# Introduction to Image processing and Image Processing Operations

***Syllabus:***

***Introduction to Image processing:*** *Overview, Nature of IP, IP and its related fields, Digital Image representation, types of images.*

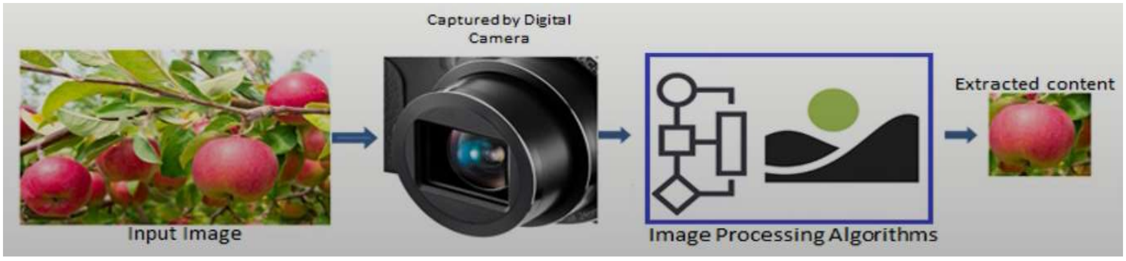
***Digital Image Processing Operations:*** *Basic relationships and distance metrics, Classification of Image processing Operations.*

***Resources:***

1. ***Text Book 2-***S. *Sridhar, Digital Image Processing, second edition, Oxford University press 2016.,* ***Chapters: 1,3***

## Overview

- Images are everywhere! Sources of Images are paintings, photographs in magazines, Journals, Image galleries, digital Libraries, newspapers, advertisement boards, television and Internet.
- Images are imitations of real-world objects.
- In image processing, the term 'image' is used to denote the image data that is sampled, quantized, and readily available in a form suitable for further processing by digital computers.
- In digital image processing, the image shall be fed in as an input to the system and the system shall interpret and understand the content to let the further actions happen.
- In other words, Image processing enables us to perform the required operations on a particular image which could help in either enhancing the image or to extract the information from the image. One can categorize image processing as one of the fields of signal processing.




## Nature of IP

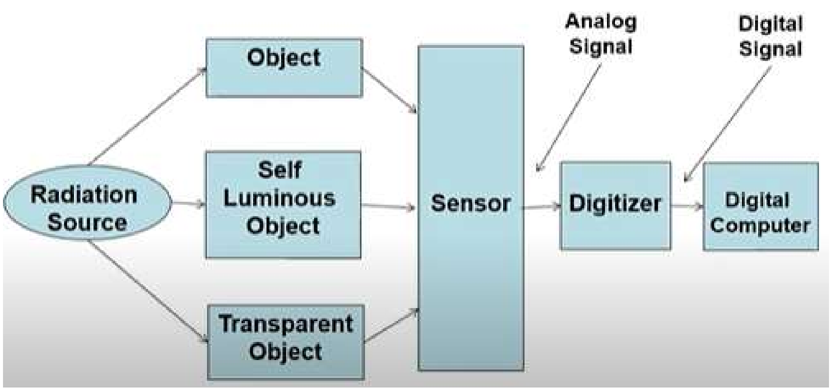



**Image Processing Environment**

- In image processing Environment, first we have a **Radiation Source**which is a light source which is essential for vieing the object.The sun, lamps, and clouds are all examples of radiation or light sources. Next, we have objects, The ***object*** is the target for which the image needs to be created.

- The object can be people, industrial components, or the anatomical structure of a patient.
- The objects can be 2-D, 3-D, or multidimensional mathematical functions involving many variables. For example, a printed document is a 2D object. Most real-world objects are 3D.

- ***Sensors*** are important components of imaging systems. They convert light energy to electric signals.
- The first major challenge in image processing is to acquire the image for further processing.
- **The 3 ways of acquiring image are:**
  1. ***Reflective mode imaging***
     - Reflective mode imaging represents the *simplest form of imaging* and uses a sensor to acquire the digital image.
     - All ***video cameras***, digital cameras, and scanners use some types of sensors for capturing the image.
  2. ***Emissive type imaging***
     - Emissive type imaging is the second type, where the images are acquired from ***self-luminous objects*** without the help of a radiation source.
     - In emissive type imaging, the objects are self-luminous.
     - The radiation emitted by the object is directly captured by the sensor to form an image.
     - ***MRI***(Magnetic Resonance Imaging) is an example of emissive type imaging.
  3. ***Transmissive imaging***
     - Transmissive imaging is the third type, where the *radiation source illuminates the object.*
     - The absorption of radiation by the objects depends upon the nature of the material.
     - Some of the radiation passes through the objects. The attenuated radiation is sensed into an image. This is called Transmissive imaging.
     - ***X-Ray*** imaging is an example of Transmissive type imaging.
  - **Figure above shows 3 types of processing:**

1. ***Optical image processing*** is an area that deals with the object, optics, and how processes are applied to an image that is available in the form of reflected or transmitted.
   1. ***Analog image processing***
   2. An analog or continuous image is a continuous function f(x, y), where x and y are two spatial coordinates. Analog signals are characterized by continuous signals varying with time. They are often referred to as pictures. The processes that are applied to the analog signal are called analog processes.
   3. ***Analog image processing*** is an area that deals with the processing of analog electrical signals using analog circuits. The imaging systems that use film for recording images are also known as analog imaging systems.
   4. The analog signal is often sampled, quantized, and converted into digital form using a ***digitizer.Digitization*** refers to the process of sampling and quantization.
      - ***Sampling*** is the process of converting a continuous-valued image f(x, y) into

a discrete image, as computers cannot handle continuous data. So the main aim is to create a **discretized version of the continuous data.** Sampling is a reversible process, as it is possible to get the original image back.

- ❑ ***Quantization*** is the process of converting the sampled analog value of the function f(x, y) into a discrete-valued integer.

2. ***3. Digital image processing*** is an area that uses digital circuits, systems, and software algorithms to carry out the image processing operations. The image processing operations may include quality enhancement of an image, counting of objects, and image analysis.

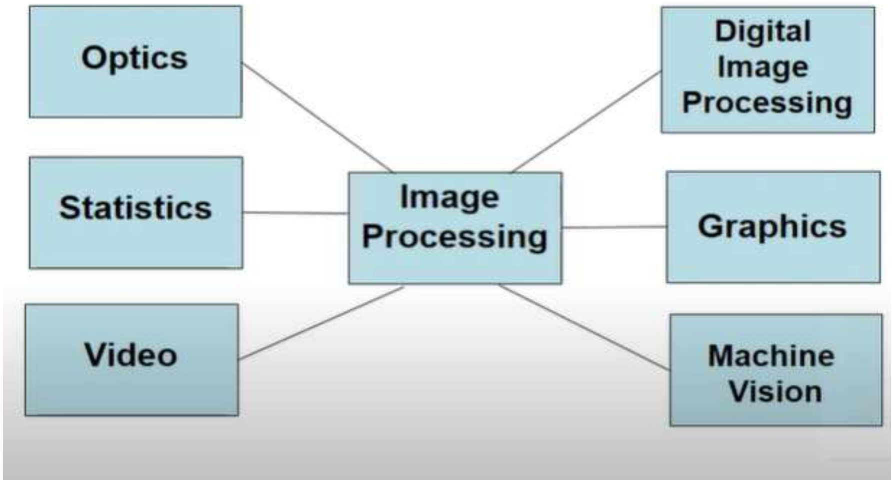
## Reasons for Popularity of DIP(Advantages)

1. It **is easy to post-process the image**.
2. **Small corrections can also be made** in the captured image using software.
3. It is **easy to store** the image in the digital memory.
4. It is possible to transmit the image over networks. So **sharing an image is quite easy**.
5. A digital image does not require any chemical process. So it is very **environment friendly**, as harmful film chemicals are not required or used.
6. It is easy to operate a digital camera.

## Disadvantages

- ➢ The disadvantages of digital images are very few.
- ➢ Some of the disadvantages are:
  - ❑ The initial cost,
  - ❑ problems associated with sensors such as high power consumption
  - ❑ potential equipment failure, and
  - ❑ other security issues associated with the storage and transmission of digital images.

## IMAGE PROCESSING AND RELATED FIELDS



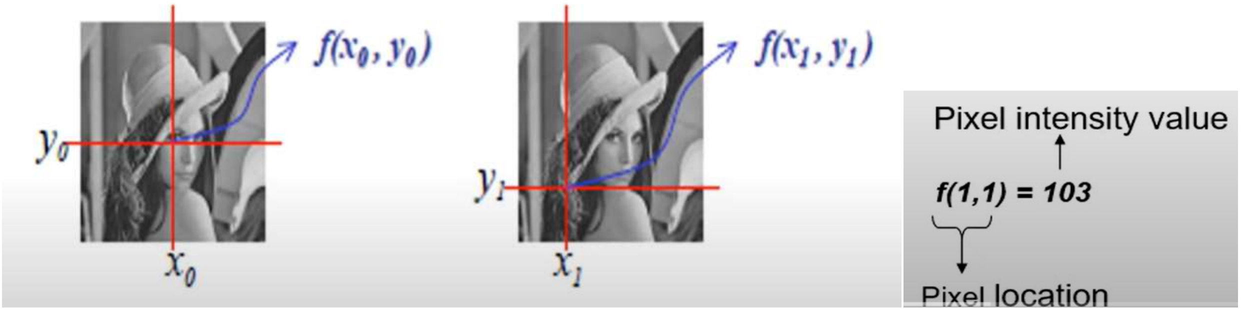

Relationship between image processing and other related fields.

- ➢ **Image processing & Computer Graphics**: Image processing deals with raster data or bitmaps, whereas computer graphics primarily deals with vector data.

- **Image processing & Signal Processing:** Digital signal processing deals with the processing of a one dimensional signal while image processing deals with visual information that is often in two or more dimensions.
- **Image processing & Machine vision:** The main goal of machine vision is to interpret the image and to extract its physical, geometric, or topological properties.
- **Image processing & Video Processing:** Image processing is about **still images**. A video can be considered as a collection of images indexed by time. Thus, video processing is an extension of image processing.
- **Image processing & Optics:** Optical image processing deals with lenses, light, lighting conditions, and associated optical circuits. The study of lenses and lighting conditions has an important role in the study of image processing.
- **Image processing & Statistics:** Statistics play an important role in image understanding and image analysis.

## Digital Image representation

- An image is a two-dimensional function that represents a measure of some characteristic such as brightness or color of a viewed scene.
- An image may be defined as a two-dimensional function ***f(x,y)***, where x and y are spatial (plane) coordinates, and the amplitude of ***f*** at any pair of coordinates ***(x,y)*** is called the **intensity** of the image at that point.




**Example: Let 1 represents white and 0 represents black color**


**Small Binary Digital Image** **Equivalent Image contents in matrix form**

- In general, the image can be written as a mathematical function **f(x, y)** as follows:

$$f(x, y) = \begin{bmatrix} f(1,1) & f(1,2) & .... & f(1,N) \\ f(2,1) & f(2,2) & .... & f(2,N) \\ \vdots & \vdots & \vdots & \vdots \\ f(M,1) & f(M,2) & .... & f(M,N) \end{bmatrix}$$

- The term **gray level** is used often to refer to the **intensity of monochrome images**.
- **Color images** are formed by a combination of individual 2-D images.
  - For example: The **RGB color** system.
- When x, y, and f (x, y) are all finite, discrete quantities, we call the image a **digital image**.
- Digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as:
  - **picture elements**
  - **image elemen**ts
  - **Pels**
  - **pixels**
- Pixel is the most widely used term.
- Image resolution depends on two factors—**optical resolution** of the lens and **spatial resolution.**
- **Optical resolution:**

  **-** Refers to the level of detail that an imaging system, such as a camera or scanner, can capture.

  **-** It is primarily determined by the physical characteristics of the imaging system, including the quality of its lenses, the size of it sensor or film, and other factors

  **-** Typically measured in terms of the number of pixels per unit length, such as pixels per inch (PPI) or pixels per millimeter (PPM).

  **-** Higher optical resolution means more pixels are used to represent the same area, resulting in finer detail and greater clarity in the captured image
- **Spatial resolution:**

  **-** Refers to the scale or size of the smallest unit of an image capable of distinguishing objects

  or

  **-**A measure of the smallest angular or linear distance to identify adjacent objects in an image.

  **-** Spatial resolution is a term utilized to describe how many pixels are employed to comprise a digital image

  **-**Depends on two parameters—

  **1.** The number of pixels of the image

  **2.** The number of bits necessary for adequate intensity resolution, referred to as the **bit depth**.

  **-** Number of pixels determines quality of digital image.

  **-**Total number pixels are measured by **X (Number of rows) x Y(Number of Columns)**

  -To represent Pixel intensity value, certain bits are required

  Example: Binary Image

  - Binary image pixels can have only two colors, usually black and white

  - Binary images are also called bi-level or two-level,

  - Pixelart made of two colours is often referred to as 1-Bit or 1bit.

  - This means that each pixel is stored as a single bit—i.e., a 0 or 1.

- Number of bits required to encode the pixel value called **Bit Depth**
- Bit Depth is **power of two, written as $2^m$**
- In **Monochrome Gray scale image**, pixel values can be rangefrom **0 to 255**
- **Eight bits** are required to represent monochrome gray scaleimage as **$2^8$ = 256 (Range from 0 to 255), -** So **Bit Depth** of Gray scale image **is 8**

- **So the total number of bits necessary to represent the image is = Number of rows *Number of columns * Bit depth**
- The concept of 2D images can be extended to 3D images also. A 3D Image is a finction (x, y, z), where x, y, and z are spatial coordinates.
- In 3D images, the term 'voxel' is used for pixel. **Voxel** is an abbreviation of **'volume element'**.

**Example 1.1** What is the storage requirement for a 1024×1024 binary image?

***Solution*** For a binary image, one bit is sufficient for representing the pixel value. So the number of bits required will be $1024 \times 1024 \times 1 = 10{,}48{,}576$ bits = 1,31,072 bytes = 131.072 Kb (Assume 1 Kb = 1000 bytes).

- Rows = 1024, Columns = 1024
- Binary image requires 1 bit for representation

No. of bits required = Rows x Columns x Bit Depth

**Example 1.2** What is the storage requirement for a 1024 × 1024 24-bit colour image?

***Solution*** Since colour images are three-band images (red, green, and blue components), the storage requirement is $1024 \times 1024 \times 3$ bytes = 31,45,728 bytes. If it is assumed that 1 Kb is 1000 bytes, the storage requirement is 3,145.728 Kb.

- Rows = 1024, Columns = 1024
- 24 Bits requires for colour representation

No. of bits required = Rows x Columns x Bit Depth

**Example 1.3** A picture of physical size 2.5 inches by 2 inches is scanned at 150 dpi. How many pixels would be there in the image?

*Solution* The relation between the physical dimensions and the spatial resolution is simple. The pixel dimensions are obtained by multiplying the physical width and height by the scanned resolution. Therefore, the pixel dimension is as follows.

$$(2.5 \times 150) \times (2 \times 150)$$

$$= 375 \times 300 = 112500 \text{ pixels would be present}$$

**Example 1.4** If a 375 × 300 grey-scale image needs to be sent across the channel of capacity 28 kbps, then how much transmission time is required?

*Solution* If the picture is grey scale, then 8 bits are used. Therefore, transmission time would be

$$= \frac{375 \times 300 \times 8}{28 \times 1000} = \frac{112500 \times 8}{28000} = 32.143 \text{ sec}$$

**Example 1.5** Given a grey-scale image of size 5 inches by 6 inches scanned at the rate of 300 dpi, answer the following:

(a) How many bits are required to represent the image?
(b) How much time is required to transmit the image if the modem is 28 kbps?
(c) Repeat the aforementioned if it were a binary image.

*Solution*

(a) Number of bits required to represent grey-scale image (uses 8 bits)

$$= 5 \times 300 \times 6 \times 300 \times 8 = 1500 \times 1800 \times 8 = 21600000 \text{ bits}$$

(b) Total time taken to transmit image

$$= \frac{\text{Total number of bits in image}}{\text{Transmission Speed}} = \frac{21600000}{28000} = 771.43 \text{ sec}$$

(c) If it is binary image, then the number of bits required to represent binary image

$$= 5 \times 300 \times 6 \times 300 \times 1 = 1500 \times 1800 \times 1 = 2700000 \text{ bits}$$

The total transmission time would be $= \frac{\text{Total number of bits}}{\text{Transmission speed}} = \frac{2700000}{28000} = 96.429$ sec

## Classification of Images

**1.Based on Nature:**

i. **Natural:** Images of natural objects obtained using devices called cameras or scanners.
ii. **Synthetic:** Images that are generated using computer programs.

**2.Based on Attributes:**

i. **Raster:** Pixel based, quality based on no. of pixels

ii. **Vector graphics:** Use basic geometric attributes such as line andcircles to describe image

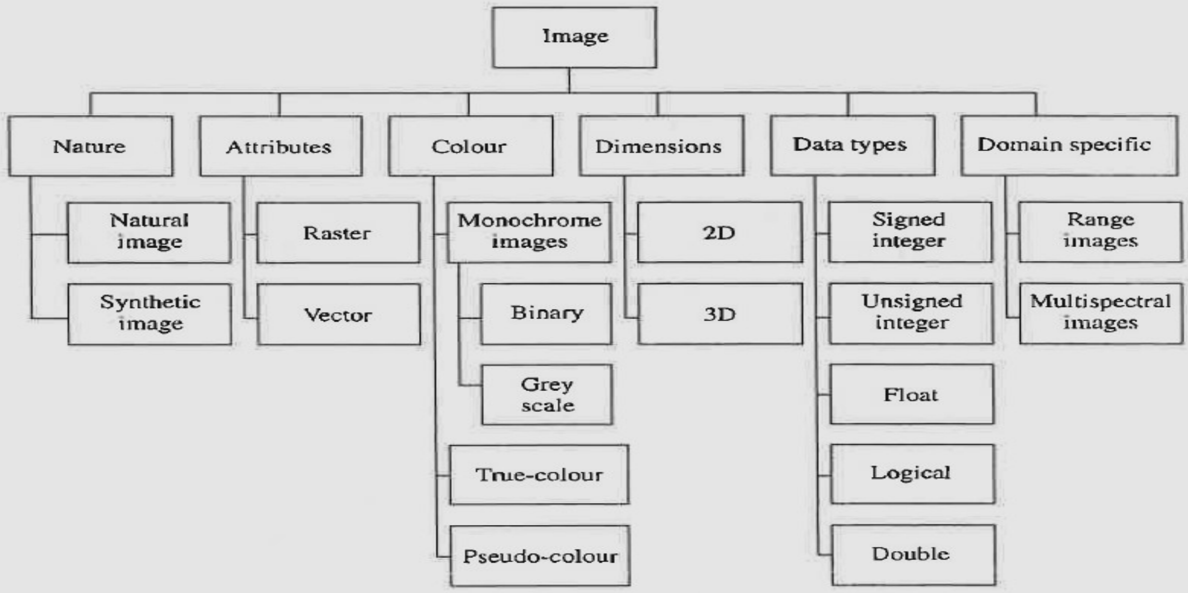



**Fig. 1.4** Classification of images

**3. Based on Colour:**

i. **Grey scale** images are different from binary images as they have many shades of grey between black and white. These images are also **called monochromatic** as there is no colour component in the image, like in binary images. Grey scale is the term that refers to the range of shades between white and black or vice versa.Gray scale use 8 bits representation, produce 28 =256 levels. Human visual system can distinguish only 32 different gray levels

ii. **In binary images,** the pixels assume a value of 0 or 1. So one bit is sufficient to represent the pixel value. Binary images are also called bi-level images.

iii. **In true colour images**, the pixel has a colour that is obtained by mixing the primary colours **red, green, and blue**. Each colour component is represented using 8 bits, so**true colour images use 24 bits** to represent all the colours. That is **$2^{24}$ = 1,67,77,216 Colours**

iv. A special category of colour images is the indexed image. In most images, the full range of colours is not used. So it is better to reduce the number of bits by maintaining a colourmap, gamut, or palette with the image.

v. Like true colour images, **Pseudo-colour images** are also used widely in image processing. **True colour** images are called **three-band** images. However, in remote sensing applications, multi-band images or multi-spectral images are generally used. These images, which are captured by satellites, contain many bands.

**4. Based on Dimensions:**

- Images can be classified based on dimensions also.
- Normally, digital images are a **2D** rectangular array of pixels.
- If another dimension, of depth or any other characteristic, is considered, it may be necessary to use a higher-order stack of images**(3D)**.
- A good example of a 3D image is a volume image, where pixels are called **voxels.**
- By **'3D image'**, it is meant that the dimension of the target in the imaging system is 3D.
- Example: CT images, MRIs, and microscopy images.

**5.Based on Data Types:**

- Sometimes, image processing operations produce images with negative numbers, decimal fractions, and complex numbers.
- **To handle negative numbers, signed and unsigned integer types are used.** In these data types, the first bit is used to encode whether the number is positive or negative.
- **Floating-point** involves storing the data in scientific notation. For example, 1230 can be represented as 0.123×10, where 0.123 is called the significant and the power is called the exponent. There are many floating-point conventions.
- The quality of such data representation is characterized by parameters such as data accuracy and precision. For this we use **double datatype.**

**6.Based on Domain specific:**

- Images can classified based on domains and applications

**i. Range Images:**

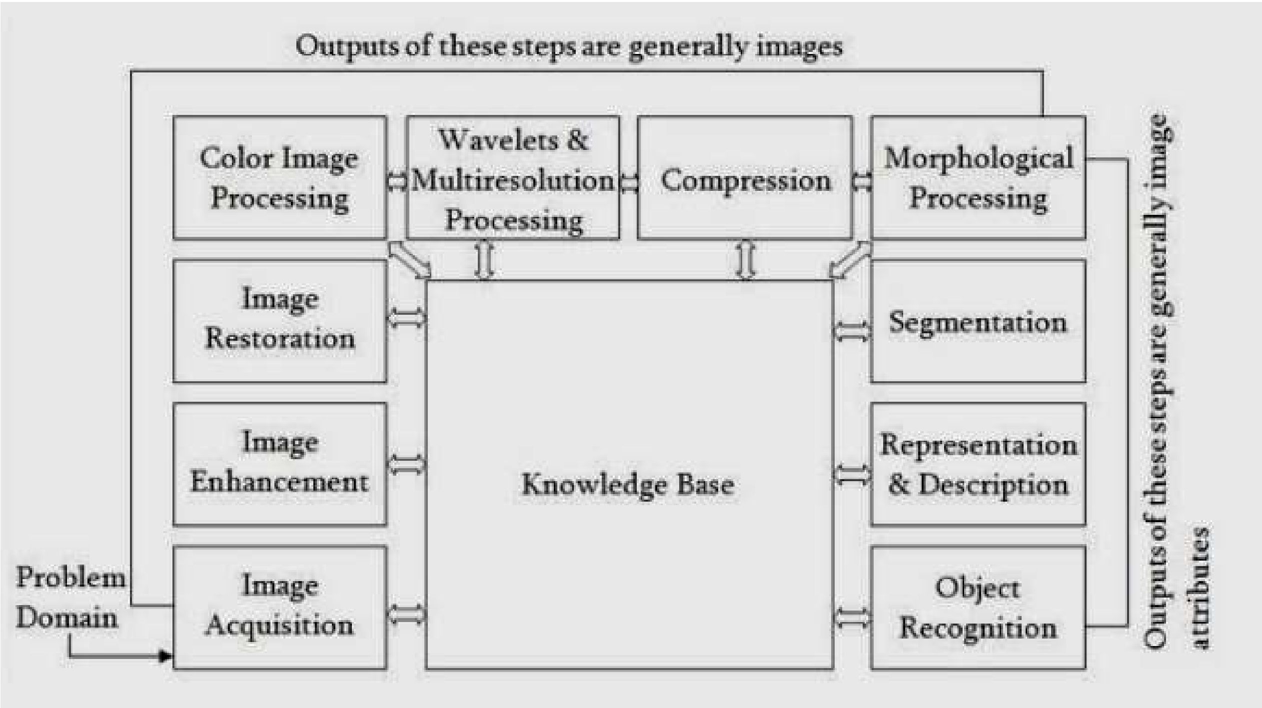
- Range images are often encountered in computer vision. In these images, Pixel values denote distance between object and camera. These images are referred to as depth images.

**ii. Multispectral Images**

- Multispectral Images are often encountered in remote sensing applications. These images may have many bands that may include infrared and ultravioletregions of EM spectrum.

## Fundamental Steps in Digital Image Processing

- The different tasks being done in DIP are divided into two broad categories based on theoutput of the study and all the DIP methods or techniques must have output of any of the shapes amongst the two categories, the categories are:
  1. Input is an image and the output is also an image.
  2. Input is the image and output is the attribute extracted from an image.
- Below is the graphical notation for the fundamental steps in image processing



**1. Image Acquisition:**

- In image processing, it is defined as the action of retrieving an image from some source, usually a hardware-based source for processing.
- It is the first step in the workflow sequence because, without an image, no processing is possible. The image that is acquired is completely unprocessed. In image acquisition using pre-processing such as scaling is done.

**2. Image Enhancement:**

- It is the process of adjusting digital images so that the results are more suitable for display or further image analysis. Usually it includes sharpening of images, brightness & contrast adjustment, removal of noise, etc.
- In image enhancement, we generally try to modify the image, so as to make it more pleasing to the eyes.
- It is subjective in nature as for example some people like high saturation images and some people like natural colour. That's why it is subjective in nature as it differs from person to person.

**3. Image Restoration:**

- It is the process of recovering an image that has been degraded by some knowledge of degraded function H and the additive noise term. Unlike image enhancement, image restoration is completely objective in nature.

**4. Color Image Processing:**

- Color image processing is an area that has been gaining its importance because of the significant increase in the use of digital images over the Internet. This may include color modeling and processing in a digital domain etc. This handles the image processing of colored images either as indexed images or RGB images.

**5. Wavelets and multiresolution processing:**

- Wavelets are small waves of limited duration which are used to calculate wavelet transform which provides time-frequency information.
- Wavelets lead to multiresolution processing in which images are represented in various degrees of resolution.

**6. Compression:**

- Compression deals with the techniques for reducing the storage space required to save an image or the bandwidth required to transmit it.
- This is particularly useful for displaying images on the internet as if the size of the image is large, then it uses more bandwidth (data) to display the image from the server and also increases the loading speed of the website.

**7. Morphological Processing:**

- It deals with extracting image components that are useful in representation and description of shape.
- It includes basic morphological operations like erosion and dilation.
- As seen from the block diagram that the outputs of morphological processing generally are image attributes.

**8. Segmentation:**

- It is the process of partitioning a digital image into multiple segments. It is generally used to locate objects and boundaries in objects.
- In general, autonomous segmentation is one of the most difficult tasks in digital image processing. A segmentation procedure brings the process a long way toward successful solution of imaging problems that require objects to be identified individually.

**9. Representation and Description:**

- **Representation** deals with converting the data into a suitable form for computer processing.
    - **Boundary representation**: it is used when the focus is on external shape characteristics e.g. corners
    - **Regional representation**: it is used when the focus in on internal properties e.g. texture
- **Description** deals with extracting attributes that
    - results in some quantitative information of interest
    - is used for differentiating one class of objects from others

**10. Recognition:**

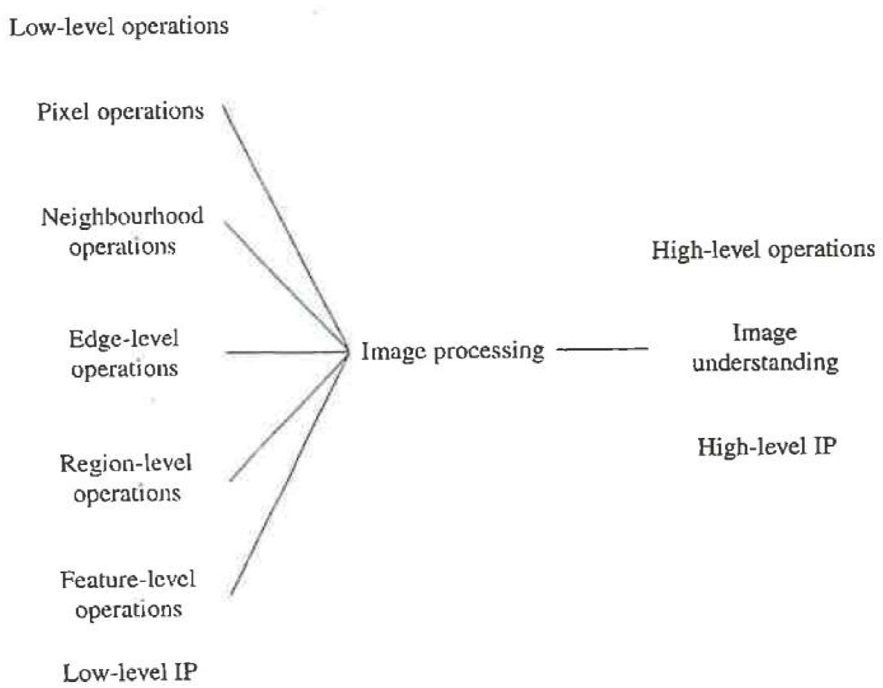
- It is the process that assigns a label (e.g. Notebook, Laptop) to an object based on its description.
- It is the last step of image processing which use artificial intelligence.

**Knowledge Base:**

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

## Digital Image Processing Operations

- Generally, image processing operations are divided into two categories as follows
    - Low-level Operations
    - High-level Operations
- Image acquisition, preprocessing and compression are consideredas **low level operations.**
- Image segmentation and feature extraction deal with the extraction of necessary image portions and analysis of image features. Hence, segmentation and feature extraction are considered to be important areas, these stage serve link between low-level and high-levelimage processing
- **High level image processing** deals with image understanding, Image interpretation in more meaningful manner. It is based on knowledge, goals and plans. Image understanding uses concept of Artificial Intelligence toimitate human cognition.




## Basic Relationships and Distance Metrics

- The various neighborhood relationships between pixels and Distancemetrics between image points(pixels) are discussed in this section.

## 1.Image Coordinate System:

- In image co-ordinate system, images can be easily represented as atwo-dimensional array matrix.
- The popularity of the matrix form is due to the fact that most of the programming languages support 2D array data structure and can easily implement matrix-level computation.
- Pixels can be visualized **logically and physically**.
- **Logical pixels** specify the points of a continuous 2D function. These are logical in the sense that they specify a location but occupy no physical area. Normally, this is represented in the Cartesian first coordinate system.
- **Physical pixels,** on the other hand, occupy a small amount of space when displayed on the output device.
- The pixels array is represented by the **Cartesian co-ordinate system.**
- **For example,** an analog image of size 3 x 3 is represented in the first quadrant of the Cartesian coordinate system as shown in Fig. 3.1.
- Figure 3.1 illustrates an image f(x, y) of dimension 3 x 3, where f(0, 0) is the bottom left corner. Since it starts from the coordinate position (0, 0), it ends with f(2, 2), that is, x=0, 1, 2, ..., M-1 and y = 0, 1, 2, ..., N-1. x and y define the dimensions of the image.

$$
\begin{array}{r|lll}
2 & f(0,2) & f(1,2) & f(2,2) \\
1 & f(0,1) & f(1,1) & f(2,1) \\
y=0 & f(0,0) & f(1,0) & f(2,0) \\
\hline
 & x=0 & 1 & 2
\end{array}
$$

**Fig. 3.1** Analog image *f(x, y)* in the first quadrant of Cartesian coordinate system

- However, in digital image processing, the discrete form of the image is often used. Discrete images are usually represented in the fourth quadrant of the Cartesian coordinate system. A discrete image f(x, y), of dimension 3 x 3, is shown in Fig. 3.2(a).
- Many programming environments including MATLAB start with an index of (1, 1). The equivalent representation of the given matrix is shown in Fig. 3.2(b).
- The coordinates used for discrete image is, by default, the fourth quadrant of the Cartesian system.

$$
\begin{array}{c|ccc}
 & n=0 & n=1 & n=2 \\
\hline
m=0 & f(0,0) & f(0,1) & f(0,2) \\
m=1 & f(1,0) & f(1,1) & f(1,2) \\
m=2 & f(2,0) & f(2,1) & f(2,2)
\end{array}
\qquad
\begin{array}{c|ccc}
 & n=1 & n=2 & n=3 \\
\hline
m=1 & f(1,1) & f(1,2) & f(1,3) \\
m=2 & f(2,1) & f(2,2) & f(2,3) \\
m=3 & f(3,1) & f(3,2) & f(3,3)
\end{array}
$$

(a) (b)

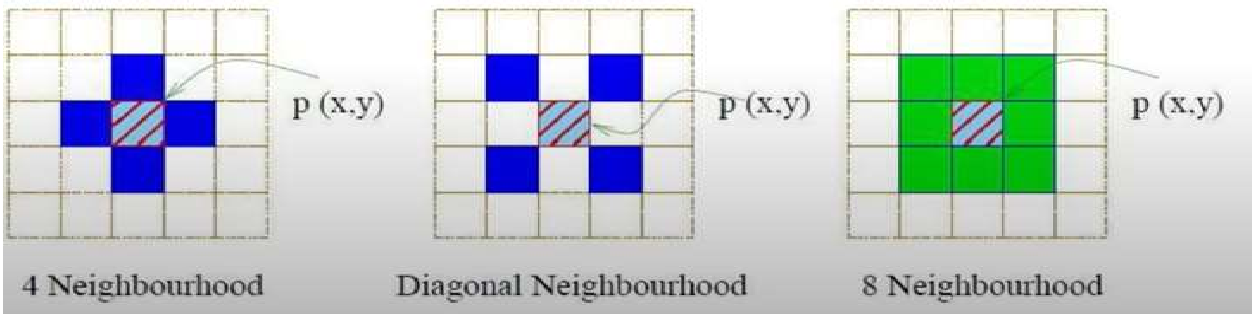
**Fig. 3.2** Discrete image (a) Image in the fourth quadrant of Cartesian coordinate system (b) Image coordinates as handled by software environments such as MATLAB

## 2. Image Topology

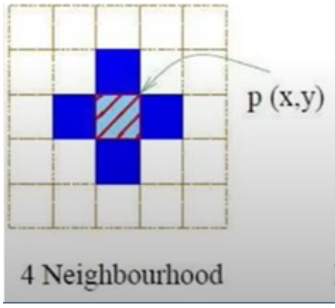
- Image topology is a branch of image processing that deals with the fundamental properties of the image such as image **neighborhood, paths among pixels, boundary, and connected components.**
- It characterizes the image with topological properties such as neighborhood, adjacency, and connectivity.

### Neighborhood

- Neighbourhood is fundamental to understanding image topology.
- In the simplest case, the neighbours of a given reference pixel are those pixels with which the given reference pixel shares its edges and corners.




- **In (4 Neighbourhood) $N_4$(p),** the reference pixel p(x, y) at the coordinate position (x, y) has two horizontal and two vertical pixels as neighbours. This is shown graphically below. Thus from figure, we have 4 neighbours of **p**are:**{(x-1,y), (x+1,y), (x,y+1), (x,y-1)}**

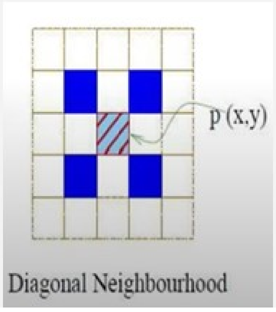



$$\begin{pmatrix} 0 & X & 0 \\ X & p(x,y) & X \\ 0 & X & 0 \end{pmatrix}$$

**Fig. 3.3** 4-Neighbourhood $N_4(p)$

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

- The **Diagonal Neighbours** of pixel p(x,y) are represented as $\mathbf{N_D(p)}$. The diagonal neighbours are: **{(x-1,y+1), (x+1,y+1), (x-1,y-1), (x+1,y-1)}**

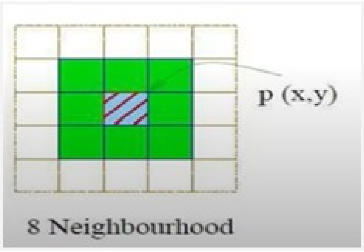



$$\begin{pmatrix} X & 0 & X \\ 0 & p(x,y) & 0 \\ X & 0 & X \end{pmatrix}$$

**Fig. 3.4** Diagonal elements $N_D(p)$

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

  - The 4-neighbourhood ($N_4$) and $\mathbf{N_D}$ are collectively called the **8-Neighbourhood ($N_8$)**. This refers to all the neighbours and pixels that share a common corner with the reference pixel **p(x, y).** These pixels are called indirect neighbours. This is represented as $\mathbf{N_8(p)}$ and is shown graphically in Fig. 3.5.
  - The set of pixels $N_8(x) = N_4(x) \cup N_D(x)$




$$\begin{pmatrix} X & X & X \\ X & p(x,y) & X \\ X & X & X \end{pmatrix}$$

**Fig. 3.5** 8-Neighbourhood $N_8(p)$

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

## Connectivity /Adjacency/Neighbors Connectivity

- The relationship between two or more pixels is defined by pixel connectivity.
- Connectivity information is used to establish the boundaries of the objects.
- The pixels p and q are said to be connected if certain conditions on pixel brightness specified by the set V and spatial adjacency are satisfied.
- For a binary image the set 'V' will be 0,1 and for gray scale images, 'V'might be any range of gray levels.

1. **4-Connectivity** The pixels p and q are said to be in **4-connectivity** when both have the same values as specified by the set V and if q is said to be in the set $\mathbf{N_4(p)}$. This implies any path from p to q on which every other pixel is 4-connected to the next pixel.

$$\boxed{p \text{ and } q \text{ are 4-connected if } q \in N_4(p)}$$

2. **8-Connectivity** It is assumed that the pixels p and q share a common grey scale value. The pixels p and q are said to be in 8-connectivity if q is in the set $\mathbf{N_8(p)}$.

$$\boxed{p \text{ and } q \text{ are 8-connected if } q \in N_8(p)}$$

3. **Mixed connectivity** Mixed connectivity is also known as m-connectivity. Two pixels p and q are said to be in m-connectivity when
   a. q is in **$N_4$(p)**or
   b. q is in **$N_D$(p)** and the intersection of **$N_4$(p)** and **$N_4$(q)** is empty.

$$\boxed{\begin{array}{l} p \text{ and } q \text{ are m-connected if } q \in N_4(p) \text{ or} \\ q \in N_D(p) \text{ and } N_4(p) \cap N_4(q) = \varnothing \end{array}}$$

**For example,** Fig. 3.6.1 shows 4-connectivity Fig. 3.6.2 shows 4-connectivity when V= {0, 1}. 4- and 8-Connectivity is shown as lines. Here, a multiple path or loop is present. In m-connectivity, there are no such multiple paths. The m-connectivity for the image in Fig. 3.6.2 is as shown in Fig. 3.7.It can be observed that the multiple paths have been removed.

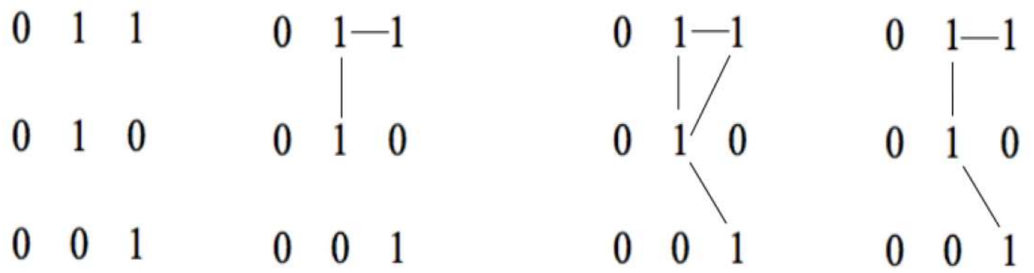


Fig: An arrangement of pixels | Fig 3.6.1: 4-connectivity of pixels | Fig 3.6.2 :8-connectivity of pixels | Fig 3.7: m-connectivity of pixels

## Relations

- A binary relation between two pixels **a** and **b**, denoted as ***aRb***, specifies a pair of elements of an image. For example, consider the image pattern given in Fig. 3.8.

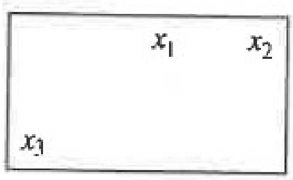



Fig. 3.8 Image pattern

- The set is given as **A= {x1, x2, x3}.** The set based on the **4-connectivity** relation is given as **A = (X1, X2).** It can be observed that x3 is ignored as it is not connected to any other element of the image by 4-connectivity.
- The following are the properties of the binary relations:

***Reflexive*** For any element $a$ in the set $A$, if the relation $aRa$ holds, this is known as a reflexive relation.

***Symmetric*** If $aRb$ implies that $bRa$ also exists, this is known as a symmetric relation.

***Transitive*** If the relations $aRb$ and $bRc$ exist, it implies that the relationship $aRc$ also exists. This is called the transitivity property.

- If all these properties hold, the relationship is called ***Equivalence relation.***

## Distance Measures

- The distance between the pixels **p** and **q** in an image can be given by distance measures such as **Euclidian distance($D_e$), $D_4$ distance, and $D_8$ distance.**
- Consider three pixels **p, q, and z.** If the coordinates of the pixels are **P(x, y), Q(s, t), and Z(u, w)** as shown in Fig. 3.9, the distances between the pixels can be calculated.

| | | | |
|---|---|---|---|
| 0 | 1 | 1 | 1(*z*) |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1(*q*) |
| 1 | 1 | 1 | 1 |
| (*p*) | | | |

Fig. 3.9 Sample image

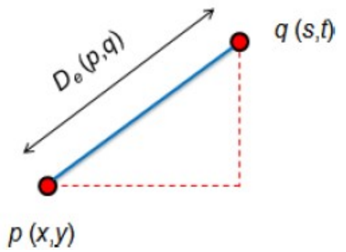
The distance function can be called metric if the following properties are satisfied:

1. $D(p, q)$ is well-defined and finite for all $p$ and $q$.
2. $D(p, q) \geq 0$ if $p = q$, then $D(p, q) = 0$.
3. The distance $D(p, q) = D(q, p)$.
4. $D(p, q) + D(q, z) \geq D(p, z)$. This is called the property of triangular inequality.

**i) Euclidian distance($D_e$)**

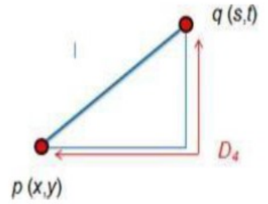
- The Euclidean Distance between p and q is definedas:

$$D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$$



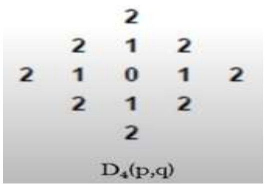

**ii) $D_4$ distance(also called city-block distance)**between p and q is definedas:

$$D_4(p,q) = |x - s| + |y - t|$$



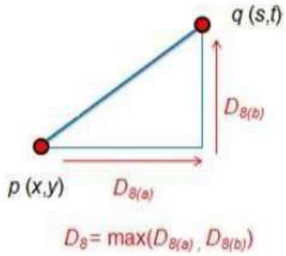

**Example:**

The pixels with distance D4≤ 2 from (x,y) form the following contours of constant distance. The pixels with D4= 1 are the 4-neighbors of (x,y)




**iii) $D_8$ distance (also called chessboard distance)** between p and q is definedas:

$$D_8(p,q) = \max(|x - s|, |y - t|)$$



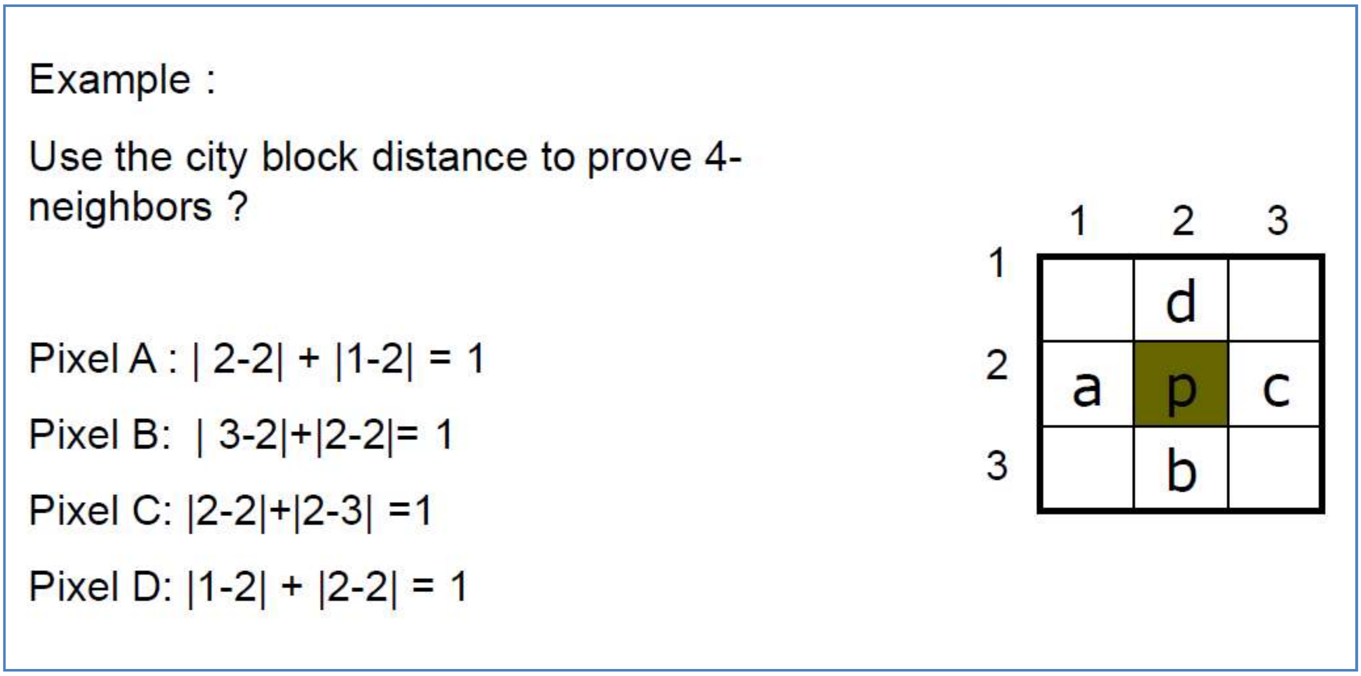

**Example:**

$D_8$ distance ≤ 2 from (x,y) form the following contours of constant distance. The pixels with D8=1 are the 8-neighbors of (x, y).

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

**iv) $D_m$ Distance:**

It is defined as the shortest m-path between the points.In this case, the distance between two pixels will depend on the values of the pixels along the path, as well as the values of their neighbors.



**Example**

Consider the following arrangement of pixels and assume that P , $P_2$ , and $P_4$ have value 1 and that $P_1$ and $P_3$ can have a value of 0 or 1:

| | | |
|---|---|---|
| | $P_3$ | $P_4$ |
| $P_1$ | $P_2$ | |
| P | | |

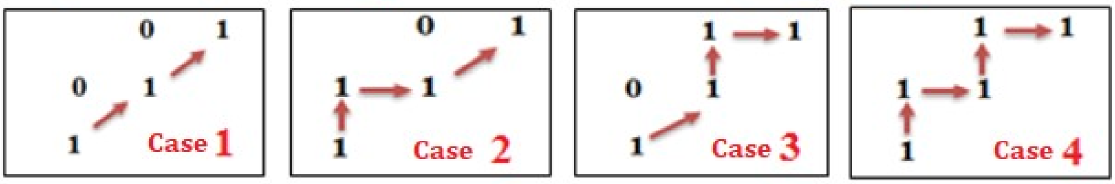
Suppose that we consider adjacency of pixels valued 1 (i.e., v={1} )

Case 1- If $P_1$ and $P_3$ are 0, The m-path($D_m$ distance between P and $P_4$ is 2

Case 2- If $P_1$ = 1 and $P_3$ = 0 The $D_m$ distance between P and $P_1$ $P_2$ $P_4$ is 3

Case 3- if $P_3$=1 and $P_1$ = 0 The $D_m$ distance between P and $P_2$ $P_3$ $P_4$ is 3

Case 4- if $P_3$=1 and $P_1$ = 1 The $D_m$ distance between P and $P_1$ $P_2$ $P_3$ $P_4$ is 4



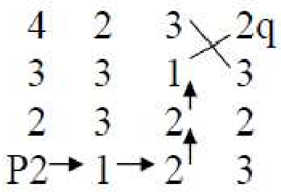

**Example:**

Consider the image segment shown in figure. Compute length of the shortest-4, shortest-8 & shortest-m paths between pixels p& q where, v={1,2}

| | | | |
|---|---|---|---|
| 4 | 2 | 3 | 2q |
| 3 | 3 | 1 | 3 |
| 2 | 3 | 2 | 2 |
| P2 | 1 | 2 | 3 |

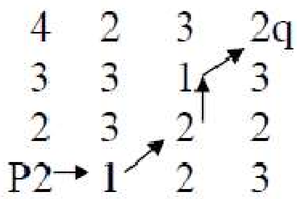
**Solution:**

**1. Shortest-4 path**



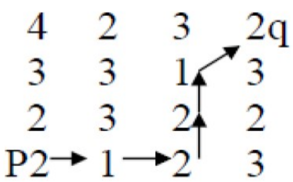
So, no path exist

**2. Shortest-8 path**



**So, shortest-8 path is 4.**

**3. Shortest-m path**



**So, shortest-m path is 5.**

**Example 3.1** Let $V = \{0, 1\}$. Compute the $D_e$, $D_4$, $D_8$, and $D_m$ distances between two pixels $p$ and $q$. Let the pixel coordinates of $p$ and $q$ be (3, 0) and (2, 3), respectively, for the image shown in Fig. 3.10.

Find the distance measures.

$$\begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 2 & 1 & 1 & 1 & 1(q) \\ 3 & 1 & 1 & 1 & 1 \\ & (p) & & & \end{array}$$

**Fig. 3.10** Sample image

***Solution*** The Euclidean distance is

$$D_e = \sqrt{(x-s)^2 + (y-t)^2} = \sqrt{(3-2)^2 + (0-3)^2}$$

$$= \sqrt{1+9} = \sqrt{10}$$

$$D_4 = |x-s| + |y-t| = |3-2| + |0-3|$$

$$= 1 + 3 = 4$$

$$D_8 = \max(|x-s|, |y-t|) = \max(|3-2|, |0-3|)$$

$$= \max(1, 3) = 3$$

**Example 3.2** Consider the following image with marked pixels $p$ and $q$.

$$\begin{array}{cccc} 3 & 1 & 2 & 1\ (q) \\ 2 & 2 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 1 & 0 & 1 & 2 \\ (p) & & & \end{array}$$

If $V = \{1, 2\}$, find the shortest $m$-path between pixels $p$ and $q$.

***Solution*** The shortest path for set $V$ {1,2} is shown here:

$$\begin{array}{cccc} 3 & 1 & 2 & 1\ (q) \\ 2 & 2 & 0 & 2 \\ 1 & 2 & 1 & 1 \\ 1 & 0 & 1 & 2 \\ (p) & & & \end{array}$$

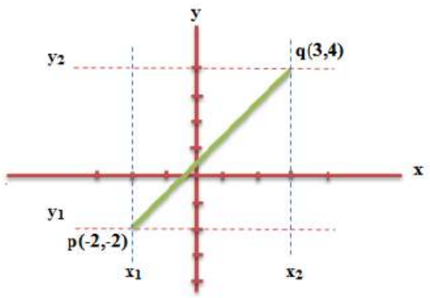
The shortest $m$-path is 1 – 2 – 1 – 2 –1. Therefore, $D_m$ distance is 4.

**Example :** **Suppose p and q are two pixels. Calculate the distance measures $D_4$ , $D_8$ and $D_e$**

1. $D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2}$

**P(3,4) and q(-2,-2)**

$$D_e(p,q) = \sqrt{((-2)-3))^2 + ((-2)-4))^2}$$
$$D_e(p,q) = \sqrt{(-5)^2 + (-6)^2}$$
$$D_e(p,q) = \sqrt{25+36}$$
$$D_e(p,q) = \sqrt{61}$$
$$D_e(p,q) = 7.8$$




2. $D_4(p,q) = |x-s| + |y-t|$

$= |(-2)-3)| + |(-2)-4|$

$= 5+6$

$= 11$

3. $D_8(p,q) = max(|(x-s)|, |(y-t)|)$

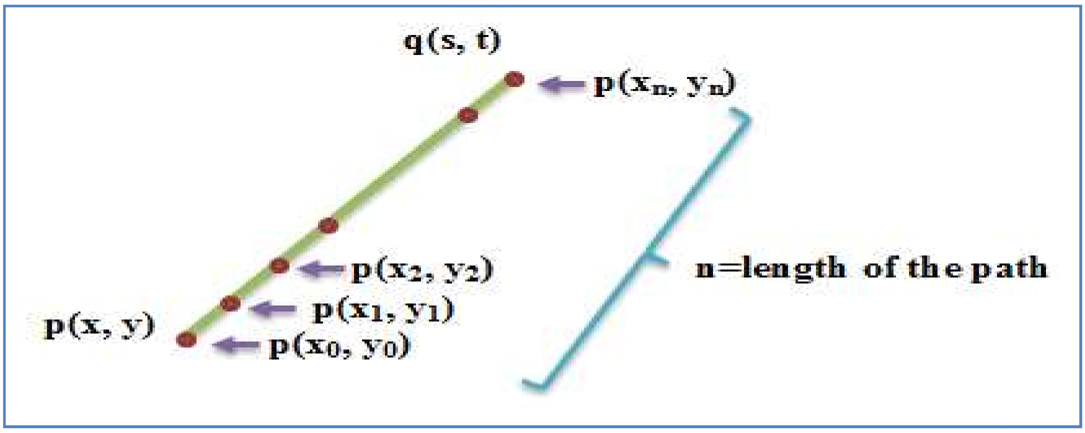
$= max(|(-2)-3|, |(-2)-4|)$

$= max(5,6)$

$= 6$

## Path

- A digital path (or curve) from pixel p with coordinate **(x, y)** to pixel q with coordinate **(s, t)** is a sequence of distinct pixels with coordinates **$(x_0, y_0)$, $(x1, y1)$, ..., $(xn, yn)$,** where**$(x_0, y_0)$= (x, y), $(x_n, y_n)$= (s, t)**
- **$(x_i, y_i)$**and **$(x_{i-1}, y_{i-1})$ are** adjacent pixel for **1≤j≤n** ,
- **n**- The length of the path.
- If **$(x_0, y_0)$= $(x_n, y_n)$**:the path is closed path.
- We can define ***4- ,8- , or m-paths*** depending on the type of adjacency specified.




## Region

- A connected set is also called a **Region.**
- Two regions (let Ri and Rj) are said to be adjacent if their union forms a connected set.

Adjacent Regions or joint regions

- Regions that are not adjacent are said to be disjoint regions.
- 4- and 8-adjacency is considered when referring to regions

- Discussing a particular region, type of adjacency must be specified.
- Fig. below the two regions are adjacent only if 8-adjacency is considered

$$
\begin{array}{ccc}
1 & 1 & 1 \\
1 & 0 & 1 \\
0 & 1 & 0 \\
\hline
0 & 0 & 1 \\
1 & 1 & 1 \\
1 & 1 & 1
\end{array}
\begin{array}{l}
\left.\begin{array}{c} \\ \\ \\ \end{array}\right\} R_i \\
\left.\begin{array}{c} \\ \\ \\ \end{array}\right\} R_j
\end{array}
$$

## Boundary

- The boundary (border or contour) of a region R is the set of points that are adjacent to the points in the complement of R.
- Set of pixels in the region that have at least one background neighbor.
- The boundary of the region R is the set of pixels in the region that have one or more neighbors that are not in R.
- Inner Border: Border of Foreground
- Outer Border: Border of Background

## Classification of image processing operations

- There are various ways to classify image operations.
- One way to characterize the operations based on neighbourhood is as follows
  1. Point operations
  2. Local operations
  3. Global operations
- **Point operations** are those whose output value at a specific co-ordinate is dependent only on the input value.
- A **local operation** is one whose output value at a specific co-ordinate is dependent on the input values in the neighborhood of that pixel.
- **Global operations** are those whose output value at a specific co-ordinate is dependent on all the values in the input image.
- Another way of categorizing the operations is as follows:
  1. Linear operations
  2. Non-linear operations
- An operator is called a linear operator if it obeys the following rules of **Additivity and Homogeneity**.

  **1. Property of Additivity**

$$H(a_1 f_1(x,y) + a_2 f_2(x,y)) = H(a_1 f_1(x,y)) + H(a_2 f_2(x,y))$$

$$= a_1 H(f_1(x,y)) + a_2 H(f_1(x,y))$$

$$= a_1 g_1(x,y) + a_2 g_2(x,y)$$

**2. Property of Homogeneity**

$$H(kf_1(x,y)) = kH(f_1(x,y)) = kg_1(x,y)$$

- A non-linear operator does not follow above operations.
- Image operations are **array operations**. These operations are done on a **pixel-by-pixel basis**.
- Array operations are different from matrix operations.
- For example, consider two images

$$F_1 = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \text{and } F_2 = \begin{pmatrix} E & F \\ G & H \end{pmatrix}$$

- The **multiplication of F1, and F2 is element-wise**, as follows:

$$F_1 \times F_2 = \begin{pmatrix} AE & BF \\ CG & HD \end{pmatrix}$$

- The multiplication of F1, and F2 is element-wise, as follows:
- In addition, one can observe that F1 * F2 = F2 * F1, whereas matrix multiplication is clearly different, since in matrices, A*B ≠B*A.
- By default, image operations are array operations only.

## Arithmetic Operations

- Arithmetic operations in images include Addition, Subtraction, Multiplication, and Division.

## 1. Image Addition

- Two images can be added in a direct manner, as given by

$$g(x,y) = f_1(x,y) + f_2(x,y)$$

- The pixels of the input images f1(x, y) and f2(x, y) are added to obtain the resultant image g(x,y).
- Similarly it is possible to add a constant value to a single image as follows to increase its brightness or intensity

$$g(x,y) = f_1(x,y) + k$$

## 2. Image Subtraction

- The Subtraction of two images can be done as follows:

$$g(x,y) = f_1(x,y) - f_2(x,y)$$

- The subtraction is a modulus operation.

$$|g(x,y) = f_1(x,y) - f_2(x,y)|$$

- Similarly it is possible to subtract with a constant value also

$$g(x,y) = |f_1(x,y) - k|$$

- To decrease the intensity or brightness.
- Some of the practical applications of image subtraction are:
  1.Background Elimination
  2.Brightness reduction
  3.Change detection

## 3. Image Multiplication

- Image Multiplication can be done in the following manner:

$$g(x, y) = f_1(x, y) * f_2(x, y)$$

- Scaling the given image by a constant 'k' can be performed as follows:

$$g(x, y) = f_1(x, y) * k$$

- If 'k' is less than 1, contrast of the image decreases, if 'k' is greater than 1, contrast increases.

## 3. Image Division

- Image Division operation can done as follows:

$$g(x, y) = f_1(x, y)/f_2(x, y)$$

- The division operation may result in floating-point numbers, hence float datatype is used in programming for this operation.
- Division using a constant(k) can also be performed as:

$$g(x, y) = f(x, y)/k$$

Table 3.1 Data type and allowed ranges

| S. no. | Data type | Data range |
|---|---|---|
| 1 | Uint8 | 0–255 |
| 2 | Uint16 | 0–65,535 |
| 3 | Uint32 | 0–4,29,49,67,295 |
| 4 | Uint64 | 0–1,84,46,74,40,73,70,95,51,615 |

**Example 3.3** Consider the following two images.

$$f_1 = \begin{pmatrix} 1 & 3 & 7 \\ 5 & 15 & 75 \\ 200 & 50 & 150 \end{pmatrix} \quad f_2 = \begin{pmatrix} 50 & 150 & 125 \\ 45 & 55 & 155 \\ 200 & 50 & 75 \end{pmatrix}$$

Perform addition, subtraction, multiplication, division, and image blending operations. Assume both the images are of the 8-bit integer type (uint8 of MATLAB type).

***Solution***

*Addition*:

$$g = f_1 + f_2 = \begin{pmatrix} 1+50 & 3+150 & 7+125 \\ 5+45 & 15+55 & 75+155 \\ 200+200 & 50+50 & 150+75 \end{pmatrix} = \begin{pmatrix} 51 & 153 & 132 \\ 50 & 70 & 230 \\ 400 & 100 & 225 \end{pmatrix}$$

If the data type uint8: is assumed, the minimum and maximum allowed values are 0 and 255, respectively. So if the value is larger than 255, it is reset to 255. Similarly, if the value is less than 0, it is reset to 0 (Table 3.1).

So the result of the image addition is

$$g = \begin{pmatrix} 51 & 153 & 132 \\ 50 & 70 & 230 \\ 255 & 100 & 225 \end{pmatrix}$$

*Subtraction*:

$$g = f_1 - f_2 = \begin{pmatrix} 1-50 & 3-150 & 7-125 \\ 5-45 & 15-55 & 75-155 \\ 200-200 & 50-50 & 150-75 \end{pmatrix} = \begin{pmatrix} -49 & -147 & -118 \\ -40 & -40 & -80 \\ 0 & 0 & 75 \end{pmatrix}$$

Since the data type is unit8, values less than 0 are reset to 0.

$$g = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 75 \end{pmatrix}$$

It can be observed that the modulus of the difference results in a different image.

*Multiplication*:

$$g = f_1 \times f_2 = \begin{pmatrix} 1\times 50 & 3\times 150 & 7\times 125 \\ 5\times 45 & 15\times 55 & 75\times 155 \\ 200\times 200 & 50\times 50 & 150\times 75 \end{pmatrix} = \begin{pmatrix} 50 & 450 & 875 \\ 225 & 825 & 11625 \\ 40000 & 2500 & 11250 \end{pmatrix}$$

Since the data type is unit8, values greater than 255 are reset to 255.

$$g = \begin{pmatrix} 50 & 255 & 255 \\ 225 & 255 & 255 \\ 255 & 255 & 255 \end{pmatrix}$$

*Division*:

$$g = f_1 / f_2 = \begin{pmatrix} 1/50 & 3/150 & 7/125 \\ 5/45 & 15/55 & 75/155 \\ 200/200 & 50/50 & 150/75 \end{pmatrix} = \begin{pmatrix} 0.02 & 0.02 & 0.056 \\ 0.11 & 0.272 & 0.484 \\ 1 & 1 & 2 \end{pmatrix}$$

$$g = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 1 & 2 \end{pmatrix}$$

This resultant image is due to truncation.

**Example 3.4** Consider the following 4 × 4, 8 level images $A$ and $B$. Find $A + B$, $A - B$, $A \times B$, and $A/B$.

$A =$

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
| 5 | 5 | 6 | 6 |
| 6 | 7 | 6 | 6 |
| 6 | 7 | 2 | 3 |

$B =$

| | | | |
|---|---|---|---|
| 1 | 3 | 5 | 7 |
| 8 | 7 | 0 | 1 |
| 3 | 5 | 6 | 7 |
| 1 | 3 | 5 | 7 |

***Solution*** The given images are 4 × 4 and 8 grey level (0 – 7) images. As the grey levels are 0 – 7, any value above 7 is reduced to 7.

$A + B =$

| | | | |
|---|---|---|---|
| 1 + 1 | 2 + 3 | 3 + 5 | 4 + 7 |
| 5 + 8 | 5 + 7 | 6 + 0 | 6 + 1 |
| 6 + 3 | 7 + 5 | 6 + 6 | 6 + 7 |
| 6 + 1 | 7 + 3 | 2 + 5 | 3 + 7 |

=

| | | | |
|---|---|---|---|
| 2 | 5 | 8 | 11 |
| 13 | 12 | 6 | 7 |
| 9 | 12 | 12 | 13 |
| 7 | 10 | 7 | 10 |

=

| | | | |
|---|---|---|---|
| 2 | 5 | 7 | 7 |
| 7 | 7 | 6 | 7 |
| 7 | 7 | 7 | 7 |
| 7 | 7 | 7 | 7 |

$A - B =$

| | | | |
|---|---|---|---|
| 1 – 1 | 2 – 3 | 3 – 5 | 4 – 7 |
| 5 – 8 | 5 – 7 | 6 – 0 | 6 – 1 |
| 6 – 3 | 7 – 5 | 6 – 6 | 6 – 7 |
| 6 – 1 | 7 – 3 | 2 – 5 | 3 – 7 |

=

| | | | |
|---|---|---|---|
| 0 | –1 | –2 | –3 |
| –3 | –2 | 6 | 5 |
| 3 | 2 | 0 | –1 |
| 5 | 4 | –3 | –4 |

=

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 6 | 5 |
| 3 | 2 | 0 | 0 |
| 5 | 4 | 0 | 0 |

$A \times B =$

| | | | |
|---|---|---|---|
| 1 × 1 | 2 × 3 | 3 × 5 | 4 × 7 |
| 5 × 8 | 5 × 7 | 6 × 0 | 6 × 1 |
| 6 × 3 | 7 × 5 | 6 × 6 | 6 × 7 |
| 6 × 1 | 7 × 3 | 2 × 5 | 3 × 7 |

=

| | | | |
|---|---|---|---|
| 1 | 6 | 15 | 28 |
| 40 | 35 | 0 | 6 |
| 18 | 35 | 36 | 42 |
| 6 | 21 | 10 | 21 |

=

| | | | |
|---|---|---|---|
| 1 | 6 | 7 | 7 |
| 7 | 7 | 0 | 6 |
| 7 | 7 | 7 | 7 |
| 6 | 7 | 7 | 7 |

$A/B =$

| | | | |
|---|---|---|---|
| 1/1 | 2/3 | 3/5 | 4/7 |
| 5/8 | 5/7 | 6/0 | 6/1 |
| 6/3 | 7/5 | 6/6 | 6/7 |
| 6/1 | 7/3 | 2/5 | 3/7 |

=

| | | | |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 6 |
| 2 | 1 | 1 | 0 |
| 6 | 2 | 0 | 0 |

## Applications of Arithmetic operations

- Arithmetic operations can be combined and put to effective use.
- For example, the image averaging operation can be used to remove noise in an image.
- Noise is a random fluctuation of pixel values, which affects the quality of an image.
- A noise image can be expresses as original image added with noise component as follows:

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

where f(x,y) is the input image and g(x,y) is the output image.

- Several instances of noise images can be taken and averaged as:

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

## Logical Operations

- Bit wise logical operations can be applied to image pixels.
- The resultant pixel is determined by the rules of the particular operation.
- Some of the logical operations that are widely used in image processing are as follows:

  **1. AND/NAND**

  **2. OR/NOR**

  **3. EXOR/EXNOR**

  **4. NOT**

## 1. AND/NAND

The truth table of the AND and NAND operators is given in Table 3.2.

Table 3.2 Truth table of the AND and NAND operators

| A | B | C (AND) | C (NAND) |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |

- The operators **AND and NAND** take two images as input and produce one output image.
- The output image pixels are the output of the logical AND/NAND of the individual pixels.
- Some of the practical applications of the AND and NAND operators are as follows:

  1. Computation of the intersection of images

  2. Design of filter masks

  3. Slicing of grey scale images

Figures 3.18(a)–3.18(d) shows the effects of the AND and OR logical operators. It illustrates that the AND operator shows overlapping regions of the two input images and the OR operator shows all the input images with their overlapping.

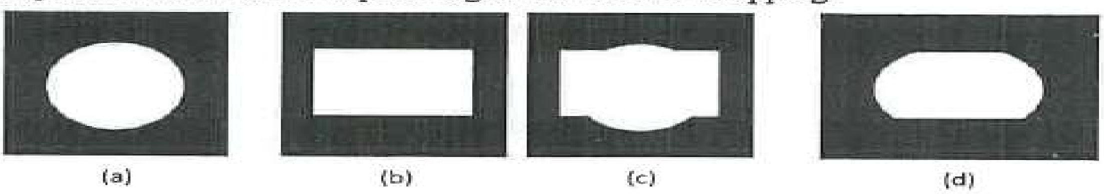



Fig. 3.18 Results of the AND and OR logical operators (a) Image 1 (b) Image 2 (c) Result of image 1 OR image 2 (d) Result of image 1 AND image 2

## 2. OR/NOR

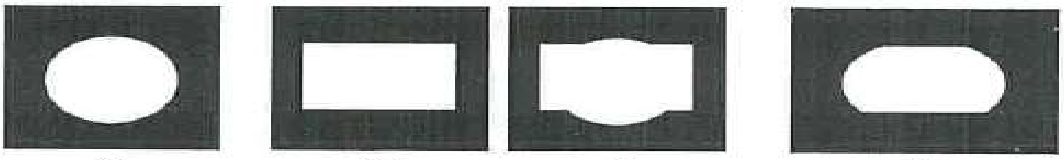
- The practical applications of the OR and NOR operators are as follows:
  1. OR is used as the union operator of two images.
  2. OR can be used as a merging operator.

The truth table of the OR and NOR operators is given in Table 3.3.

Table 3.3 Truth table of the OR and NOR operators

| A | B | C (OR) | C (NOR) |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 |

Figures 3.18(a)–3.18(d) shows the effects of the AND and OR logical operators. It illustrates that the AND operator shows overlapping regions of the two input images and the OR operator shows all the input images with their overlapping.



(a) (b) (c) (d)

Fig. 3.18 Results of the AND and OR logical operators (a) Image 1 (b) Image 2 (c) Result of image 1 OR image 2 (d) Result of image 1 AND image 2

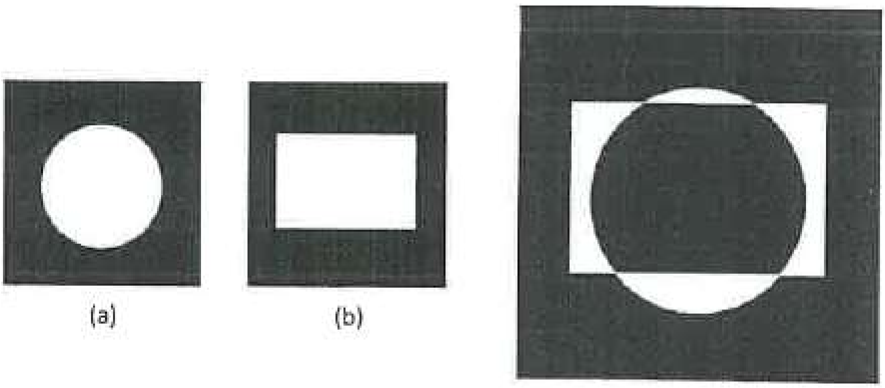
## 3. EXOR/EXNOR

The truth table of the XOR and XNOR operators is given in Table 3.4.

Table 3.4 Truth table of the XOR and XNOR operators

| A | B | C (XOR) | C (XNOR) |
|---|---|---|---|
| 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

- The practical applications of the XOR and XNOR operators are as follows:
  1. Change detection
  2. Use as a subcomponent of a complex imaging operation. XOR for identical inputs is zero. Hence it can be observed that the common region of image 1 and image 2 in Figs (a) and (b), respectively, is zero and hence dark. This is illustrated in Fig. 3.19.



(a) (b)

Fig. 3.19 Result of the XOR operation

## 4. Invert/Logical NOT

Table 3.5 Truth table of the NOT operator

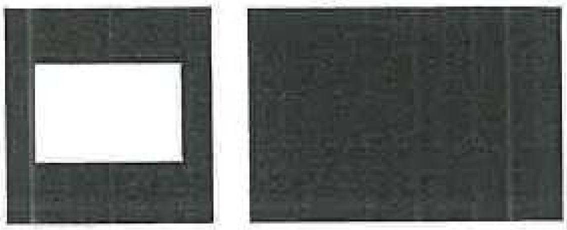
| A | C (NOT) |
|---|---|
| 0 | 1 |
| 1 | 0 |

- For grey scale values, the inversion operation is described as

  **g(x, y)=255-f(x, y)**

- The practical applications of the inversion operator are as follows:

  1. Obtaining the negative of an image. Figure 3.20 (b) shows the negative of the original image shown in Fig. 3.20(a).
  2. Making features clear to the observer
  3. Morphological processing



(a) (b)

Fig. 3.20 Effect of the NOT operator (a) Original Image (b) NOT of original Image

**Example 3.5** Consider the following two images.

$$f_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} f_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Perform the logical AND, OR, NOT

***Solution***

AND

$$f_1 \text{ AND } f_2 = \begin{pmatrix} 1 \wedge 1 & 0 \wedge 1 & 0 \wedge 1 \\ 1 \wedge 1 & 1 \wedge 1 & 1 \wedge 1 \\ 0 \wedge 1 & 0 \wedge 1 & 1 \wedge 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

OR

$$f_1 \text{ OR } f_2 = \begin{pmatrix} 1 \vee 1 & 0 \vee 1 & 0 \vee 1 \\ 1 \vee 1 & 1 \vee 1 & 1 \vee 1 \\ 0 \vee 1 & 0 \vee 1 & 1 \vee 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

NOT

$$\mathrm{NOT}(f_1) = \begin{pmatrix} \neg 1 & \neg 0 & \neg 0 \\ \neg 1 & \neg 1 & \neg 1 \\ \neg 0 & \neg 0 & \neg 1 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

Difference

$$f_1 \,\mathrm{AND}\, (\neg f_2) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

and

$$f_2 \,\mathrm{AND}\, (\neg f_1) = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$

## Geometric operations

### 1. Translation

- Translation is the movement of an image to a new position.
- Let us assume a point at the co-ordinate position ***X=(x, y)*** of the matrix ***F*** is moved to the new position ***X ′***, whose co-ordinate position is ***(x′, y′).***
- Mathematically this can be stated as a translation of a point ***X*** to the new position ***X′***.
- The translation is represented as

$$x' = x + \delta x$$
$$y' = y + \delta y$$

- In a Homogeneous Co-ordinate system, in matrix form as:

$$[x', y', 1] = \begin{pmatrix} 1 & 0 & \delta x \\ 0 & 1 & \delta y \\ 0 & 0 & 1 \end{pmatrix} [x, y, 1]^T$$

### 2. Scaling

- Scaling means enlargement and shrinking.
- In the homogeneous co-ordinate system, the scaling of the point (x,y) of the image F to the new point (x′, y′) of the image F′ is described as:

$$x' = x * Sx$$
$$y' = y * Sy$$

$$[x', \quad y'] = \begin{pmatrix} S_x & 0 \\ 0 & S_y \end{pmatrix} [x, \quad y]$$

- Where Sx and Sy are called scaling factors along the x and y axis, respectively.
- If the scale factor is 1, the object would appear larger.
- If the scaling factors are less than 1, the object would shrink.
- In the homogeneous co-ordinate system, it is represented as:

$$[x', \quad y', \quad 1] = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} [x, \quad y, \quad 1]^T$$

## 3. Mirror or Reflection Operation

- This function creates the reflection of the object in a plane mirror.
- The function returns an image in which the pixels are reversed.
- The reflected object is of same size of the original object, but the object is in opposite quadrant.
- The reflection is also described as rotation by $180^0$
- The reflection along the x-axis is given by:

$$F' = [-x, \quad y] = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} [x, \quad y]^T$$

- Similarly the reflection along the y-axis is given by

$$F' = [x, \quad -y] = \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} [x, \quad y]^T$$

## 4. Shearing

- Shearing is a transformation that produces a distortion of the shape.
- This can be applied either in X-axis or in Y-axis.
- In this transformation, the parallel and opposite layers of the object are simply sided with respect to each other.

Shearing can be done using the following calculation and can be represented in the matrix form as

$$x' = x + ay$$

$$y' = y$$

$$X_{\text{shear}} = \begin{pmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ (where } a = sh_x\text{)}$$

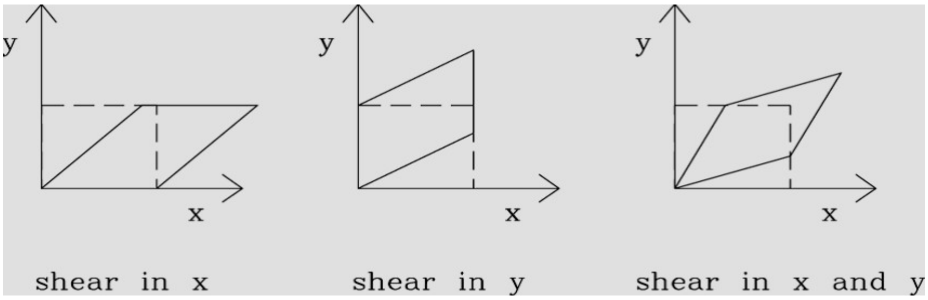
Similarly, $Y_{\text{shear}}$ can be given as

$$x' = x$$

$$y' = y + bx$$

$$Y_{\text{shear}} = \begin{pmatrix} 1 & 0 & 0 \\ b & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \text{ (where } b = sh_y\text{)}$$

where $sh_x$ and $sh_y$ are shear factors in the $x$ and $y$ directions, respectively.




## 4. Rotation

- An image can be rotated by various degrees such as $90^0$, $180^0$, $270^0$.
- In the matrix form it is given as:

$$[x', \quad y'] = \begin{pmatrix} Cos\theta & -Sin\theta \\ Sin\theta & Cos\theta \end{pmatrix} [x, \quad y]^T$$

In the homogeneous coordinate system, rotation can be represented as

$$[x', y', 1] = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} [x, y, 1]^T$$

- This can be represented as $F' = RA$.
- The parameter θ is the angle of rotation with respect to the x-axis.
- It is assumed that the object rotation is about the origin.
- The value of θ can be Positive or negative.
- A positive angle represents counter clockwise rotation and a negative angle represents clockwise rotation.

**Example 3.6** Consider an image point [2, 2]. Perform the following operations and show the results of these transforms.

(a) Translate the image right by 3 units.

(b) Perform a scaling operation in both *x*-axis and *y*-axis by 3 units.

(c) Rotate the image in *x*-axis by 45°.

(d) Perform horizontal skewing by 45°.

(e) Perform mirroring about *x*-axis.

(f) Perform shear in *y*-direction by 30 units.

***Solution***

(a) Translation of the image right by 3 units means that

$$\delta_x = 3 \text{ and } \delta_y = 0$$

So the translation matrix is given as

$$T = \begin{pmatrix} 1 & 0 & \delta_x \\ 0 & 1 & \delta_y \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Therefore,

$$x' = T \times x$$

$$= \begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times [2, 2, 1] = [5 \quad 2 \quad 1]^T$$

(b) Scaling by 3 units in both the directions means that

$$S = \begin{pmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$S = \begin{pmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times [2, 2, 1]^T = [6 \quad 6 \quad 1]^T$$

(c) Rotating the image in $x$-axis by 45°

$$R = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos 45 & -\sin 45 & 0 \\ \sin 45 & \cos 45 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R = \begin{pmatrix} 0.707 & -0.707 & 0 \\ 0.707 & 0.707 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times [2, 2, 1]^T = [0 \quad 2.828 \quad 1]^T$$

The fraction 2.828 will be rounded to 3. This is determined by the interpolation technique used.

(d) Performing horizontal skewing by 45°

$$Skew = \begin{pmatrix} 1 & 0 & 0 \\ \tan\varphi & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ \tan 45 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$Skew = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \times [2, 2, 1]^T = [2, 4, 1]^T$$

## Affine Transformation

The transformation that maps the pixel at the coordinates $(x, y)$ to a new coordinate position is given as a pair of transformation equations. In this transform, straight lines are preserved and parallel lines remain unchanged. It is described mathematically as

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

$T_x$ and $T_y$ are expressed as polynomials. The linear equation gives an affine transform.

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

This is expressed in matrix form as

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = \begin{pmatrix} a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

- The affine transform is a compact way of representing all transformations.
- The given equation represents all transformations.
- Translation → **$a_0 = 1, a_1 = 0, a_2 = \delta x, b_0=1, b_1=1, b_2= \delta y$;**
- Scaling transformation → **$a_0 = S_x, a_1 = 0, a_2 = 0, b_0=0, b_1= S_y, b_2= 0$;**
- Rotation → **$a_0 = \cos(\Theta), a_1 = -\sin(\Theta), a_2 = 0, b_0=\sin(\Theta), b_1=\cos(\Theta), b_2= 0$;**
- and horizontal shear is performed when **$a_0 = 1, a_1 = Sh_x, a_2 = 0, b_0=1, b_1= Sh_y, b_2= 0$;**

## Inverse Transformation

The purpose of inverse transformation is to restore the transformed object to its original form and position. The inverse or backward transformation matrices are given as follows:

$$\text{Inverse transform for translation} = \begin{pmatrix} 1 & 0 & -\delta x \\ 0 & 1 & -\delta y \\ 0 & 0 & 1 \end{pmatrix}$$

$$\text{Inverse transform for scaling} = \begin{pmatrix} \frac{1}{S_x} & 0 & 0 \\ 0 & \frac{1}{S_y} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Inverse transform for rotation can be obtained by changing the sign of the transform term. For example, the following matrix performs inverse transform.

$$\begin{pmatrix} \cos\theta & +\sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

## 3D Transforms

The purpose of inverse transformation is to restore the transformed object to its original form and position. The inverse or backward transformation matrices are given as follows:

$$\text{Inverse transform for translation} = \begin{pmatrix} 1 & 0 & -\delta x \\ 0 & 1 & -\delta y \\ 0 & 0 & 1 \end{pmatrix}$$

$$\text{Inverse transform for scaling} = \begin{pmatrix} \frac{1}{S_x} & 0 & 0 \\ 0 & \frac{1}{S_y} & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Inverse transform for rotation can be obtained by changing the sign of the transform term. For example, the following matrix performs inverse transform.

$$\begin{pmatrix} \cos\theta & +\sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Rotation along $x$-axis

$$R_{x,\theta} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta & 0 \\ 0 & \sin\theta & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Rotation along $y$-axis

$$R_{y,\theta} = \begin{pmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Compactly, these can be represented as

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = C \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Similarly, the reflection transform matrices can be given as

$$\text{Reflection}_{xy\text{-}plane} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{Reflection}_{xz\text{-}plane} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\text{Reflection}_{yz\text{-}plane} = \begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

The shear matrices for the shear quantities $a$ and $b$ can be given as

$$\text{Shear}_z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ a & b & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \qquad \text{Shear}_x = \begin{pmatrix} 1 & a & b & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

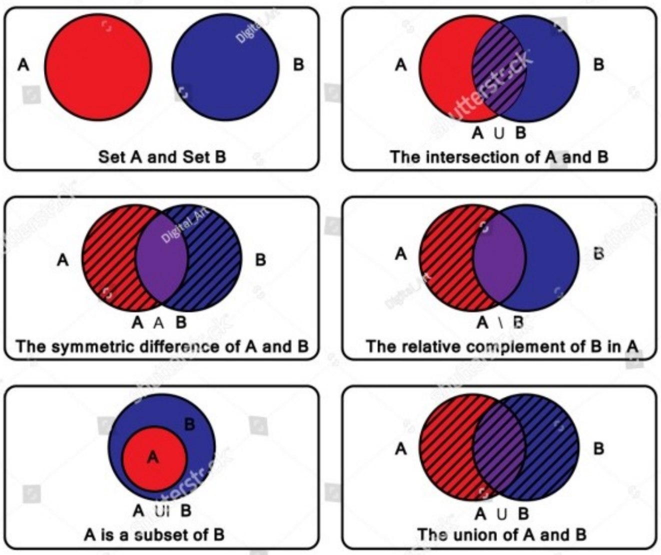
$$\text{Shear}_y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ a & 1 & b & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

## Set operations

- A set like B in Z with elements $a = (a_1, a_2)$ is defined as

$$B = \{w \mid w = (a_1, a_2) \ \text{for} \ a_1, a_2 \in Z\}$$

- Union, Intersection, Set Difference, and Complement are defined as shown in the figure.




- ➢ Morphology is a collection of operations based on set theory, to accomplish various tasks such as extracting boundaries, filling small holes present in the image, and removing noise present in the image.
- ➢ Mathematical morphology is a very powerful tool for analysing the shapes of the objects that are present in the images. The theory of mathematical morphology is based on set theory.
- ➢ The set operators such as intersection, union, inclusion, and complement can then be applied to the images.

- The two most widely used **Morphological** operations are **Erosion and Dilation.**

## 1. Erosion

- Erosion shrinks the image pixels, or erosion removes pixels on object boundaries.
- Let us assume that A and B are a set of pixel coordinates. The dilation of A by B can be denoted as:

$$A \ominus B = \{(x, y) - (u, v) : (x, y) \in A, (u, v) \in B\}$$

where **x** and **y** correspond to the **set A**, and **u** and **v** correspond to the **set B**. The coordinates are subtracted and the **intersection is carried out to create the resultant se**t.

## 2. Dilation

- Dilation expands the image pixels, or it adds pixels on object boundaries.
- It can be applied to binary as well as grey scale images.
- The basic effect of this operator on a binary image is that it gradually increases the boundaries of the region, while the small holes that are present in the images become smaller.
- Let us assume that A and B are a set of pixel coordinates. The dilation of A by B can be denoted as:

$$A \oplus B = \{(x, y) + (u, v) : (x, y) \in A, (u, v) \in B\}$$

where **x** and **y** correspond to the **set A**, and **u** and **v** correspond to the **set B**. The coordinates are added and the **union is carried out to create the resultant se**t.

**Example 3.7** Consider the following binary image. Show the results of the dilation and erosion operations.

$$F = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 2 & 0 & 1 & 1 \end{array}$$

Let the structured element $S$ be [1 1] with coordinates $\{(0, 0), (0, 1)\}$. Show the results of dilation and erosion.

***Solution*** The image $F$ can be written as

$$F = \{(0, 2), (1, 2), (2, 1), (2, 2)\}$$
$$S = \{(0, 0), (0, 1)\}$$

The dilation operation is done as follows:

First add the coordinates (0, 0) of $S$ to all the coordinate points of the image set $F$, followed by the second point of the set $S$.

$$F \text{ Dilation } S = \{(0, 2), (1, 2), (2, 1), (2, 2), (0, 3), (1, 3), (2, 2), (2, 3)\}$$

Remove the repetitions; the union of these sets results in dilation. This results in

$$S = \{(0, 2), (0, 3), (1, 2), (1, 3), (2, 1), (2, 2), (2, 3)\}$$

The erosion is the intersection of these sets.
First subtract the coordinates (0, 0) of $S$ from all the coordinate points of the image set $F$, followed by the second point of the set $S$.

$$F \text{ Erosion } S = \{(0, 2), (1, 2), (2, 1), (2, 2), (0, 1), (1, 1), (2, 0), (2, 1)\}$$

The erosion is the intersection operation. Find the common element. This results in

$$S = (2, 1)$$

If the coordinates are $(x, y)$ and $(s, t)$, the result would be $(x + s, y + t)$ for dilation and $(x - s, y - t)$ for erosion.

The result of this numerical calculation is shown in Fig. 3.26(a).

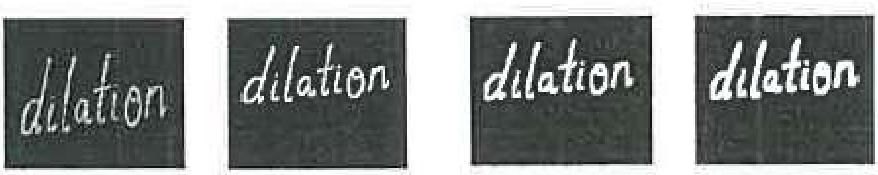
Dilation

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \Rightarrow \begin{array}{c|cccc} & 0 & 1 & 2 & 3 \\ \hline 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 \\ 2 & 0 & 1 & 1 & 1 \end{array}$$

Erosion

$$\begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \end{bmatrix} \Rightarrow \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 2 & 0 & 1 & 0 \end{array}$$

(a)




(b) (c) (d) (e)

Fig. 3.26 Dilation operation (a) Effects of dilation and erosion for a numerical example (b) Original large image (c) Dilation operation with structural element (of order 3 × 3) (d) Dilation operation with structural element (of order 9 × 9) (e) Dilation operation with structural element (of order 13 × 13)

## Statistical operations

- Statistics play an important role in Digital Image processing.
- An image can be assumed as a set of discrete points.
- Statistical operations can be applied to an image to get the desired results such as manipulation of brightness and contrast.
- Some of the very useful statistical operations include mean, median, mode and mid-range. The measures of data dispersion also include quartiles, inter-quartile range, and variance.
- Some of the common statistical measures as follows:

***Mean*** Mean is the average of all the values in the sample (population) and is denoted as $\bar{X}$.

$$\bar{X} = \frac{X_1 + X_2 + \cdots + X_n}{n} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

The overall brightness of the grey scale image is measured using the mean. This is calculated by summing all the values of the pixels of an image and dividing it by the number of pixels in the image.

$$\mu = \frac{1}{n}\sum_{i=0}^{n-1} I_i$$

Sometimes the data is associated with a weight. This is called weighted mean. The problem of mean is its extreme sensitivity to noise. Even small changes in the input affect the mean drastically.

***Median*** Median is the value where the given $X_i$ is divided into two equal halves, with half of the values being lower than the median and the other half higher. The procedure for obtaining the median is to sort the values of the given $X_i$ in ascending order. If the given sequence has an odd number of values, the middle value is the median. Otherwise, the median is the arithmetic mean of the two middle values.

***Mode*** Mode is the value that occurs most frequently in the dataset. The procedure for finding the mode is to calculate the frequencies for all of the values in the data. The mode is the value (or values) with the highest frequency. Normally, based on the mode, the dataset is classified as unimodal, bimodal, and trimodal. Any dataset that has two modes is called bimodal.

***Standard deviation and variance*** By far, the most commonly used measures of dispersion are variance and standard deviation. The mean does not convey much more than a middle point. For example, the following datasets, {10, 20, 30} and {10, 50, 0}, both have a mean of 20. The difference between these two sets is the spread of data. Standard deviation is the average distance from the mean of the dataset to each point. The formula for standard deviation is given by

$$\sigma = \sqrt{\sum_{i=1}^{N} (Xi - \mu)^2 / N - 1}$$

Sometimes, we divide the value by $N - 1$ instead of $N$. The reason is that in a larger, real-world scenario, division by $N - 1$ gives an answer that is closer to the actual value. In image processing, it is a measure of how much a pixel varies from the mean value of the image. The mean value and the standard deviation characterize the perceived brightness and contrast of the image. Variance is another measure of the spread of the data. It is the square of standard deviation. While standard deviation is a more common measure, variance also indicates the spread of the data effectively.

## Questions on Unit-4

1. Write a Program to read a digital image. Split and display image into 4 quadrants, up, down, right and left.
2. Write a program to show rotation, scaling, and translation on an image.
3. In detail explain the fundamental steps involved in digital image processing systems.
4. Explain in detail the classification of images.
5. Illustrate the relationship between image processing and other related fields.
6. Given a grey-scale image of size 5 inches by 6 inches scanned at the rate of 300 dpi, answer the following:
   (a) How many bits are required to represent the image?
   (b) How much time is required to transmit the image if the modem is 28 kbps?
   (c) Repeat (a) and (b) if it were a binary image.
7. Explain Digital image representation. A picture of physical size 2.5 inches by 2 inches is scanned at 150 dpi. How many pixels would be there in the image?
8. Explain Distance measure. Compute the Euclidean Distance (D1), City-block Distance (D2) and Chessboard distance (D3) for points p and q, where p and q be (5, 2) and (1, 5) respectively. Give answer in the form (D1, D2, D3).
9. Describe the fundamental steps in image processing?
10. Describe the basic relationship between the pixels
    a. Neighbours of a pixel
    b. Adjacency, Connectivity, Regions and Boundaries
    c. Distance measures
11. All solved problems in notes.
12. Summarize the Arithmetic operations on digital images with relevant expressions.
13. Summarize the Logical operations on digital images with relevant expressions.
14. Explain 2D Geometric transformation with equations and homogeneous matrix.
15. Consider two pixels x and y whose coordinates are (0, 0) and (6, 3) .Compute De, D4, D8 distance between x and y
16. Consider the following two images. Perform the arithmetic operations: addition, multiplication, division. Assume that all the operations are uint8.

$$f_1 = \begin{pmatrix} 10 & 40 & 30 \\ 40 & 100 & 90 \\ 90 & 80 & 70 \end{pmatrix} \qquad f_2 = \begin{pmatrix} 40 & 140 & 90 \\ 140 & 100 & 90 \\ 90 & 80 & 190 \end{pmatrix}$$

17. Consider the images f1 and f2 in above question. What is the result of image subtraction and image absolute difference? Is there any difference between them?

18.

Consider the following binary image. Show the results of the dilation and erosion operations.

$$F = \begin{array}{c|ccc} & 0 & 1 & 2 \\ \hline 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 2 & 0 & 1 & 1 \end{array}$$

19. Consider the following two images. The addition and subtraction of images are given by f1+f2 and f1−f2. Assume both the images are of the 8-bit integer type.

$$f1 = \begin{array}{ccc} 100 & 100 & 100 \\ 50 & 50 & 50 \\ 200 & 150 & 150 \end{array} \quad \text{and} \quad f2 = \begin{array}{ccc} 50 & 50 & 25 \\ 40 & 40 & 50 \\ 50 & 50 & 75 \end{array}$$

20. Consider the following two images. Perform the logical operations AND, OR, NOT and difference.

$$f_1 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix} \quad f_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

<u>Solution</u>

<u>AND</u>

$$f_1 \,\&\, f_2 = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

<u>OR</u>

$$f_1 \mid f_2 = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

<u>NOT</u>

$$!\, f_1 = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix} \quad !\, f_2 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

<u>DIFFERENCE:</u>

$$f_1 - f_2 = (f_1) \,\&\, (!f_2) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$f_2 - f_1 = (f_2) \,\&\, (!f_1) = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$$