

1. DIGITAL IMAGE FUNDAMENTALS & IMAGE TRANSFORMS

Image: Image is a two dimensional function $f(x, y)$ where (x, y) are spatial coordinates.

Example: Here (x_0, y_0) (x_1, y_1) are called spatial coordinate in digital image.

→ Images are very powerful tool in communication

→ image may be white & black (or) colour

amplitude of any point $f(x, y)$ gives intensity (or) gray level at that point.

→ At different points we have different intensity or gray level values of an image.

	x_1	$f(x_1, y_1)$
y_0	x_0	$f(x_0, y_0)$

* A simple image model

* Image must suitable for computer processing for this an image $f(x, y)$ must be digitalized both in spatial and amplitude.

* Digitalized of spatial coordinates (x, y) is called sampling (or) image sampling.

* Digitalization of amplitude value is called quantization or gray level quantization.

* Image processing: It is a method to convert an image into digital form and perform some operations on it in order to get an enhanced image, it is a process in which input is and o/p may be image or characters associated with that image.

* Image processing steps.

It follows three steps

Step 1: Importing the image via image acquisition tool

Step 2: analyzing & manipulating the image

Step 3: output is the last stage in which result can be altered image.

* Digital image

* When all values of (x, y) & amplitude of f are finite & discrete values then it is called as digital image

* Intensity at each point of image is finite & discrete value.

* Digital image processing

- * It refers to the processing of digital image to digital computers.
- * Digital image consists of finite no. of elements with particular locations and values. These elements are commonly called as image elements or pixel elements.
- * Each pixel is represented by k -bits. ^{black white} a pixel can have 2^k different values.
Ex: 8 bit image $2^8 = 256$ (0-255)
- * Human vision is the most advanced sensor, but this sensor is limited to vision band of electromagnetic spectrum.
- * To overcome this disadvantage, image machines are used.
- * This machine covers all the bands in the EM spectrum.
- * Hence we can say that a dip covers wide & variety fields of applications due to above reasons.
- * **Image**: Any visual object that's modified or altered by a computer or an imaginary object created using a computer.
- * **picture**: A drawing, painting, or artwork created on a computer. A picture is also describes anything created using a camera or scanner.
- * **Advantages of DIP**

color image processing	wavelets of multi resolution processing	Image compression	Morphological image processing
Image restoration			Image segmentation
Image enhancement	knowledge base		Representation & Description
Image acquisition			object recognition

There are two categories of image involved in the image processing.

1. Methods whose i/p's & o/p's are images
 2. Methods whose o/p's are attributes extracted from those image (i/p image)
1. **Image Acquisition**: It is acquiring the image that is already in digital form that is digital image.
If it is not in digital form we need to convert the image into digital form by using image processing.
Image acquisition stage involves preprocessing such as scaling etc.

2. Image Enhancement: It is a process of adjusting or modifying manipulating the digital image, so that results are more pleasant for display. Ex: Brightness, contrast & adjustment, removal of noise etc. It is subjective in nature as some may like high saturation images & some may like natural colours.
3. Image Restoration: It is a process of recovery and image that has been degraded. It is completely objectively in nature.
4. Colour image processing: It is a stage where processing of an colour images are done. It handles the image processing of colour images indexed images or RGB images.
5. Wavelet & multiresolution processing: Wavelets are small waves of limited duration which are used to calculate wavelet transform which provides time & frequency information of an image. Wavelets leads to multiresolution processing where images are represented in various degrees of resolution.
6. Image compression: It is a technique used for reducing storage space required to save an image. If the image size is large it requires more Bandwidth to display an image. Hence image compression can be used to reduce Bandwidth's.
7. Morphological image processing: It deals with extracting of image components that are useful in representation & description of shapes. It includes basic morphological operations like erosion & dilation.
8. Image segmentation: It is a process that of partition of an image into multiple segments. It is used to locate object & boundaries in an object.
9. Object Recognition: It is a process that assigns label to an object in a image, Ex: If image is consisting of a car this page should process image & assigning a label car which identifies a car is present in the image.
10. Knowledge base: knowledge about problem domain is coded in image processing in the form of knowledge base, so all the information related to image processing is present in knowledge base.

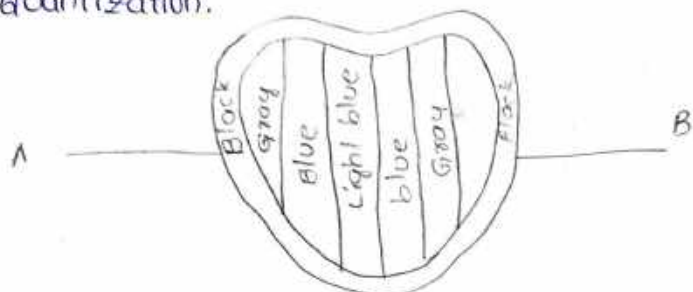
* The o/p of the image sensors are continuous in nature

* For digital processing & image must be in digital form, so we need to convert continuous s/l into digital form. For this we are having two main processes

1. Sampling 2. Quantization

1. Sampling: A digitalizing the spatial coordinate values is nothing but sampling

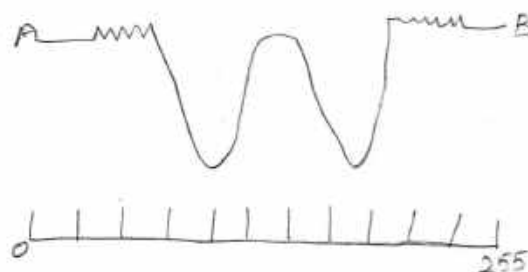
2. Quantization: Digitizing the amplitude values of the image is known as quantization.



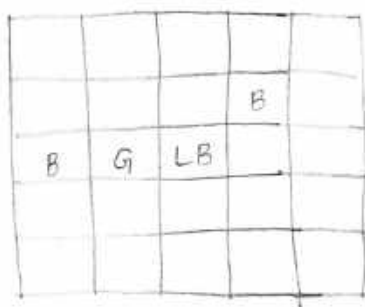
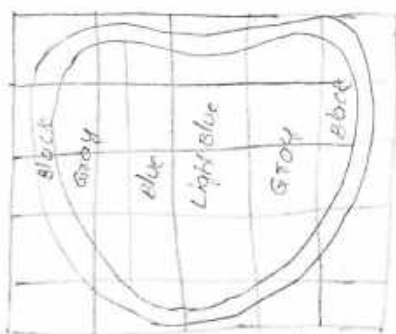
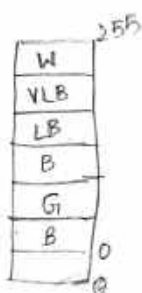
Let us take a line through a image in order to understand the concept of sampling & quantization, we move from point A to B.

* Sampling: To understand the graph and digitize the image, we will divide entire system into equal parts

* Dividing the image into equal parts or coordinate values into equal parts is called sampling.

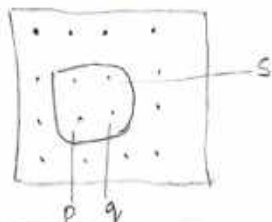


* Quantization



* Relationship b/w pixels.

→ consider an image $f(x,y)$ consists of some pixel values have shown in below fig.



* Where p & q are particular pixels

* A group of pixels is called as subset it is denoted by s .

* An image is represented as follows

$y+1$			
y		x, y	
$y-1$			

$x-1, y+1$	$x, y+1$	$x+1, y+1$
$x-1, y$	x, y	$x+1, y$
$x-1, y-1$	$x, y-1$	$x+1, y-1$

* Relationship b/w pixels is classified into 3 types

1. 4-Neighbourhood pixels (or) 4-neighbours

If a pixel $P(x, y)$ is surrounded by two vertical & two horizontal neighbours then it is 4-Neighbourhood pixels.

	$x, y+1$	
$x-1, y$	x, y	$x+1, y$
	$x, y-1$	

* It is denoted by the symbol
 $N_y(P) = (x, y+1), (x, y-1), (x-1, y), (x+1, y)$

2. Diagonal Neighbours $N_d(P)$: If 4 pixels from form A with respect to the middle pixel it is called as the diagonal Neighbour.

$x-1, y+1$		$x+1, y+1$
	x, y	
$x-1, y-1$		$x+1, y-1$

$$N_d(P) = (x-1, y+1), (x+1, y+1), (x-1, y-1), (x+1, y-1)$$

3. 8-Neighbour $N_8(P)$: It is a combination of 4-Neighbours and diagonal neighbours

$x-1, y+1$	$x, y+1$	$x+1, y+1$
$x-1, y$	x, y	$x+1, y$
$x-1, y-1$	$x, y-1$	$x+1, y-1$

$$N_8(P) = N_y(P) + N_d(P)$$

* Connectivity / Adjacency

* If two pixels that are neighbours and have some gray level value & Adjacency taken be connected

* connectivity is the most important in the image process to connect the pixels.

They are classified into 3 types

1. 4-adjacency

2. 8 - "

3. M - "

1. Binary Image

$$\begin{array}{ccc} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{array}$$
 8 adjacency
 "

$$\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array}$$
 Not connected
 (or)
 Not adjacency

Gray level Image

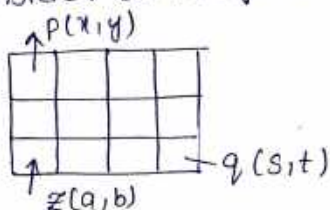
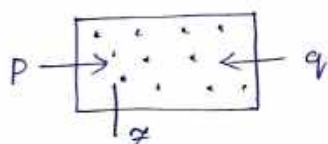
$$V = \{1, 2, 3, \dots, 100\}$$

$$\begin{array}{ccccc} 54 & 10 & 100 & 5 \\ 81 & 150 & 2 & 34 \\ 201 & 200 & 3 & 45 \\ 70 & 147 & 56 \end{array}$$

Not connected or Not adjacent

5. M- Adjacency

- * It is also known as onixed adjacency
- * It is used to eliminate the ambtutity (difficulty) that occurs during 8 adjacency usage.
- * It says that we should give top priority to straight line connect but not the diagonal connect.
- * Distance measure: consider an image $f(x,y)$ having three pixel values as shown in below



P, Q, Z = particular pixels

* properties:

$$1. D(P, Q) \geq 0 \quad 2. D(P, Q) = 0 ; P = Q \quad 3. D(P, Q) = D(Q, P)$$

$$4. D(Z, P) \leq D(P, Q) + D(Q, Z)$$

* There are three distance measure methods

1. Euclidean Distance

$$D_E(P, Q) = [(x-s)^2 + (y-t)^2]^{1/2}$$

2. cityblock distance

$$D_H(P, Q) = |x-s| + |y-t|$$

$$\Rightarrow$$

4		2		4
2	1	0	1	2
		1		
4		2		4

3. chessboard :

$$D_8(P, Q) = \max \{ |x-s|, |y-t| \} \Rightarrow$$

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

* Arithmetic operations b/w images

$$1. s(x, y) = f(x, y) + g(x, y)$$

$$2. b(x, y) = f(x, y) - g(x, y)$$

$$3. p(x, y) = f(x, y) * g(x, y)$$

$$4. d(x, y) = f(x, y) \% g(x, y)$$

* Important points

1. If the result is floating point of 'P' value
2. If the result is above the pixel range select the max range value.
3. If the result is below the pixel range select the min range values.
4. If the result is infinity write it as zero.

$$A = \begin{bmatrix} 1 & 5 & 100 \\ 2 & 111 & 120 \\ 8 & 0 & 5 \end{bmatrix}$$

$$B = \begin{bmatrix} 10 & 100 & 5 \\ 2 & 8 & 10 \\ 1 & 100 & 10 \end{bmatrix}$$

$$A \div B = \begin{bmatrix} 0 & 0 & 20 \\ 1 & 14 & 12 \\ 8 & 0 & 1 \end{bmatrix}$$

$$A+B = \begin{bmatrix} 11 & 105 & 105 \\ 4 & 119 & 130 \\ 9 & 100 & 15 \end{bmatrix}$$

$$A-B = \begin{bmatrix} 0 & 0 & 95 \\ 0 & 103 & 110 \\ 7 & 0 & 0 \end{bmatrix}$$

$$A * B = \begin{bmatrix} 120 & 205 & 255 \\ 255 & 255 & 255 \\ 85 & 255 & 90 \end{bmatrix}$$

* Image Transforms

1. Hadamard Transform (H):

It is a non sinusoidal transfer function

DCT & DFT are sinusoidal transfer function

It is a symmetric matrix i.e. $A = A^T$

The smallest order of the matrix is $H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

$$H_{2N} = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix}$$

If $N=2$

$$H_{2 \times 2} = \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} \Rightarrow H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

If $N=4$

$$H_{2 \times 4} = \begin{bmatrix} H_4 & H_4 \\ H_4 & -H_4 \end{bmatrix} \Rightarrow H_8 = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}$$

Find the Hadamard Transform $f(m,n) = \begin{bmatrix} 1 & 1 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 1 & 3 & 2 & 1 \\ 2 & 1 & 2 & 1 \end{bmatrix}$

$$F = H_4^T F \cdot H_4$$

$$F = H_4 \cdot F = \begin{bmatrix} 1 & 1 & 2 & 1 \\ 2 & 1 & 1 & 2 \\ 1 & 3 & 2 & 1 \\ 2 & 1 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 6 & 6 & 7 & 5 \\ -2 & 2 & 1 & -1 \\ 0 & -2 & -1 & 1 \\ 0 & -2 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

$$F = \begin{bmatrix} 4 & 4 & 8 & 4 \end{bmatrix}$$

* Walsh Transform

* It is a non sinusoidal transfer function

* It is a symmetric matrix

In Walsh transform we need to check the sign changes on each row

For this we consider 4×4 Hadamard matrix

$$H_4 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{matrix} \rightarrow 0 \\ \rightarrow 3 \\ \rightarrow 1 \\ \rightarrow 2 \end{matrix}$$

In order to get the Walsh transform matrix we will arrange the sign changes in ascending order

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

Formulas for Walsh Transform

$$1D \rightarrow F = F \cdot W$$

$$2D \rightarrow F = W^T F W$$

Q. If $f(x) = \{1, 2, 0, 3\}$ Find the Walsh transform for it

We need to convert row matrix into column matrix

$$f(x) = \{1, 2, 0, 3\} = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 3 \end{bmatrix}$$

We know that $F = F \cdot W$

$$F = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 3 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} = \begin{bmatrix} 6 \\ 0 \\ 2 \\ -4 \end{bmatrix}$$

$$\text{If } f(x, y) = \begin{bmatrix} 2 & 1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 2 \end{bmatrix}$$

Apply Walsh transform for it We know that $F = W^T F W$

$$F = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} 2 & 1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

* Discrete cosine Transform

It is a sinusoidal transfer function

In DCT we convert time domain into frequency domain

The main difference b/w DFT & DCT is that the DFT's have complex terms where as DCT will have real terms

$$\text{For 1D} \rightarrow F = C \cdot R$$

$$\text{2D} \rightarrow F = C^T \cdot F \cdot C$$

Where C is the cosine transform matrix or kernel matrix

We have

$$C = \begin{cases} \frac{1}{\sqrt{N}} & u=0, 0 \leq v \leq N-1 \\ \sqrt{\frac{2}{N}} \cos \pi \left[\frac{(2v+1)u}{2N} \right] & 1 \leq u \leq N-1, 0 \leq v \leq N-1 \end{cases}$$

For 2x2 matrix we consider $N=2$

$$\begin{array}{c|cc} u \backslash v & 0 & 1 \\ \hline 0 & 0.707 & 0.707 \\ 1 & 0.707 & -0.707 \end{array} ; C_2 = \begin{bmatrix} 0.707 & 0.707 \\ 0.707 & -0.707 \end{bmatrix}$$

For 4x4 matrix we consider $N=4$

$$\begin{array}{c|cccc} u \backslash v & 0 & 1 & 2 & 3 \\ \hline 0 & 0.5 & 0.5 & 0.5 & 0.5 \\ 1 & 0.65 & 0.27 & -0.27 & -0.65 \\ 2 & 0.5 & -0.5 & -0.5 & 0.5 \\ 3 & 0.27 & -0.65 & 0.65 & -0.27 \end{array}$$

$$C_4 = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}$$

* Find the DCT for the image $f(x,y) = \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 1 & 2 & 1 \\ 1 & 2 & 2 & 1 \\ 2 & 1 & 2 & 1 \end{bmatrix}$ 16/08/23

$$F = C^T \cdot P \cdot C$$

$$C = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}$$

$$C^T = \begin{bmatrix} 0.5 & 0.65 & 0.5 & 0.27 \\ 0.5 & 0.27 & -0.5 & -0.65 \\ 0.5 & -0.27 & -0.5 & 0.65 \\ 0.5 & -0.65 & 0.5 & -0.27 \end{bmatrix}$$

$$F = C^T \cdot C \cdot P$$

$$= \begin{bmatrix} 0.5 & 0.65 & 0.5 & 0.27 \\ 0.5 & 0.27 & -0.5 & -0.65 \\ 0.5 & -0.27 & -0.5 & 0.65 \\ 0.5 & -0.65 & 0.5 & -0.27 \end{bmatrix} \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix} \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 1 & 2 & 1 \\ 1 & 2 & 2 & 1 \\ 2 & 1 & 2 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.5 & 0.65 & 0.5 & 0.27 \\ 0.5 & 0.27 & -0.5 & -0.65 \\ 0.5 & -0.27 & -0.5 & 0.65 \\ 0.5 & -0.65 & 0.5 & -0.27 \end{bmatrix} \begin{bmatrix} 3 & 3 & 4 & 2 \\ -0.38 & +0.38 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -0.92 & 0.92 & 0 & 0 \end{bmatrix} \begin{bmatrix} 2.84 & 2.92 & 3.84 & 1.92 \\ -0.76 & -0.38 \\ 0.76 & -0.76 \\ -0.84 \end{bmatrix}$$

1 1.9

5.75 -0.95 -0.04 0.92
-1.10 0.14 -0.14 -0.41
1.10 -0.14 0.14 0.41
0.38 -0.26 -0.73 -1.06

* Find the DCT of the sequence $f(x) = \{1, 2, 4, 7\}$

$$F = C \cdot P$$

$$C = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 7 \end{bmatrix} = \begin{bmatrix} 7 \\ -4.44 \\ 1 \\ -0.32 \end{bmatrix}$$

* KL TRANSFORM

17/08/23

* Karhunen Loeve Transform or Hotelling Transform or Eigen vector Transform

* The transformation matrix is not fixed

* In DCT / DFT we have fixed matrix but in KL Transform the transformation matrix depends on the image

* It is a statistical method used for image compression

Ex:

3	0	0	0	0
2	0	1	1	0
1	0	0	1	1
0	0	0	0	0
	0	1	2	3

$\rightarrow x_i$

* step 1: Form a vector of the object (Finding the location of one's in the image)

$$X = \left[\begin{pmatrix} 1 \\ 2 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \end{pmatrix} \begin{pmatrix} 2 \\ 2 \end{pmatrix} \begin{pmatrix} 3 \\ 1 \end{pmatrix} \right]$$

* step 2: Find the mean of the vector

$$\mu_x = \begin{bmatrix} \frac{1+2+2+3}{4} \\ \frac{2+1+2+1}{4} \end{bmatrix} = \begin{bmatrix} 8/4 \\ 6/4 \end{bmatrix} = \begin{bmatrix} 2 \\ 1.5 \end{bmatrix}$$

* step 3: Find the covariance matrix

$$C_x = E[(x_i - \mu_x)(x_i - \mu_x)^T] \quad i = 1, 2, 3, 4$$

$$C_1 = E \begin{bmatrix} 1-2 \\ 2-1.5 \end{bmatrix} = \begin{bmatrix} -1 \\ 0.5 \end{bmatrix} \begin{bmatrix} -1 & 0.5 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & -0.5 \\ -0.5 & 0.25 \end{bmatrix}$$

$$C_2 = E \begin{bmatrix} 2-2 \\ 1-1.5 \end{bmatrix} = \begin{bmatrix} 0 \\ -0.5 \end{bmatrix} \begin{bmatrix} 0 & -0.5 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 0.25 \end{bmatrix}$$

$$C_3 = \begin{bmatrix} 2-2 \\ 2-1.5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix} \begin{bmatrix} 0 & 0.5 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0.25 \end{bmatrix}$$

$$C_4 = \begin{bmatrix} 3-2 \\ 1-1.5 \end{bmatrix} = \begin{bmatrix} 1 \\ -0.5 \end{bmatrix} \begin{bmatrix} 1 & -0.5 \end{bmatrix} = \begin{bmatrix} 1 & -0.5 \\ -0.5 & 0.25 \end{bmatrix}$$

$C_X = \text{mean of } C_1, C_2, C_3, C_4$

$$C_X = \begin{bmatrix} \frac{1+0+1+0}{4} & \frac{-0.5+0+0+0.5}{4} \\ \frac{-0.5+0+0+0.5}{4} & \frac{-0.25+0.25+0.25+0.25}{4} \end{bmatrix} = \begin{bmatrix} 0.5 & -0.25 \\ -0.25 & 0.25 \end{bmatrix}$$

* Step 4: Find the Eigen values $[C_X - \lambda I] = 0$

$$\begin{bmatrix} 0.5-\lambda & -0.25 \\ -0.25 & 0.25-\lambda \end{bmatrix} = 0 \quad (0.5-\lambda)(0.25-\lambda) - (0.25)(0.25) = 0$$

$$= \lambda^2 - 0.75\lambda + 0.0625 = 0$$

$$\lambda_1 = 0.6545 \quad \lambda_2 =$$

* Step 5: Eigen vectors for $\lambda_1 = 0.6545$

$$[C_X - \lambda I][x] = 0$$

$$\begin{bmatrix} 0.5-\lambda & -0.25 \\ -0.25 & 0.25-\lambda \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} 0.5-0.6545 & -0.25 \\ -0.25 & 0.25-0.6545 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} -0.1545 & -0.25 \\ -0.25 & -0.4045 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$-0.1545x_1 - 0.25x_2 = 0 \quad (1)$$

$$-0.25x_1 - 0.4045x_2 = 0 \quad (2)$$

$$-0.25x_1 = 0.4045x_2$$

$$x_1 = \frac{-0.4045}{0.25}x_2 \quad x_1 = -1.618x_2$$

$$\text{If } x_2 = 1 \quad e_0 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -1.618 \\ 1 \end{bmatrix}$$

Eigen vector for $\lambda_2 = 0.095495 \quad [Cx - \lambda I][x] = 0$

$$\begin{bmatrix} 0.5 - 0.095495 & -0.25 \\ -0.25 & 0.25 - 0.095495 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$\begin{bmatrix} 0.4045 & -0.25 \\ -0.25 & 0.1545 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0$$

$$0.4045x_1 - 0.25x_2 = 0$$

$$x_1 = \frac{0.25}{0.4045} x_2 = 0.618 x_2$$

$$e_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.618 \\ 1 \end{bmatrix}$$

* step 6: Transition matrix A

$$\lambda_1 = e_0 [\text{Highest } \lambda_1]$$

$$\lambda_2 = e_0 [\text{Highest } \lambda_2]$$

$$A = [\lambda_1 \quad \lambda_2] = \begin{bmatrix} -1.618 & 0.618 \\ 1 & 1 \end{bmatrix}$$

$$\boxed{Y = A[X - \mu_r]}$$

* slant Transform

* It is orthogonal and very fast matrix

* It is widely used in image compression

* slant Transform of order 2×2 is defined as $S_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$

* In General $N \times N$ matrix can be given as $S_N = A \times B$

$$A = \begin{bmatrix} \begin{pmatrix} 1 & 0 \\ a_n & b_n \end{pmatrix} & 1 & \begin{pmatrix} 1 & 0 \\ -a_n & b_n \end{pmatrix} & 0 \\ 0 & I_{N/2-2} & 0 & I_{N/2-2} \\ \begin{pmatrix} 0 & 1 \\ -b_n & a_n \end{pmatrix} & 0 & \begin{pmatrix} 0 & -1 \\ b_n & a_n \end{pmatrix} & 0 \\ 0 & I_{N/2-2} & 0 & I_{N/2-2} \end{bmatrix} \quad B = \begin{bmatrix} S_{N/2} & 0 \\ 0 & S_{N/2} \end{bmatrix}$$

$$S_N = AXB$$

FOR $N=4$

$$N = 2^n \quad n=2$$

$$N = 2^2 = 4$$

$$\frac{1}{\sqrt{2}} \left[\begin{array}{cc|cc} \begin{pmatrix} 1 & 0 \\ a_n & b_n \end{pmatrix} & \begin{pmatrix} 1 & 0 \\ -a_n & b_n \end{pmatrix} & 0 & 0 \\ \hline 0 & I_{N/2-2} & 0 & I_{N/2-1} \\ \hline \begin{pmatrix} 0 & 1 \\ -b_n & a_n \end{pmatrix} & 0 & \begin{pmatrix} 0 & -1 \\ b_n & a_n \end{pmatrix} & 0 \\ \hline 0 & I_{N/2-2} & 0 & I_{N/2-2} \end{array} \right] \begin{bmatrix} S_{N/2} & 0 \\ 0 & S_{N/2} \end{bmatrix}$$

To next

$$S_4 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ a_2 & b_2 & -a_2 & b_2 \\ 0 & 1 & 0 & -1 \\ -b_2 & a_2 & b_2 & a_2 \end{bmatrix} \begin{bmatrix} S_2 & 0 \\ 0 & S_2 \end{bmatrix}$$

$$\text{Where } b_n = (1 + 4a_{n-1}^2)^{-1/2}$$

$$a_n = 2 \times b_n \times a_{n-1}$$

$$\text{Put } n=2 \quad a_1 = 1$$

$$b_2 = (1 + 4a_1^2)^{-1/2} = (1 + 4(1))^{-1/2} \quad b_2 = \frac{1}{\sqrt{5}}$$

$$a_2 = 2 \times b_2 \times a_{n-1} = 2 \times \frac{1}{\sqrt{5}} \times 1 = \frac{2}{\sqrt{5}}$$

$$= \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 1 & 0 \\ \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} & -\frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} \\ 0 & 1 & 0 & -1 \\ -\frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{5}} & \frac{2}{\sqrt{5}} \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

$$= \frac{1}{2} \begin{bmatrix} 1 & 0 & 1 & 0 \\ \frac{3}{\sqrt{5}} & \frac{1}{\sqrt{5}} & -\frac{1}{\sqrt{5}} & -\frac{3}{\sqrt{5}} \\ 0 & -1 & -1 & 1 \\ \frac{1}{\sqrt{5}} & -\frac{3}{\sqrt{5}} & \frac{3}{\sqrt{5}} & -\frac{1}{\sqrt{5}} \end{bmatrix}$$

* Haar Transform

Haar proposed Haar Transform in 1910

Due to its low computing it is mainly used for image processing and pattern recognition.

properties

It is orthogonal and very fast

It is symmetric

It is based on a class of orthogonal matrices whose elements $\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}$ are multiplied by $\sqrt{2}$

procedure

1. Find order $N \cdot n = \log_2 N$

2. Determine p & q values

p ranges from 0 to $n-1$

$p=0$ then $q=0$ & $q=1$

$p \neq 0$ then $1 \leq q \leq 2^p$

3. Determine k

$k = 2^p + q - 1 \cdot \{z = 0/2, 1/2, \dots, (N-1)/2\}$

4. If $k=0$

$$h_0(z) = h_{00}(z) = \frac{1}{\sqrt{N}} \text{ For } z \in (0,1)$$

$$\text{else } h_k(z) = h_{pq}(z) = \frac{1}{\sqrt{N}} \left[2^p \text{ if } \frac{q-1}{2^p} \leq z \leq \frac{q}{2^p} \right]$$

$$= -\frac{1}{\sqrt{N}} \left[-2^{p/2} \text{ if } \frac{q-1}{2^p} \leq z \leq \frac{q}{2^p} \right]$$

$$= 0 \text{ For } z \in (0,1)$$

Ex: $N=2$

1. Order $N=2$ $n = \log_2 2 = 0.3010 = 1$

2. $p=0$ $q=0$ & $q=1$

3. $k = 2^0 + 0 - 1 = 0$ If $q=0$

$k = 2^0 + 1 - 1 = 1$ If $q=1$

4. If $k=0$

$$h_0(z) = \frac{1}{\sqrt{2}}$$

If $k=1$

$$h_k(z) = \frac{1}{\sqrt{2}} [2^0 \cdot i^k]$$

$$h_1(0) = -\frac{1}{\sqrt{N}} = -\frac{1}{\sqrt{2}}$$

$$A_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

2x2 Haar

Transform matrix

Similarly

$$A_4 = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

* Image Transform

21/08/23

→ Most of the sl's or images are in time domain, it can be measured with the function of time.

→ An image Transform can be applied to an image to convert it from one domain to another domain. so that identification of features will be easy.

Ex: subtraction b/w two images. to identify the changes b/w two images.

* DFT

An image is a 2D sl so 2D Transforms are required to represent image in terms of discrete set of images. so it is necessary to have discrete formulation of the Fourier transform.

Fourier Transform gives relationship b/w spatial and frequency domain.

It is used for image enhancement in frequency domain

DFT $\rightarrow$ 1DFT $\rightarrow x(n) \rightarrow x(k)$ $n = \text{NO. of samples}$

$$x(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi \frac{nk}{N}} \cdot 0 \leq k \leq N-1$$

IDFT $x(k) \rightarrow x(n)$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} x(k) e^{j2\pi kn/N} \quad 0 \leq n \leq N-1$$

Fourier spectrum

$$F(u) = [R^2(u) + I^2(u)]^{1/2}$$

phase angle

$$\phi = \tan^{-1} \left[\frac{I(u)}{R(u)} \right]$$

power spectrum

$$p(u) = |F(u)|^2 = R^2(u) + I^2(u)$$

2DFT

$$F(x, y) \rightarrow F(u, v)$$

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

IDFT

$$F(u, v) \Rightarrow F(x, y)$$

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

Fourier spectrum

$$F(u, v) = [R^2(u, v) + I^2(u, v)]^{1/2}$$

phase angle

$$\phi = \tan^{-1} \left[\frac{I(u, v)}{R(u, v)} \right]$$

power spectrum

$$p(u, v) = |F(u, v)|^2 = R^2(u, v) + I^2(u, v)$$

* FFT (Fast Fourier Transform)

* It uses a strategy known as successive doubling method

* A FFT is a algorithm that computes DFT of a sequence or its inverse

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} F(x) \cdot W_N^{ux}$$

$$W_N = e^{-j2\pi/N}$$

FFT Algorithm is based on the principle of Division & conquer rule.

If the problem is big it is divided into sub problems this problem is known as division then the problems are solved and individual solutions are combined to get final solution this process is known as conquer.

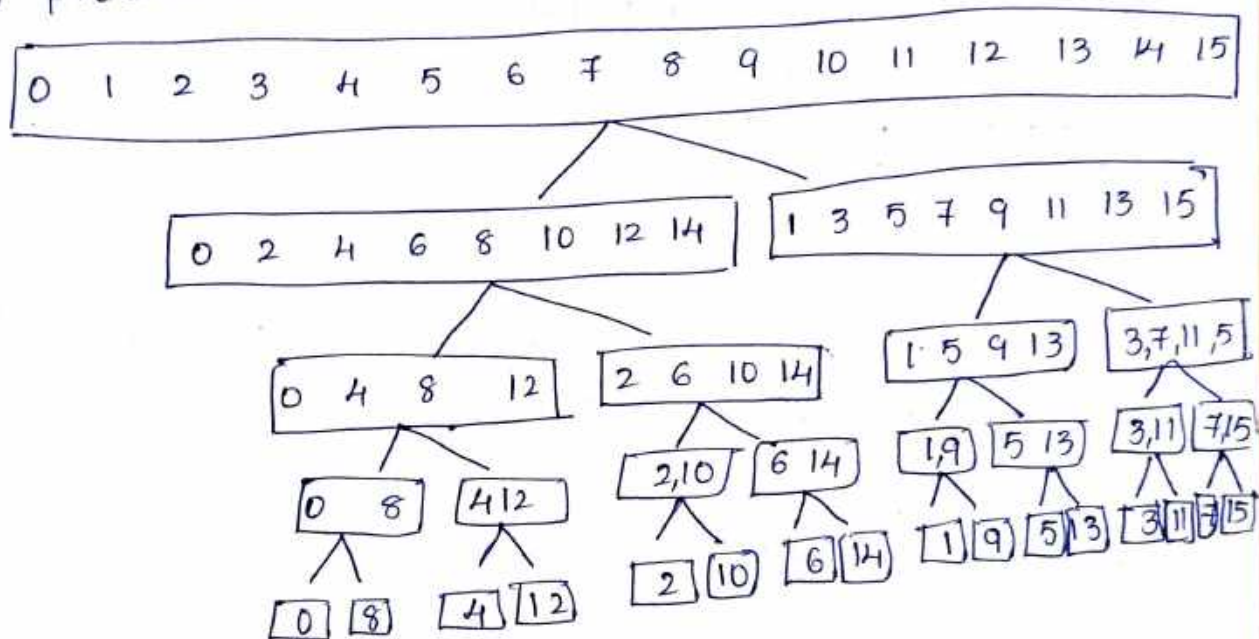


IMAGE ENHANCEMENT

* It is a process of manipulating or adjusting the image in order to improve the quality.

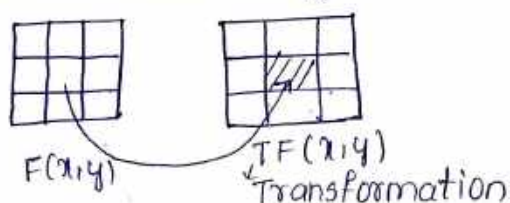
Need of image enhancement

1. To highlight the important features or details in the image
2. To remove the noise and to make an image more pleasant

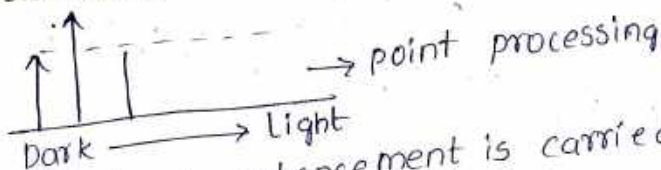
* Image Enhancement methods are classified into 3 types

1. spatial domain: It is nothing but a image plane it self.

In this method direct manipulation of pixels are done to achieve desired enhanced image.



* Simplest form: It is denoted with the symbol 'S'. $S = T(r)$ Where
 S is old image pixel value, r is new image pixel value 'T' Gray level
 Transformation



spatial domain enhancement is carried out in two ways

1. point processing
2. Neighbourhood processing

1. point processing: In point processing we work on a single pixel at a time.

- Ex 1. Digital Negative 2. contrast stretching 3. Gray level slicing
 4. bit plane slicing

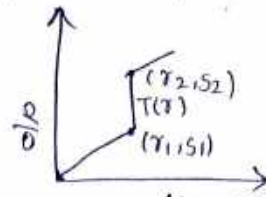
1) Digital Negative: It inverse the gray level values i.e if black is in original image we get white in old image and vice versa. It is mainly used in displaying of the x-ray image.

Digital Negative image can be obtained by using a simple transformation $S = 255 - r$

2) contrast stretching: contrast is nothing but difference b/w highest & lowest gray level values of an image.

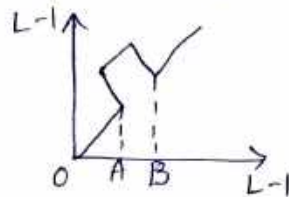
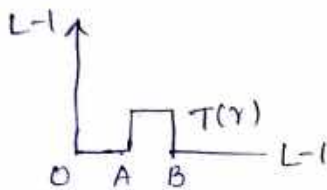
We can increase contrast by making darker portion more darker & brighter portion more brighter.

If $s_1 = r_1$ & $s_2 = r_2$ then it will get linear transformation



* If we consider intermediate values of (r_1, s_1) & (r_2, s_2) then we will get various degrees of gray level values.

3. Gray level slicing : It is used to highlight specific range of gray level values in a given image



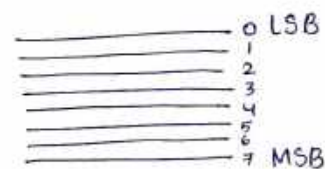
4. Bit plane slicing : It highlights the contribution made to the total image appearance by the specific bits

Eq: consider an 8-bit Transform

MSB will have more data components

Compare to LSB.

It is mainly used in image compression



2. Neighbourhood processing : It consists of a 3 simple neighbourhood operations

1. Minimum = set the pixel values to minimum in the neighbourhood.

2. Maximum = set " " " " maximum " " "

3. Median = It is a set of no's and the bit coin value in the set.

Eq:

1	2	3
4	5	6
7	8	9

If we replace 5 with 1 it is mini neighbourhood

" " " " " 9 " " maxi "

" " " " " 5 " " median "

* HISTOGRAM EQUILIZATION

It is a graphical representation of data

In case of image processing it used to represent the data related to digital images.

It also gives representation of frequency of occurrence of various gray level values

It is mainly used in manipulating contrast & brightness of a image

Ex: i/p image

4	4	4	4	4
3	4	5	4	3
3	5	5	5	5
3	4	5	4	3
4	4	4	4	4

0 → 0

6 → 0

1 → 0

7 → 0

2 → 0

3 → 6

4 → 14

5 → 5



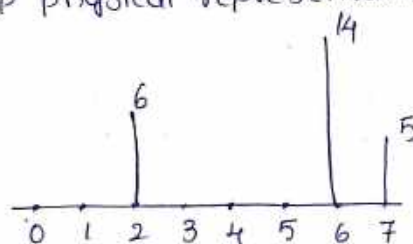
Gray level values	No. of pixels (N _k)	PDF = $\frac{N_k}{\text{Sum}}$	CDF = S _k	S _k × 7	H.E
0	0	0/25 = 0	0	0	0
1	0	0/25 = 0	0	0	0
2	0	0/25 = 0	0	0	0
3	6	6/25 = 0.24	0.24	1.68	2
4	14	14/25 = 0.56	0.5	5.6	6
5	5	5/25 = 0.2	1.0	7	7
6	0	0/25 = 0	1	7	7
7	0	0/25 = 0	1	7	7

sum = 25

o/p image

6	6	6	6	6
2	6	7	6	2
2	7	7	7	2
2	6	7	6	2
6	6	6	6	6

o/p physical representation



Gray level	No. of Pixels	PDF = $\frac{N_k}{\text{Sum}}$	CDF = S _k	S _k × 7	H.E
0	790	0.19	0.19	1.33	1
1	1023	0.24	0.48	3.01	3
2	850	0.20	0.63	4.41	4
3	656	0.15	0.78	5.46	5
4	329	0.08	0.86	6.02	6
5	245	0.05	0.91	6.37	6
6	122	0.02	0.93	6.51	7
7	81	0.01	0.94	6.58	7

sum = 4096

* padding { pixel padding
zero padding

* Median Filtering

ilp:

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

1) pixel padding

step 1:

4	4	3
4	4	3
3	3	1

4 4 3 4 4 3 3 3 1

Ascending order:

4 4 4 4 ③ 3 3 3 1

step 1: 3

step 2:

4	3	2
4	③	2
3	1	2

→ 4 3 2 4 3 2 3 1 2

A.O: 4 4 3 3 ③ 2 2 2 1

step 2: 3

step 3:

3	2	1
3	②	1
1	2	4

→ 3 2 1 3 2 1 1 2 4

A.O: 4 3 3 2 ② 2 1 1 1

step 3: 2

step 4:

2	2	1
2	②	1
2	4	4

2 2 1 2 2 1 2 4 4

A.O: 4 4 2 2 ② 2 2 1 1

step 4: 2

step 5:

4	3	2
3	①	2
5	5	1

4 4 3 3 3 1 5 5 1

A.O: 5 5 4 4 ③ 3 3 1 1

step 5: 3

step 6:

4	3	2
3	①	2
5	1	6

4 3 2 3 1 2 5 1 6

A.O: 6 5 4 3 ③ 2 2 1 1

step 6: 3

step 7:

3	2	1
1	2	4
1	6	2

3 2 1 1 2 4 1 6 2

A.O: 6 4 3 2 ② 2 1 1 1

step 7: 2

step 8:

2	1	1
2	④	4
6	2	2

 2 1 1 2 4 4 6 2 2
A.O: 6 4 4 2 ⑤ 2 2 1 1

step 8: 2

step 9:

3	3	1
5	⑥	1
2	2	3

 3 3 1 5 5 1 2 2 3
A.O: 5 5 3 3 ③ 2 2 1 1

step 9: 3

step 10:

3	1	2
5	①	6
2	3	5

 3 1 2 5 1 6 2 3 5
A.O: 6 5 5 3 ③ 2 2 1 1

step 10: 3

step 11:

1	2	4
1	⑥	2
3	5	6

 1 2 4 1 6 2 3 5 6
A.O: 6 6 5 4 ③ 2 2 1 1

step 11: 3

step 12:

2	4	4
6	②	2
5	6	6

 2 4 4 6 2 2 5 6 6
A.O: 5 6 6 5 ④ 4 2 2 2

step 12: 4

step 13:

5	5	1
2	②	3
2	2	3

 5 5 1 2 2 3 2 2 3
A.O: 5 5 3 3 ② 2 2 2 1

step 13: 2

step 14:

5	1	6
2	③	5
2	3	5

 5 1 6 2 3 5 2 3 5
A.O: 6 5 5 5 ③ 3 2 2 1

step 14: 3

step 15:

1	6	2
3	⑤	6
3	5	6

 1 6 2 3 5 6 3 5 6
A.O: 6 6 6 5 ⑤ 3 3 2 1

step 15: 5

step 16:

6	2	2
5	⑥	6
5	6	6

 6 2 2 5 6 6 5 6 6
A.O: 6 6 6 6 ⑥ 5 5 2 2

step 16: 6

o/p:

3	3	2	2
3	3	2	2
3	3	3	4
2	3	5	6

ii) zero padding

IP

0	0	0	0	0	0
0	4	3	2	1	0
0	3	1	2	4	0
0	5	1	6	2	0
0	2	3	5	6	0
0	0	0	0	0	0

step 1:

0	0	0
0	4	3
0	3	1

 . 0 0 0 0 4 3 0 3 1
4 3 3 1 0 0 0 0 0

step 1: 0

step 2:

0	0	0
4	3	2
3	1	2

0 0 0 4 3 2 3 1 2

4 3 3 2 2 1 0 0 0

step 2: 2

step 3:

0	0	0
3	2	1
1	2	4

0 0 0 3 2 1 1 2 4

4 3 3 2 2 1 0 0 0

step 3: 1

step 4:

0	0	0
2	1	0
0	2	0

0 0 0 2 1 0 0 4 0

6 2 2

step 4: 0

step 5:

0	4	2
0	3	2
0	5	6

0 4 3 0 3 1 0 5 1

5 4 3 3 1 0 0 0

step 5: 1

step 6:

4	3	2
3	1	2
5	1	6

4 3 2 3 1 2 5 1 6

6 5 4 3 3 2 2 1 1

step 6: 3

step 7:

3	1	2
1	2	4
1	6	2

3 1 2 1 2 4 1 6 2

6 4 3 2 2 2 1 1 1

step 7: 2

step 8:

2	1	0
2	4	0
6	2	0

2 1 0 2 4 0 6 2 0

6 4 2 2 2 1 0 0 0

step 8: 2

step 9:

0 0 0 3 5 2 1 1 3

5 3 3 2 1 0 0 0

step 9: 1

step 10:

3 5 2 1 1 3 2 6 5

6 5 5 3 3 2 2 1 1

step 10: 3

Step 11: 1 1 3 2 6 5 4 2 6

6 6 5 4 ③ 2 2 1 1

Step 12: 2 6 5 4 2 6 0 0 0

6 6 5 4 ② 2 0 0 0

Step 13: 0 0 0 5 2 0 1 3 0

5 3 2 2 1 ① 0 0 0 0

Step 14: 5 2 0 1 3 0 6 5 0

6 5 5 3 ② 1 0 0 0

Step 15: 1 3 0 6 5 0 2 6 0

6 6 5 3 ② 1 0 0 0

Step 11: 3

Step 12: 2

Step 13: 0

Step 14: 2

Step 15: 2

Image - 1

Gray level	0	1	2	3	4	5	6	7
NO. OF pixels	80	100	90	60	30	20	10	10

Image - 2

Gray level	0	1	2	3	4	5	6	7
NO. OF pixels	0	0	0	60	80	100	80	70

For histogram matching we need to consider two points

1. Take 1st & last columns of table 2

2. Take last two columns

Table - 1:

Gray level	NO. of pixels	PDF = $\frac{N_k}{\text{sum}}$	CDF = $\sum N_k$	$\sum k \times f_k$	H.E	
0	80	$80/390 = 0.20$	0.20	1.4	1	80
1	100	$100/390 = 0.25$	0.45	3.15	3	100
2	90	$90/390 = 0.23$	0.68	4.76	5	90
3	60	$60/390 = 0.15$	0.83	5.81	6	} 60+30=90
4	30	$30/390 = 0.07$	0.90	6.3	6	
5	20	$20/390 = 0.05$	0.95	6.65	7	} 20+10+0=30
6	10	$10/390 = 0.02$	0.97	6.79	7	
7	0	$0/390 = 0$	0.97	6.79	7	

Sum = 390

Table 2:

Gray level	No. of pixels (N_k)	PDF = $\frac{N_k}{\text{Sum}}$	CDF = S_k	$S_k \times 7$	H.E
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	60	$60/390 = 0.15$	0.15	1.05	1
4	80	$80/390 = 0.20$	0.35	2.45	2
5	100	$100/390 = 0.25$	0.60	4.2	4
6	80	$80/390 = 0.20$	0.85	5.95	6
7	70	$70/390 = 0.17$	0.97	6.79	7

sum = 390

Table-3 1. Take 1st and last column from Table-2
2. Take last 2 columns from table 1

G.L	H.E	H.E $\rightarrow$ G.L	New N_k
0	0	1	80
1	0	3	100
2	0	5	
3	1	6	90
4	2	6	
5	4	7	80
6	6	7	
7	7	7	

Result:

G.L	0	1	2	3	4	5	6	7
No. of pixels N_k	0	0	0	80	80	100	90	30

*Image-1

G.L	0	1	2	3	4	5	6	7
No. of pixels N_k	790	1023	890	656	329	245	122	81

Table-1

G.L	No. of pixels N_k	PDF = $\frac{N_k}{\text{sum}}$	CDF = S_k	$S_k \times 7$	H.E	New N_k
0	790	0.19	0.19	1.33	1	790
1	1023	0.24	0.413	3.01	3	1023
2	850	0.20	0.63	4.41	4	850
3	656	0.15	0.78	5.46	5	656
4	329	0.08	0.86	6.02	6	329 + 574 = 245
5	245	0.05	0.91	6.37	6	
6	122	0.02	0.93	6.51	7	122 + 81 = 203
7	81	0.01	0.94	6.58	7	

sum = 4096

Table:2

G.L	No. of pixels N_k	PDF = $\frac{N_k}{\text{sum}}$	CDF = S_k	$S_k \times 7$	H.E
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	614	0.14	0.14	0.98	1
4	819	0.19	0.33	2.31	2
5	1230	0.30	0.63	4.41	4
6	819	0.19	0.82	5.74	6
7	614	0.14	0.96	6.72	7

Table:3

G.L	H.E	H.E $\rightarrow$ G.L	New N_k	G.L	No. of Pixel
0	0	1	790	0	0
1	0	3		1	0
2	0	4	850	2	0
3	1	5		3	790
4	2	6	574	4	790
5	4	6		5	850
6	6	7	203	6	574
7	7	7		7	203

* Image Enhancement in Frequency Domain

1. In spatial domain direct manipulation of pixels is done.
2. In frequency domain we will convert spatial domain into frequency domain, then it is processed.
3. Here mainly we are using two types of filters i.e. smoothing & sharpening by removing high & low frequencies components.
4. If we apply frequency domain filter on the image then the total image will be changed.
5. Finally Inverse transform is applied to convert it into again spatial/time domain.

* Types of filters

1. Low pass Filter (Smoothing)

It is used to remove noise from a image

It is used for blurring of an image

2. High pass Filter [Sharpening]

It is mainly used to remove background of an image.

These filters again, they are classified into 3 types

* Ideal Filter * Butterworth Filter * Gaussian Filter

Ideal LPF

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

$D_0 \rightarrow$ Non-negative value

$D(u, v) \rightarrow$ Distance b/w u, v

Ideal HPF

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

2. If we consider an image $F(x, y)$ with $M \times N$ size $D(u, v)$ can be given as

$$\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2$$

3. Major demerit of this filter is after processing we will get blurred images so we can't differentiate an object & other data present in the image.

2. If we consider an image $F(x, y)$ with $M \times N$ size $D(u, v)$ can be given as

$$\left(u - \frac{M}{2}\right)^2 + \left(v - \frac{N}{2}\right)^2$$

3.

Butterworth LPF

$$1. H(U,V) = \frac{1}{1 + \left(\frac{D(U,V)}{D_0}\right)^{2n}}$$

here n value varies from 0 to ∞

2. When ' n ' value is increases this filter acts as ideal filter, so we have to choose ' n ' value as slow as possible, so that it can remove blurring edges and edges can be defined clearly.

Gaussian LPF

$$1. H(U,V) = e^{-\frac{D^2(U,V)}{2D_0^2}}$$

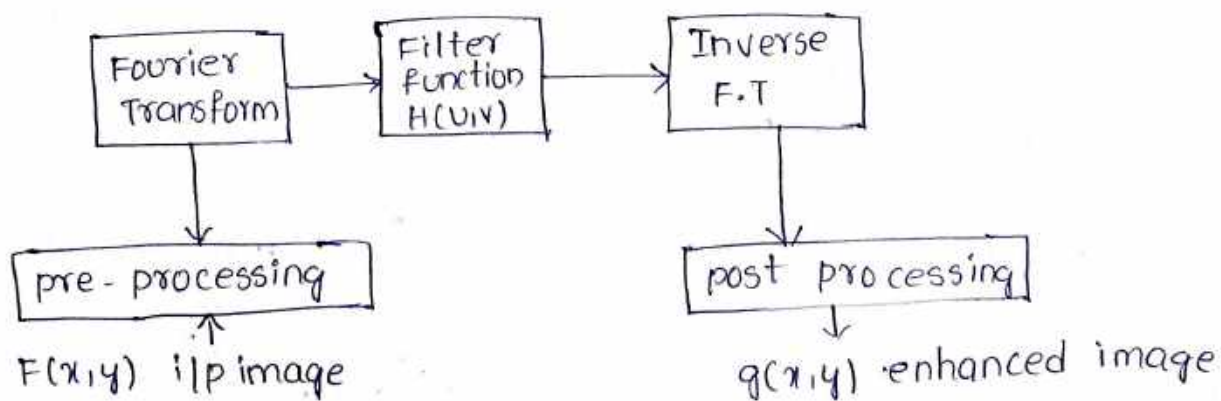
Butterworth HPF

$$1. H(U,V) = \frac{1}{1 + \left(\frac{D_0}{D(U,V)}\right)^{2n}}$$

Gaussian HPF

$$1. H(U,V) = 1 - e^{-\frac{D^2(U,V)}{2D_0^2}}$$

* STEPS FOR FILTERING IN FREQUENCY DOMAIN



* Gray level Transformation

8/09/23

It is given by $s = T(r)$

Here we are mapping r value with the s value with the help of a lookup Table. If we consider 8-bit image then the lookup table consists of 256 entities.

The Transformation function will have values from 0 to 255.

There are 3 basic transformation functions that are frequently used in image enhancement.

1. Linear (Negative & Identity)
2. Logarithmic (log & inverse log)
3. Power law (nth power & nth root)

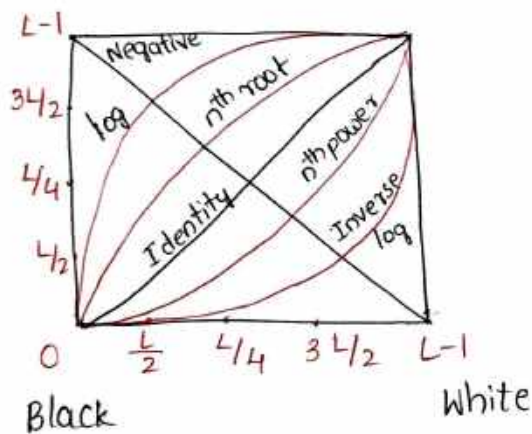
1. Negative $s = L-1-r$
Identity $\rightarrow$ same o/p

2. $s = \log(1+r)$ $r \geq 0$

c - constant

3. power law (gamma law)
 $s = c r^{\gamma}$ $\gamma > 1$ nth power
 $\gamma < 1$ nth root

White



* Linear Transformation

1. Identity

It is less useful in DIP because the o/p image is same as i/p image.

2. Image Negative

It is given by $s = L-1-r$

Intensity level (0 - L-1)

If we give low i/p value then we will get high o/p value & vice versa.

* Logarithmic Transformation

It is given by $s = \log(1+r)$ $r \geq 0$

c = constant

We can compress or expand the gray level value of o/p image
We can also get high/low contrast image depending upon the input value.

If i/p value is low we will get high contrast image as o/p.
& Vice versa

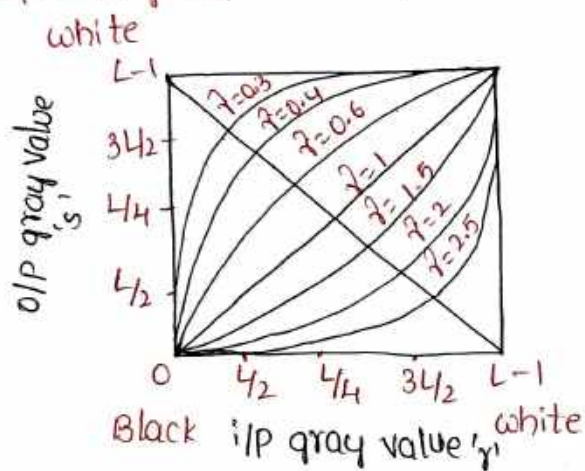
* power law

It is given by $s = c r^\gamma$

If $\gamma > 1$ it is n^{th} power law

If $\gamma < 1$ it is n^{th} root law

here also we can get high/low contrast image as o/p depending upon the γ values.



UNIT-03

IMAGE RESTORATION;

Introduction:

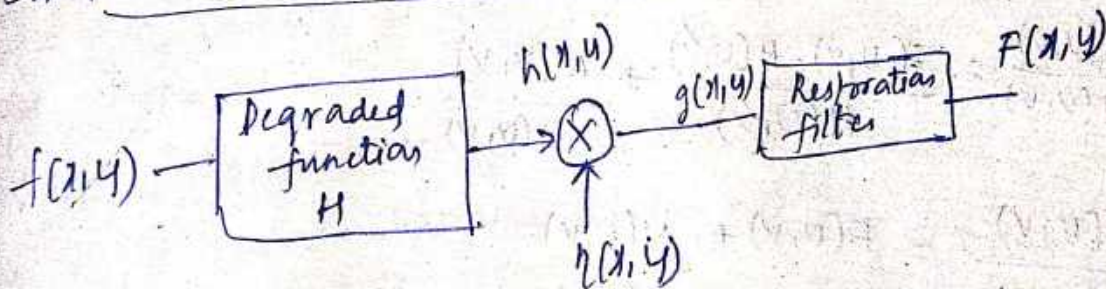
- It is used to Restore/Recover/Reconstruct images that have been degraded, with the help of prior knowledge of ~~degraded~~ function degradation
- It is similar to Image Enhancement but it is objective in nature.
- In Image Enhancement, we are enhancing the image without depending on the noise whereas an Image Restoration can be applied based on the noise.

IMAGE RESTORATION TECHNIQUES

They are classified into two types

- i) Spatial Domain Technique - It is applied when 'Additive Noise' is present in the image
- ii) Frequency Domain Technique - It is applied when there is a blur in the image

SAMPLE Model For IMAGE RESTORATION:



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

where $*$ is convolution operation

- Degraded Image in frequency domain is given by

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

SOURCES OF NOISE

1. Image acquisition
2. Digitalization
3. Image Sensors (Environment factor)
4. Transmission

INVERSE FILTERING:

- ↳ Blur Image
- ↳ Some Noise

Process of removing blurs and noise from image we are using Inverse filtering

From the degradation model, we have

$$g(x,y) = f(x,y) * h(x,y) + n(x,y) \quad \text{--- (1)}$$

$$G(u,v) = F(u,v) \cdot H(u,v) + N(u,v) \quad \text{--- (2)}$$

When $G(u,v)$ is passed through Restoration filter, we get estimated output ie $\hat{F}(u,v)$

Now divide Eqn (2) with $H(u,v)$

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

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

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

for Inverse filtering, Noise $N \approx 0$

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

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

DRAWBACKS OF INVERSE FILTERING

1. It is sensitive to the presence of noise (Applicable only when noise is almost 0)
2. It is unstable (If noise $\uparrow$, filter becomes unstable)
3. It is difficult to build

ADAPTIVE FILTERS

It is used to enhance or restore the data by removing noise without significantly blurring the structures in the image.
 50-60%.

They are classified into 2 types -

- i) Adaptive local noise reduction filter
- ii) Adaptive medium noise filter

* Adaptive Local Noise reduction filter

To measure a random variable image we have to calculate mean and variance

- Mean is a measure of gray level values
- Variance is a measure of contrast values
- These filters based on four quantities

* $g(x,y)$ → Value of noise image at x,y

* σ_g^2 → variance of noise corrupting $f(x,y)$ to form $g(x,y)$

* μ_L → Local mean of the pixel

* σ_L^2 → Local variance of the pixel

$$P^n(x,y) = g(x,y) - \frac{\sigma_g^2}{\sigma_L^2} [g(x,y) - \mu_L]$$

* Adaptive Medium Noise filter:

Medium filter works very well only when the impulse noise is small (or) medium.

Adaptive median filter can't handle impulse noise of larger values

Advantage of adaptive median filter is, it preserves detail of image by smoothing impulse noise

Let $z_{min} \rightarrow$ Min. value of gray level in S_{xy}

$z_{max} \rightarrow$ Max. value of gray level in S_{xy}

$z_{med} \rightarrow$ Med value of gray level in S_{xy}

$z_{max} \rightarrow$ Max allowed size of S_{xy}

At level A:

$$A_1 = z_{med} - z_{min}$$

$$A_2 = z_{med} - z_{max}$$

$A_1 > 0$ & $A_2 < 0$ Go to level B

else

increase window size

If window size $\leq S_{max} \rightarrow$ Repeat

else

Output is z_{xy}

At level B:

$$B_1 = z_{xy} - z_{min}$$

$$B_2 = z_{xy} - z_{max}$$

$$B_1 > 0 \text{ \& \> } B_2 < 0$$

old is z_{xy}

else

z_{median}

WIENER FILTER

- This filter is proposed by scientist ~~Wien~~ Weiner.
- It is an optimum filter.
- It not only restores the image but also removes noise from the image.
- It is assumed that partial knowledge of the degradation function is available.
- In order to design a Weiner filter an estimation of original image and ~~additive~~ ^{additive} noise is required.
- Estimate $\hat{f}(x,y)$ of the original signal $f(x,y)$ such that mean square error is minimized.
- The minimized error is given as

$$e^2 = E \left\{ (F(x,y) - \hat{f}(x,y))^2 \right\}$$

where $E\{ \}$ - Expected value

- The estimation in frequency domain

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

where $S_n(u,v)$ - Power Spectrum of Noise

$S_f(u,v)$ - Power spectrum of image

$$\text{Let } \frac{S_n(u,v)}{S_f(u,v)} = K$$

'K' = Specified constant

$$F(u,v) = \left[\frac{1}{H(u,v)} \cdot \frac{|H(u,v)|^2}{|H(u,v)|^2 + K} \right] G(u,v)$$

• Based on ' γ ' (gamma) value, we get the following combinations

If $\gamma = 0$ — Inverse filter

$\gamma = 1$ — Weiner filter

NOTE: This filter also called as Least Mean Square filter or LMS filter.

IMAGE SEGMENTATION

SEGMENTATION:

It is a process of dividing an image into sub images (Regions) until the region of interest is extracted from a digital image.

Conditions:

1. All the pixel values must lie inside the region
2. All the pixel points present in a region should be connected in 4 (or) 8 connected manner.
3. Two regions must be separated from each other

$$R_i \cap R_j = \phi$$

4. If all the pixels in a region containing the same intensity values then they are represented by a single value.

REGION BASED SEGMENTATION

1. Segmentation is a process in which ^{we} divide the image into subimages and then we apply different rules on it.

2. Separating regions and forming operations on it. With the help of a given rule so as to make suitable Image segmentation.

3. There are four types of Region based Segmentation

i) Region Growing System

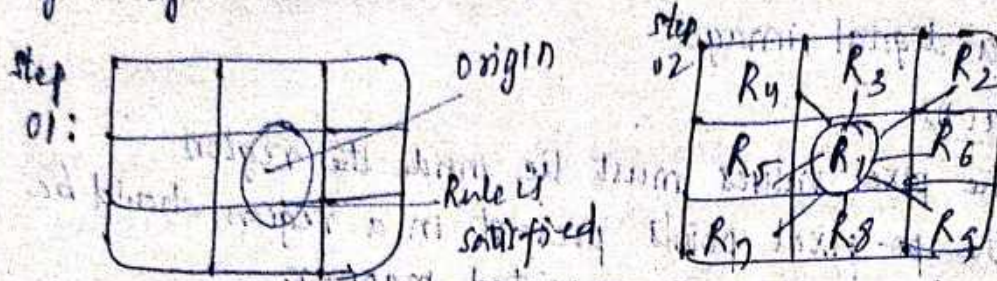
ii) Region Splitting

iii) Region Merging

iv) Region Splitting and Merging

i) Region Growing Segmentation

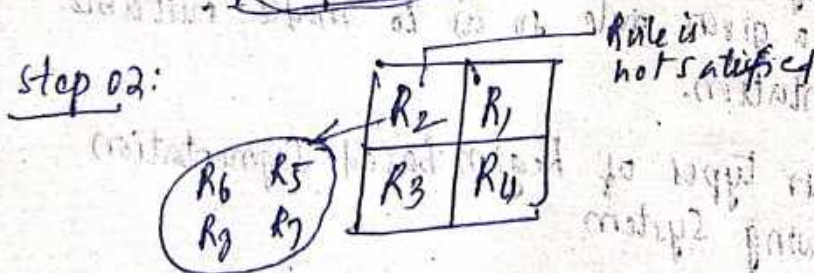
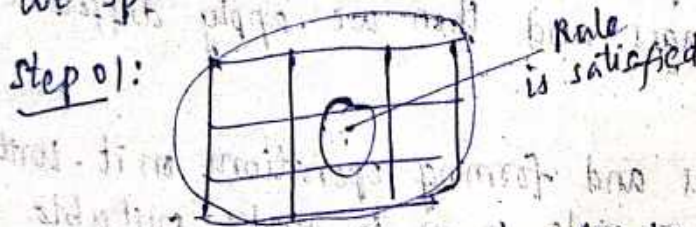
In this segmentation we consider a origin of the image and according to the following rules, if the rule is satisfied then the region will be growing until the rule is not satisfied.



ii) Region Splitting

In this region segmentation we usually splits the region based on the rule. If rule is satisfied, then the total image is considered as single image or region.

If rule is not satisfied on particular region then we split that region.



iii) Region Merging

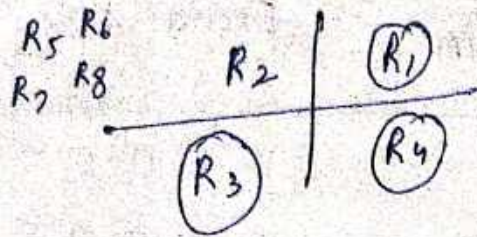
In this segmentation, we usually deal with each and every pixel as a Region.

No. of pixels = No. of Regions in an image.

Ex: If R_1, R_2, R_3 follows the rule then we can merge R_1, R_2, R_3 into single region.

iv) Region Splitting and Merging

Combination of Region Splitting and Region Merging.
In one Region, Region Merging takes place, in other region Region Splitting takes place.



EROSION AND DILATION:

* Erosion: It shrinks foreground and enlarge background

• It is also called as "fits into"

• It is the set of all the points in the image where structuring elements "fits into"

• If each pixel in input image matches with structuring element completely then we write 1 or else '0' at that position.

$$\text{Image} \ominus \text{Structuring element}$$

Application: It shrinks the object

* Dilation: It enlarges foreground and shrinks background

• It is also known as "Touches" or "hits"

• It is a set of all the points in the image where structuring elements "touches" or "hits" the foreground of the image.

• Each pixel in the input image matches completely with the pixel value or even one match with the structuring element is found then we write 1 at that position or else we will write '0'

• Dilation represented with $A \oplus B$

Applications:

- It enlarges the object.
- It repairs the breaks.

Example of Erosion and Dilation

Image(A)

Structuring
element (B)

Erosion $A \ominus B$

```
0 0 0 0 0
0 1 1 1 0
0 1 1 1 0
0 1 1 1 0
0 0 0 0 0
```

```
1 1 1
1 1 1
0 0 0
```

```
0 0 0 0 0
0 0 0 0 0
0 1 0 0 0
0 0 0 0 0
0 0 0 0 0
```

Image(A)

structuring
element (B)

Dilation $A \oplus B$

```
0 0 0 0 0
0 1 1 1 0
0 1 1 1 0
0 1 1 1 0
0 0 0 0 0
```

```
1 1 1
1 1 1
0 0 0
```

template

HIT AND MISS TRANSFORMATION

Used when a particular pattern need to search or extract in the foreground and background of the image.

- If there is a match it is called HIT Transformation.
- If there is no match it is called MISS Transformation.

* HIT:

- for HIT scenario consider all the other ones in the structuring element should match with all the ones in input image.
- Similarly consider all the zeroes in the structuring element should match with zeroes in i/p image. It is called HIT Transformation or True transformation.

Example:

<u>SE</u>	<u>Image</u>
0 0 0	0 0 0
x ① x	1 ① 0
1 1 1	1 1 1

* MISS:

- for MISS scenario consider all the ones in structuring element should match with all the 1's in i/p image.
- Similarly consider all the 0's should not match with 0's in i/p image. It is called MISS Transformation or False Transformation.

Example:

<u>SE</u>	<u>Image</u>
0 0 0	0 0 1
x ① x	x ① 0
1 1 1	1 1 1

But 0's not matched.

MORPHOLOGICAL OPENING & CLOSING OPERATION

* Opening Operation:

- It is used to remove small objects from the foreground of image
- Opening operation is nothing but Erosion followed by Dilation.
- It is represented by $A \circ B = (A \ominus B) \oplus B$

* Closing Operation:

- It is ~~close~~ used to remove small holes from the foreground of the image.
- Closing operation is nothing but dilation followed by Erosion.
- It is represented with $A \bullet B = (A \oplus B) \ominus B$

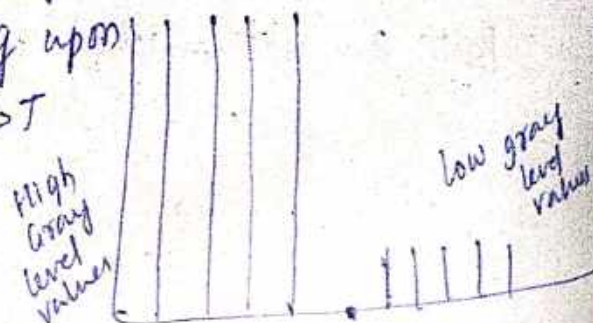
SEGMENTATION BASED ON TRESHOLDING:

- Thresholding is one of type of Image segmentation.
- In thresholding, we are manipulating or changing the pixel values so that it makes easier for the analysis of an image.
- Operating speed of thresholding is very high.
- Segmentation based on thresholding is divided into 3 types:

- 1) Global Thresholding
- 2) Multiple
- 3) Adaptive

Global Thresholding:

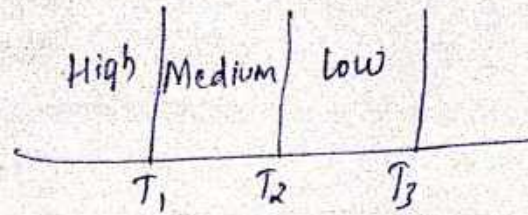
Image histogram is segmented with help of Global Threshold Value (T) depending upon values of Gray level i.e. $\leq T$ or $> T$



Multiple Thresholding

When we want to separate out more than one object from the background of image we choose multiple thresholding.

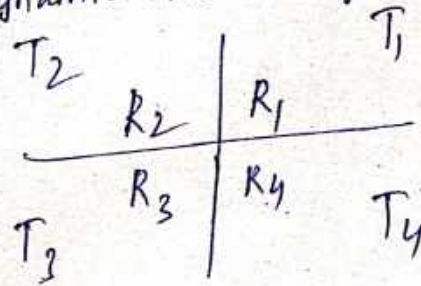
Example:



Adaptive Thresholding

In some cases it is very difficult to separate an object from the background of image then we go for adaptive thresholding. We will partition the entire image into some sub images and we assign different threshold values for different regions. It is also called as local or Dynamic Thresholding.

Example:



Discontinuity in a Digital Image

- A point on the image whose gray level values are different from the image background it is called as Discontinuity or Discontinuity point
- Most common approach to find discontinuity in the image is to run $n \times n$ mask over each point in the image
- A point is the most basic type of discontinuity in a digital image
- There are 3 types of Discontinuity in Digital Image
 - i) Point Discontinuity (or) Point Detection
 - ii) Line Discontinuity (or) Line Detection
 - iii) Edge Discontinuity (or) Edge Detection

Point Discontinuity:

- An isolated point is a point whose gray level value is significantly different from its background.
- A mask is superimposed on the image then Convolution is applied to get the response of the mask.

Example:

Image		
w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

Mask		
z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

$$R_i = \sum w_j z_j$$

- To identify the isolated availability of isolated point, we need to check Threshold value (T) (which is used to identify the points).

$R \geq T \rightarrow$ a point is detected

Sample Mask

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

Line Detection:

There are 4 types of masks are used to get the responses of line Detection.

i) Horizontal line Detection

-1	-1	-1
2	2	2
-1	-1	-1

ii) Vertical line Detection

-1	2	-1
-1	2	-1
-1	2	-1

iii) +45 line Detection

-1	-1	2
-1	2	-1
2	-1	-1

iv) -45 line Detection

2	-1	-1
-1	2	-1
-1	-1	2

Edge Detection:

It is a point on the image where pixels of two different gray level values get connected

Edges provides outline of the image by performing the derivation operation on an image is Magnitude (Contrast of the image). and Dilation (Range of Image)

Some of the commonly encountered edges in Image processing are -

i) Step Edge Detection - Sudden change in intensity

ii) Ramp Edge Detection - Slow and gradual change in intensity

iii) Spike Edge Detection - ~~fast~~ Simultaneous change in intensity value



Morphological Gradient

- Diff. b/w dilation and erosion of an image is known as Morphological gradient
- Denoted with the symbol ' g '

$$g = (a \oplus b) - (a \ominus b)$$

where a = Image

b = structuring Element

- Dilation increases the density of the region and Erosion compresses the density of the region.



IMAGE COMPRESSION

Introduction:

• COMPRESSION-

It is a process of reducing the size of a file without disturbing the quality of a original file, so that storage space required is reduced, and transmission speed is increased. We are doing compression, so that they are suitable for upgraded technologies.

• IMAGE COMPRESSION-

- It is also called as Data Compression.
- It is a process of reducing the amount of data required (or) reducing the no. of pixels so that storage space is reduced and transmission speed is increased.

• REDUNDANCY:

- Sometimes there is irrelevant or repetitive information present in the data it is called as Redundancy.
- They are classified into 3 types
 - ① Code Redundancy (or) Coding Redundancy
 - ② Psycho visual Redundancy
 - ③ Inter pixel Redundancy

• CODE REDUNDANCY:

When images are represented in the form of code. If there is a repetition in the code and when we are removing that code from the Image, that is called Code Redundancy.

PSYCHO VISUAL REDUNDANCY:

- It is completely subjective in nature which depends on visual perception.
- There will be more information loss in this redundancy.

INTER PIXEL REDUNDANCY:

- The pixel values must be isolated from its neighbouring pixel values.
- It is also called as Geometric Redundancy.

FIDELITY CRITERIA:

- It is used for removal of Psycho visual Redundant data
- It is divided into two types
 1. Objective fidelity criteria
 2. Subjective fidelity criteria

OBJECTIVE FIDELITY CRITERIA:

- The information loss can be expressed in terms of compressed and decompressed image
- Let us assume, $F(x,y)$ as input and $\hat{F}(x,y)$ as output

$$e(x,y) = F(x,y) - \hat{F}(x,y)$$

- The total error between $M \times N$ size image is given by

$$e(x,y) = e_{\text{total}} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F(x,y) - \hat{F}(x,y)$$

- The RMS value between input and output is given by

$$e_{\text{rms}} = \sqrt{\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [F(x,y) - \hat{F}(x,y)]^2}$$

$$e_{\text{rms}} = \sqrt{\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [F(x,y) - \hat{F}(x,y)]^2}$$

SUBJECTIVE FIDELITY CRITERIA

- We know that human perception based on subjective quality so it is more suitable to use this fidelity criteria to evaluate the information loss.

↳ Concept - It is implemented by showing a typically decompressed image to the no. of viewers.

- Their evaluation is averaged to take a decision.
- These evaluations may be done using a rating scale (0-5), side by side comparisons etc.

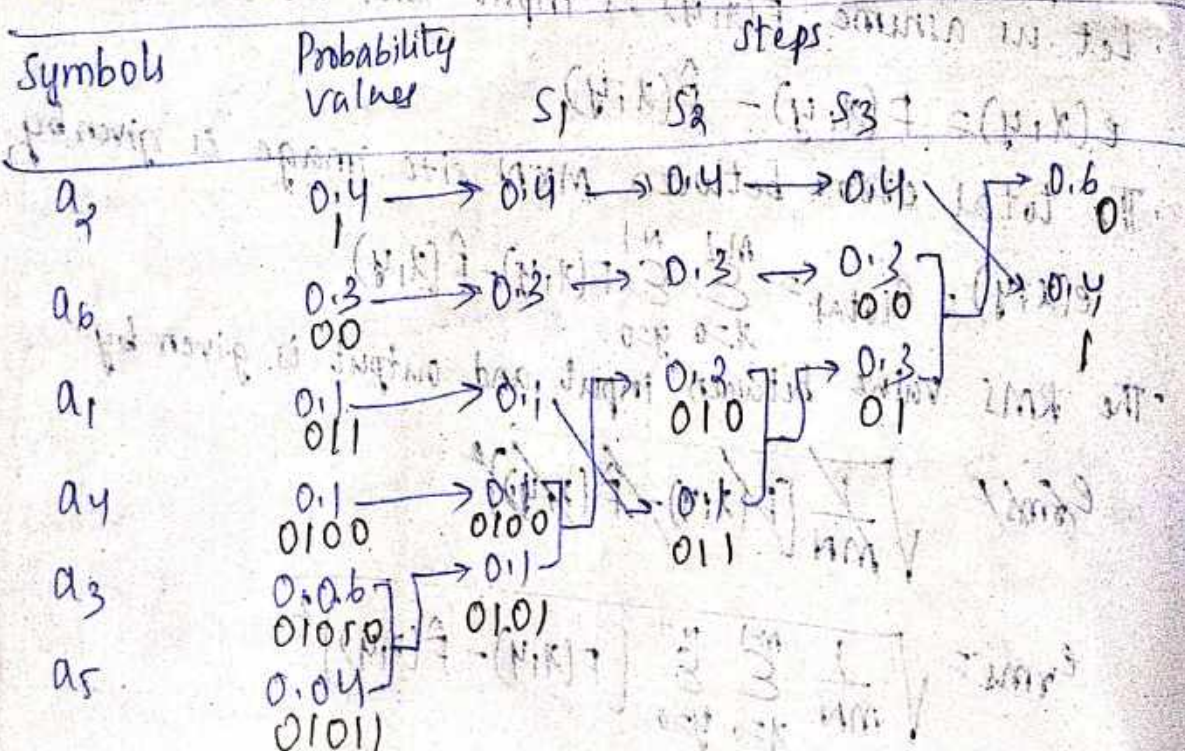
* HUFFMAN CODING

- It is a lossless data compression algorithm.
- It was developed by David Huffman.
- It is useful to compress the data of frequently occurring characters.

Example:

$a_2=40$ $a_6=30$ $a_1=10$ $a_4=10$ $a_3=6$ $a_5=4$ (symbols)

$a_2=0.4$ $a_6=0.3$ $a_1=0.1$ $a_4=0.1$ $a_3=0.06$ $a_5=0.04$ (Probability values)



on some fields of

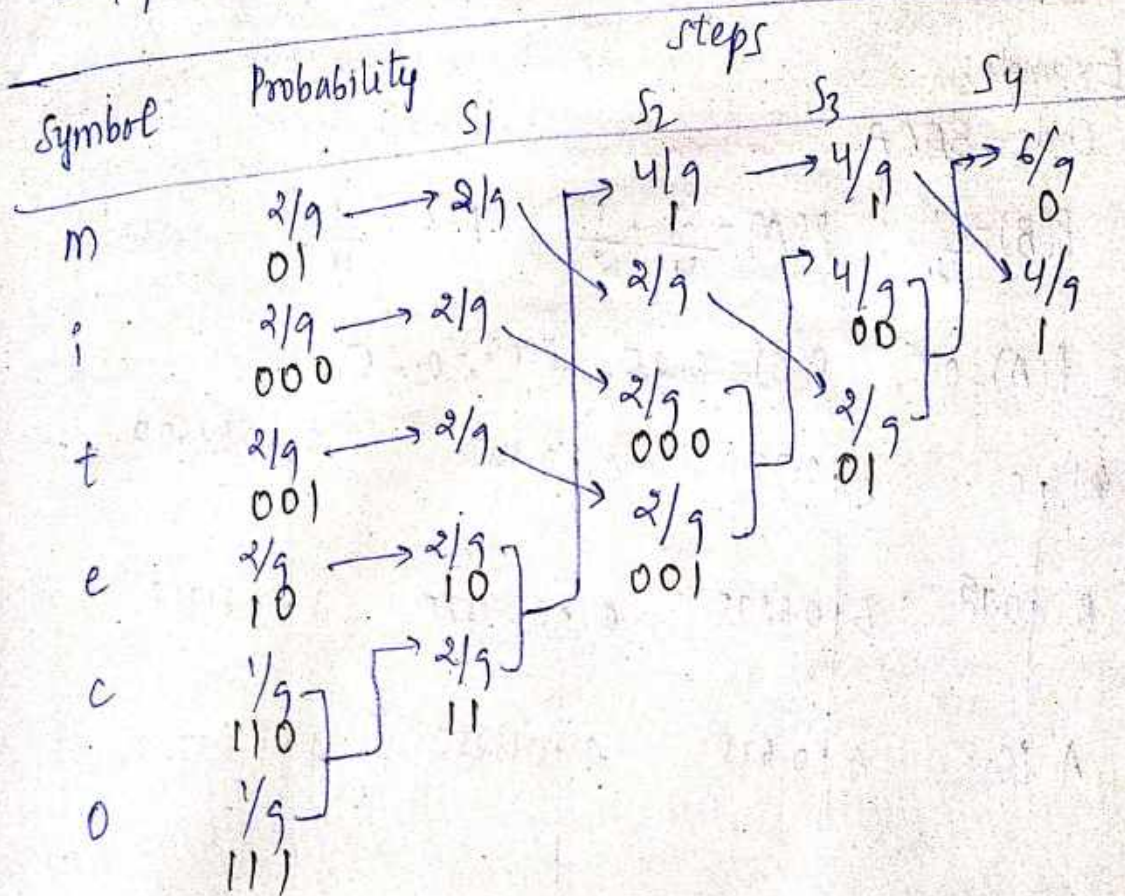
101

Output:

Symbol	Probability	Codeword	Codelength
a_2	0.4	1	1
a_6	0.3	00	2
a_1	0.1	011	3
a_4	0.1	0100	4
a_3	0.06	01010	5
a_5	0.04	01011	5

Example 02:

$$c = 1/9 \quad o = 1/9 \quad m = 2/9 \quad i = 2/9 \quad t = 2/9 \quad e = 2/9$$



Output:

Symbol	Probability	Codeword	Code Length
m	$\frac{2}{9}$	01	2
i	$\frac{2}{9}$	000	3
t	$\frac{2}{9}$	001	3
e	$\frac{2}{9}$	10	2
c	$\frac{1}{9}$	110	3
o	$\frac{1}{9}$	111	3

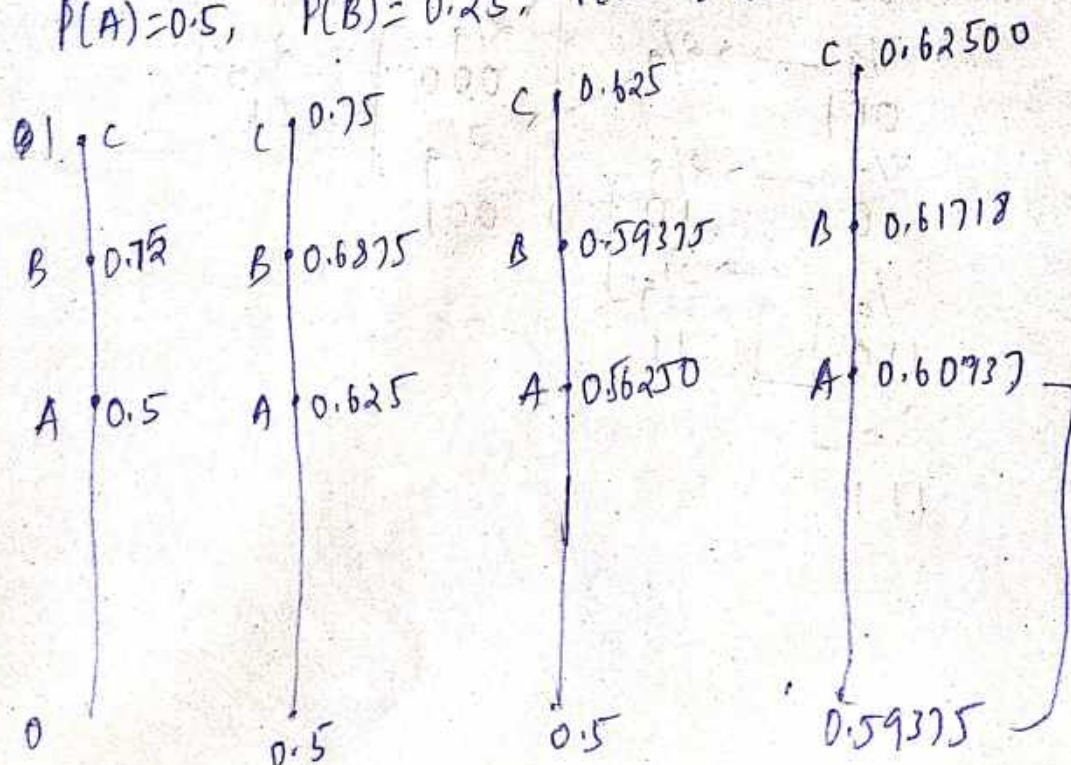
ARITHMETIC CODING:

Example:

Code = BACA

$$P(B) = \frac{1}{4}, \quad P(A) = \frac{2}{4} = \frac{1}{2}; \quad P(C) = \frac{1}{4}$$

$$P(A) = 0.5, \quad P(B) = 0.25, \quad P(C) = 0.25$$



$$\text{Tag} = \frac{\text{Upper limit} + \text{Lower limit}}{2}$$

$$= \frac{0.60937 + 0.59375}{2}$$

$$= 0.60156$$

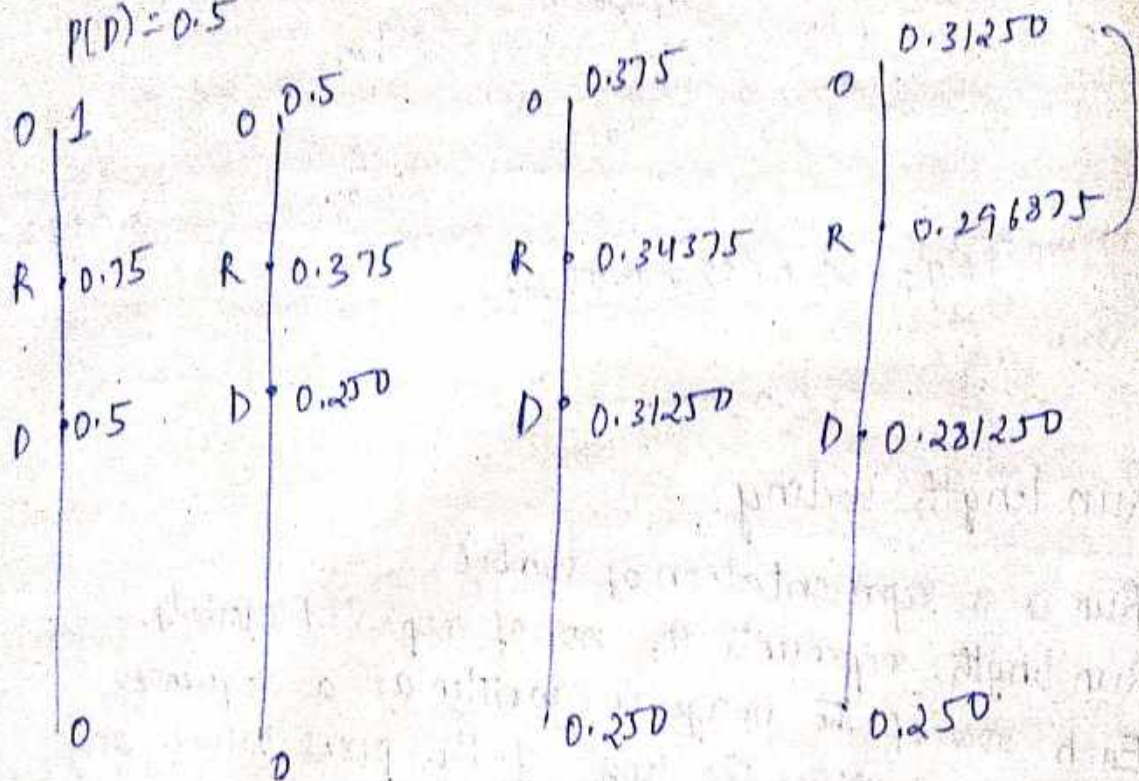
Example 02:

Code - DRDD

$$P(D) = \frac{2}{4} = \frac{1}{2} \quad P(R) = \frac{1}{4} \quad P(O) = \frac{1}{4}$$

$$R(D) = 0.025 \quad P(R) = 0.25 \quad P(D) = 0.25$$

$$P(D) = 0.5$$



$$0.31250$$

$$\text{Tag} = \frac{0.31250 + 0.296875}{2}$$

$$= 0.3046875$$

$$0.296875$$

Alphabet

Characters = 4

$\frac{\text{Upper} - \text{lower}}{4} = 1 \text{ part}$

Here $D = 2 \text{ parts}$

$D = 1 \text{ pt} + 1 \text{ pt}$

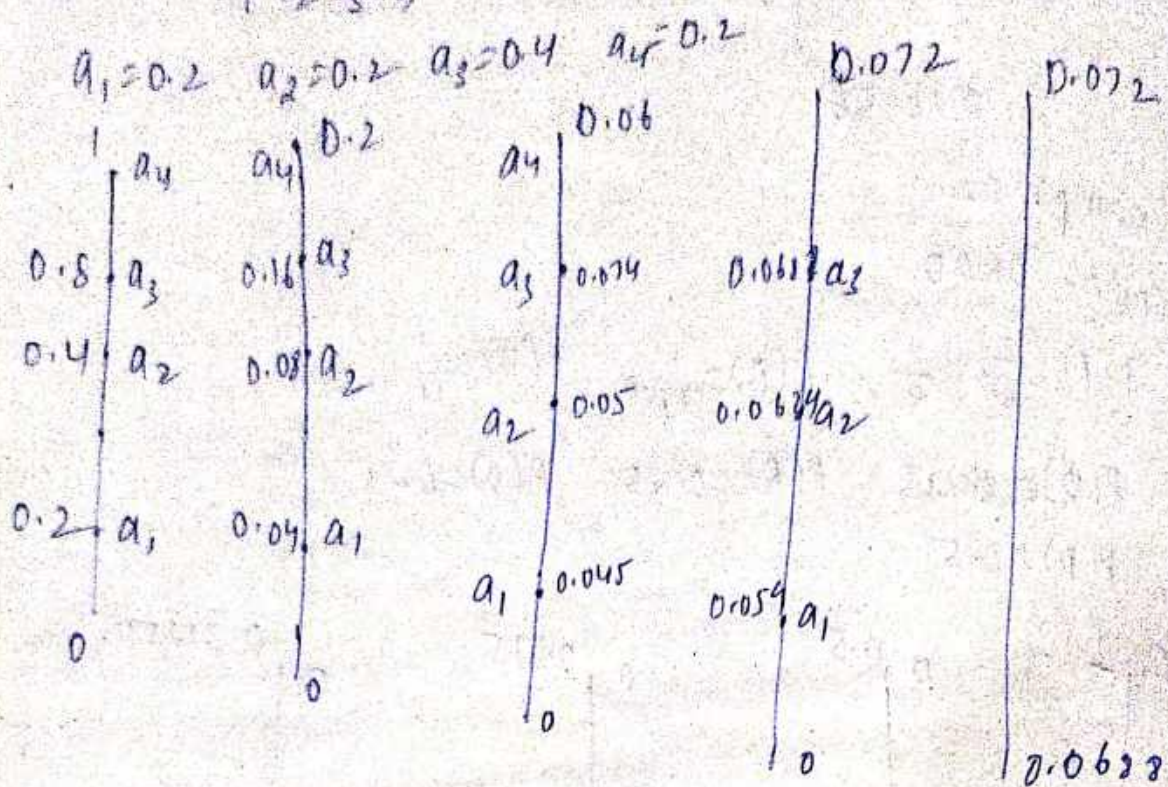
Then for $D = \text{lower pt}$

For $R = D + 1 \text{ pt}$



Example

Code - $a_1 a_2 a_3 a_4$



$$\text{Tag} = \frac{0.0688 + 0.0722}{2}$$
$$= 0.0704$$

Run length Coding:

- Run is a representation of symbol.
- Run length represents the no. of repeated symbols.
- Each row of the image is written as a sequence.
- The RLC identifies the length of the pixel values and encodes the image in the form of Run.
- The length is represented as a Run of Black and White pixels.
- RLC classified into 2 types:
 - 1) 1D RLC
 - 2) 2D RLC

→ 1D RLC:

- It consists of only horizontal correlation between the pixels on the scan line.
- In this technique individual encoding of each scan line takes place.

Example:

0	0	0	0	0	—	(0,5)
0	0	0	1	1	—	(0,3) (1,2)
1	1	1	1	1	—	(1,5)
1	1	1	1	1	—	(1,5)
1	1	1	1	1	—	(1,5)

No. of ~~vec~~ vectors = 6

Max. length = 5

No. of bits required = 3

No. of bits required per pixel = 1

Total No. of pixels = No. of vectors $\times$ (No. of bits required + No. of bits required per pixel)

$$= 6 \times (3+1)$$

$$= 24$$

Total No. of pixels = 25

So, Image is compressed.

→ 2D RLC:

- It consisting of both horizontal and vertical correlation between the pixel of the neighbouring scan

Example:

0	0	0	0	0	(0,2) (1,3)
0	0	0	1	1	(0,2) (1,3)
1	1	1	1	1	(0,2) (1,3)
1	1	1	1	1	(0,1) (1,4)
1	1	1	1	1	(0,1) (1,4)

No. of vectors = 10

Max. length = 4

No. of bits required = 3

No. of bits required per pixel = 1

Total No. of pixels = No. of vector $\times$ (No. of bits required per pixel)

$$= 10 \times (3+1)$$

$$= 10 \times 4$$

$$= 40$$

Total No. of pixels = 25

So, Image is Enhanced

JPEG :- (Applicable for lower and medium bit rate)

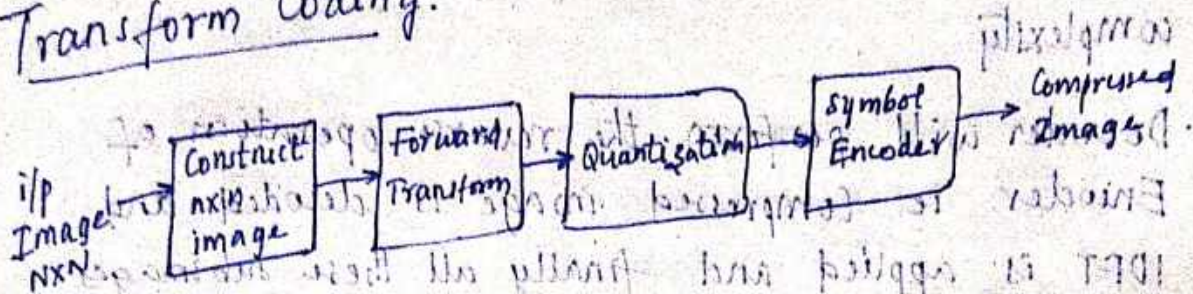
- JPEG means Joint Photographic Expert Group.
- This is the first compression standard developed by ISO (International Standard Organisation).
- It is mainly applicable to Image formats only.
- It is utilizing 2^{24} colours combination of an image.
- Sometimes when more no. of changes occurs in some pixel values makes that part of the image to have high frequency values, generally it is high for color images and low for black and white images.
- It achieves good degree of compression

JPEG 2000 standard :- (for high or more quality higher bit rate)

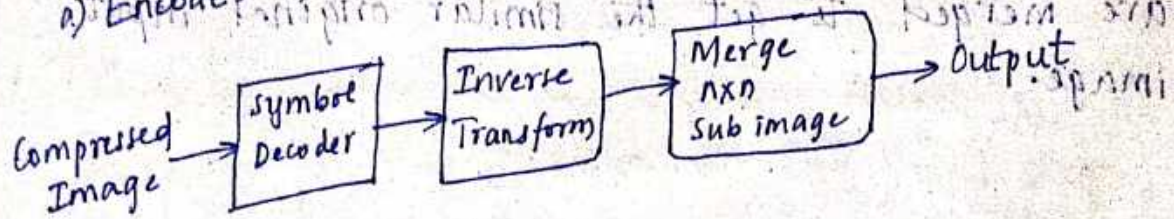
- It is used to increase features like compression, scalability
- Its design has ability to handle large range of bit rate.
- In JPEG standard, the resolution of the input image is reduced before Encoding, whereas JPEG 2000 standard the resolution is automatically reduced with the help of Multiresolution Decomposition structure

- It gives superior Compression Performance.
- Image can be represented in Multiple resolution.
- Both Lossless and Lossy compressions can be achieved in a single compression architecture.

Transform Coding:



a) Encoder



b) Decoder

- Transform coding is a lossy compression technique.
- In this coding, input image $N \times N$ is subdivided into $n \times n$ sub images.
- These sub images are allowed for transformation i.e. DFT in order to decorrelate the pixels of each sub image and store large amount of information into smaller coefficients.
- Some of the small coefficients stores huge amount of data and some of the coefficients stores less amount of data.
- When we apply Quantization, the ~~are~~ coefficients that have least or low information are removed and the coefficients that have huge or high information are allowed to pass during the quantization stage.

