* which determines the size of the object being imaged. Amplitude features can be extracted easily by intensity window slicing or by the more general point transformations

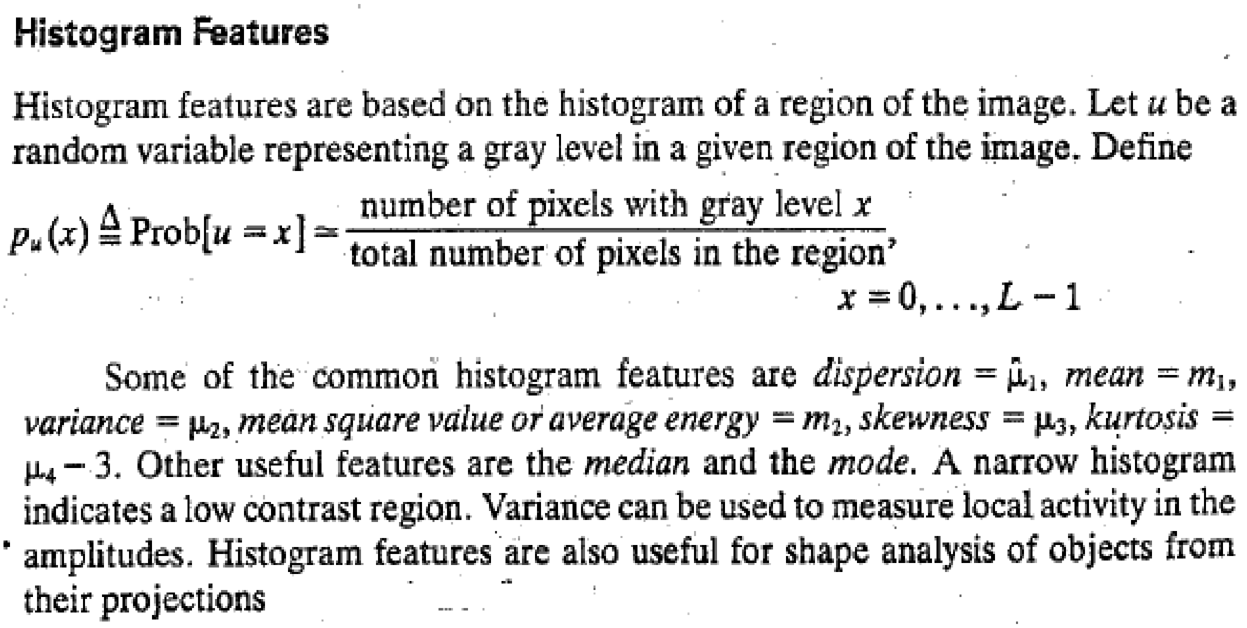
Amplitude features: e.g. the brightness levels can identify regions of interest in the image:

Amplitude features may be discriminative enough if intensity is enough to distinguish wanted info from the rest of the scene

=> defining the best parameters of the transformation for feature extraction – most difficult

=> amplitude feature space representation is not necessarily binary; just that unwanted parts of the scenes should be represented uniquely (i.e. black) in the feature space

=> sometimes adaptive thresholding/adaptive grey scale slicing is needed.



Histogram Features

Histogram features are based on the histogram of a region of the image. Let u be a random variable representing a gray level in a given region of the image. Define

$$p_u(x) \triangleq \text{Prob}[u = x] = \frac{\text{number of pixels with gray level } x}{\text{total number of pixels in the region}}, \quad x = 0, \dots, L-1$$

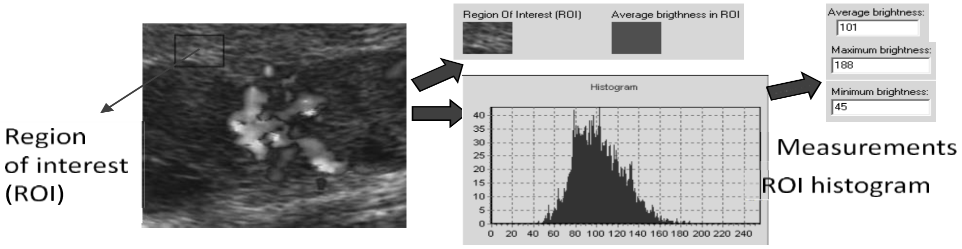
Some of the common histogram features are *dispersion* $= \hat{\mu}_1$, *mean* $= m_1$, *variance* $= \mu_2$, *mean square value or average energy* $= m_2$, *skewness* $= \mu_3$, *kurtosis* $= \mu_4 - 3$. Other useful features are the *median* and the *mode*. A narrow histogram indicates a low contrast region. Variance can be used to measure local activity in the amplitudes. Histogram features are also useful for shape analysis of objects from their projections

Histogram based features

Local histogram = a local statistical description of the image;

If u = an image pixel; x=a grey level => p_u(x)=the probability of appearance of the grey level x in the image region = a value in the **normalized histogram**

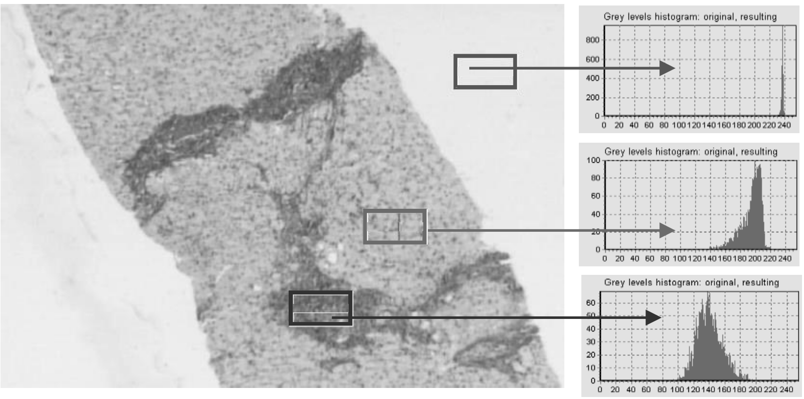
=> One can compute: the standard deviation; the entropy; the median; percentiles, of $p_u(x)$.



Average brightness: 101

Maximum brightness: 188

Minimum brightness: 45



Tissue of interest is well
discriminated from the
by the
of the

microscopic slide
standard deviation
local histogram

Transform-feature extraction techniques are also important when the source data originates in the transform coordinates. For example, in optical and optical-digital (hybrid) image analysis applications, the data can be acquired directly in the Fourier domain for real-time feature extraction in the focal plane.

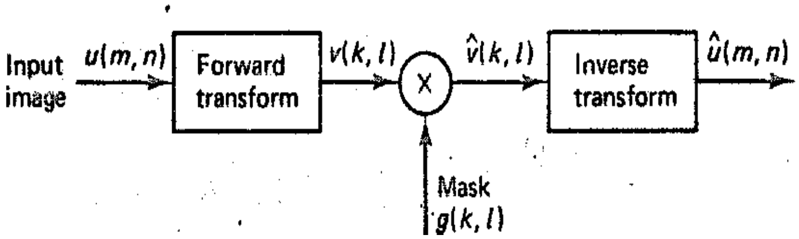


Image transforms provide the frequency domain information in the data. Transform features are extracted by zonal-filtering the image in the selected transform space (Fig. 9.4). The zonal filter, also called the feature mask, is simply a slit or an aperture.

Generally, the high-frequency features can be used for edge and boundary detection, and angular slits can be used for detection of orientation.

A combination of an angular slit with a bandlimited low-pass, band-pass or high-pass filter can be used for discriminating periodic or quasiperiodic textures. Other transforms, such as Haar and Hadamard, are also potentially useful for feature extraction.

EDGE DETECTION

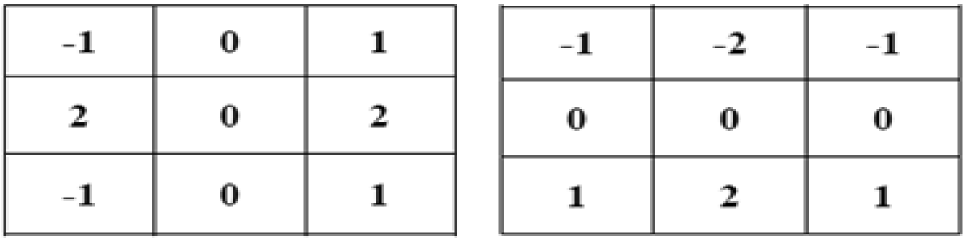
- Edge detection significantly **reduces the amount of data and filters out useless information**, while **preserving the important structural properties in an image**.
- Edges are **boundaries between different textures**.
- Edge also can be defined as **discontinuities in image intensity from one pixel to another**.
- The edges for an image are always the important characteristics that offer an **indication for a higher frequency**.
- Detection of edges for an image may help for **image segmentation, data compression,** and also help for well matching, such as **image reconstruction** and so on.
- Edge detection is **difficult in noisy images**, since both the noise and the edges contain high-frequency content.

Edge Detectors

- Robert
- Sobel
- Prewitt
- Laplacian and
- Canny

Sobel Edge Detector

- The salient features of the sobel edge detectors are listed as follows
- It has two 3x3 convolution kernels or masks, Gx and Gy, as shown in fig 1. both Gx and Gy can be joined together to find the absolute magnitude and the orientation of the gradient.



-1	0	1
2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

Fig. 1 3x3 convolution kernels or masks, Gx and Gy

$$|G| = \sqrt{(G_x^{\ 2} + G_y^{\ 2})}$$

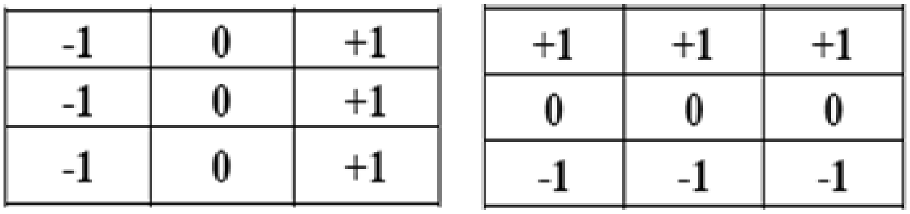
- Used to detect edges along the horizontal and vertical axis
- Based on convolving the image with a small, integer valued filter (3×3 kernel) in both horizontal and vertical direction. So this detector requires less computation
- The sobel edge detection masks search for edges in horizontal and vertical directions and then combine this information into a single metric
- In this, image intensity is calculate at every pixel (pixel) and presenting the direction of the maximum possible change from white (light) to black (dark) and the rate of change in that direction.
- Simplicity
- Detection of edges and their orientations

- Sensitivity to noise
- Inaccurate

PREWITT EDGE DETECTOR

The salient features of the Prewitt Edge Detector are listed as follows

- This edge detector is very similar of sobel operator
- Simplicity
- Detection of horizontal and vertical edges and their orientations
- Sensitivity to noise
- Inaccurate
- The kernel used in the Prewitt detector is shown in fig 2.



-1	0	+1
-1	0	+1
-1	0	+1

+1	+1	+1
0	0	0
-1	-1	-1

Fig. 2 Kernel of prewitt detector

LAPLACIAN OF GAUSSIAN EDGE DETECTOR

The salient features of the Laplacian of Gaussian edge detector is listed as follows

- Proposed by Marr and Hildreth in 1980
- This is a **combination of the Gaussian filtering and Laplacian gradient operator**.
- **Laplacian gradient operator determines the regions where the rapid intensity changes**. So it is best suit for edge detection.
- After the laplacian process is over, the **image is given to Gaussian filter to remove the noise pixels.**
- The laplacian gradient of an image is given by

$$L(x,y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2}$$

Where I(x, y) = pixel intensity values in image

- In this, an image is divided where the intensity varies to detect the edges effectively.
- It is very difficult to find the orientation of edges due to laplacian filter.
- Used to determine exact location of edges
- Does not produce good result where the gray level function varies (corners, curves)
- Not useful for finding the orientation of edges

The two 3X3 kernels used for Laplacian edge detector is shown in fig 3.

$$X = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$Y = \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

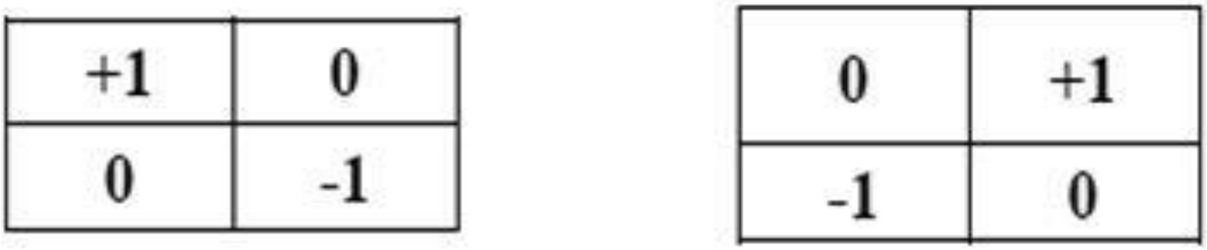
Fig. 3 kernel of laplacian edge detector

ROBERTS EDGE DETECTOR

The salient features of the Roberts Edge Detector is listed as follows

- This detector has two 2x2 convolution kernels in which one of the kernels is rotated by 90^0
- Fast computation
- Performs a simple and fast two dimensional spatial gradient measurement on an image.
- Each point (pixels) in the output image represents the expected absolute magnitude of the spatial gradient of the input Image at that point.

Convolution mask is shown in fig 4.



+1	0
0	-1

0	+1
-1	0

Fig. 4 kernel of Roberts's edge detector

$$\Delta f = grad(f) = \begin{bmatrix} Gx \\ Gy \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

The Magnitude of the vector is defined as

$$M(x,y) = \sqrt{(G_x{}^2 + G_y{}^2)}$$

CANNY EDGE DETECTOR

- The canny edge detector can be used to identify a wide range of real edges in images.
- The detector eliminates the unwanted noise pixels by the process of smoothening edges in an image because noise pixels generate false edges. \
- In this edge detection, the signal to noise ratio is improved as compared to other methods.
- This is the reason why this detector is extensively used for edge detection in image processing.

The procedure to find edges in images is explained as follows.

- Initially the image is smoothened using a suitable filter such as mean filter, Gaussian filter etc., to reduce the effect of noise.

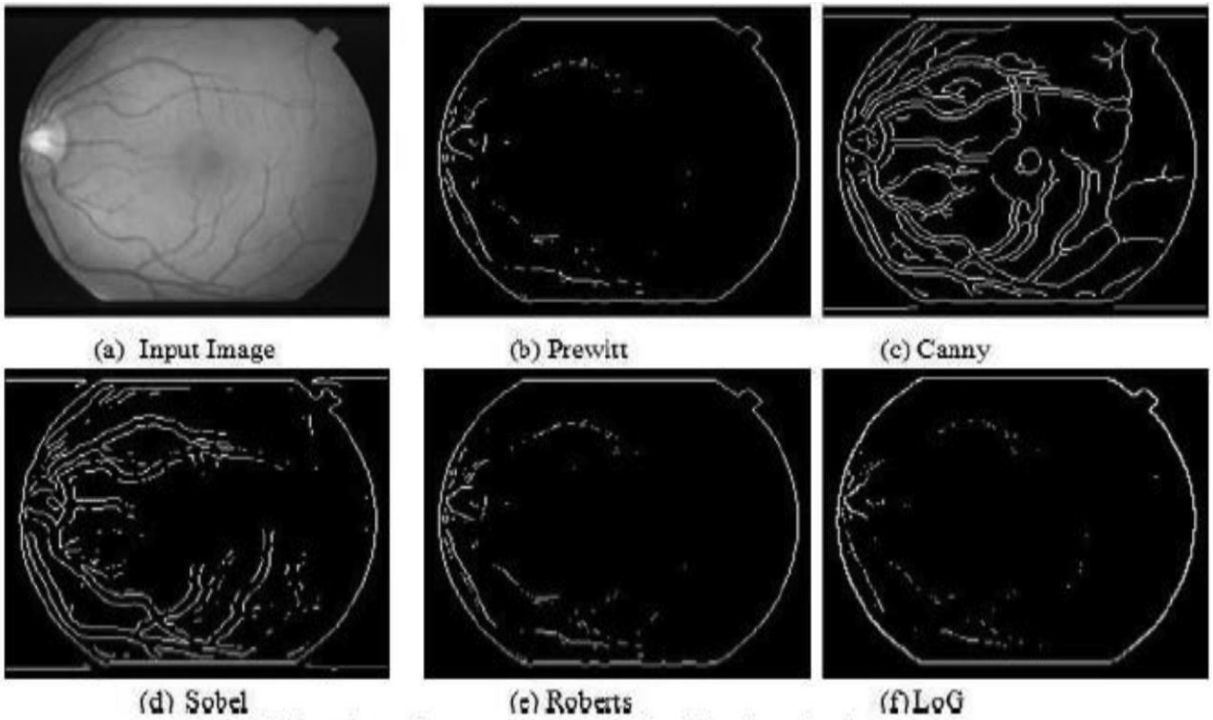
- Then local gradient and edge direction is calculated for every point. This point has a maximum strength in the direction of the gradient.

- These edge points give rise to ridges in the gradient magnitude image.

- The edge detector tracks along the top of these ridges and make all the pixels to zero that are not actually on the top of the ridge. No a thin line is generated in the output.

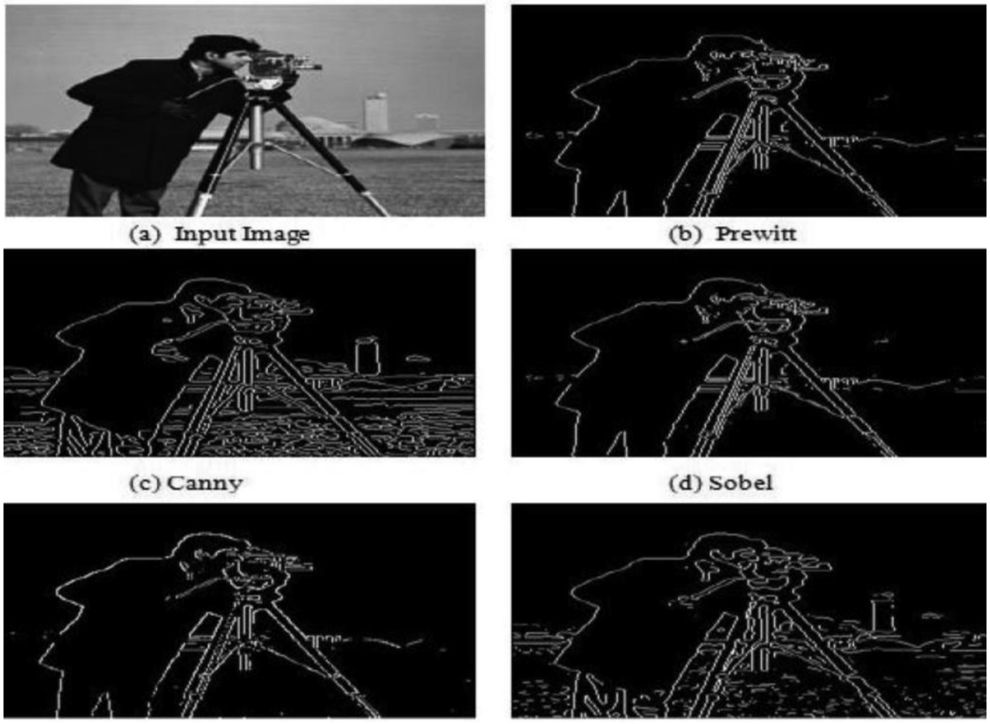
  - These ridge pixels are threshold using two threshold values: upper threshold (T2) and lower threshold (T1).
  - Ridge pixels are classified as strong edge pixels if ridge pixel values are greater than upper threshold (T2) and ridge pixels are classified as weak edge pixels if ridge pixel values are between the lower threshold (T1) and the upper threshold (T2).
  - Finally, the edges in the image are linked by integrating the weak pixels which are connected to the strong pixels.

EDGE DETECTORS EXAMPLES


(a) Input Image (b) Prewitt (c) Canny

(d) Sobel (e) Roberts (f) LoG

Fig. 6.Edge detection result for medical (retina fundus.jpg) image



(a) Input Image

(b) Prewitt

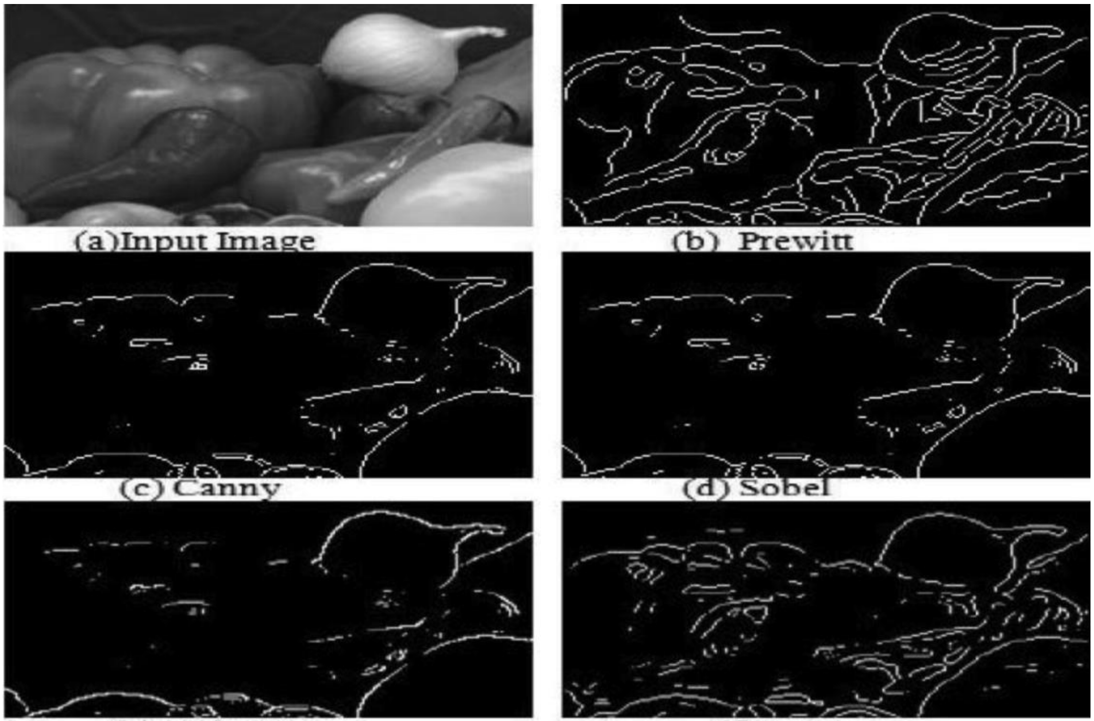
(c) Canny

(d) Sobel

(e) Roberts

(f) LoG

Fig. 7.Edge detection result for cameraman.tiff image



(a)Input Image

(b) Prewitt

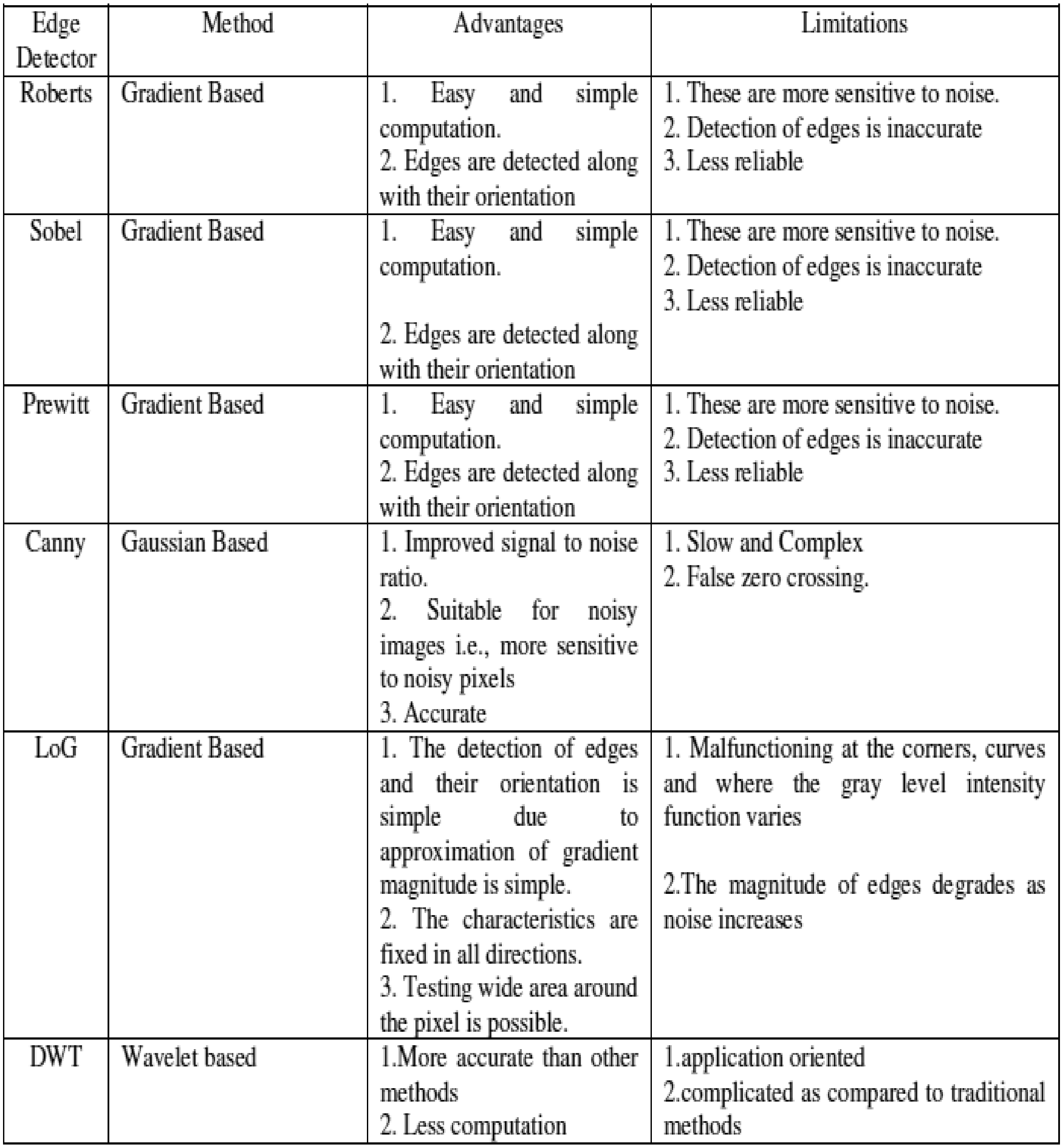
(c) Canny

(d) Sobel

(e) Roberts

(f) LoG

Fig. 8.Edge detection result for onion.png



Edge Detector	Method	Advantages	Limitations
Roberts	Gradient Based	1. Easy and simple computation. 2. Edges are detected along with their orientation	1. These are more sensitive to noise. 2. Detection of edges is inaccurate 3. Less reliable
Sobel	Gradient Based	1. Easy and simple computation. 2. Edges are detected along with their orientation	1. These are more sensitive to noise. 2. Detection of edges is inaccurate 3. Less reliable
Prewitt	Gradient Based	1. Easy and simple computation. 2. Edges are detected along with their orientation	1. These are more sensitive to noise. 2. Detection of edges is inaccurate 3. Less reliable
Canny	Gaussian Based	1. Improved signal to noise ratio. 2. Suitable for noisy images i.e., more sensitive to noisy pixels 3. Accurate	1. Slow and Complex 2. False zero crossing.
LoG	Gradient Based	1. The detection of edges and their orientation is simple due to approximation of gradient magnitude is simple. 2. The characteristics are fixed in all directions. 3. Testing wide area around the pixel is possible.	1. Malfunctioning at the corners, curves and where the gray level intensity function varies 2.The magnitude of edges degrades as noise increases
DWT	Wavelet based	1.More accurate than other methods 2. Less computation	1.application oriented 2.complicated as compared to traditional methods

BOUNDARY EXTRACTION AND REPRESENTATION

Boundaries are linked edges that characterize the shape of an object. They are useful in computation of geometry features such as size or orientation.

Connectivity

Conceptually, boundaries can be found by tracing the connected edges. On a rectangular grid a pixel is said to be *four-* or *eight-connected* when it has the same properties as one of its nearest four or eight neighbors, respectively (Fig. 9.14).

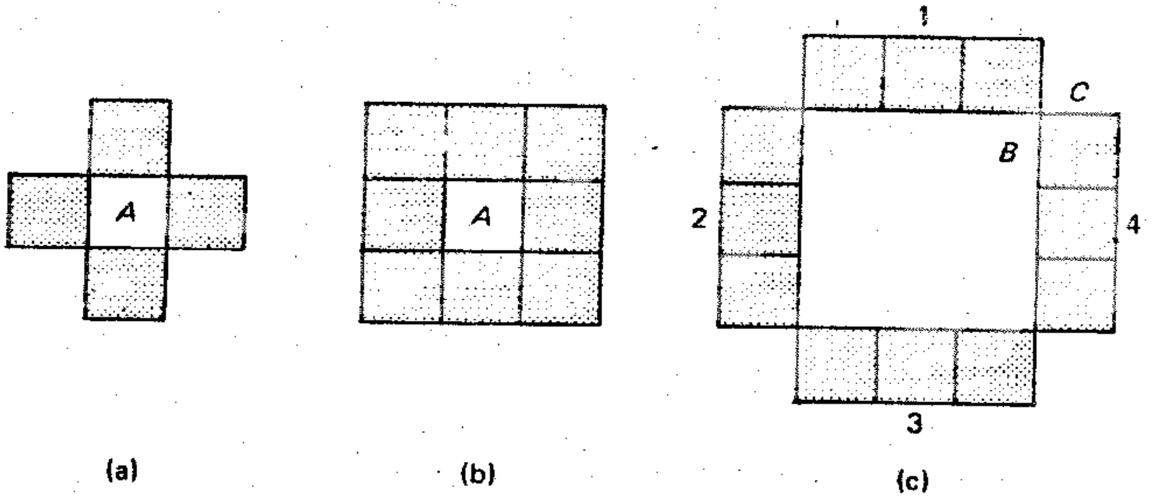
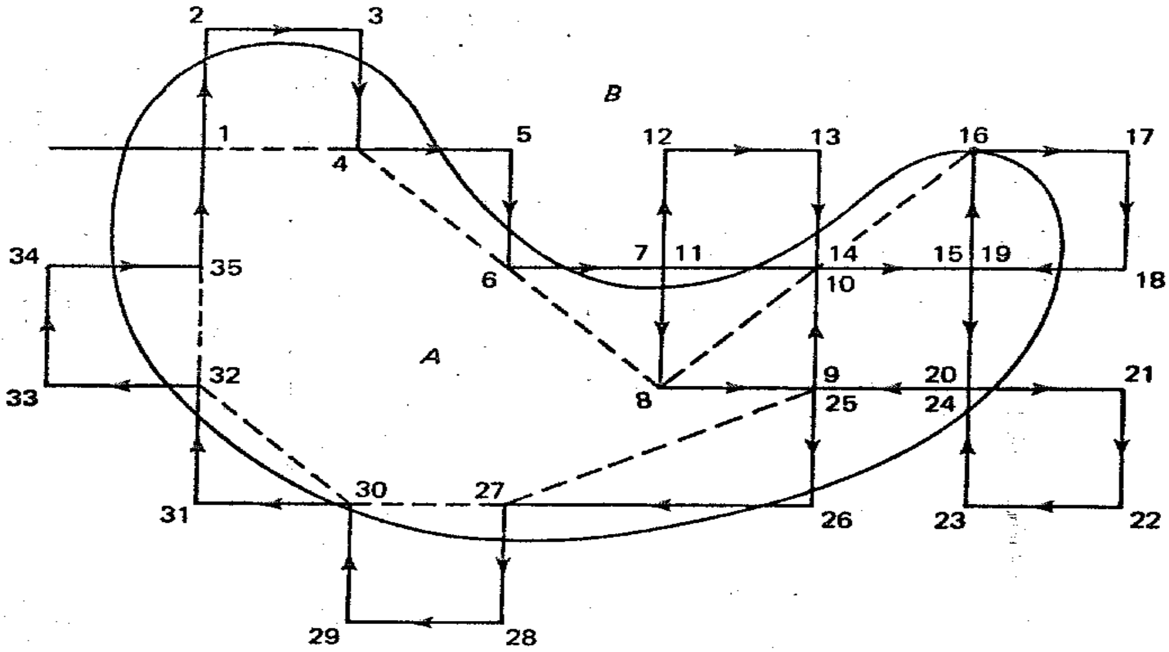


Figure 9.14 Connectivity on a rectangular grid. Pixel A and its (a) 4-connected and (b) 8-connected neighbors; (c) connectivity paradox: "Are B and C connected?"

A modified version of this contour-following method is called the *crack-following algorithm* . In that algorithm each pixel is viewed as having a square-shaped boundary, and the object boundary is traced by following the edge-pixel boundaries.

As the name suggests, contour-following algorithms trace boundaries by ordering successive edge points. A simple algorithm for tracing closed boundaries in binary images is shown in Fig. . This algorithm can yield a coarse contour, with some of the boundary pixels appearing twice.



Algorithm

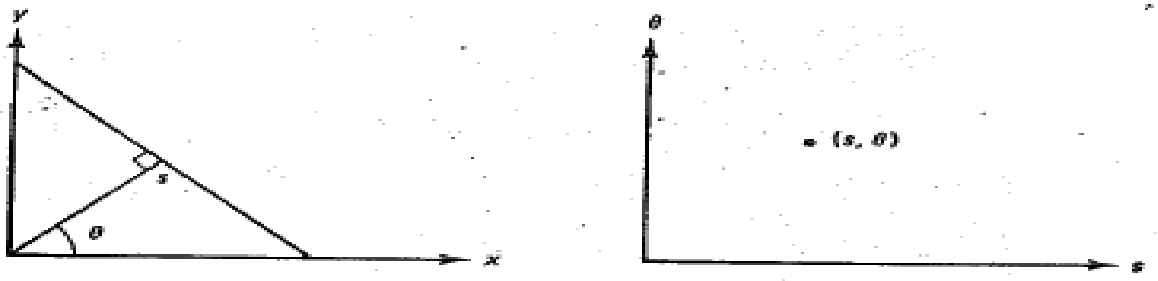
1. Start inside A (e.g., 1)
2. Turn left and step to next pixel if in region A, (e.g., 1 to 2), otherwise turn right and step (e.g., 2 to 3)
3. Continue until arrive at starting point 1

Figure 9.15 Contour following in a binary image.

Boundary Extraction and Representation using Hough Transform

A straight line at a distance s and orientation θ (Fig. 9.19a) can be represented as

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



(a) Straight line

(b) Hough transform

Figure The Hough transform.

The Hough transform of this line is just a point in the (s, θ) plane; that is, all the points on this line map into a single point This fact can be used to detect straight lines in a given set of boundary points. Suppose we are given boundary points (x_i, y_i), $i = 1, \dots, N$. For some chosen quantized values of parameters s and θ, map each (x_i, y_i) into the (s, θ) space and count $C(s, \theta)$, the number of edge points that map into the location (s, θ), that is, set

$$C(s_k, \theta_l) = C(s_k, \theta_l) + 1, \quad \text{if} \quad x_i \cos\theta + y_i \sin\theta = s_k \quad \text{for} \quad \theta = \theta_l$$

Then the local maxima of $C(s, \theta)$ give the different straight line segments through the edge points.

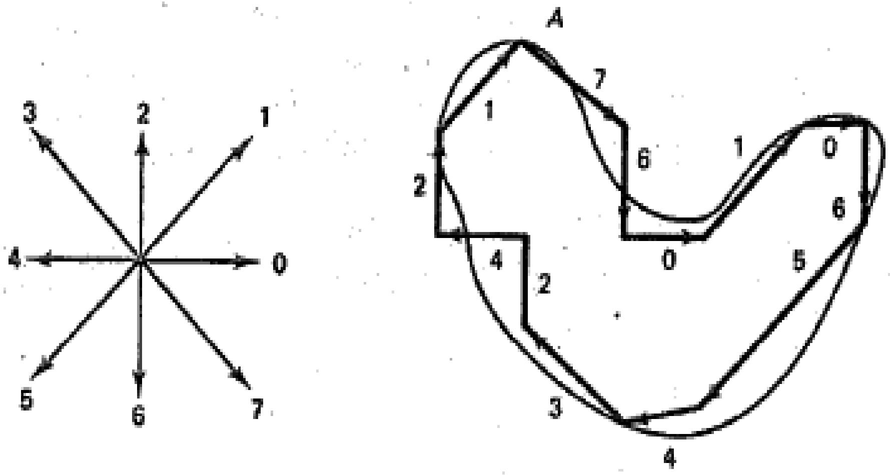
Boundary Representation

Proper representation of object boundaries is important for analysis and synthesis of shape. Shape analysis is often required for detection and recognition of objects in a scene. Shape synthesis is useful in computer-aided design (CAD) of parts and assemblies, image simulation applications such as video games, cartoon movies,

environmental modeling of aircraft-landing testing and training, and other computer graphics problems.

Chain Codes

In chain coding the direction vectors between successive boundary pixels are encoded. For example, a commonly used chain code employs eight directions, which can be coded by 3-bit code words. Typically, the chain code contains the start pixel address followed by a string of code words. Such codes can be generalized by increasing the number of allowed direction vectors between successive boundary pixels..



Algorithm:

1. Start at any boundary pixel, *A*.
2. Find the nearest edge pixel and code its orientation. In case of a tie, choose the one with largest (or smallest) code value.
3. Continue until there are no more boundary pixels.

(b) Contour

Boundary pixel orientations: (A), 76010655432421

Chain code: A 111 110 000 001 000 110 101 101 110 011 010 100 010 001

Figure Chain code for boundary representation.

Fourier Descriptors

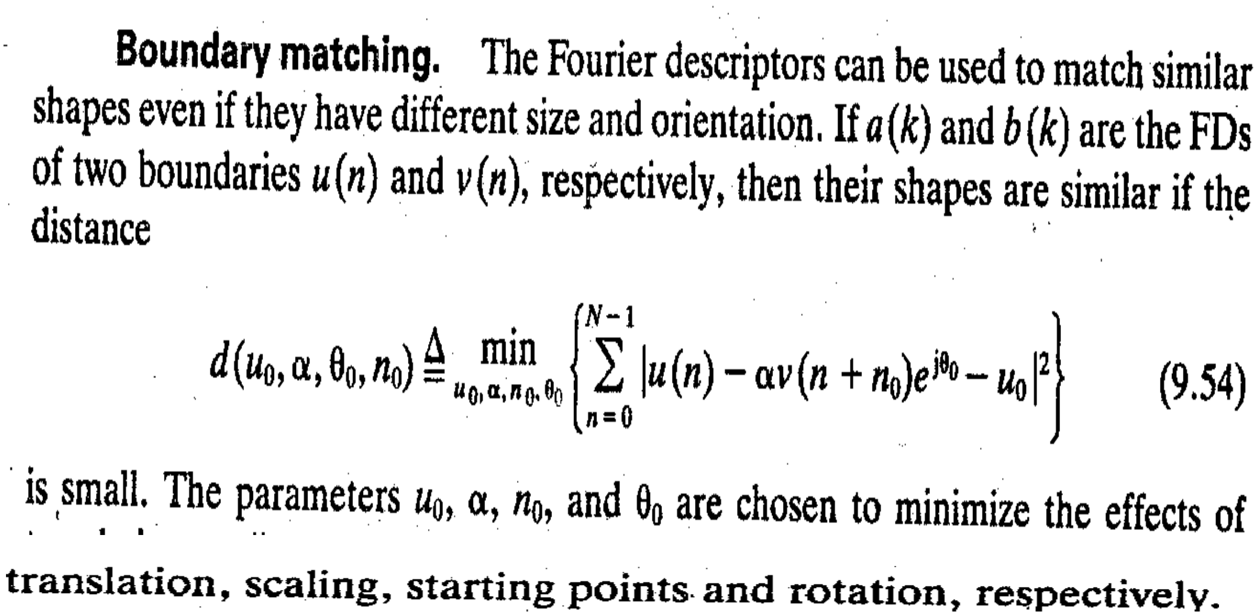
Once the boundary trace is known, we can consider it as a pair of waveforms $x(t), y(t)$. Hence any of the traditional one-dimensional signal representation techniques can be used. For any sampled boundary we can define

$$u(n) \overset{\Delta}{=} x(n) + \mathrm{j}y(n), \qquad n = 0, 1, \ldots, N-1 \tag{9.49}$$

which, for a closed boundary, would be periodic with period N. Its DFT representation is

$$\begin{aligned} u(n) &\overset{\Delta}{=} \frac{1}{N}\sum_{k=0}^{N-1} a(k)\exp\left(\frac{\mathrm{j}2\pi kn}{N}\right), \qquad 0 \le n \le N-1 \\ a(k) &\overset{\Delta}{=} \sum_{n=0}^{N-1} u(n)\exp\left(\frac{-\mathrm{j}2\pi kn}{N}\right), \qquad 0 \le k \le N-1 \end{aligned} \tag{9.50}$$

The complex coefficients $a(k)$ are called the *Fourier descriptors* (FDs) of the boundary. For a continuous boundary function, $u(t)$, defined in a similar manner to (9.49), the FDs are its (infinite) Fourier series coefficients. Fourier descriptors have been found useful in character recognition problems [32].



Boundary matching. The Fourier descriptors can be used to match similar shapes even if they have different size and orientation. If $a(k)$ and $b(k)$ are the FDs of two boundaries $u(n)$ and $v(n)$, respectively, then their shapes are similar if the distance

$$d(u_0,\alpha,\theta_0,n_0) \triangleq \min_{u_0,\alpha,n_0,\theta_0} \left\{ \sum_{n=0}^{N-1} \left|u(n) - \alpha v(n+n_0)e^{j\theta_0} - u_0\right|^2 \right\} \qquad (9.54)$$

is small. The parameters u_0, α, n_0, and θ_0 are chosen to minimize the effects of translation, scaling, starting points and rotation, respectively.

- Stochastic textures. Texture images of stochastic textures look like noise: colour dots that are randomly scattered over the image, barely specified by the attributes minimum and maximum brightness and average colour. Many textures look like stochastic textures when viewed from a distance. An example of a stochastic texture is roughcast.

- $tructured textures. These textures look like somewhat regular patterns. An example of a structured texture is a stonewall or a floor tiled with paving stones.

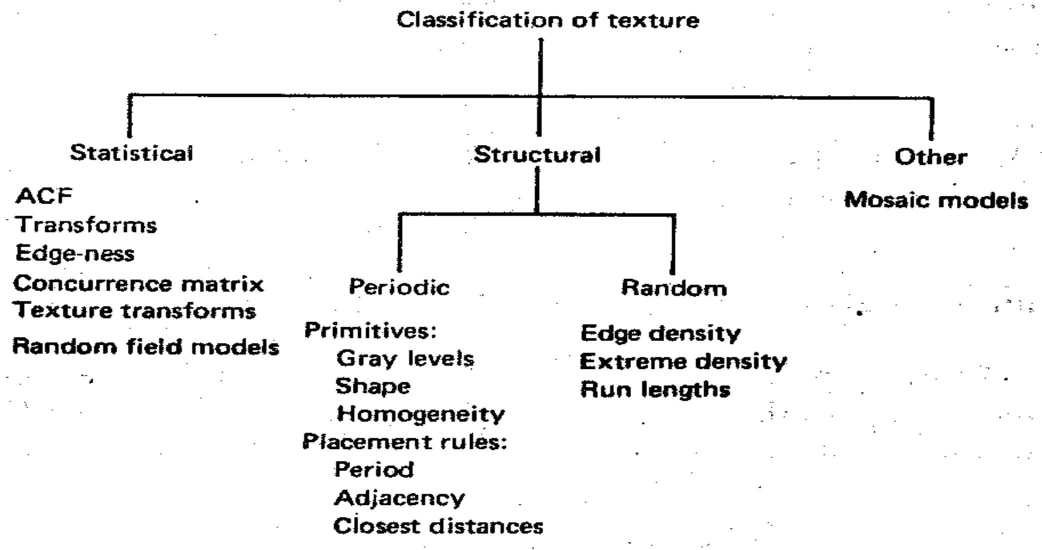


IMAGE SEGMENTATION

There are many definitions:

- Segmentation subdivides an image into its constituent regions or objects (Gonzales, pp567)
- Segmentation is a process of grouping together pixels that have similar attributes (Efford , pp250)
- Image Segmentation is the process of partitioning an image into non intersecting regions such that each region is homogeneous and the union of no two adjacent regions is homogeneous (Pal, pp1277)

Segmentation is typically associated with pattern recognition problems. 't is considered the first phase of a pattern recognition process and is sometimes also referred to as object isolation.

Why segmentation is difficult?

It can be difficult for many reasons:

- Non- uniform illumination
- No control of the environment
- Inadequate model of the object of interest
- Noise

Segmentation algorithms have been used for a variety of applications. Some examples are :

- Optical character recognition(OCR)
- Automatic Target Acquisition
- Colorization of Motion Pictures
- Detection and measurement of bone, tissue, etc, in medical images.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF ELECTRICAL AND ELECTRONICS

DEPARTMENT OF ELECTRONICS AND COMMMUNICATION ENGINEERING

UNIT - IV

ADVANCED DIGITAL IMAGE PROCESSING – SECA7022

UNIT 4 MORPHOLOGICAL PROCESSING & COMPRESSION

Morphological Image Processing – Logic Operations involving Binary

Images; Dilation and Erosion;

Opening and Closing;

Basic Morphological Algorithms – Boundary Extraction, Region Filling, Thickening,

Thinning; Image Compression – Compression Model, Huffman Coding, Arithmetic Coding.

Basic Morphology Concepts

- Mathematical Morphology is based on the algebra of non-linear operators operating on object shape and in many respects supersedes the linear algebraic system of convolution.
- It performs in many tasks – pre-processing, segmentation using object shape, and object quantification – better and more quickly than the standard approach.
- Mathematical morphology tool is different from the usual standard algebra and calculus.
- Morphology tools are implemented in most advanced image analysis.
- Mathematical morphology is very often used in applications where shape of objects and speed is an issue—example: analysis of microscopic images, industrial inspection, optical character recognition, and document analysis.
- The non-morphological approach to image processing is close to calculus, being based on the point spread function concept and linear transformations such as convolution.
- Mathematical morphology uses tools of non-linear algebra and operates with point sets, their connectivity and shape.
- Morphology operations simplify images, and quantify and preserve the main shape characteristics of objects.
- Morphological operations are used for the following purpose:
    - Image pre-processing (noise filtering, shape simplification).
    - Enhancing object structure (skeleton zing, thinning, thickening, convex hull, object marking).
    - Segmenting objects from the background.
    - Quantitative description of objects (area, perimeter, projections, Euler-Poincare characteristics).

- Mathematical morphology exploits point set properties, results of integral geometry, and topology.
- The real image can be modeled using point sets of any dimension; the Euclidean 2D space and its system of subsets is a natural domain for planar shape description.
- Computer vision uses the digital counterpart of Euclidean space – sets of integer pairs **(∈)** for binary image morphology or sets of integer triples**(∈)** for gray-scale morphology or binary 3D morphology.
- Discrete grid can be defined if the neighborhood relation between points is well defined. This representation is suitable f r b th rectangular and hexagonal grids.
- A morphological transformation is givenby the relation of the image with another small point set ***B*** called stru ct ring element. ***B*** is expressed with respect to a local origin .
- Structuring element is a small image-used as a moving window-- whose support delineates pixel neighborhoods in the image plane.
- It can be of any shape, size, or connectivity (more than 1 piece, have holes).
- To apply the morphologic transformation **()** to the image means that the structuring element ***B*** is moved systematically across the entire image.
- Assume that B is positioned at some point in the image; the pixel in the image corresponding to the representative point ***O*** of the structuring element is called the current pixel.
- The result of the relation between the image ***X*** and the structuring element *B* in the current position is stored in the output image in the current image pixel position.

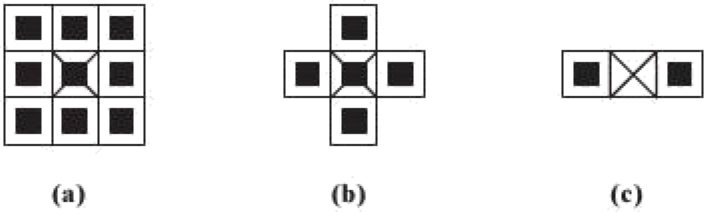


Fig 1: Typical structuring elements.

- The duality of morphological operations is deduced from the existence of the set complement; for each morphological transformation () there exists a dual transformation *()

() =*()

- The translation of the point set by the vector h is denoted by ; it is defined by

= , = + h !" #!$% &.

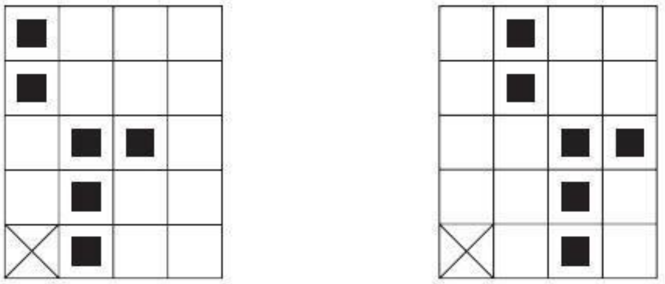


Fig 2: Translation by a vector

Binary Dilation and Erosion

- The sets of black and white pixels constitute a description of a binary image. Assume only black pixels is considered, and the others are treated as a background.
- The primary morphological operations are dilation and erosion, and from these two, more complex morphological operations such as opening, closing, and shape decomposition can be constituted.

Dilation

- The morphological transformation dilation $\oplus$ combines two sets using vector addition (e.g., (a, b) +(c, d) = (a+c, b+d))

  Example:
  A={(1,0),(1,1),(1,2),(2,2),(0,3),(0,4)},

  1= {(0, 0), (1, 0)},

  A$\oplus$ **1**={(1,0),(1,1),(1,2),(2,2),(0,3),(0,4),(2,0),(2,1),(2,2),(3,2),(1,3),(1,4)}

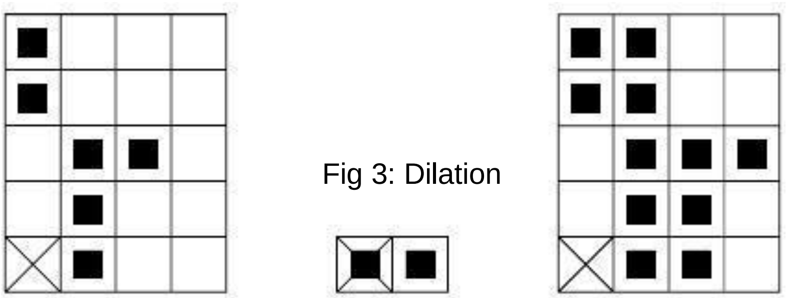


Fig 3: Dilation

- Fig 4 shows 256x256 original image on the left. A structuring element size 3x3 is used.
- The result of dilation is shown on the right side of Fig 4. In this case the dilation is an isotropic expansion (Fill or Grow).

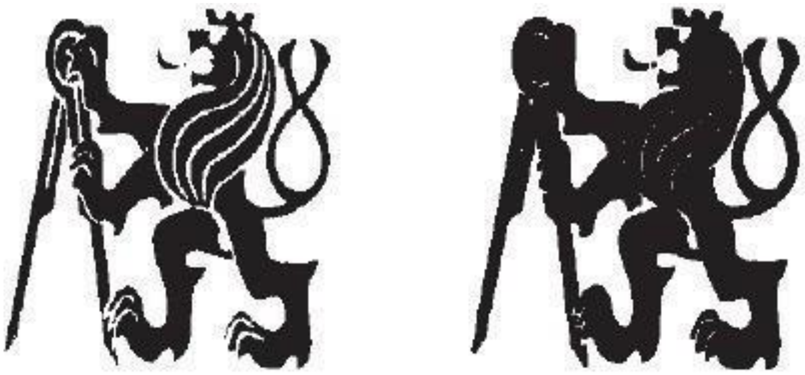


Fig 4: Dilation as isotropic expansion

- Dilation with an isotropic 3x3 structuring element might be described as a transformation which changes all background pixels neighboring the object to object pixels.

- Dilation is used to fill small holes and narrow gulfs in objects. Iincreases the object size if the original size needs to be preserved, and then dilation is combined with erosion.

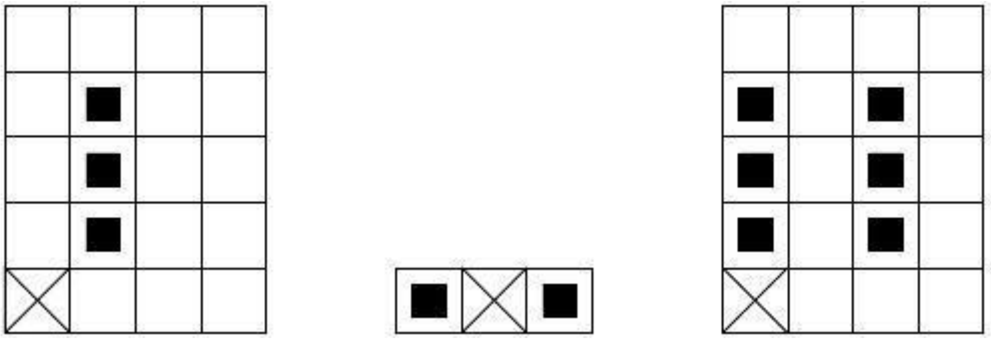


Fig 5: Dilation where the representative point is not a member of the structuring element.

- Fig 5 illustrates the result of dilation if the representative point is not a member of the structuring element B, if this structuring element is used; the dilation result is substantially different from the input set.

Erosion

- Erosion ⊖ combines two sets using vector subtraction of set elements and is dual operator of dilation.

⊖ 1 = ∈ : = + 3 ∈ !" %?%"@ 3 1}.

- This formula says that every point from the image is tested; the result of the erosion is given by those points for which all possible **+ 3** are in X.

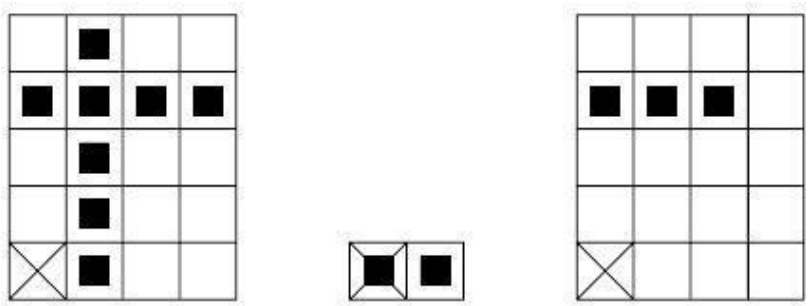


Fig 6: Erosion

- The result of the erosion is shown in the right side of the Fig 7. Erosion with an isotropic structuring element is called as shrink or reduce.

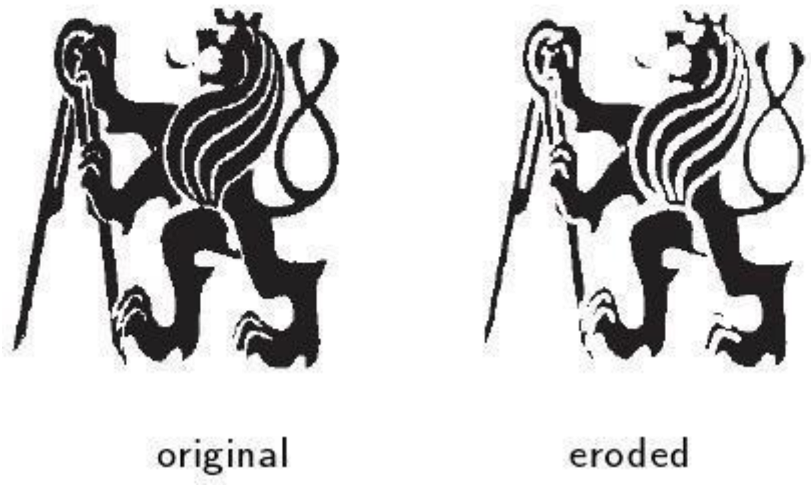


Fig 7: Erosion as isotropic shrink

- Basic morphological transformations can be used to find the contours ofobjects in an image very quickly. This can be achieved, for instance, by subtraction from the original picture of its eroded version as in Fig 8.

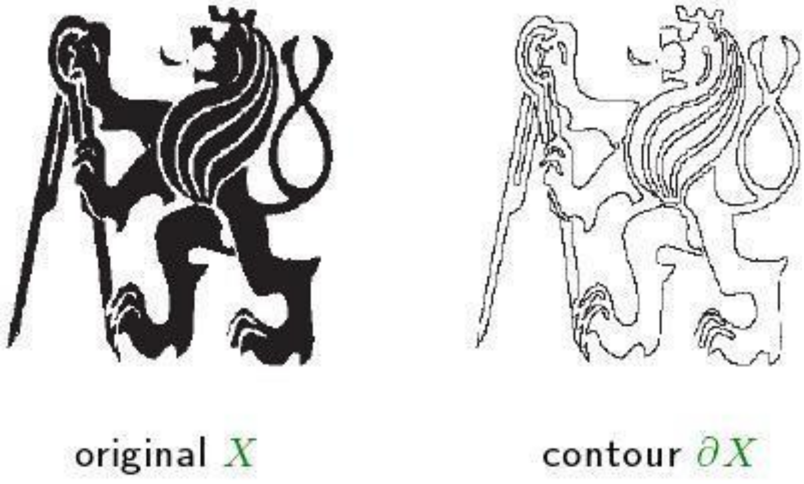
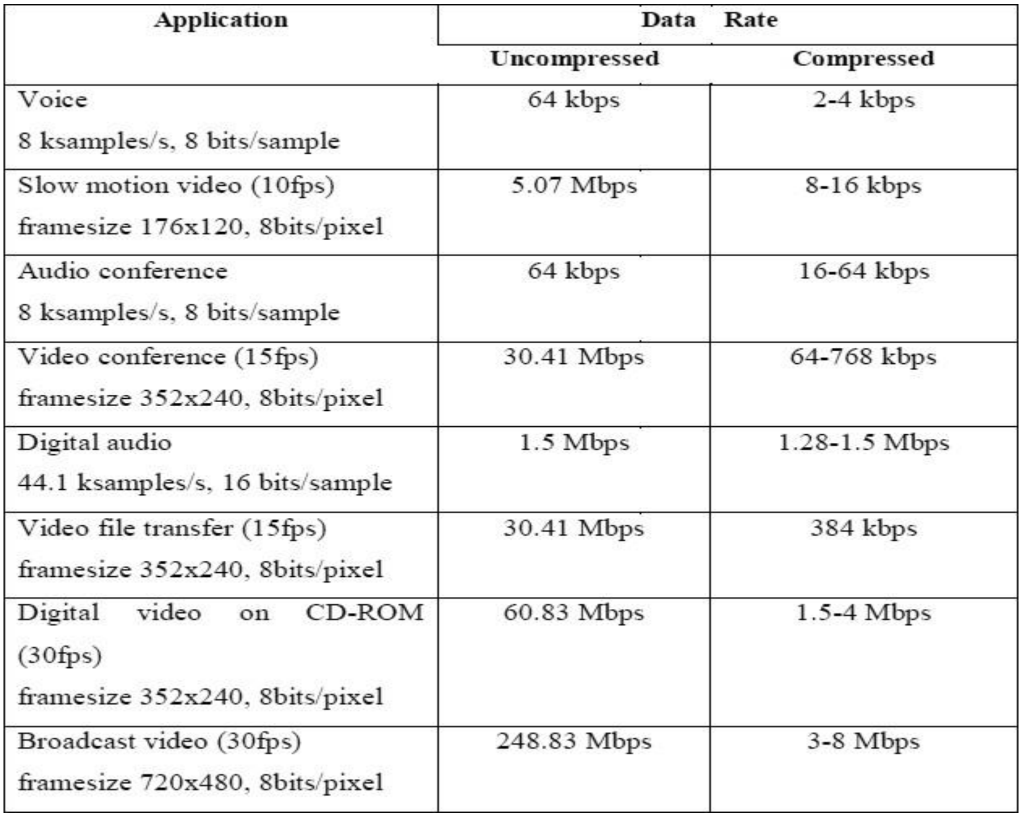
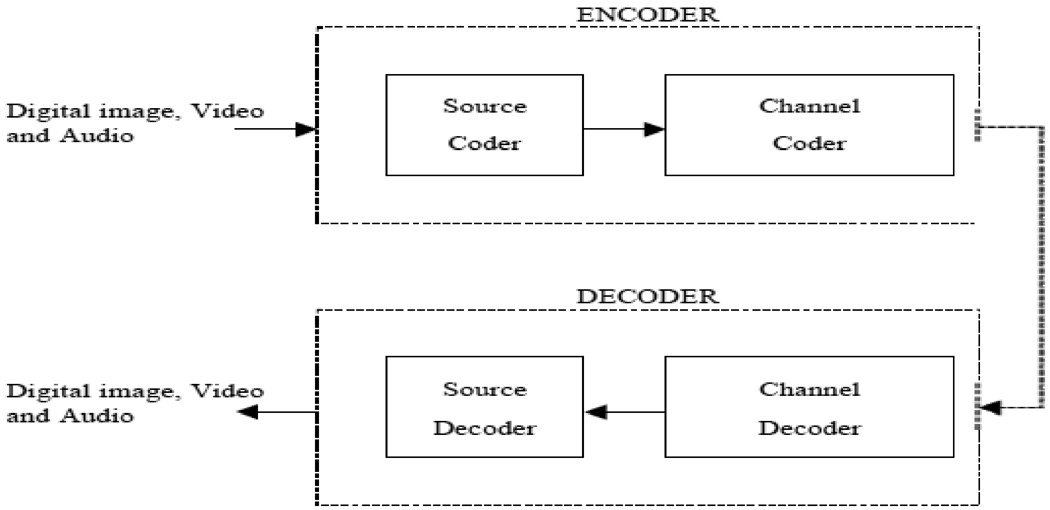

original X contour ∂X

Fig 8: Contours obtained by subtraction of an eroded image from the original (left).



Application	Data Rate	
	Uncompressed	Compressed
Voice 8 ksamples/s, 8 bits/sample	64 kbps	2-4 kbps
Slow motion video (10fps) framesize 176x120, 8bits/pixel	5.07 Mbps	8-16 kbps
Audio conference 8 ksamples/s, 8 bits/sample	64 kbps	16-64 kbps
Video conference (15fps) framesize 352x240, 8bits/pixel	30.41 Mbps	64-768 kbps
Digital audio 44.1 ksamples/s, 16 bits/sample	1.5 Mbps	1.28-1.5 Mbps
Video file transfer (15fps) framesize 352x240, 8bits/pixel	30.41 Mbps	384 kbps
Digital video on CD-ROM (30fps) framesize 352x240, 8bits/pixel	60.83 Mbps	1.5-4 Mbps
Broadcast video (30fps) framesize 720x480, 8bits/pixel	248.83 Mbps	3-8 Mbps



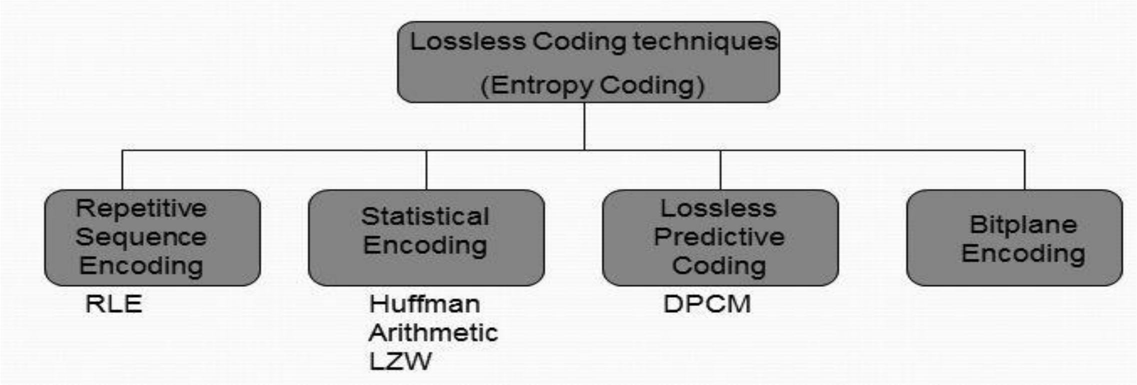
Types of image compression

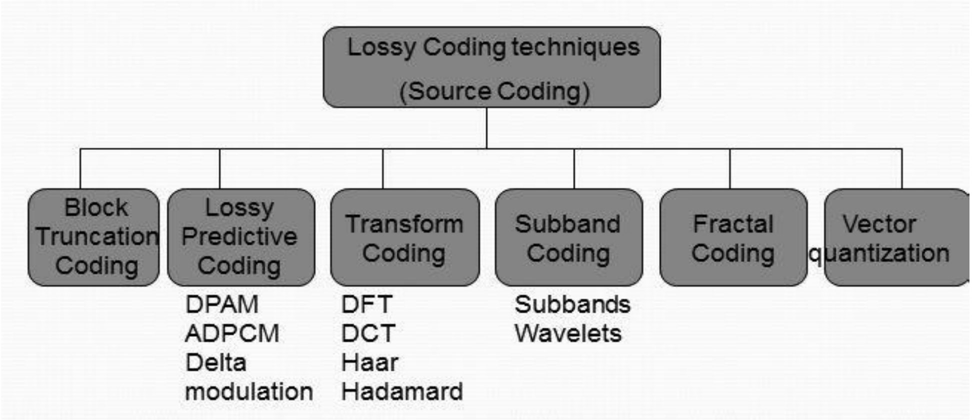
Image compression can be:

- **Reversible** (loss less), with no loss of information.
  - A new image is identical to the original image (after decompression).
  - Reversibility is necessary in most image analysis applications.

  - The compression ratio is typically 2 to 10 times.
  - Examples are Huffman coding and run length coding.
- **Non reversible** (lossy), with loss of some information.
  - Lossy compression is often used in image communication, video,WWW, etc.
  - It is usually important that the image visually is still *nice*.
  - The compression ratio is typically 10 to 30 times.

$$SNR = 10\log_{10}\frac{encoderinput\ signal\ energy}{noisesignalenergy}$$


Lossless Coding techniques
(Entropy Coding)
Repetitive Sequence Encoding
RLE
Statistical Encoding
Huffman
Arithmetic
LZW
Lossless Predictive Coding
DPCM
Bitplane Encoding



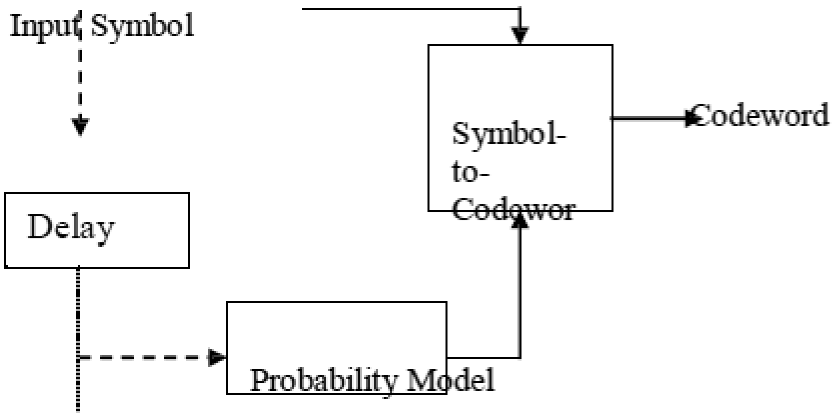
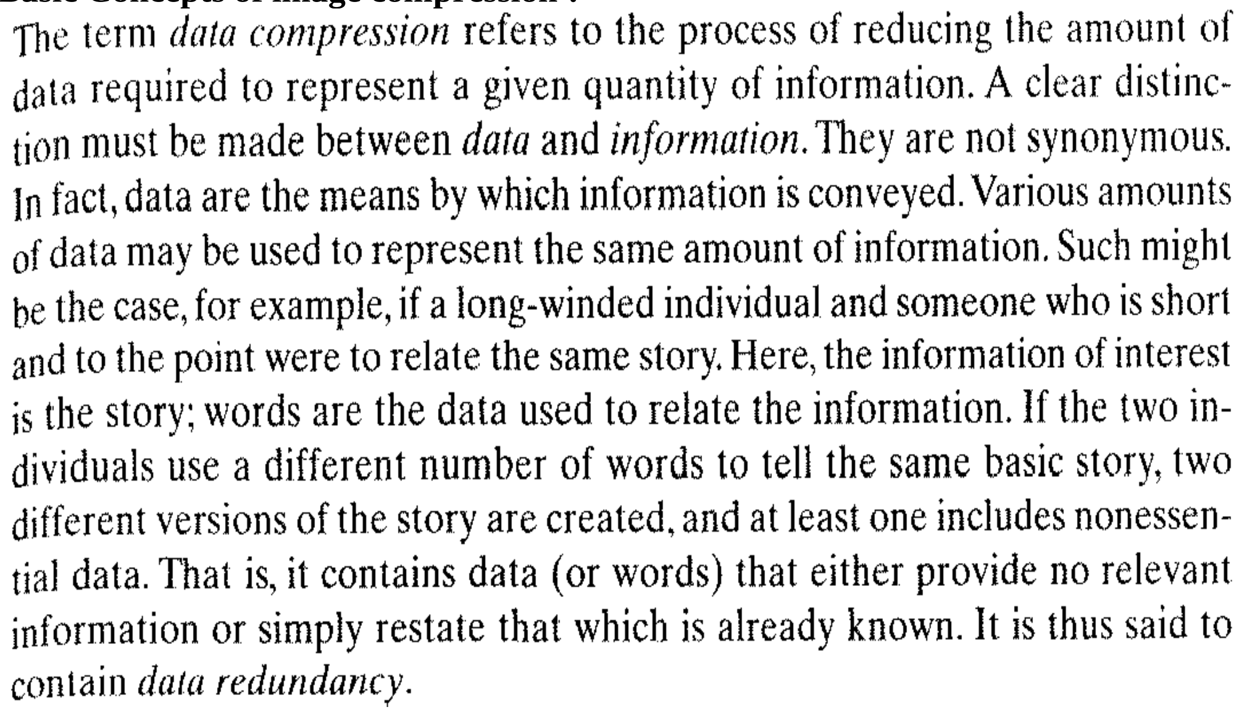
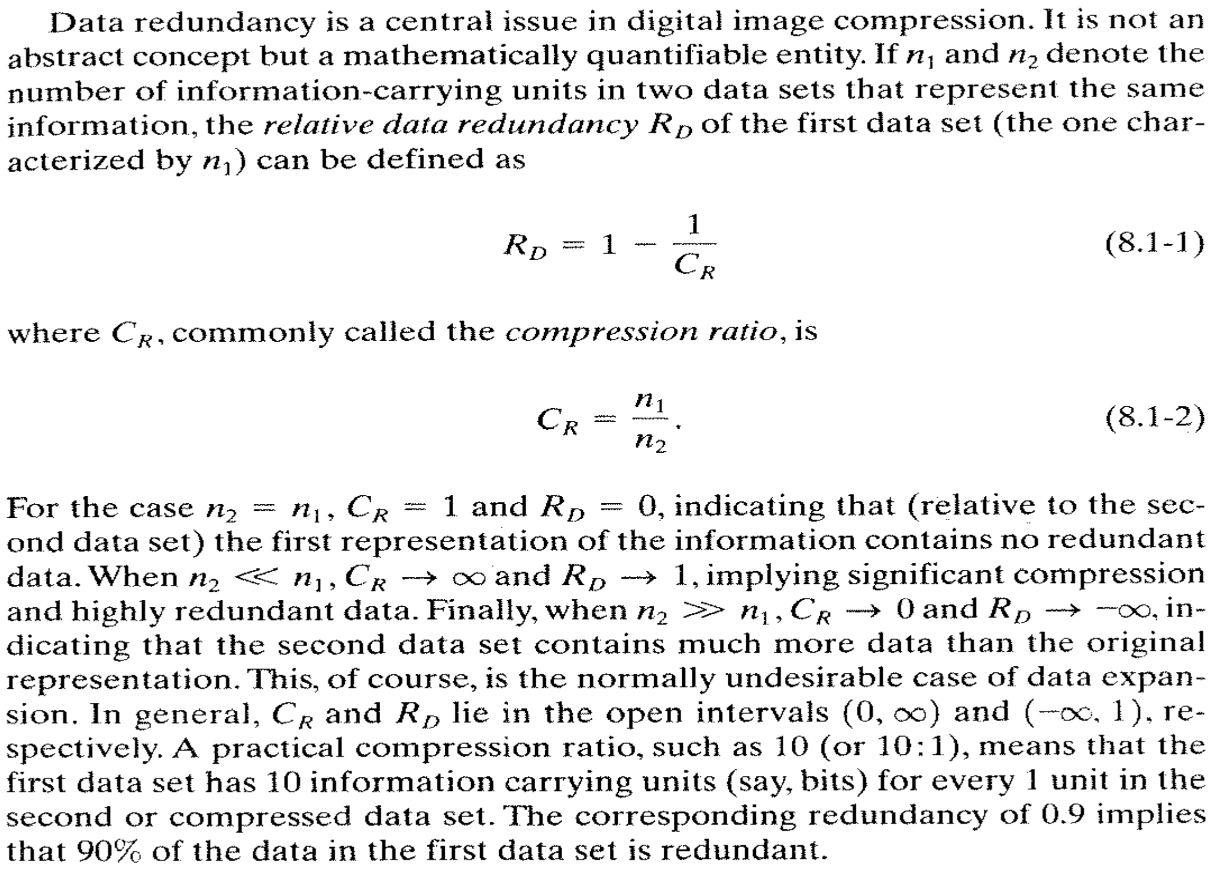


Figure 1.1: A generic model for lossless compression

Basic Concepts of image compression :



The term *data compression* refers to the process of reducing the amount of data required to represent a given quantity of information. A clear distinction must be made between *data* and *information*. They are not synonymous. In fact, data are the means by which information is conveyed. Various amounts of data may be used to represent the same amount of information. Such might be the case, for example, if a long-winded individual and someone who is short and to the point were to relate the same story. Here, the information of interest is the story; words are the data used to relate the information. If the two individuals use a different number of words to tell the same basic story, two different versions of the story are created, and at least one includes nonessential data. That is, it contains data (or words) that either provide no relevant information or simply restate that which is already known. It is thus said to contain *data redundancy*.



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

$$R_D = 1 - \frac{1}{C_R} \qquad (8.1\text{-}1)$$

where C_R, commonly called the *compression ratio*, is

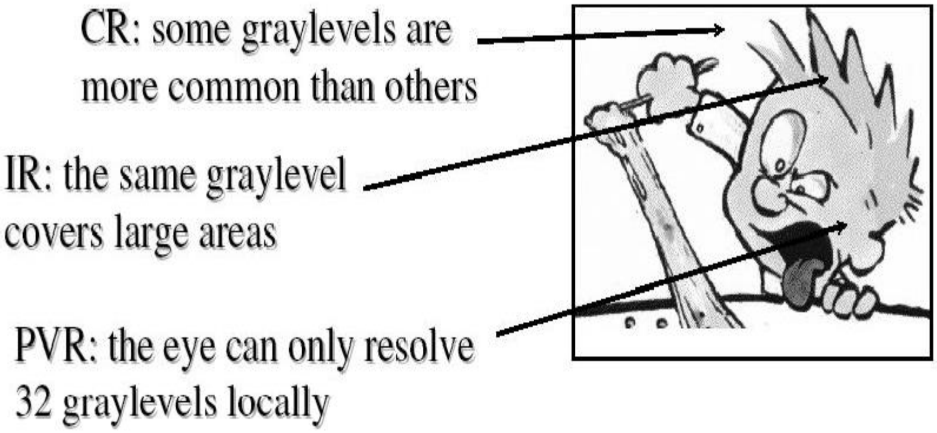
$$C_R = \frac{n_1}{n_2}. \qquad (8.1\text{-}2)$$

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

- Data that provide no relevant information=*redundant data* or *redundancy*.
- Image compression techniques can be designed by reducing or eliminating the Data Redundancy
- Image coding or compression has a goal to reduce the amount of data by reducing the amount of redundancy.

Three basic data redundancies

- Coding Redundancy
- Interpixel Redundancy
  - Psychovisual Redundancy



CR: some graylevels are more common than others

IR: the same graylevel covers large areas

PVR: the eye can only resolve 32 graylevels locally

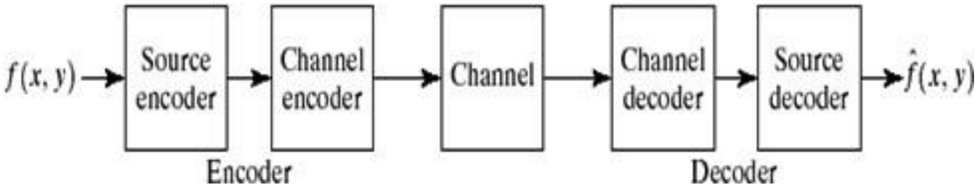
- Coding
  - if grey levels of image are coded in such away that uses more symbols than is necessary
- Inter-pixel
  - can guess the value of any pixel from its neighbours
- Psyco-visual
  - some information is less important than other info in normal visual processing

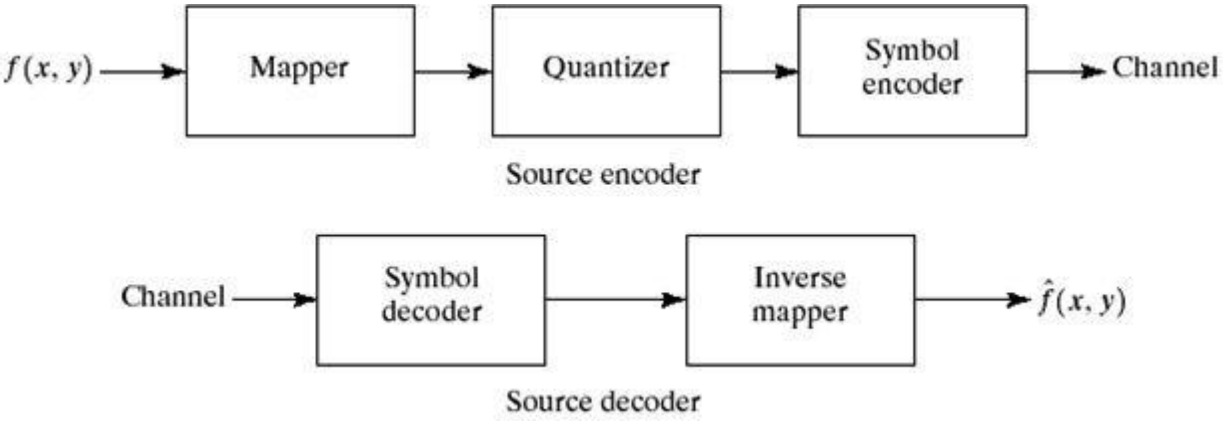
- **Coding redundancy**
  - Fewer bits to represent frequent symbols
  - Huffman coding : Lossless

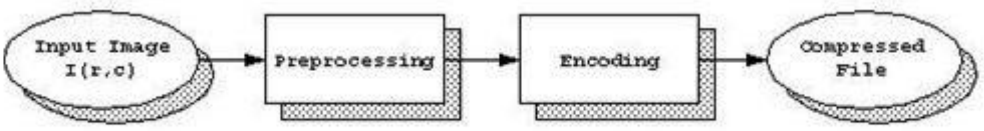
- **Interpixel redundancy**
  - Neighboring pixels have similar values
  - Predictive coding : Lossless

- **Psychovisual redundancy**
  - Quantization : Lossy
  - Remove information that human visual system cannot perceive
  - Removal of high frequency data : Lossy

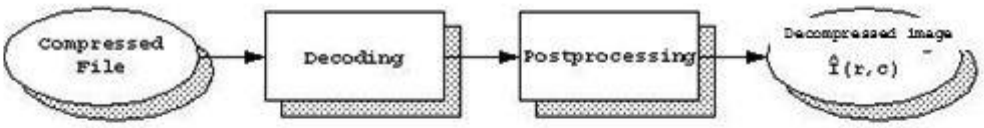
General compression model







a) Compression

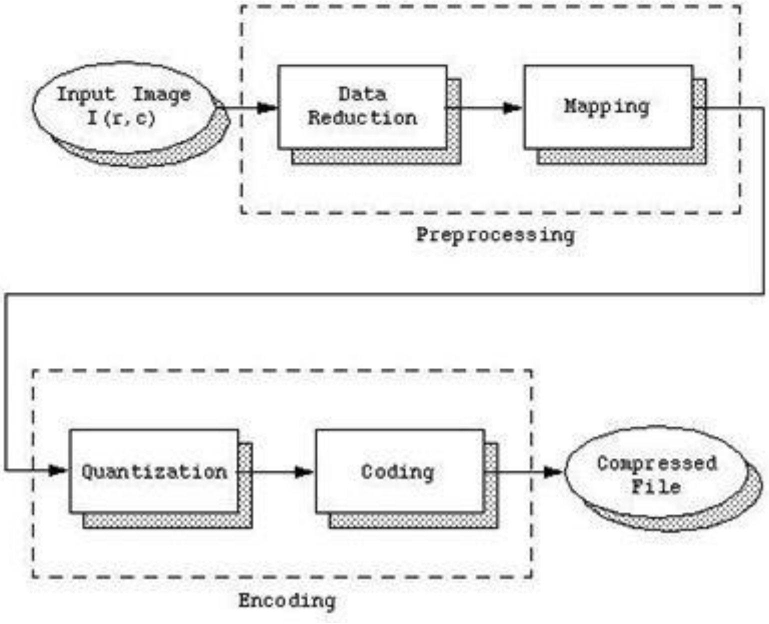


b) Decompression

- Before encoding, preprocessing is performed to prepare the image for the encoding process, and consists of any number of operations that are application specific
- After the compressed file has been decoded, postprocessing can be performed to eliminate some of the potentially undesirable artifacts brought about by the compression process

3. *Quantization*: Involves taking potentially continuous data from the mapping stage and putting it in discrete form
4. *Coding*: Involves mapping the discrete data from the quantizer onto a code in an optimal manner

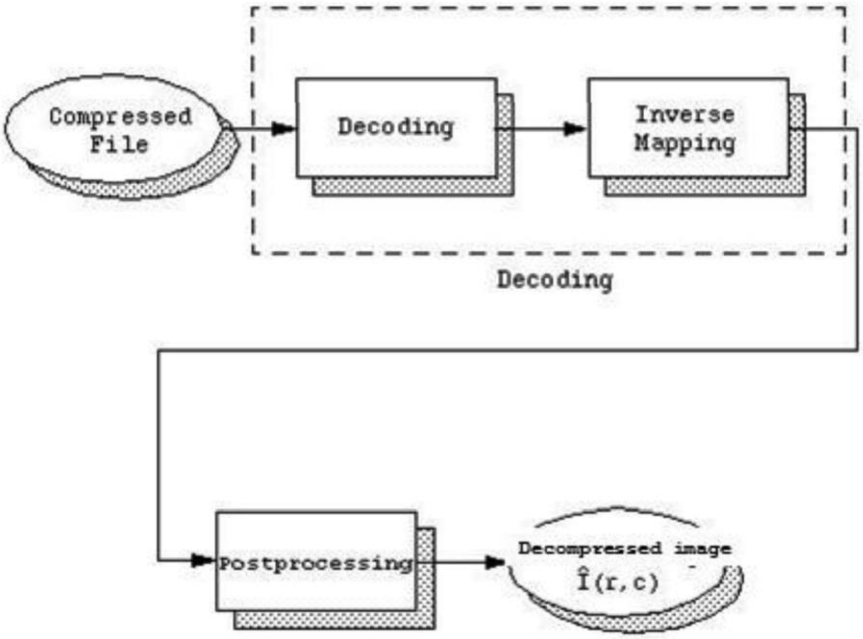
- A compression algorithm may consist of all the stages, or it may consist of only one or two of the stages



The decompressor can be broken down into following stages:

1. *Decoding*: Takes the compressed file and reverses the original coding by mapping the codes to the original, quantized values
2. *Inverse mapping*: Involves reversing the original mapping process
3. *Postprocessing*: Involves enhancing the look of the finalmage

- This may be done to reverse any preprocessing, for example, enlarging an image that was shrunk in the data reduction process
- In other cases the postprocessing may be used to simply enhance the image to ameliorate any artifacts from the compression process itself



- The mapping process is important because image data tends to be highly correlated
- Specifically, if the value of one pixel is known, it is highly likely that the adjacent pixel value is similar
- By finding a mapping equation that decorrelates the data this type of data redundancy can be removed

Huffmann Coding

- The Huffman code, developed by D. Huffman in 1952, is a *minimum length code*
- This means that given the statistical distribution of the gray levels (the histogram), the Huffman algorithm will generate a code that is as close as possible to the *minimum bound, the entropy*
- The method results in an *unequal* (or *variable*) *length code*, where the size of the code words can vary
- For complex images, Huffman coding alone will typically reduce the file by 10% to 50% (1.1:1 to 1.5:1), but this ratio can be improved to 2:1 or 3:1 by preprocessing for irrelevant information removal

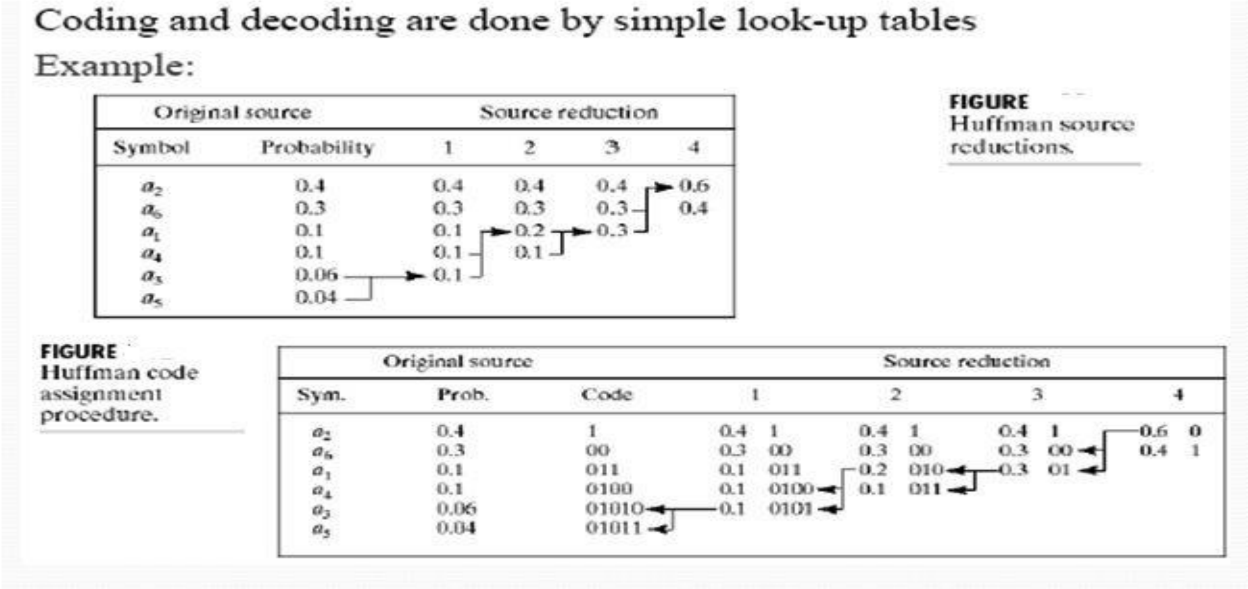
The Huffman algorithm can be described in five steps:

1. Find the gray level probabilities for the image by finding the histogram
2. Order the input probabilities (histogram magnitudes) from smallest to largest
3. Combine the smallest two by addition
4. GOTO step 2, until only two probabilities are left
5. By working backward along the tree, generate code by alternating assignment of 0 and 1

The most popular technique for removing coding redundancy; yields the smallest possible number of code symbols per source symbol

Huffman Coding Algorithm

- Arrange the symbol probabilities p_i in a decreasing order; consider them (p_i) as leaf nodes of a tree
- While there is more than one node:
  - *merge the two nodes with smallest probability to form a new node whose probability is the sum of the two merged nodes*
  - *Arrange the combined node according to its probability in the tree*
  - *Repeat until only two nodes are left*
- Starting from the top, arbitrarily assign 1 and 0 to each pair of branches merging into a node
- Continue sequentially from the root node to the leaf node where the symbol is located to complete the coding



Coding and decoding are done by simple look-up tables

Example:

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

FIGURE
Huffman source reductions.

FIGURE
Huffman code assignment procedure.

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

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

SCHOOL OF ELECTRICAL AND ELECTRONICS

DEPARTMENT OF ELECTRONICS AND COMMMUNICATION ENGINEERING

UNIT - V

ADVANCED DIGITAL IMAGE PROCESSING – SECA7022

UNIT 5 CLASSIFICATION AND APPLICATIONS

Object Recognition and Classification, Statistical classification, Structural /Syntactic Classification,

3D Image Processing, 3DVisualization: Surface rendering, Volume rendering;

Applications: Motion Analysis, Image Fusion, Image Classification

Sources of 3D Data;

Acquisition from 2D images:

3D data acquisition and object reconstruction can be performed using stereo image pairs. Stereo photogrammetry or photogrammetry based on a block of overlapped images is the primary approach for 3D mapping and object reconstruction using 2D images

Acquisition from acquired sensor data:

Semi-Automatic building extraction from LIDAR Data and High-Resolution Images is also a possibility. Again, this approach allows modelling without physically moving towards the location or object. From airborne LIDAR data, digital surface model (DSM) can be generated and then the objects higher than the ground are automatically detected from the DSM. Based on general knowledge about buildings, geometric characteristics such as size, height and shape information are then used to separate the buildings from other objects. The extracted building outlines are then simplified using an orthogonal algorithm to obtain better cartographic quality. Watershed analysis can be conducted to extract the ridgelines of building roofs. The ridgelines as well as slope information are used to classify the buildings per type. The buildings are then reconstructed using three parametric building models (flat, gabled, hipped)

Acquisition from on-site sensors:

LIDAR and other terrestrial laser scanning technology offers the fastest, automated way to collect height or distance information. LIDAR or laser for height measurement of buildings is becoming very promising. Commercial applications of both airborne LIDAR and ground laser scanning technology have proven to be fast and accurate methods for building height extraction. The building extraction task is needed to determine building locations, ground elevation, orientations, building size, rooftop heights, etc. Most buildings are described to sufficient details in terms of general polyhedra, i.e., their boundaries can be represented by a set of planar surfaces and straight lines. Further processing such as expressing building footprints as polygons is used for data storing in GIS databases.

Slicing the Data set;

In fields employing interface design skills, **slicing** is the process of dividing a single 2D user interface composition layout (comp) into multiple image files (digital assets) of the graphical user interface (GUI) for one or more electronic pages. It is typically part of the client side development process of creating a web page and/or web site, but is also used in the user interface design process of software development and game development.

The process involves partitioning a comp in either a single layer image file format or the multi-layer native file format of the graphic art software used for partitioning. Once partitioned, one would save them as separate image files, typically in GIF, JPEG or PNG format in either a batch process or one at a time. Multi-layered image files may include multiple versions or states of the same image, often used for animations or widgets.

Volumetric display;

A volumetric display device is a graphic display device that forms a visual representation of an object in three physical dimensions, as opposed to the planar image of traditional screens that simulate depth through a number of different visual effects. One definition offered by pioneers in the field is that volumetric displays create 3D imagery via the emission, scattering, or relaying of illumination from well-defined regions in (x,y,z) space.

3D Image Processing

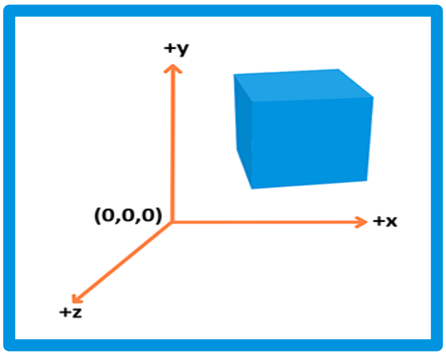
What is Digital Image Processing?

Digital Image Processing is used to manipulate the digital images by the use of computer system. It is also used to enhance the images, to get some important information from it.

For example: Adobe Photoshop, MATLAB, etc.

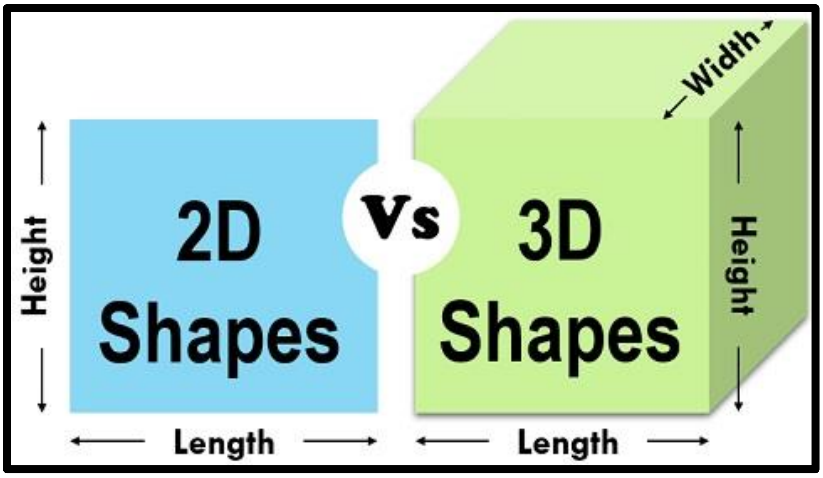
3D Image:

Every pixel is represented with 3 co-ordinates with each having their own values.



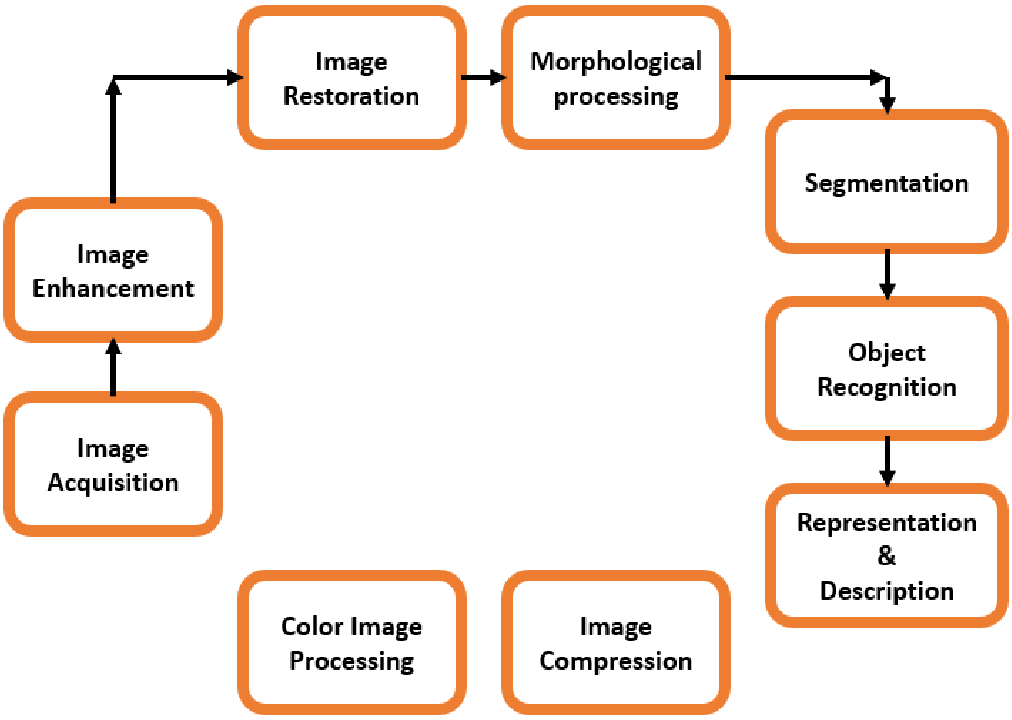
2D vs 3D Image

The 2D shapes have length and height. The 3D images have length, height and width.



Difference between 2D and 3D images

2D	3D
Uses a flat image	Shape of the image is Considered in totality.
Visible light is enough	Infra-red light + Visible light
Uses 20 to 30 reference points	Uses 1000s of reference points
Less Accurate	More Accurate
Uses 2D image Database	Uses 3D image database or existing 2D image database



Key stages of Digital Image Processing:

It is common to both the 2D and 3D images.

1) Image Acquisition

- It is basically **capturing** an **image.**
- Generally, the image acquisition stage involves **pre-processing**, such as scaling, etc.

2) Image Enhancement

- It is the process of **filtering image** (removing noise, increasing contrast, etc) to **improve the quality.**
- The resulting image will be more suitable than the **original image**.

3) Image Restoration

- It is the process of **improving appearance** (reducing blurring etc) of an image by **mathematical or probabilistic models.**

4) Morphological Processing

- It is the process for **extracting** image components that are useful in the **representation** and **description** of shape.

5) Segmentation

- It is the process of partitioning the image into **multiple segments.**

6) Recognition

- It is the process of **assigning labels** to an object based on its **description**.

7) Representation and Description

- It involves representing an image in various forms:
- **Boundary Representation** — It focuses on the **external shape** characteristics such as corners and inflections.
- **Regional Representation** — It focuses on **internal properties** such as texture and skeletal shape.
- **Description** and **Feature selection** helps in **extracting useful information**.

8) Compression

- It involves the techniques for **reducing the size** of the image with **minimum deterioration** in its **quality**.

Some of the sources of 3D image

- Computerized Tomography (CT)
- 3D laser scanning
- 3D facial morphometry (3DFM)
- Magnetic Resonance Imaging (MRI)

Necessity and Applications for 3D Image Processing

The 3D images have found its applications along various domains. Hence processing it to make it better suitable for our application has become more necessary.

1) 3D Vision Inspection Machine

- Parts identification
- To identify defects

2) **Medical imaging** – To process and analyse images obtained from CT and MRI scan

3) **Aviation** – Air traffic mapping

4) 3D modelling

5) 3D Printing

6) **Entertainment applications** – Gaming, Animation Films and Cartoon Series

7) 3D Facial Recognition

Why we are switching towards 3D facial recognition these days is that, facial recognition depends on various factors like,

- Environmental factors, for example, the lighting conditions, background scene
- Head pose
- Facial hair, Use of cosmetics, Jewelry
- Plastic surgery
- Long-term processes like aging and weight gain.

2D facial recognition faces a serious problem with respect to all these factors whereas 3D Facial Recognition solves most of this issue.

Application Areas of 3D Face Recognition:

- **Access Control** - Companies often want to maintain high levels of security as to who can enter their office buildings.
- **Criminal Identification** - An image of a suspect can be taken from CCTV footage for example, and the face searched for in the database of known criminals.

There are mainly 3 steps of rendering:

✓ Volume Formation

✓ Classification

Image Formation

There are2 methods of rendering:

Surface Rendering:

✓ This is a binary, not a continuous classification technique.

✓ Volumes can be visualized by generating an iso surface.

Volume Rendering:

✓ This is a percentage classificationtechnique.

Maximum Intensity Projection is a volume rendering technique

It involves the following steps:

✓ Forming of an RBGA volume from thedata

✓ Reconstruction of a continuous function

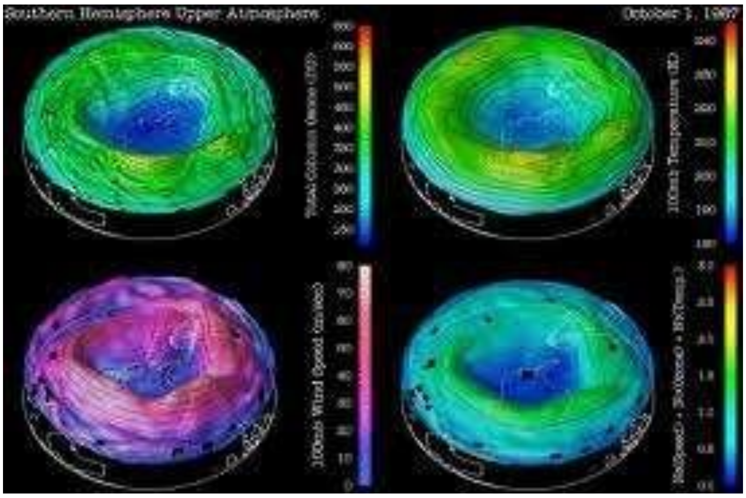
✓ Projecting it onto the 2d viewingplane

There are two implementations of volume rendering:

- ✓ Ray casting
- ✓ Splatting

Isocontouring

It is a technique where one constructs a boundary



It is a natural extension from colour mapping.
There are twosteps:

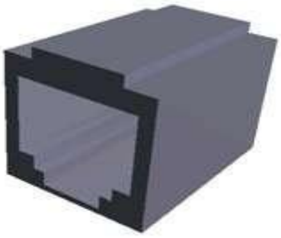
✓ Explore the dataspace

✓ Connect the points

Hole Detection in 3d Models

□ Retrieval speed can be improved

□ More meaningful results can beachieved

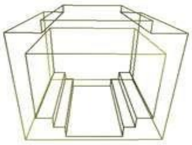


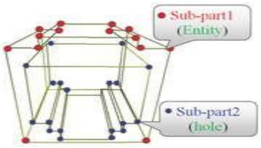
There are two methods for hole detection:

✓ Ray-Scanning

✓ X-Ray inspection

There are three primary stages as follows for detecting holes inside 3D models:





✓ Plane Detection

✓ Contour Extraction

✓ Hole Identification

Image fusion refers to the process of combining two or more images into one composite image, which integrates the information contained within the individual images. The result is an image that has a higher information content compared to any of the input images. The goal of the fusion process is to evaluate the information at each **pixel location** in the input images and retain the information from that image which best represents the true scene content or enhances the utility of the fused image for a particular application. Image fusion is a vast discipline in itself, and refers to the fusion of various types of imagery that provide complementary information.

IMAGE FUSION IN IMAGE PROCESSING

WHAT IS IMAGE FUSION:

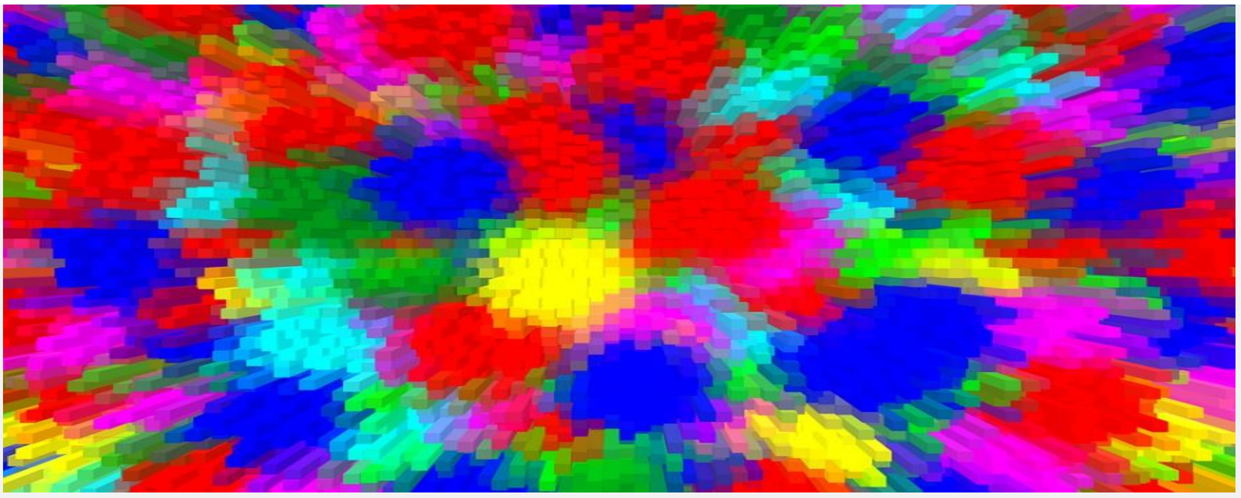


Image fusion is a process or method in image processing which is responsible to combine the specifications or features from different images into one to get an accurate and more informative image.

What this basically means is that while performing the image processing, different images which are most similar are taken and the information of each of the image is gathered into one image. Hence the process called fusion (combining one or many into one)

This is done to obtain a single image that is more accurate and informative than all of the images combined. It is also done to reduce the size, obviously, when you combine only the features of many images into one, the resultant image will have more-accurate characteristics.

We don’t have to waste space in storing all of them, just this one image will do the trick. Hence, compactness is secured of the image and the database where it will be stored.

For instance, say that there are two images, one coloured and another, black & white. Both the images are of the same scene or thing. The only difference between them is the way they are presented. Now there are some things which are not visible in coloured or are not clear or are missed, while the black & white image has those things which the coloured doesn't, but lack some other information or distortion.

Now each of the images is incomplete. To get the best and accurate image, this image fusion is done. The best characters from the coloured and black & white image will be taken and fused together to get one perfect image.

This resultant image won't have any faults or missed areas like the two before. That is the whole purpose of image fusion to get one improved image from two imperfect images.

Now that you know what image fusion is all about, let's see how it is done. There are 3 levels in which the image fusion takes place.

· At Pixel Level

· At feature Level

· At Decision level

As we all know, the pixel of an image is the smallest unit. So, the first process of fusion is at the pixel level.

At the pixel level, the pixel of the first image is already registered and related to the second image in the database and then the same pixel from the second image is analysed. It is done at a very minute level.

At the feature level, the features of any object in the image (say person) is matched with the features of the object (same person) from another image and then the features are fused to get a new better image.

At Decision level, at this level, both the images are analysed separately and the information regarding each of the image, say features and characteristics, are stored and then that collected information is fused to get a new complete and fused image.

Methods Used for Image Fusion:

There are a number of methods that are used for image fusion. Such as,

· Multiplicative algorithm

· Subtractive method

· PCA (Principle Component Analysis)

· IHS (Intensity Hue Saturation) method

· High pass filter method

· Brovey Transform method

· Wavelet method

Multiplicative Algorithm is a method where a high-quality image and a low-quality image is taken and fused together to get a perfect image which is more informative than the two.

The subtractive method is a method where the overlapping of the two images is done based on subtractive algorithms and the resultant image will have the colour specifications of the coloured image but the detail of the black & white image. It has good quality and viewing crisp.

PCA method is a mathematical method where the pixel values are modified to achieve the final resultant image. Here the storing of similar data from the images is reduced to the maximum extent without reducing the quality of the image.

IHS method is a method that is mainly used for sharpening the colour image. It only involves the basic three colours which are red, green and blue. It converts the RGB image into IHS space. All these three characteristics of the image are improved and the image is enhanced in the intensity, hue and saturation department.

High Pass Filter method is the method in which the high-resolution information in the black & white image is combined with the low-resolution information in the coloured image and the final image is real and smooth which sacrifices the sharpness and hence the object in the image is not sharp enough.

Brovey Transform method is a method where data from different sensors have the ability to preserve the pixels. Pixels of each image are gathered but the resultant image can have more brightness than the two combined. It is a simple method to combine data from different image sensors.

Wavelets method is a method where the colour is taken from the coloured image and is poured on to the black & white image. The resultant image is not well-formed and the corners are not clear. Hence the image is not that perfect.

Application of Image Fusion:

Image fusion is used to check the following:

1) Object identification: Suppose that the resolution of the coloured image is low and the person or object in the image is small, then it is fused with a black & white image which has higher resolution, to get a high-resolution image.

The resultant image will have a high-resolution and the size of the object in that image will also be larger. Hence it becomes easy to identify.

2) Classification: The second application of image fusion is to classify the image. Meaning, when a coloured image is fused with a black & white image, the final image will have high resolution and thus it becomes easy to classify them according to the different categories of data.

3) Change detection: The third and final application of image fusion is to differentiate the change between the two images. Say that the image is kept for a long duration of time, now it can be that the image will have some distortion or some portion may be faded or missing. To check that kind of change the image fusion is done.

MOTION ANALYSIS OF IMAGES

- Motion analysis is a measuring technique used in computer vision, image processing and high-speed photography applications to detect a movement.
- The objectives of motion analysis are:

i. To detect motion within an image,
ii. Track an object's motion over time,
iii. Group objects that move together and
iv. Identify the direction of motion.

- Specific techniques for implementing motion or movement analysis include electromyography, background segmentation and differential equation models.

- Motion analysis allows us to extract much useful information from a scene:
  - Object locations and tracks
  - Camera motion
  - 3D geometry of the scene
- Motion analysis software can help organizations implement aspects of motion analysis with minimal effort. Once connected to motion capture devices, the software can visually display object tracking information for further examination and manipulation.

How motion analysis works:

- The basic function of motion analysis is to compare two or more consecutive images captured by sensors or cameras to return information on the apparent motion in the images. This is usually done by programming the recording device to produce binary images based on movement.

- All of the image points, or pixels, that correspond with motion are set to a value of 1 while stationary pixels are set to 0. The resulting image can be processed even further to remove noise, label objects and group neighboring values of 1 into a singular object.
- The data produced by motion analysis tools often correlates to a specific image at a specific point in time based on its position in the sequence.
- Therefore, the motion capture data is time-dependent, which is a crucial component in most tracking applications.

- Motion analysis is an overarching task that can be broken down into many components.
- Object detection
- Object localization
- Object segmentation
- Tracking
- Motion detection
- Pose estimation
- Each of the mentioned components is a noteworthy field that requires another medium article. Several research efforts have gone into the exploration of methodologies of solving each component efficiently.

OBJECT DETECTION:

- Object detection algorithms act as a combination of image classification and object localization. It takes an image as input and produces one or more bounding boxes with the class label attached to each bounding box.
- Object detection as a computer vision task is defined as **recognizing the presence** of an object of interest from a specific class in image or sequential images.
- The task of object detection usually involves illustrating bounding boxes around the instances of the object of interest and recognizing the class the detected object belongs to.
- In specific scenarios, we are focused on identifying one object within an image; in some other cases, we are interested in multiple objects for detection.
- It should be noted that object detection can be further broken down to object segmentation. Object segmentation algorithms and techniques produce as outputs highlighted pixels of object presence as opposed to bounding boxes.

OBJECT LOCALIZATION:

- This algorithm locates the presence of an object in the image and represents it with a bounding box. It takes an image as input and outputs the location of the bounding box in the form of (position, height, and width).
- The task of object localization is to predict the object in an image as well as its boundaries.
- The difference between object localization and object detection is subtle.
- Simply, object localization aims to locate the main object in an image while object detection tries to find out all the objects and their boundaries.

OBJECT SEGMENTATION:

- Object segmentation is a further extension of object detection in which we mark the presence of an object through pixel-wise masks generated for each object in the image.
- This technique is more granular than bounding box generation because this can helps us

in determining the shape of each object present in the image. This granularity helps us in various fields such as medical image processing, satellite imaging, etc.

- There are primarily two types of segmentation:

i. **Instance segmentation:** identifying the boundaries of the object and label their pixel with different colors.

ii. **Semantic segmentation:** labeling each pixel in the image with different colors based on their category class or class label.

TRACKING:

- Goal – to detect and track objects moving independently to the background.
- Two situations to be considered:

i. Static background

ii. Moving background

- Object tracking is the process of:

i. Taking an initial set of object detections

ii. Creating a unique ID for each of the initial detections

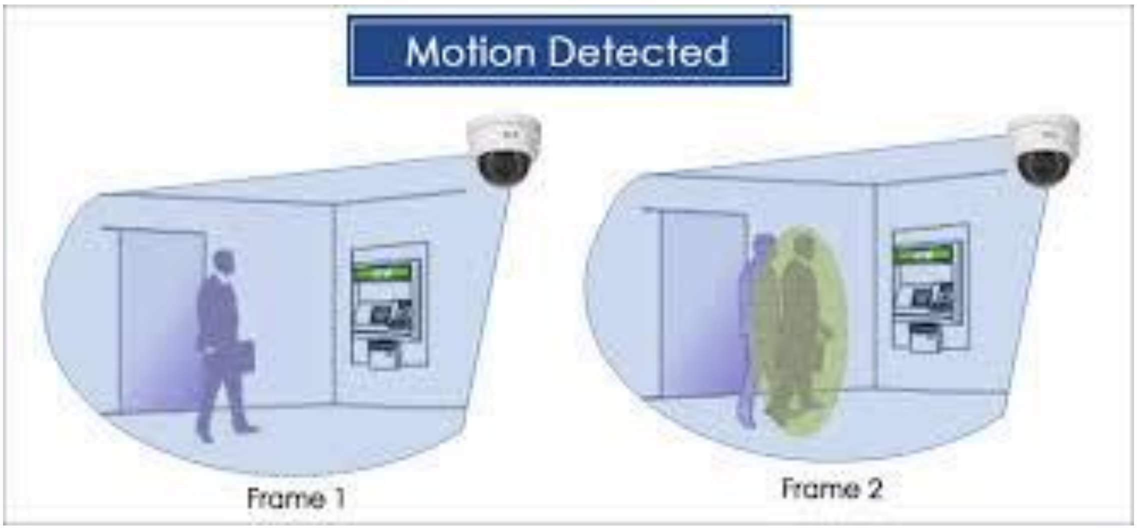
iii. And then tracking each of the objects as they move around frames in a video, maintaining the assignment of unique IDs

- Furthermore, object tracking allows us to apply a unique ID to each tracked object, making it possible for us to count unique objects in a video.

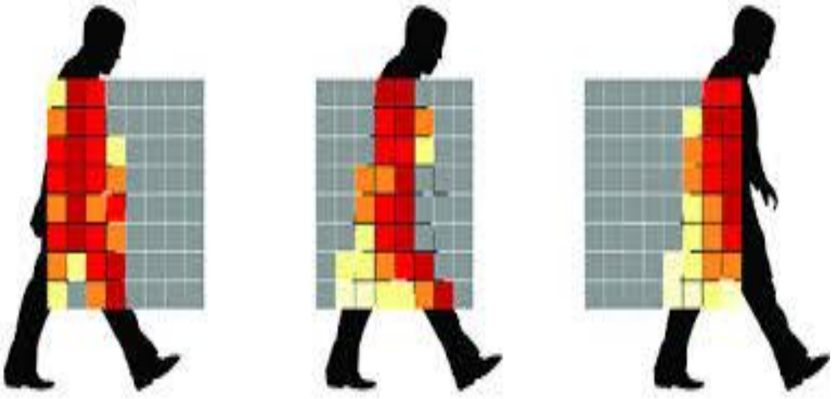
MOTION DETECTION:

- Motion detection is a fundamental aspect of motion analysis because detection of motion is required to determine what and where to conduct analysis upon.
- It is the process whereby, an image containing a moving object is subject to image processing techniques, that enable the tracking of motion through either differential methods or background segmentation, in which the moving aspects within the images are extracted by discarding the motionless parts of the images, to isolate the moving part.
- This technique is mostly dependent on the observation of the intensity of pixels in an image to determine if the pixel belongs to a background class or not.
- By identifying the progression of the intensity of pixels between frames or pictures, a deduction of motion can be drawn where there is a notable change in intensity.

Motion detection:



How detection works:



POSE ESTIMATION:

- Pose estimation is the process where the position and orientation of the vital body parts and joints of a body are derived from an image or sequence of images.
- The output of pose estimation is a 2D or 3D rigid representation of the configuration of the body posture within the images.
- The output of a pose estimation for motion analysis is generally an image, that is generated by the algorithm, that depicts the location of the significant body parts and joints of subjects of interest within the original image.
- Pose estimation has developed within computer vision and become a prominent field in its own right.
- This is typically done by identifying, locating, and tracking a number of **key-points** on a given object or person.

▪ Pose estimation refers to computer vision techniques that detect human figures in images and videos, so that one could determine, for example, where someone’s elbow shows up in an image.

▪ To be clear, this technology is not recognizing who is in an image. The algorithm is simply estimating where key body joints are.

▪ The key points detected are indexed by "part ID", with a confidence score between 0.0 and 1.0, 1.0 being the highest.

APPLICATIONS OF MOTION ANALYSIS:

▪ Motion analysis provides complete 3D full-body motion capture solutions for a wide range of industries, including

i. Movement analysis,
ii. Animation,
iii. Medicine,
iv. Broadcast,
v. Industrial, and
vi. Robotics.

IMAGE CLASSIFICATION

Introduction

Today, with the increasing volatility, necessity and applications of artificial intelligence, fields like machine learning, and its subsets, deep learning and neural networks have gained immense momentum. The training needs software's and tools like classifiers, which feed huge amount of data, analyse them and extract useful features. The intent of the classification process is to categorize all pixels in a digital image into one of several classes. Normally, multi-spectral data are used to perform the classification and, indeed, the spectral pattern present within the data for each pixel is used as the numerical basis for categorization. The objective of image classification is to identify and portray, as a unique gray level (or colour), the features occurring in an image in terms of the object these features actually represent on the ground. Image classification is perhaps the most important part of digital image analysis. Classification between objects is a complex task and therefore image classification has been an important task within the field of computer vision. Image classification refers to the labelling of images into one of a number of predefined classes. There are potentially n number of classes in which a given image can be classified. Manually checking and classifying images could be a tedious task especially when they are massive in number and therefore it will be very useful if we could automate this entire process using computer vision. The advancements in the field of autonomous driving also serve as a great example of the use of image classification in the real-world. The applications include automated image organization, stock photography and video websites, visual search for improved product discoverability, large visual databases, image and face recognition on social networks, and many more; which is why, we need classifiers to achieve maximum possible accuracy.

STRUCTURE FOR PERFORMING IMAGE CLASSIFICATION

1. **Image Pre-processing**: The aim of this process is to improve the image data (features) by suppressing unwanted distortions and enhancement of some important image features so that the computer vision models can benefit from this improved data to work on. Steps for image pre-processing includes Reading image, resizing image, and Data Augmentation (Gray scaling of image, Reflection, Gaussian Blurring, Histogram, Equalization, Rotation, and Translation).

2. **Detection of an object**: Detection refers to the localization of an object which means the segmentation of the image and identifying the position of the object of interest.

3. **Feature extraction and training**: This is a crucial step wherein statistical or deep learning methods are used to identify the most interesting patterns of the image, features that might be unique to a particular class and that will, later on, help the model to differentiate between different classes. This process where the model learns the features from the dataset is called model training.

4. **Classification of the object**: This step categorizes detected objects into predefined classes by using a suitable classification technique that compares the image patterns with the target patterns.

SUPERVISED CLASSIFICATION

Supervised classification is based on the idea that a user can select sample pixels in an image that are representative of specific classes and then direct the image processing software to use these training sites as references for the classification of all other pixels in the image. Training sites (also known as testing sets or input classes) are selected based on the knowledge of the user. The user also sets the bounds for how similar other pixels must be to group them together. These bounds are often set based on the spectral characteristics of the training area. The user also designates the number of classes that the image is classified into. Once a statistical characterization has been achieved for each information class, the image is then classified by examining the reflectance for each pixel and making a decision about which of the signatures it resembles most. Supervised classification uses classification algorithms and regression techniques to develop predictive models. The algorithms include linear regression, logistic regression, neural networks, decision tree, support vector machine, random forest, naive Bayes, and k-nearest neighbor.

UNSUPERVISED CLASSIFICATION

Unsupervised classification is where the outcomes (groupings of pixels with common characteristics) are based on the software analysis of an image without the user providing sample classes. The computer uses techniques to determine which pixels are related and groups them into classes. The user can specify which algorithm the software will use and the desired number of output classes but otherwise does not aid in the classification process. However, the user must have knowledge of the area being classified when the groupings of pixels with common characteristics produced by the computer have to be related to actual features on the ground. Some of the most common algorithms used in unsupervised learning include cluster analysis, anomaly detection, neural networks, and approaches for learning latent variable models.