

(10)

Image:

classmate

Date _____

Page _____

Chapter-1

Introduction To Digital Image Processing

(4 hours)

Introduction:

The pictorial representation of an object is called formation of an image

A digital image is a 2-D light intensity function $f(x, y)$ where x and y denotes spatial co-ordinates and value of function f at any point (x, y) is proportional to the brightness (gray level) of an image at that point (x, y) .

Digital Image Processing refers to the processing of digital image by using digital computer on the basis of digitization (sampling and quantization).

Digital Image Representation:

A digital image is an image $f(x, y)$ that has been discretized both in spatial co-ordinates and in brightness

It is represented by 2-D integer array. The digitized brightness value is called gray level values.

Digital image can be represented as a matrix as shown in figure below:

$$f(x, y) = \begin{bmatrix} f(0,0), f(1,0) & \dots \\ \dots & f(x_{\max}, y_{\max}) \end{bmatrix}$$

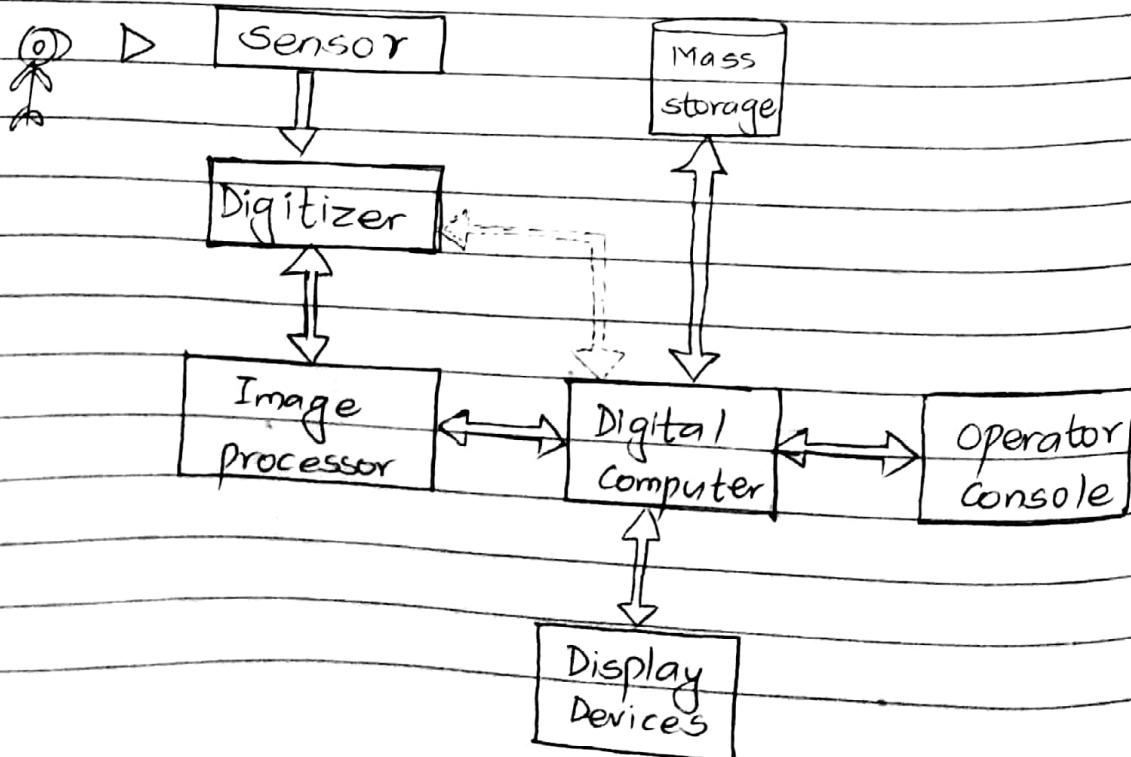
(fig: Representation of Digital Image As Matrix)

Here,

the row and column identify the point in the image and the corresponding matrix elements values identify the gray level at that point.

The element in digital image is called Pixel or pels or Picture Points.

Elements (Components) of DIP:



① Sensor :

It intercepts / the energy propagation from the scene and transform it to produce an intensity image.

camera and scanner are the examples of photo sensor.

② Digitizer :

The digitizer produce an image by the digitization technique.

Normally, a digitizer converts an intensity image into numerical representation which is suitable for import into a digital computer.

③ Image Processor / Pre-Processor :

It interface with digital computer to provide easy of programming. For this purpose, we can use micro-computer or mainframe computer.

Specially, it does functions like image acquistion, low level processing, scaling and rotating.

④ Digital Computer :

It always digitize the required image by the help of image pre-processor.

⑤ Mass Storage :

It is used to store digital image line by line. The storage devices may be magnetic tape, hard disk etc.

⑥ Display Devices:

It produce and shows a visual of numerical values stored in a computer.

The example of display devices are: CRT monitor, printer etc.

⑦ Operator Console:

It controls the full mechanism of image processing system.

⑧ Applications of Digital Image Processing:

① Office Automation:

~~BTO MEC I~~ EX: OCR (Optical character Reader) & document processing etc.

② Industrial Automation:

EX: Robotics, Automatic assembly etc.

③ Bio-Medical Automation:

EX: ECG, EEG, X-Ray etc.

④ Information Technology (IT):

EX: Video phone, fax, video conferencing, etc.

⑤ Entertainment:

Ex: Video game, visual Animations, etc.

⑥ Military Application:

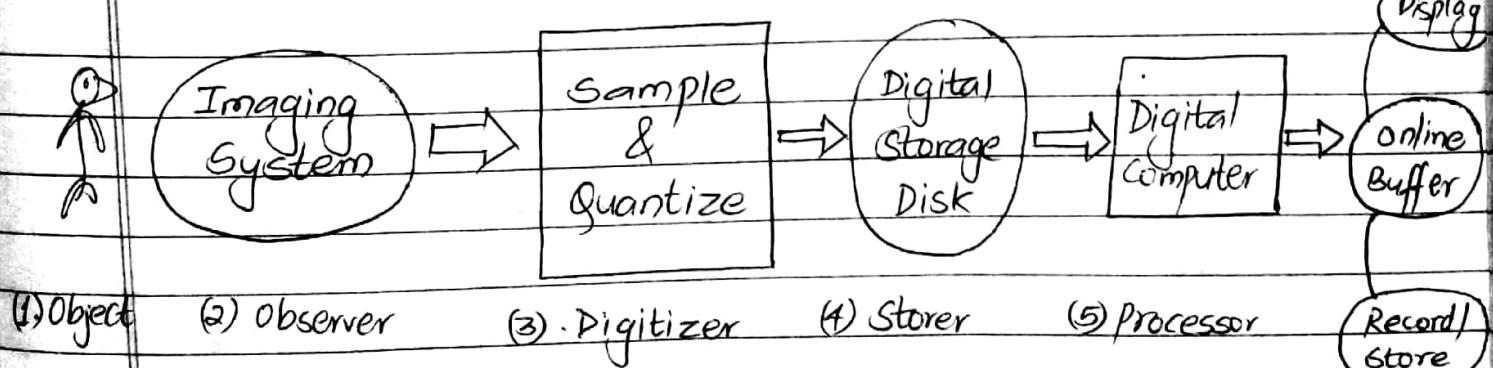
Ex: Target Identification and missile guidance and detection.

⑦ Criminology:

Ex: Bio-metric detection such as finger print detection, face detection, retina detection, etc.



A typical Image Processing Sequence:



(fig: A typical Image processing sequence)

In typical image processing sequence involves various components.

The image of an object travels along the different blocks.

Firstly, the object is sensed by the ^{imaging} system and image is formed. This image is digitized by the process of sampling and quantization.

Now, image is in a digital form. A digital image is stored in digital storage system. The digital computer is connected to digital storage disk where all the manipulation of digital image is done.

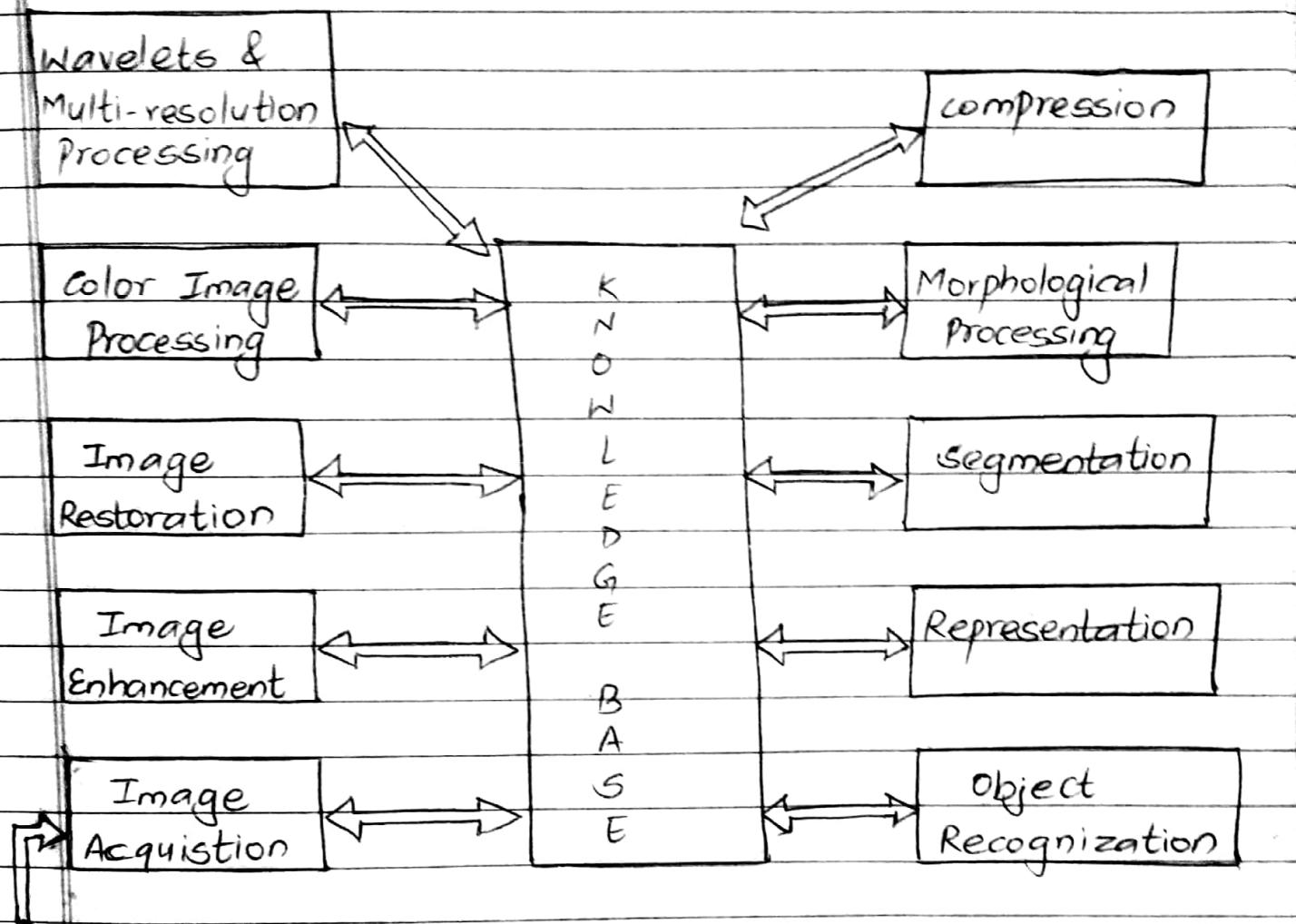
Here, also online buffer mechanism is available where the information are stored as memory.

The image is displayed through the display block which may be color monitor, color TV etc. Finally, the image is recorded on a recording block.

But sometimes some problem may occur when the image is processing sequentially into a digital computer.

(2015 fall)

Fundamental steps in DIP:



problem
domain

(fig: knowledge Base fundamental steps in DIP)

① Image Acquistion:

It is the first process in which pre-processing activities like scaling and rotating are done.

② Image Enhancement:

In this stage, the original image is

processed in such a manner that the resultant image will be more suitable with compared to original image.

③ Image Restoration:

It is an area that also deals with improving appearance of an image. Basically, image enhancement is subjective process but image restoration is objective process.

④ Color Image Processing:

With the help of color, object identification and abstraction from the scene is processed.

⑤ Wavelets & Multi-Resolution Processing:

Foundation for the representing image in various degree of resolution deals to the wavelets and multi-resolution.

Sometimes, it is also used for image compression.

⑥ Compression:

It deals with the technique for reducing the storage required to save an image to transmit it.

⑦ Morphological Processing:

It is the combination of structure. So, it deals with the tools for extraction of an image component that are useful in representation and description of shape.

⑧ Segmentation:

It deals with the partition of an image into different parts and objectives.

⑨ Representation:

After an image has been segmented into sub-regions then the resultant segment-ed pixels is represent and described in a suitable form for further analysis.

⑩ Object Recognition:

It process that. the level to an object based on its descriptors.

(2021 fall)

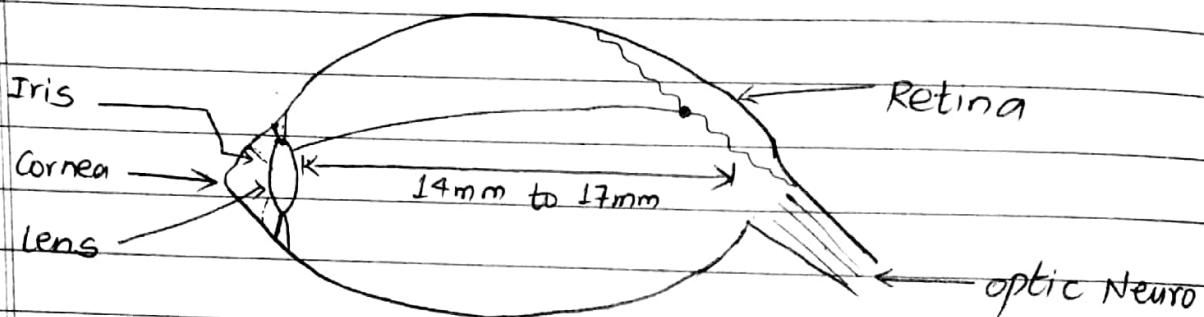
classmate

Date _____

Page _____



Elements of Visual Perception:

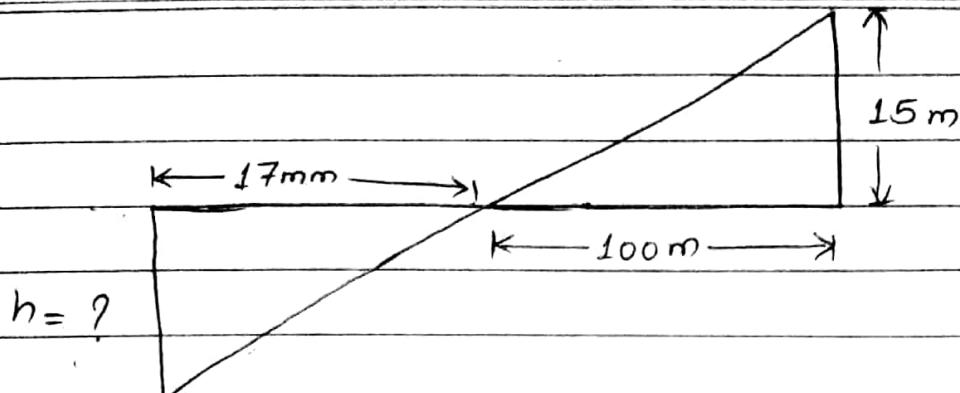


(fig: cross section of Human Eye)

Image analysis plays central role in the image processing. Therefore, the study of human visual perception is important which is represented as cross-section of human eye as shown in figure above.

The human eye nearly a sphere with an average diameter of 14mm to 17 mm. The lens of the eye is flexible than the ordinary optical lens.

let, the observer is looking at a tree 15m height at the distance of 100m. Find the height of retrieval image.



Here,

$$\frac{h}{17 \text{ mm}} = \frac{15 \text{ m}}{100 \text{ m}}$$

$$\Rightarrow h = \frac{15 \times 17}{100} = 2.55 \text{ mm}$$

\therefore Height of Retrieval Image (h) = 2.55 mm

(V.Imp)
#

Sampling & Quantization:

A physical image is changed into digital image by performing sampling and quantization.

This process is also known as Digitization:

Sampling:

(It is the process where measurement are taken at regular space intervals.)

Digitizing the spatial co-ordinates value (x, y) is called Sampling.

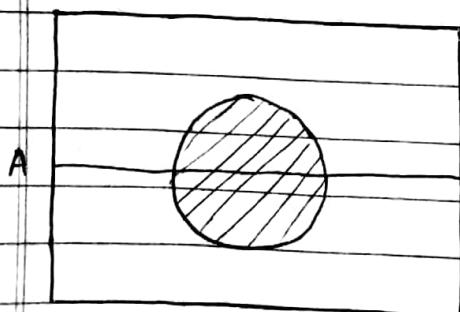
It is the process by which image is formed over a patch (domain) in a continuous domain is mapped into a discrete point with integer co-ordinates.

Hence, Sampling can be defined as a selection of discrete location in the continuous 2-D space.

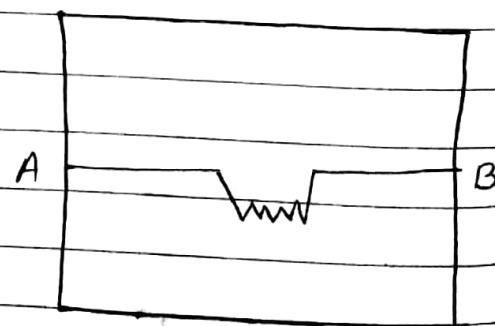
Quantization:

It is the process of mapping the measured intensity to one of the finite number of discrete levels.

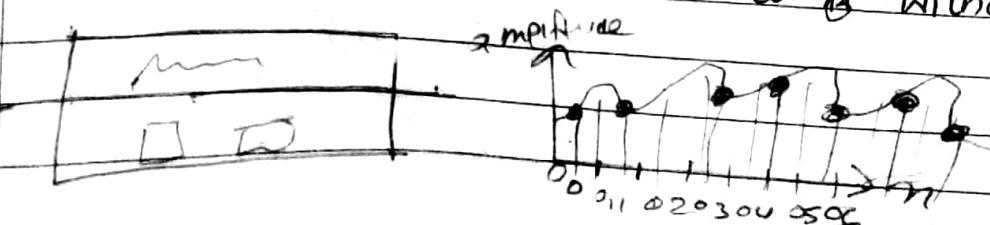
Hence, digitizing the amplitude values is called quantization or gray level quantization.

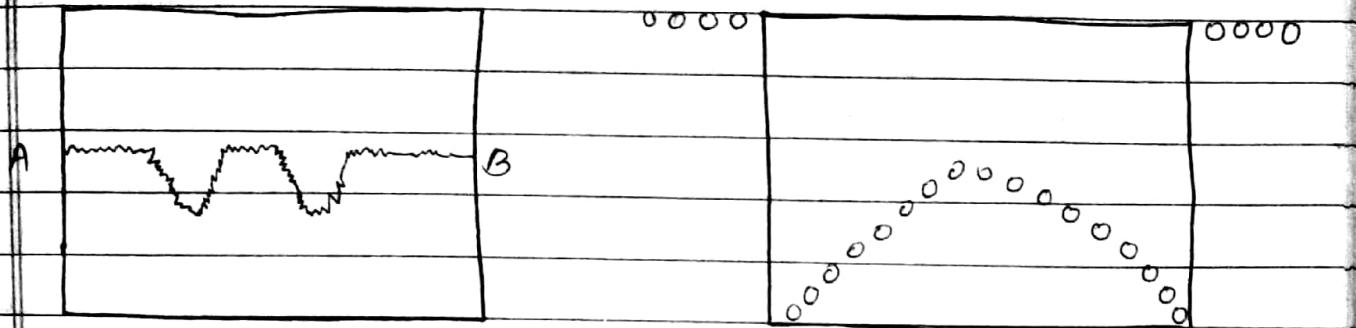


(1) Continuous Type



(2) Ideal Scan line A to B without noise

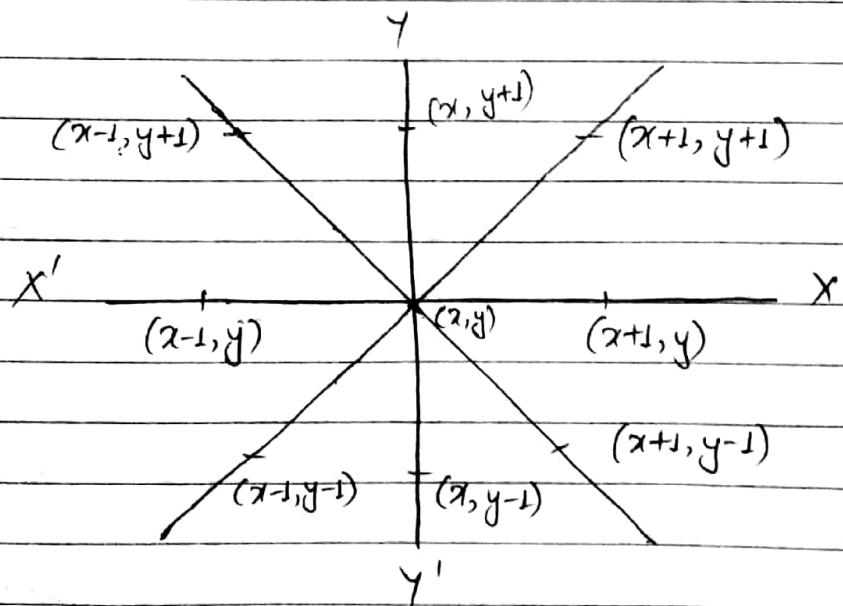




(3) Scan Line A to B
in image with noise

(4) Sampling & quantization

Neighbours of pixels:
(Arrangement of pixel in Image)



(fig: Neighbours of pixel)

A pixel at co-ordinate (x, y) has a four horizontal and four vertical neighbours whose co-ordinates are given by $(x-1, y+1)$, $(x, y+1)$, $(x+1, y+1)$, $(x+1, y)$, $(x+1, y-1)$, $(x, y-1)$, $(x-1, y-1)$, $(x-1, y)$.

These set of pixel is called four neighbours of horizontal and four neighbours of vertical.

Each pixel is at unit distance from (x, y) and called as $N_8(P)$.

To create a digital image, we need to convert the continuous sensed data into digital form.

This involves two processes : Sampling & Quantization

An image may be continuous with respect to x - and y -coordinates, and also in amplitude.

To convert it to digital form, we have to sample the function in both co-ordinates & in amplitude.

Digitizing the coordinate value is called Sampling:

And, Digitizing the amplitude value is called Quantization.

(20)

classmate

Date _____

Page _____

Chapter - 2

Image Enhancement In Spatial Domain

[7 hours]

Introduction:

Image enhancement refers to the sharpening or to improve the quality of an image by enhancement of pixel or point in a spatial domain.

Some basic gray level transformation is needed to enhance the image in a spatial domain.

Some basic gray level transformation is listed as below:

- (1) Point Operation
- (2) Contrast Stretching
- (3) Thresholding
- (4) Digital Negative
- (5) Intensity Level Slicing (Gray level slicing)
- (6) Bit Plane Slicing

[P.T.O]

~~(pre-university)~~ ~~(classmate)~~ ~~Date _____~~ ~~Page _____~~ ⑦ Point Operation:

Point operation is defined as a function that are performed on each pixel of an image.

Point operation is a unary point operation if single image is modified.

The operation can be binary which means that two images combined in the same manner.

The operation can be ternary if three images are used in same manner at operation.

Unary Operation can be defined as,

$$T_{xy} = \text{function}(P(x,y))$$

Where,

T_{xy} = Transformed Pixel value

function = Numeric Transform

$P(x,y)$ = Input pixel value

This operation can be used in many situations and for different purposes.

such as contrast stretching, noise clipping, window slicing and histogram modeling.

So, in point operation a given gray level $u \in [0, L-1]$ is mapped into a gray level $v \in [0, L-1]$ according to a transformation

$$\text{i.e. } v = f(u)$$

$$(L-1 = 255) \\ (L \rightarrow \text{Max. Gray Level})$$

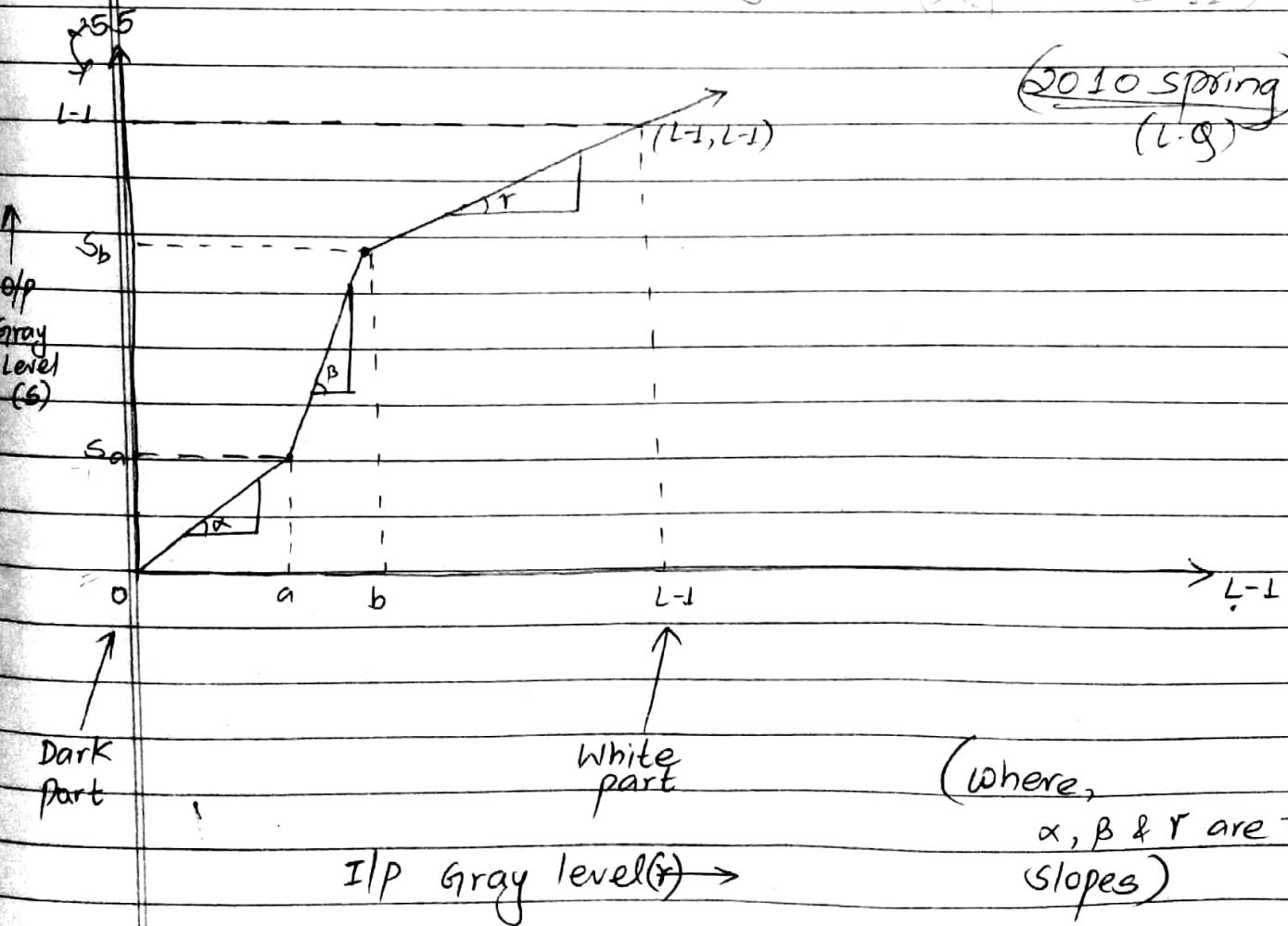
(IMP)



Contrast Stretching:

(B.E(n → 2012))

(2010 spring)
(I.G)



(fig: Contrast Stretching Transformation)

In many times, we obtain low contrast image due to poor illumination or due to the wrong setting of the lens aperture.

~~(ref)~~ The idea behind the contrast stretching is to increase the dynamic range of gray level in the image.

The main idea is to increase the contrast of the image by making the dark portion darker and bright portion brighter.

To obtain the contrast stretching we should have some formulations as below:

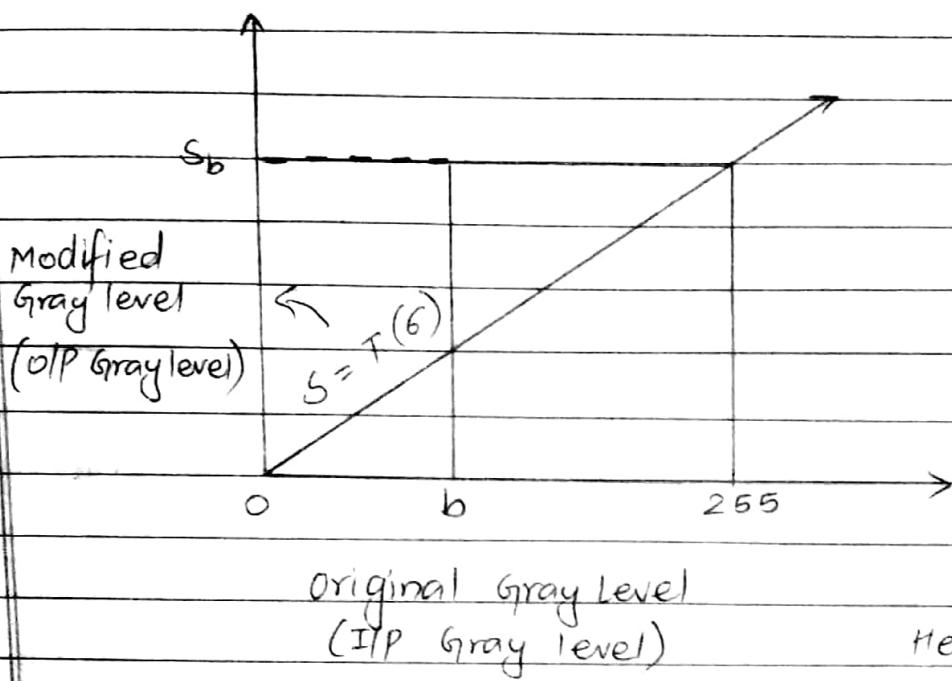
$$S = \begin{cases} \alpha r & , 0 \leq r < a \\ \beta(r-a) + S_a & , a \leq r < b \\ \gamma(r-b) + S_b & , b \leq r < k-1 \end{cases}$$

In the figure, 'r' represents the horizontal axis (i.e I/P gray level) and 's' represents the vertical axis (i.e O/P gray level.)

Here,

[darker gray level] is obtained by making a slope < 1 and [brighter] is obtained by making a slope > 1 .

Thresholding (Binary Image)



Here,

$$b = 100 \text{ (threshold value)}$$

(fig: Thresholding)

Thresholding is the simplest method of the image processing. It can be used to create binary image.

If we observe the contrast stretching diagram, we notice that the first and last slopes are made to zero.

$$\text{If } r_1 = r_2 \quad \& \quad S_1 = 0, \quad S_2 = L-1$$

Now, The thresholding function is given below:

$$S = 0, \quad \text{if } r < b \Rightarrow 0$$

$$S = L-1, \quad \text{if } r \geq b \Rightarrow 1$$

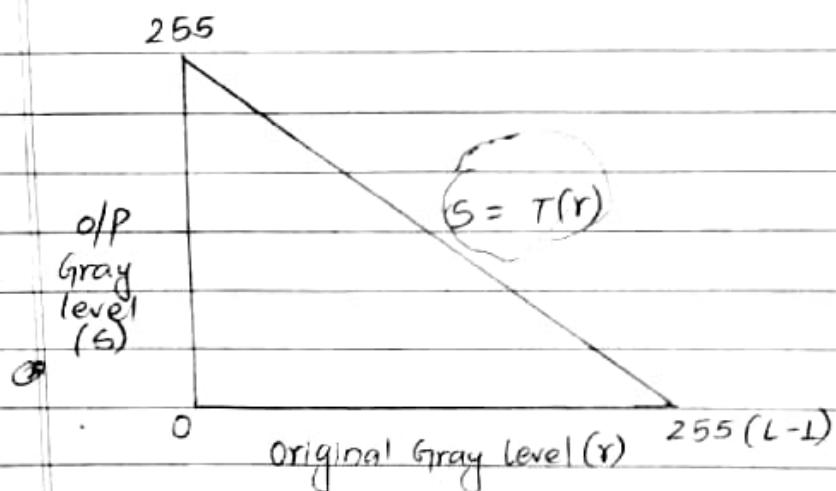
Where,

b = Thresholding value

L = Maximum Gray level

Important notice is that, threshold image has maximum contrast on it having black and white gray level only.

(Imp) ~~#~~ Digital Negative (X-ray): (2011 fall)



(fig: Digital Negative)

It is useful in lots of applications. The common example of digital negative is X-ray image.

As the name suggests, [negative] means simply inverting the gray level i.e. black into white and vice-versa.

So, the negative of digital image

with the gray level in the range from 0 to $L-1$ is obtained by using negative transformation.

i.e

$$S = T(r) = [(L-1) - r]$$

Where,

r = Input pixel value

L = No. of Gray level or Maximum Gray level

$T(r)$ = Transformation Function

S = output Gray level.

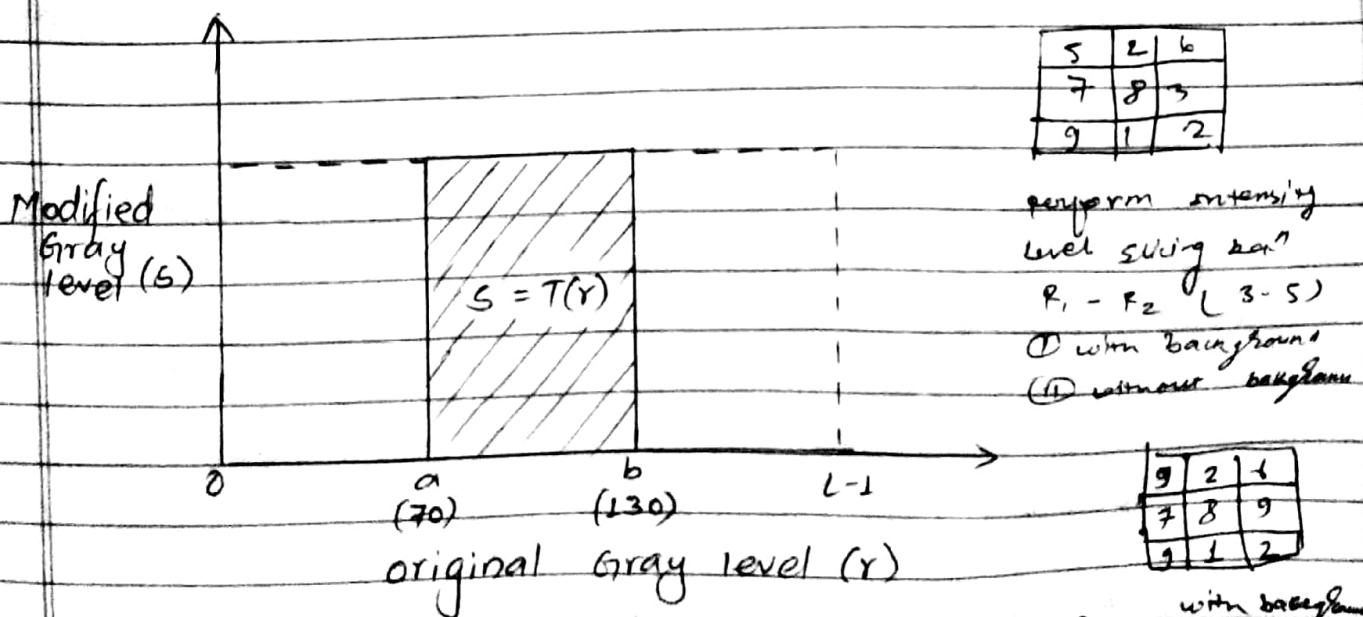
Hence,

$r = 0, S = 255$ (White)

$r = 255, S = 0$ (Black)

(imp)
#

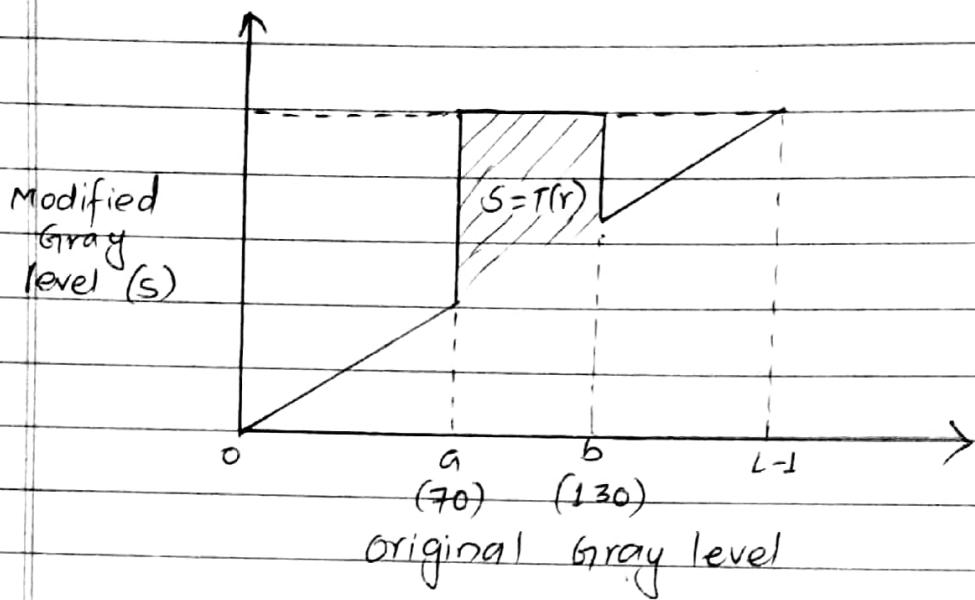
Intensity Level Slicing:
[Gray Level Slicing]



(fig: Slicing without Background)

9	0	0
0	0	9
0	0	0

without background



(fig: Slicing with Background)

Thresholding splits the gray level into two parts. But many times we need highlight a specific range of gray level.

In such a situation, use a transformation function on a gray level called as Gray level slicing.

There are two types of slicing:

(1) Slicing without background:

Here,

We have completely lost the background and this can be implemented by using the following function:

$$S = L-1, \text{ if } a < r \leq b$$

$$S = 0, \text{ otherwise}$$

(2) Slicing With Background:

Here,

in some applications we do not only need of enhancement to the band of gray level but also need to retain the background.

This can be implemented by using the following formation:

$$S = L - 1, \text{ if } a \leq r \leq b \\ S = r, \text{ otherwise}$$



Bit Plane Slicing (clipping):

(2011 fall)

+ highlight
• interesting aspect
→ is any
↑ reference
bit

It is used to enhance features with an image. We must provide a threshold level to determine how the clipping occurs.

↓ The threshold level or value is a reference value where other values get changed.

In general, it is performed for maximum and minimum operators. An expression that contain an image with threshold level.

For example:

If we have image variable and want to slice it to $>$ or $=$ to 125, then the

resulting image stated as:

$$\text{Clipped Image} = (\text{Image (original)}) > 125$$

Here,

it means if the element value is less than 125 then it is set as equal to 125. Otherwise, it is unchanged.

(V. IMP)

Introduction to Histogram:

Histogram of an image provides a global description of the appearance of an image.

The information obtained from histogram is innermost and hence histogram modeling is important in digital image processing (DIP).

By the definition,

the histogram of an image represents the relative frequency of occurrence of various gray level in an image.

Histogram of an image can be plotted as:

- (i) The X-axis has the gray level and Y-axis has the number of pixel in each gray level.

(ii) The x-axis represent the gray level and y-axis represents the probability of occurrence of that gray level.

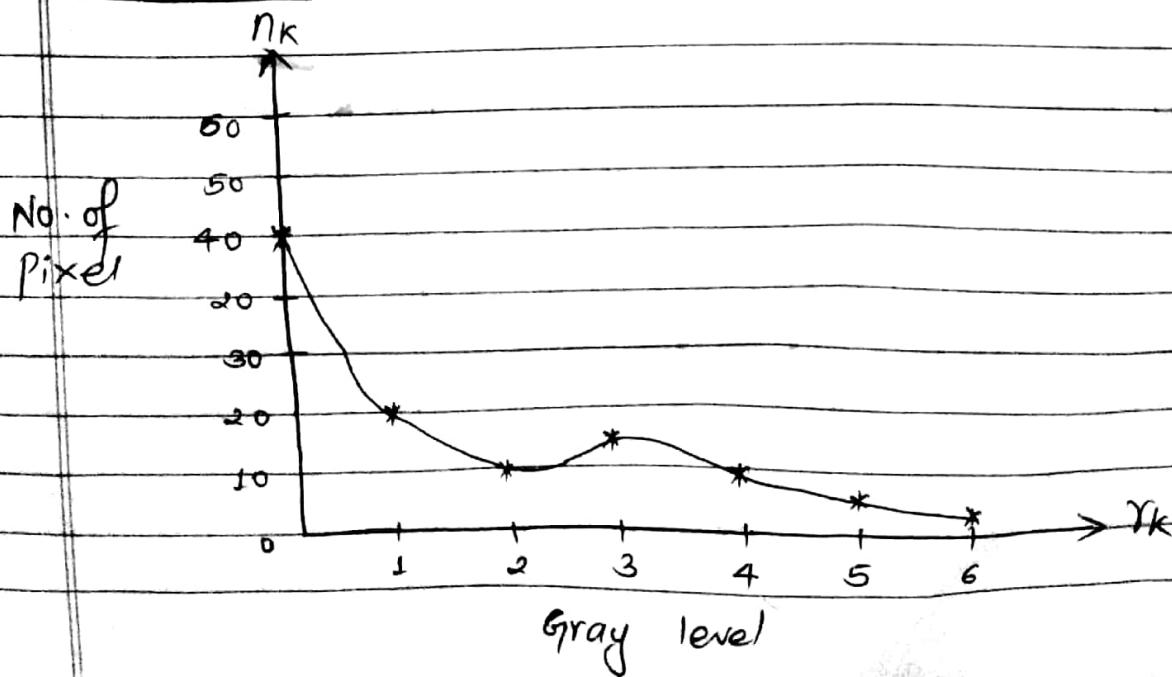
The method of histogram is shown as below:

Step(1): Original Image

Gray level (r_k)	No. of pixel (n_k)
0	40
1	20
2	10
3	15
4	10
5	3
6	2

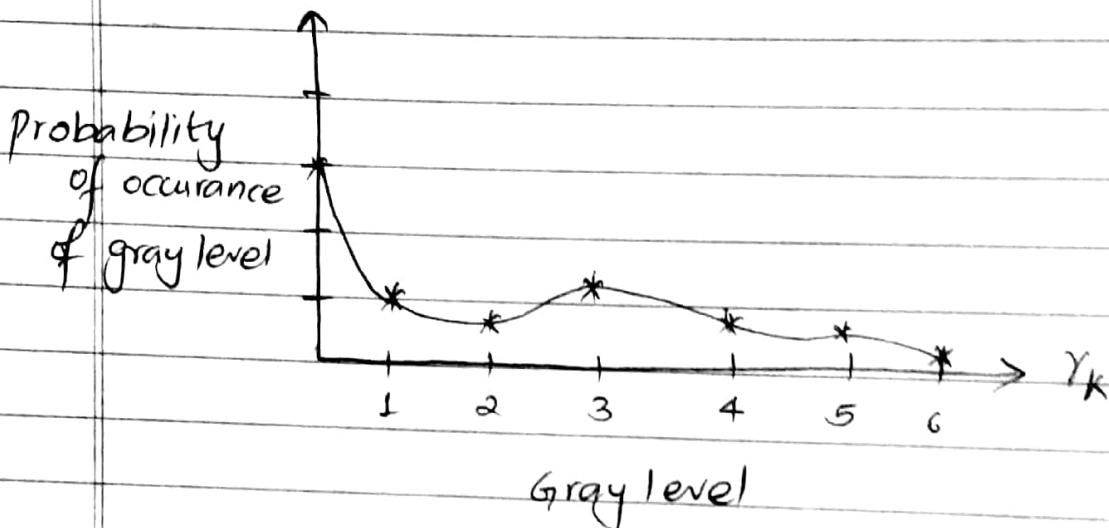
Step(2): Histogram Modeling

Method (1):



Method (2) :

$$P(r_k) = \frac{n_k}{n}$$



(fig: Histogram Representation by 2-methods)

Here,

r_k = k^{th} Gray level

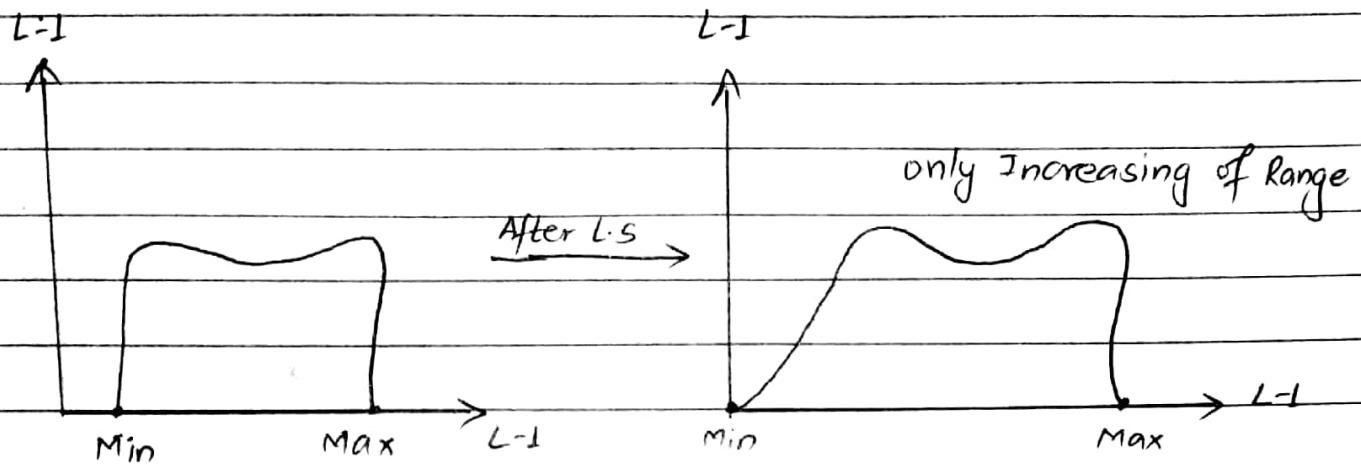
n_k = No. of pixel in k^{th} Gray level

n = Total no. of pixel

Histogram Techniques:

- (1) Histogram stretching / Linear stretching
- (2) Histogram Equalization / Histogram linearization
- (3) Histogram specification / Histogram Matching

① Histogram stretching:



(fig: Linear stretching)

To increase the dynamic range by using a technique is known as Histogram stretching.

In this method, we do not change basic shape of the histogram but spread it over the entire dynamic range.

For this, we use this technique on the basis of slope. It can be given as:

$$\text{slope (m)} = \frac{S_{\max} - S_{\min}}{r_{\max} - r_{\min}} \quad \text{(i)}$$

For transformation of histogram stretching then, transformation can be defined as:

$$S = T(r) = \frac{S_{\max} - S_{\min}}{r_{\max} - r_{\min}} (r - r_{\min}) + S_{\min}, \quad \text{(ii)}$$

$$S = m(r - r_{\min}) + S_{\min}$$

Where,

S_{\max} = Max^m gray level in o/p image

S_{\min} = Min^m gray level in o/p image

r_{\max} = Max^m gray level in i/p image

r_{\min} = Min^m gray level in i/p image

This transformation function stretches or increase the dynamic range of given image.

Example: stretch the contrast of histogram over the entire range.

[2011 fall]

(r)	Gray Level	0	1	2	3	4	5	6	7
	No. of pixel	0	0	50	60	50	20	10	0

Now, perform histogram stretching. show that the new image has a dynamic range of entire range ie [0, 7]

Soln: Given,

$$r_{\min} = 2$$

$$S_{\min} = 0$$

$$r_{\max} = 6$$

$$S_{\max} = 7$$

Now, we perform transformation function on histogram stretching as:

$$S = T(r) = \frac{S_{\max} - S_{\min}}{r_{\max} - r_{\min}} (r - r_{\min}) + S_{\min}$$

(S)

Note:

Now,

$$\text{When } r=2 : S = \frac{7-0}{6-2} (2-2) + 0 = 0$$

$$\therefore [r=2, S=0]$$

$$\text{When } r=3 : S = \frac{7-0}{6-2} (3-2) + 0 = 1.75 \approx 2$$

$$\therefore [r=3, S=2]$$

$$\text{When } r=4 : S = \frac{7-0}{6-2} (4-2) + 0 = 3.5 \approx 4$$

$$\therefore [r=4, S=4]$$

$$\text{When } r=5 : S = \frac{7-0}{6-2} (5-2) + 0 = 5.2 \approx 5$$

$$\therefore [r=5, S=5]$$

$$\text{When } r=6 : S = \frac{7-0}{6-2} (6-2) + 0 = 7$$

$$\therefore [r=6, S=7]$$

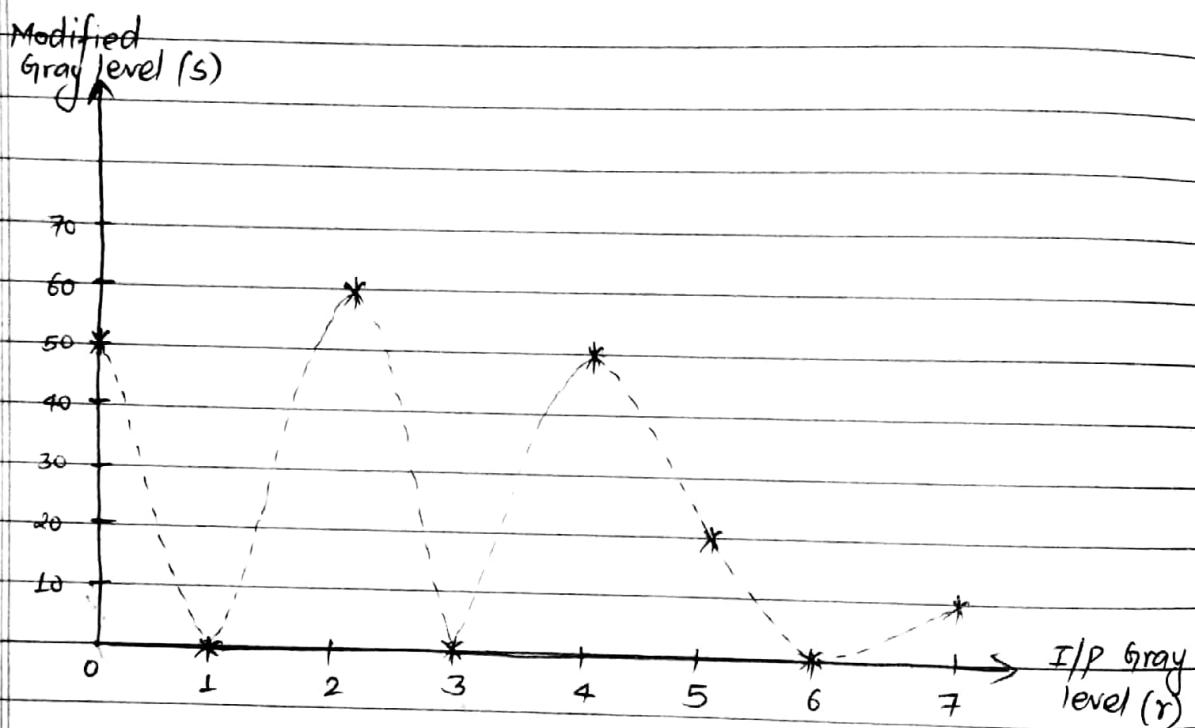
Hence,

Modified histogram or stretched range can be illustrated as:

(S)	Gray Level	0	1	2	3	4	5	6	7
	No. of pixel	50	0	60	0	50	20	0	10

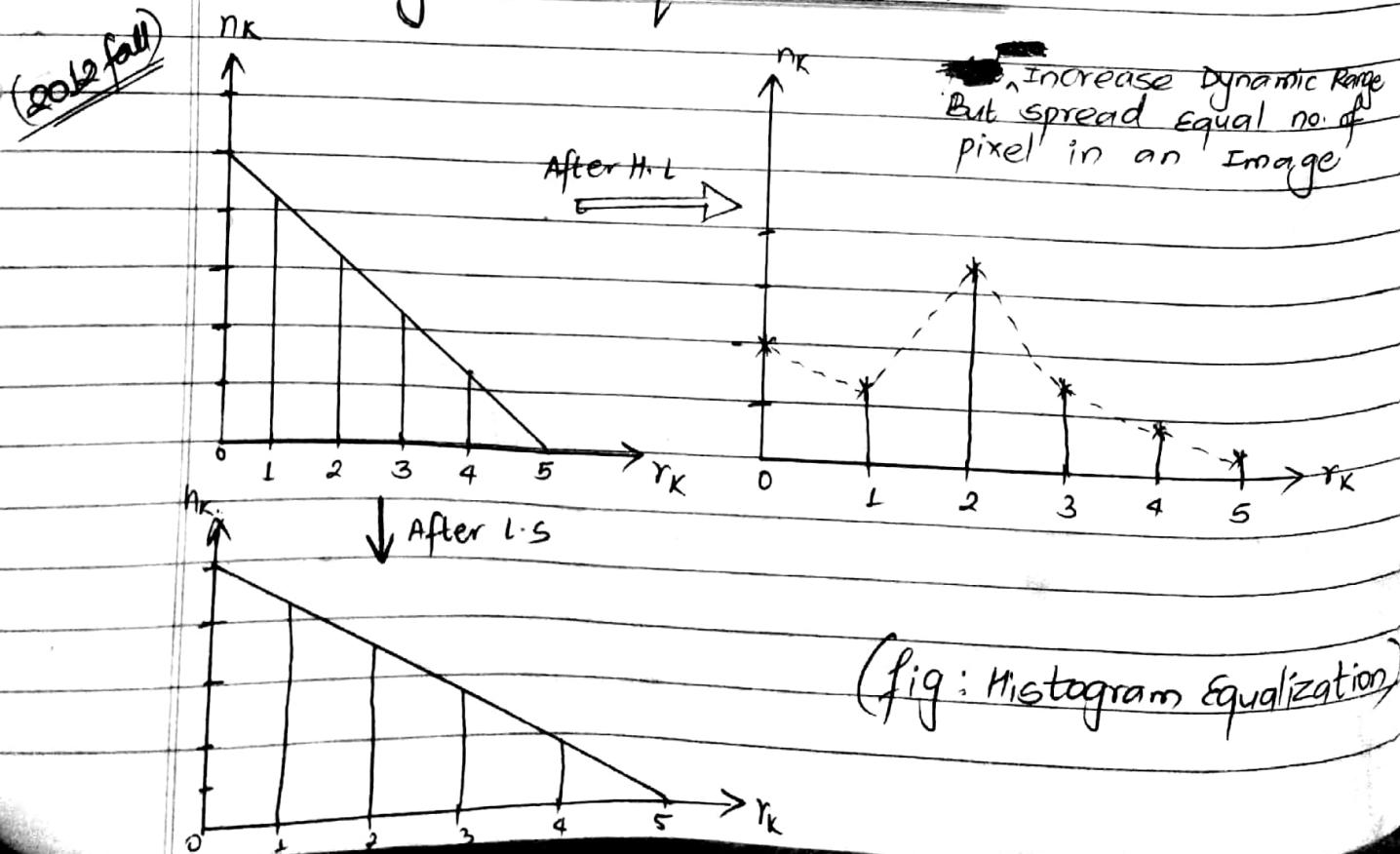
Note: $[S=0] \Rightarrow$ old table $\text{HT}[r=2]$ at value $\overset{\circ}{\text{for}}$
 $\overset{\circ}{\text{at}}$ value New Table $S=0$ HT $\overset{\circ}{\text{at}}$ 20% and so on

Now, The output histogram image is shown as below:



(fig: stretched / Modified / o/p Histogram Image)

~~(Imp)~~ ② Histogram Equalization:



③ ~~x~~ The gray level for continuous variable can be characterized by probability density function (Pdf) ie $P_r(r)$ and $P_s(s)$.

We know that, we need to find the transformation which could be the flat histogram.

① ~~x~~ Linear stretching is a good technique but shape remain same. There are many applications where we need a flat histogram so that this cannot be achieved by histogram stretching.

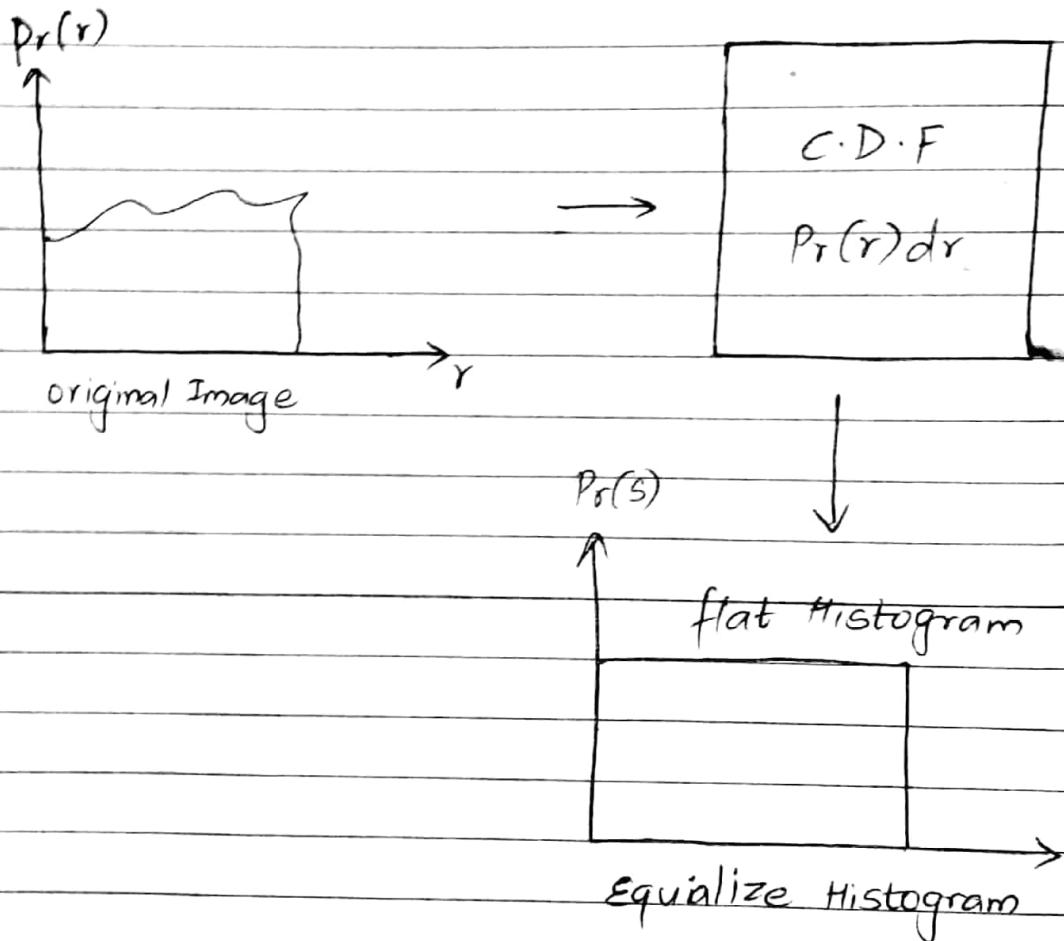
② ~~x~~ Hence, a technique can be used to obtain a ~~a~~ uniform histogram is known as Histogram Equalization or Histogram linearization.

~~x~~ A perfect image is one which has equal number of pixels in all gray level. Therefore, In this technique,
 → image is not only spread over the dynamic range but also to have equal number of pixel in all gray level.

We know that,

$$S = T(r) = \int_0^r P_r(r) dr, \quad 0 \leq r \leq 1$$

Above equation shows the transformation function of cumulative pdf (c.d.f) and it can be shown as:



Example:

(~~N. IMP~~)

Equalize the given histogram:

Gray level	0	1	2	3	4	5	6	7
No. of pixel	790	1023	850	656	329	245	122	81

~~Sol:~~ Here,

Step (1):

Gray level	n_k	P.D.F $Pr(r_k) = n_k/n$	C.D.F $S_k = \sum P_i (r_k)$	$7 * S_k$	Rounding off
0	790	0.19	0.19	1.33	1
1	1023	0.25	0.44	3.08	3
2	850	0.21	0.65	4.55	5
3	656	0.16	0.81	5.67	6
4	329	0.08	0.89	6.23	6
5	245	0.06	0.95	6.65	7
6	122	0.03	0.98	6.86	7
7	81	0.02	1	7	7
$n = 4096$					

Step (2): Representing New Gray Level

old Gray Level	No. of pixel	New Gray Level
0	790	1
1	1023	3
2	850	5
3	656	6
4	329	6
5	245	7
6	122	7
7	81	7

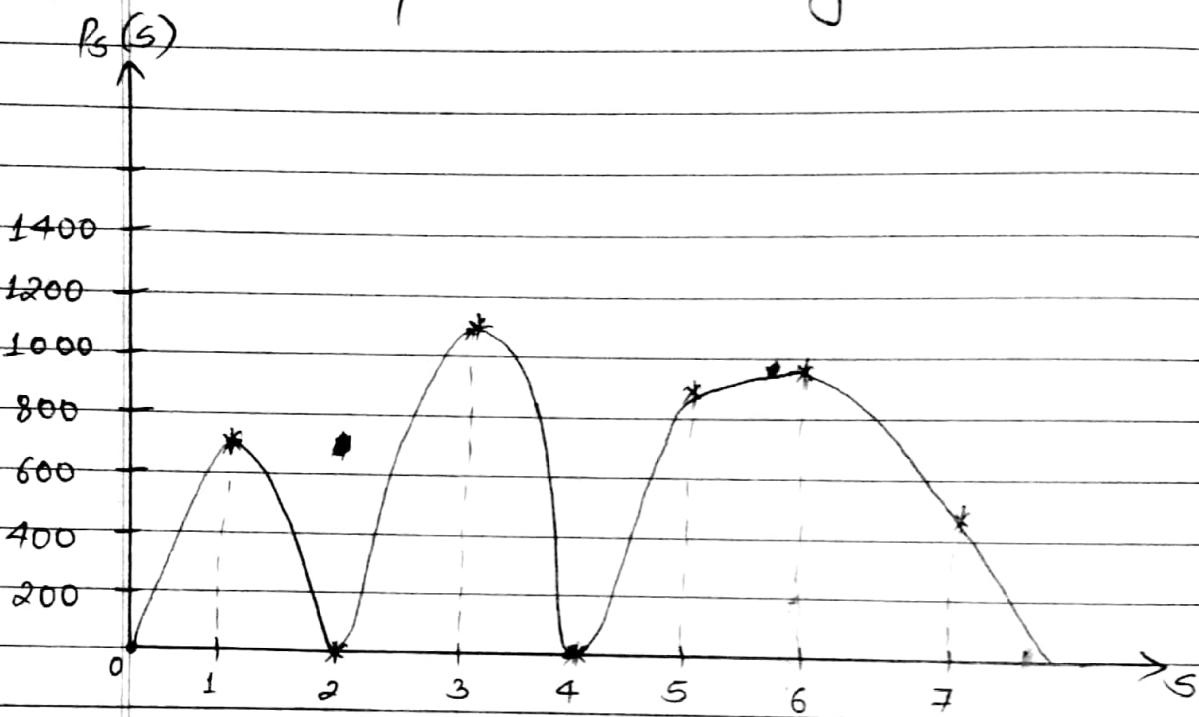
Step (3): Modified Histogram

Gray level	0	1	2	3	4	5	6	7
No. of pixel	0	790	0	1023	0	850	985	448

(656 + 329)

(245 + 122 + 81)

Hence, Equalized Histogram:



(fig: Equalized Histogram)

③ Histogram specification:

~~fh + Nm~~
Histogram specification is not interactive because it always gives one result.

i.e an approximation to an uniform histogram.

It is at time desirable to have an interactive methods in which certain gray levels are highlighted.

let, us suppose that $P_r(r)$ is the original Pdf (Probability density function) and $P_d(z)$ is the ~~original~~ desired pdf.

Suppose that, histogram equalization is first applied on the original image.

i.e

$$S = T(r) = \int_0^r p_r(r) dr$$

Histogram equalization of desired image 'z' is that:

$$V = G(z) = \int_0^z p_z(z) dz$$

Now,

The inverse process $z = G^{-1}(S)$ which have the desired pdf that provide histogram specification.

$$\text{i.e } z = G^{-1}(S) = G^{-1}(T(r)) = G^{-1} \int_0^r p_r(r) dr$$

Example :

Matching the given histogram:

Histogram for image (a) :

Gray Level	0	1	2	3	4	5	6	7
No. of pixel	790	1023	850	656	329	245	122	81

Histogram for image (b) :

Gray level	0	1	2	3	4	5	6	7
No. of pixel	0	0	0	614	819	1230	819	614

~~SOP~~: Here,

Step(1):

Equalized Histogram of Image (a) :

Gray level	0	1	2	3	4	5	6	7
No. of pixel	0	790	0	1023	0	850	985	448

Now,

Equalized Histogram for Image (b) is:

Gray level	n_k	P _r (r _k) = n _k /n	C.D.F S _k = $\sum P_r(r_k)$	S _k * 7	Rounding off
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	614	0.149	0.149	1.05	1
4	819	0.20	0.35	2.45	2
5	1230	0.30	0.65	4.55	5
6	819	0.2	0.85	5.77	6
7	614	0.15	1	7	7
$n = 4096$					

Equalized Histogram of Image (b) is:

Gray level	0	1	2	3	4	5	6	7
No. of pixel	0	614	819	0	0	1230	819	614

Step(2):

To obtain histogram specification, we apply inverse transformation comparing both equalized histogram.

Image (b) \rightarrow Gray level \rightarrow Rounding off no.
 Then, Image (a) \rightarrow same gray level
 Then, No. of pixel of Image (a)

classmate

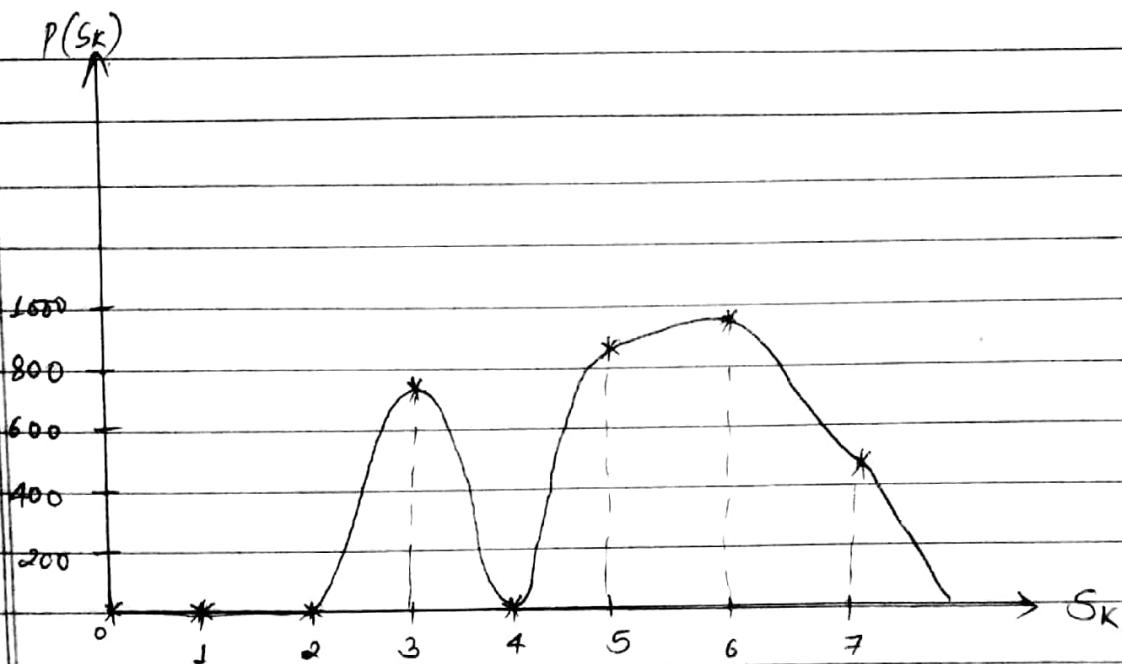
Date _____

Page _____

Gray level	0	1	2	3	4	5	6	7
No. of pixel	0	0	0	790	0	850	985	448

Hence,

The matched or resultant histogram is shown as below :



(fig: Specified Image)

(S.N) (#)

Spatial Operations:

Many image enhancement techniques are based in spatial operations. Most of the spatial operations are performed on the basis of local neighbourhood of the input pixel.

Some of the example of spatial operation for image enhancement are as below :

- (1) Neighbourhood processing
- (2) Spatial Averaging
- (3) Smoothing spatial Filter
- (4) Zooming
- (5) Low Pass, High Pass and Band Pass

(1) Neighbourhood Processing :

L				
-1 0 1				
K	-1	$g(x-1, y+1)$	$g(x, y+1)$	$g(x+1, y+1)$
	0	$g(x-1, y)$	$g(x, y)$	$g(x+1, y)$
	1	$g(x-1, y-1)$	$g(x, y-1)$	$g(x+1, y-1)$

(fig: 3×3 neighbourhood pixel)

We change the value of pixels based on the values of its 8 neighbourhood as shown in figure above.

Instead of 3×3 neighbourhood, we can also use 5×5 , 7×7 etc.

Neighbourhood processing deals to the point operation. The above neighbourhood pixel can be masked into window as shown in the figure below:

L

	w_1	w_2	w_3
w_4	w_5	w_6	
w_7	w_8	w_9	

(fig : 3×3 window)

The above figure is a one type of tablet which is also called as mask or window.

To achieve the neighbourhood processing, we place this 3×3 mask on the image corresponding to each of the component and get some information at the centre.

If g is the original image and f is the modified image then, f is derived as :

$$\begin{aligned}
 f(x, y) = & g(x-1, y+1) * w_1 + g(x, y+1) * w_2 + \\
 & g(x+1, y+1) * w_3 + g(x-1, y) * w_4 + \\
 & g(x, y) * w_5 + g(x+1, y) * w_6 + \\
 & g(x-1, y-1) * w_7 + g(x, y-1) * w_8 + \\
 & g(x+1, y-1) * w_9
 \end{aligned}$$

Hence,

the sum of products of the mask coefficients with corresponding pixel takes place.

(2) Spatial Averaging:

[Averaging Filtering]

~~(2010
spring)~~
In spatial averaging, each pixel is replaced by the weighted average of its neighbourhood pixel i.e

$$v(m, n) = \sum_{(k, l) \in W} a(k, l) y(m-k, n-l)$$

where,

$y(m-k, n-l)$ = i/p image

$v(m, n)$ = o/p image

W = chosen window

$a(k, l)$ = Filter Weight

A common class of spatial average filter has all equal weight i.e

$$v(m, n) = \frac{1}{N_w} \sum_{(k, l) \in W} y(m-k, n-l)$$

where,

$a(k, l) = \frac{1}{N_w}$; N_w is the no. of pixel in window.

Now,

Each pixel is replaced by its

average with the average of its nearest pixel value.

Spatial average is used for noise smoothing, low pass filtering and subsampling of the image.

Example of spatial averaging can be represented as :

$$\begin{array}{c}
 \begin{array}{ccc}
 & \overbrace{-1 \quad 0 \quad 1}^L \\
 \begin{array}{c} \curvearrowleft \\ K \end{array} & \left\{ \begin{array}{cc} -1 & 1/9 \\ 0 & 4/9 \\ 1 & 1/9 \end{array} \right. & \left. \begin{array}{cc} 1/9 & 1/9 \\ 4/9 & 1/9 \\ 1/9 & 1/9 \end{array} \right\} \begin{array}{c} \curvearrowright \\ k=2 \end{array} & \overbrace{1/4 \quad 1/4}^L \\
 \end{array} \\
 \end{array}$$

(fig: 2×2 window)

(fig: 3×3 window)

(3) Smoothing Spatial Filter : [Median & Mean Filter]

Median Filter:

The median filter is also a spatial filter but it replaces the central value in the window with median of all the pixel values in the window.

The kernel or template or mask is usually square but can be initialized in different window.

An example of median filtering is

i.e

unfiltered values in order like

0, 1, 2, 3, 4, 6, 10, 15, 19, 97

Now,

$$\text{O/P value} = \left(\frac{n+1}{2} \right)^{\text{th}}$$

or
Median value

$$= \left(\frac{9+1}{2} \right)^{\text{th}}$$

$$= 5^{\text{th}}$$

$$= 6$$

Hence,

6	2	0	Median filtering	6	2	0
3	97	4		3	6	4
19	15	10		19	15	10

Before filtering

After filtering

The center value 97 is replaced by the median value of all nine values i.e 6.

Formula for median filtering is :

$$V(m, n) = \text{Median} \{ y(m-k, n-l) | (k, l) \in W \}$$

Mean Filter:

Mean filter is a spatial filter that replaces the center value in the window

with the average of all the pixel values in the window.

Here, the window can be of any shape but can be initiated as a square mask. Therefore, average of smoothing filter shows the properties of lowpass filter. so, it reduces the noise from an image.

An example of mean filter is:
Unfiltered values in order like:

5, 3, 6, 2, 1, 9, 8, 4, 7

Now,

$$\text{O/P value} = \frac{\sum x}{\text{Mean value}} = 5$$

Hence,

5	3	6		5	3	6
2	(1)	9	Mean filtering	2	(5)	9
8	4	7		8	4	7

Before filtering

After filtering

The center value '1' is replaced by the mean value '5'

~~(V. IMP)~~

(4) Zooming :

(2012)

[Magnification & Interpolation]

Another important application is image enhancement where spatial domain neighbourhood operation is used.

Such type of image enhancement technique is known as zooming.

Actually, it is the process of making an image larger. It involves two steps:

- (1) Creating new pixel location.
- (2) Assigning gray level to those new location.

Different types of zooming technique can be used to zoom. They are:

- (i) Replication
- (ii) linear Interpolation

(i) Replication:

In replication, we simply replicate each pixel and then replicate row.

Let, us consider an image as:

1	2	3	4
5	6	7	8
9	8	6	7
0	1	2	3

As we start from the first row, first we replicate each pixel and then replicate each row.

Now,

The first row looks as:

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

We will now replicate this row then it looks as:

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

Now,

Performing replication operation on an entire image then we obtain output image as:

1	1	2	2	3	3	4	4
1	1	2	2	3	3	4	4
5	5	6	6	7	7	8	8
5	5	6	6	7	7	8	8
9	9	8	8	6	6	7	7
9	9	8	8	6	6	7	7
0	0	1	1	2	2	3	3
0	0	1	1	2	2	3	3

Here,

4x4 image is zoomed to 8x8 image.

This method can be repeated to get larger image but no new data is added.

Zooming By zero-Interlace Technique:

Step(1): Adding zero ~~to~~ every after pixel of the first row then we obtain as:

$$\begin{bmatrix} 1 & 0 & 2 & 0 & 3 & 0 & 4 & 0 \end{bmatrix}$$

Step(2): Adding zero at every column.
Now,

Insert row of zero and then this is known as zero-Interlacing.

$$\begin{bmatrix} 1 & 0 & 2 & 0 & 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Step(3): Now,

Performing zero-Interlace technique on an entire image then we obtain output image as:

$$\begin{bmatrix} 1 & 0 & 2 & 0 & 3 & 0 & 4 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 0 & 6 & 0 & 7 & 0 & 8 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 9 & 0 & 8 & 0 & 6 & 0 & 7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 2 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

(fig: 8×8 final Pseudo Image)

Hence, this image is known as zero-Interlace Image.

(ii) Linear Interpolation:

In this method, instead of replicating each pixel, average of the two adjacent pixel along the row is taken and placed between the two pixels.

Accordingly, same operation is then performed along the column.

Linear Interpolation is also derived from the zero-interlace technique.

Step(1): zero-Interlace

1	0	2	0	3	0	4	0
0	0	0	0	0	0	0	0
5	0	6	0	7	0	8	0
0	0	0	0	0	0	0	0
9	0	8	0	6	0	7	0
0	0	0	0	0	0	0	0
0	0	1	0	2	0	3	0
0	0	0	0	0	0	0	0

(Step 2): Interpolate row

Step(3): Interpolate column

1	1.5	2	2.5	3	3.5	4	2
3	3.5	4	4.5	5	5.5	6	3
5	5.3	6	6.5	7	7.5	8	4
7	7	7	6.25	6.5	7	7.5	3.25
9	8.5	8	7	6	6.5	7	3.5
4.5	4.5	4.5	4.5	4	4.5	5	2.5
0	0.5	1	1.5	2	2.5	3	1.5
0	0.25	0.5	0.75	1	1.25	1.5	0.7

(5) Filtering:

(i) Low Pass Filtering:

It is employed to remove ~~high frequency~~ (unnecessary noise) from a digital image.

This type of filtering is usually used to ~~attenuate~~ the image noise.

Here, an image is smoothed by decreasing the disparity between the pixel value by average value.

Example of mask or kernel or window of low pass filter can be shown as below:

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

(fig: 3x3 mask)

$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$
$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$
$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$
$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{1}{16}$

(fig: 4×4 mask)

(ii) High Pass Filtering: (Sharpening filters)

A high pass filtering is the basis for most sharpening methods. Sharpening is the opposite of smoothing.

The sharpening can make it possible to highlight the borders between homogeneous region.

This is used to return high frequency information.

The kernel of the high pass filter is designed to increase the brightness of the center pixel.

Example of mask or kernel or window of high pass filter is as shown below:

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	-1	-1	-1
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	-1	8	-1
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$	-1	-1	-1

(fig: 3×3 mask high pass filter)

(iii) Band Pass Filtering :

This filter is used to remove selected frequency region between low and high frequency.

It is also used for image restoration. It passes a limited range of frequency.

Calculating spatial averaging for band pass filtering is:

$$\vartheta(m, n; \theta) = \frac{1}{N_\theta} \sum_{(k, l) \in W_\theta} y(m-k, n-l)$$

Where,

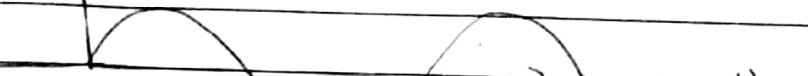
W_θ = Neighbourhood mask along direction θ

N_θ = No. of pixel within this neighbourhood

fourier series

* A series of cosine and sine term that represent the periodic signal is called fourier series

$f(n)$ → magnitude



$f(n)$ → periodic signal

magnitude

sine wave

cosine wave

time

Chapter - 3

Image Enhancement In Frequency Domain

[6 hours]

Fourier Transform :

Fourier Transform is a mathematical tool to analyze and design linear system. It is used to reduce the number of calculation to a fraction.

It also helps to quantify the effects of digitizing system.

The application of Fourier transform is:

- Image Enhancement
- X-Ray
- Enhancement of TV transmission
- and video signal.

(A) 1-D Fourier Transform:

Let, $f(x)$ be a continuous real variable of x .

Then,

$$\text{Fourier Transform} = F[f(x)] = F(u)$$

$$\therefore F(u) = \int_{-\infty}^{\infty} f(x) \cdot e^{-j2\pi ux} dx$$



And,

Inverse Fourier Transform is :

$$F^{-1}[F(u)] = f(x)$$

$$\therefore f(x) = \int_{-\infty}^{\infty} F(u) \cdot e^{j2\pi ux} dx$$

(B) 2-D Fourier Transform:

A 2-D function $f(x)$ has 2-D transform
and,

$$F[f(x,y)] = F(u,v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x,y) \cdot e^{-j2\pi(ux+vy)} dy dx$$

Now,

Inverse of fourier transform is :

$$F^{-1}[F(u,v)] = f(x,y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u,v) \cdot e^{j2\pi(ux+vy)} dy dx$$

Properties of Fourier Transform:

(1) Addition Theorem:

$$F[f(x,y) + g(x,y)] = F(u,v) + G(u,v)$$

(2) Shift Theorem:

$$F[f(x-a, y-b)] = F(u,v) e^{-j2\pi (u_a, v_b)}$$

(3) Similarity Theorem:

$$F[f(ax, by)] = \frac{1}{|ab|} F(u/a, v/b)$$

(4) Convolution Theorem:

$$F[f(x,y) * g(x,y)] = F(u,v) * G(u,v)$$

(5) Rayleigh's Theorem:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |f(x,y)|^2 dx dy = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u,v)|^2 du dv$$

(6) Evenness & oddness:

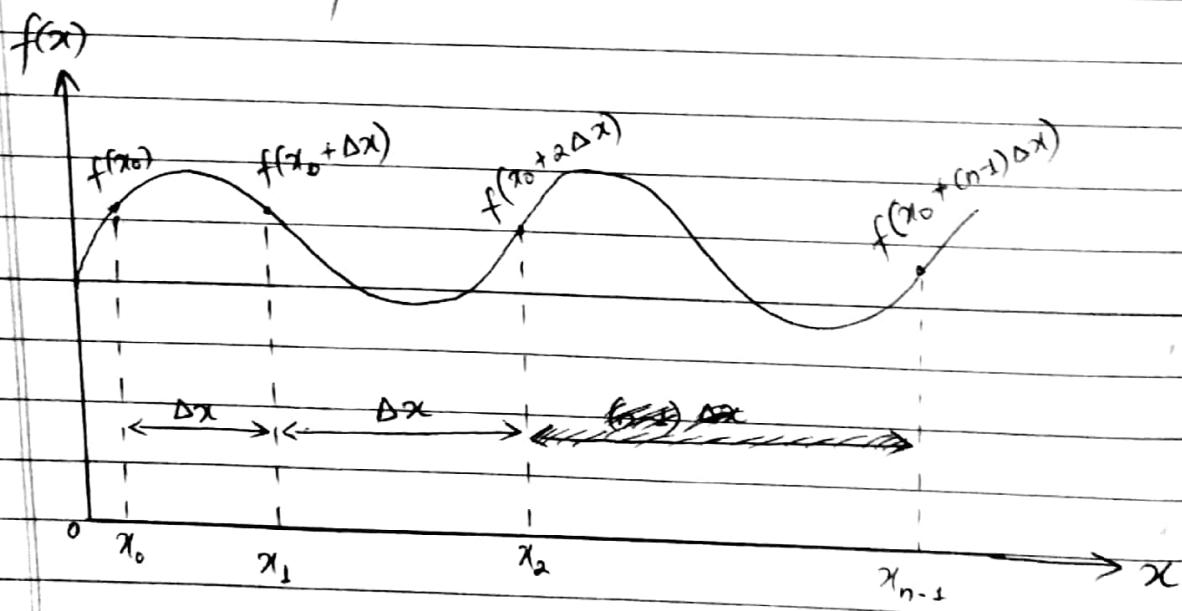
$$f_e(x) \Rightarrow f(x) = f(-x) \text{ and}$$

$$f_o(x) \Rightarrow f(-x) = -f(x)$$

(7) Power Theorem:

$$\int \int_{-\infty}^{\infty} f(x, y) * g^*(x, y) dx dy = \int \int_{-\infty}^{\infty} F(u, v) * G^*(u, v) du dv.$$

Discrete Fourier Transform (DFT) and Its Importance:



(fig: DFT)

Here,

$f(x)$ is a continuous function to be discretized by taking 'n' sample i.e

$$\{f(x_0), f(x_0 + \Delta x), f(x_0 + 2\Delta x), \dots, f(x_0 + (n-1)\Delta x)\}$$

It is defined as :

$$F(u) = \frac{1}{n} \sum_{x=0}^{n-1} f(x) \cdot e^{-j2\pi \frac{ux}{n}}$$

Inverse Discrete Fourier Transform is defined as :

$$f(x) = \sum_{v=0}^{n-1} F(v) \cdot e^{j2\pi \frac{vx}{n}}$$

These two given equation are discrete fourier transform and it is used to calculate the following :

(1) Fourier spectrum:

$$|F(v)| = \sqrt{R^2(v) + I^2(v)}$$

(2) Power spectrum:

$$|P(v)| = |F(v)|^2 = R^2(v) + I^2(v).$$

(3) Phase Angle (spectrum):

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

Where,

'v' is the transform frequency variable
'x' is capital or Image Variable.

Discrete Fourier Transform is most widely used orthogonal transform in the field of image processing and it can

diagonilized circulant matrix.

However, in this transform, it takes n^2 multiplication and $n(n-1)$ addition to calculate 1-D discrete fourier transform of 'n' data points.

Discrete Cosine Transform And Its properties : (DCT)

~~(2010
spring)~~

Discrete cosine transform is a Fourier related transform similar to DFT but it is only concerned with real number only.

It can be defined as :

$$\text{DCT} \Rightarrow C(u) = \alpha(u) \sum_{x=0}^{N-1} f(x) \cdot \cos \left[\frac{(2x+1)u\pi}{2N} \right]$$

Where,

$$u = 0, 1, \dots, N-1$$

$$\alpha(u) = \sqrt{\frac{1}{N}} = \frac{1}{\sqrt{N}} \text{ for } u = 0$$

$$= \sqrt{\frac{2}{N}} \text{ for } u = 1, \dots, N-1$$

\therefore Inverse DCT is defined as:

$$\text{IDCT} \Rightarrow f(x) = \sum_{v=0}^{N-1} \alpha(v) \cdot \cos \left[\frac{(2x+1)v\pi}{2N} \right]$$

Where,

$$v = 0, 1, \dots, N-1$$

For 2-D:

$$\text{DCT} \Rightarrow c(u, v) = \alpha(u, v) + \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cdot \cos \left[\frac{(2x+1)u\pi}{2N} \right] \cos \left[\frac{(2y+1)v\pi}{2N} \right]$$

Where,

$$\alpha(u, v) = \frac{1}{\sqrt{N}} \quad \text{for } u, v = 0$$

$$= \sqrt{\frac{2}{N}} \quad \text{for } u, v = 1, \dots, N-1$$

$$\text{IDCT} \Rightarrow f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u, v) \cos \left[\frac{(2x+1)u\pi}{2N} \right] \cos \left[\frac{(2y+1)v\pi}{2N} \right]$$

where,

$$x, y = 0, \dots, N-1$$

Orthogonal: $A^T A = I$
symmetric: $A^T = A$

classmate

Date _____
Page _____

✓ Properties of DCT:

- (1) The cosine Transform is real and orthogonal.

i.e

$$C = C^* \Rightarrow C^{-1} = C^T$$

Where,

C^* = conjugate of C

C^T = Transform of C

- (2) The Cosine Transform is not real part of Unitary DFT.

i.e

$$T^{-1} = T^* T$$

- (3) It is a fast transform. The cosine transform of ~~N~~ N elements can be calculated in $O(N \log N)$ operation through N point FFT (fast fourier Transform).

- (4) It has excellent energy compaction for highly co-related data.

✓ Uses of DCT:

DCT is used in image processing and signal processing especially for loose data compression. ~~sense~~

Since it has strong energy compaction features.

It is also used for JPEG image compression, MPEG ~~video~~ compression and MJPEG video compression.



Hadamard Transform:

(2015 fall)
(2012 Fall)

Hadamard Transform is a fast transform which is real, symmetric and orthogonal.

1-D Hadamard Transform can be implemented in (Big oh) $O(N \log N)$ addition and subtraction. It has only binary values i.e 1 or -1 in its kernel matrix.

It is defined as :

$$H(U) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x) \cdot (-1)^{\sum_{i=0}^{n-1} b_i(x) \cdot b_i(U)}$$

Kernel

For $N=2$:

$$N = 2^n \Rightarrow 2 = 2^1 \Rightarrow n=1$$

Where, $U = 0, \dots, N-1$

Now,

Hadamard Matrix can be defined as :

$$H_n = \frac{1}{\sqrt{N}} \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix}$$

where,

$$\frac{1}{\sqrt{N}} = \text{sample value}$$

$$\begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -(H_{n-1}) \end{bmatrix} = \text{Indicate kernel}$$

$$H_0 = I$$

Hadamard Matrix at 1: (H_1) i.e $n=1$

$$H_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} H_{1-1} & H_{1-1} \\ H_{1-1} & -(H_{1-1}) \end{bmatrix}$$

$$= \frac{1}{\sqrt{2}} \begin{bmatrix} H_0 & H_0 \\ H_0 & -H_0 \end{bmatrix}$$

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

Also,

For $N=4$, $N=2^n \Rightarrow 4=2^n \Rightarrow n=2$

$$\therefore H_2 = \frac{1}{\sqrt{4}} \begin{bmatrix} H_{2-1} & H_{2-1} \\ H_{2-1} & -(H_{2-1}) \end{bmatrix}$$

$$\Rightarrow \frac{1}{2} \begin{bmatrix} (H_1) & H_1 \\ H_1 & -H_1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

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

Properties of Hadamard Transform:

- (1) It is real, symmetric and orthogonal
i.e.

$$H = H^* \Rightarrow H^{-1} = H^T$$

- (2) It is a fast transform and 1-D Hadamard Transform can be implemented in $O(N \log_2 N)$ addition and subtraction.

- (3) It has only binary values i.e 1 or -1 in its kernel matrix. No multiplications are required in the transform.

- (4) It is used for digital image processing and digital signal processing.

- (5) It has good energy compaction for highly co-related image.

- (6) It is also used in digital hardware implementation of image processing algorithm.

~~(S.N.)~~ ~~#~~ Haar Transform:

2010 Spring
S.N. 2012 Fall

Haar Transform is the orthogonal transform which is derived from the Haar Matrix.

It is also a fast transform and can be implemented in $O(N \log N)$ operation.

The main advantage of this transform is that it samples the input data sequence into fine resolution and takes the difference between the adjacent pairs.

The resolution increases by a power of 2 and as a result, in the transform domain, differential energy is more and more localized.

It only exist for $N=2^n$, where n is an integer.

Haar Transform is defined as:

$$H_T = H_k(x)$$

Where, $k = 0, \dots, N-1$

k is given as:

$$k = 2^p + q - 1$$

And,

$$0 \leq p \leq (n-1)$$

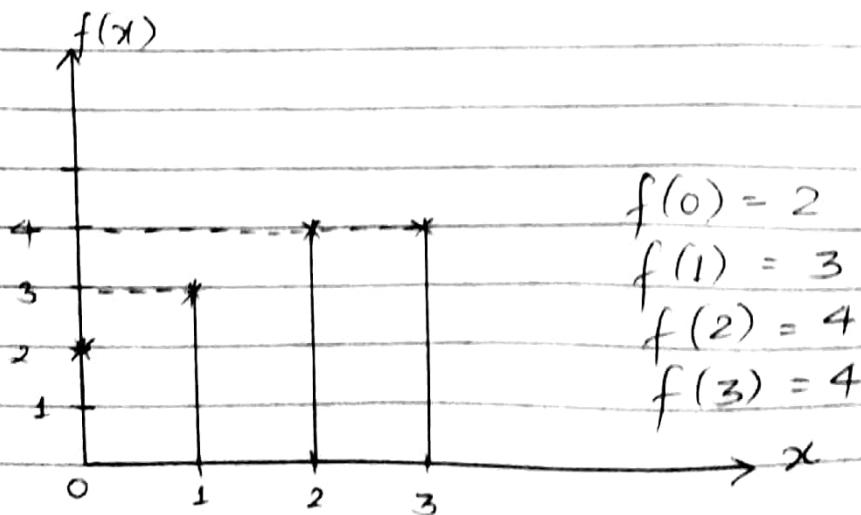
$$1 \leq q \leq 2^p, \text{ for } p \neq 0$$

$$q = 0, 1, \text{ for } p = 0$$

Properties of Haar Transform:

- (1) It is symmetric, separable unitary transform that uses haar function for its basis.
- (2) It is orthogonal and real.
ie $T^{-1} = T^T \Rightarrow H_r = H_r^*$
- (3) It is a fast transform and can be implemented in $O(N)$ operation. Where, N is a number of samples.
- (4) It exist for $N = 2^n$, where 'n' is an integer.
- (5) It has poor energy compaction property.

Q Find the Fourier spectrum of the function at point 0 and 1 as shown in figure below:



~~Sol:~~ Here,

Given that:

$$f(0) = 2, f(1) = 3, f(2) = 4, f(3) = 4$$

We know that:

$$\text{DFT} \Rightarrow F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cdot e^{-j \frac{2\pi u x}{N}}$$

Where,

$$N = 4 (0, 1, 2, 3)$$

Now,

For $u=0$:

$$F(0) = \frac{1}{4} \sum_{x=0}^3 f(x) \cdot e^{-j \frac{2\pi 0 \cdot x}{4}}$$

$$= \frac{1}{4} \sum_{x=0}^3 f(x)$$

$$= \frac{1}{4} [f(0) + f(1) + f(2) + f(3)]$$

$$= \frac{1}{4} [2 + 3 + 4 + 4]$$

$$= 13/4$$

∴ Fourier spectrum at point 0 is:

$$|F(0)| = \sqrt{R^2(0) + I^2(0)}$$

$$= \sqrt{\left(\frac{13}{4}\right)^2 + 0}$$

$$= \frac{13}{4} \quad \text{Ans}$$

Note:

Power spectrum at point 0 is:

$$|P(0)| = |F(0)|^2$$

$$= R^2(0) + I^2(0)$$

$$= \left(\frac{13}{4}\right)^2$$

$$= \frac{169}{16}$$

Now,

For $U=1$:

$$F(1) = \frac{1}{4} \sum_{x=0}^3 f(x) \cdot e^{-j\omega \frac{1}{4}x}$$

$$= \frac{1}{4} \sum_{x=0}^3 f(x) \cdot e^{-j\frac{\pi}{2}x}$$

$$= \frac{1}{4} \left[f(0) \cdot e^{-j\frac{\pi \cdot 0}{2}} + f(1) \cdot e^{-j\frac{\pi \cdot 1}{2}} + f(2) \cdot e^{-j\frac{\pi \cdot 2}{2}} + f(3) \cdot e^{-j\frac{\pi \cdot 3}{2}} \right]$$

$$= \frac{1}{4} \left[f(0) \cdot e^0 + f(1) \cdot e^{-j\frac{\pi}{2}} + f(2) \cdot e^{-j\pi} + f(3) \cdot e^{-j\frac{3\pi}{2}} \right]$$

$$= \frac{1}{4} \left[2 \times 1 + 3 \times \left(\cos \frac{\pi}{2} - j \sin \frac{\pi}{2} \right) + 4 \times \left(\cos \pi - j \sin \pi \right) + 4 \times \left(\cos \frac{3\pi}{2} - j \sin \frac{3\pi}{2} \right) \right]$$

$$= \frac{1}{4} \left[2 + 3 \times (0 - j \cdot 1) + 4 \times ((-1) - j \cdot 0) + (0 - j \cdot (-1)) \right]$$

$$= \frac{1}{4} [2 - 3j - 4 + 4j]$$

$$= \frac{1}{4} [j - 2]$$

$$= \frac{1}{4} j - \frac{1}{2}$$

$$\therefore F(1) = \frac{1}{4} j - \frac{1}{2}$$

Now,

Fourier spectrum at point 1 is:

$$|F(1)| = \sqrt{R^2(1) + I^2(1)}$$

$$= \sqrt{\left(\frac{-1}{2}\right)^2 + \left(\frac{1}{4}\right)^2}$$

$$= \sqrt{\frac{1}{4} + \frac{1}{16}}$$

$$= \sqrt{\frac{5}{16}}$$

$$= \frac{\sqrt{5}}{4}$$

$$\therefore |F(1)| = \underline{\underline{\sqrt{5}/4}} \quad \text{Ans}$$

(#) FFT (Fast Fourier Transform)

(10)

classmate

Date _____

Page _____

Chapter- 4

Image Restoration

[4 hours]

Introduction:

The objective of image restoration is to improve a given image in some pre-defined sense.

✓ Image Enhancement is a subjective process whereas Image Restoration is an objective process.

✓ Image Restoration is to re-construct or recover an image that has been degraded by using a prior knowledge of the degradation phenomena.

So, this technique is a process of degradation and applying the inverse process in order to recover the original image.

✓ If there is a presence of noise and image blurring we use the concept of image restoration.

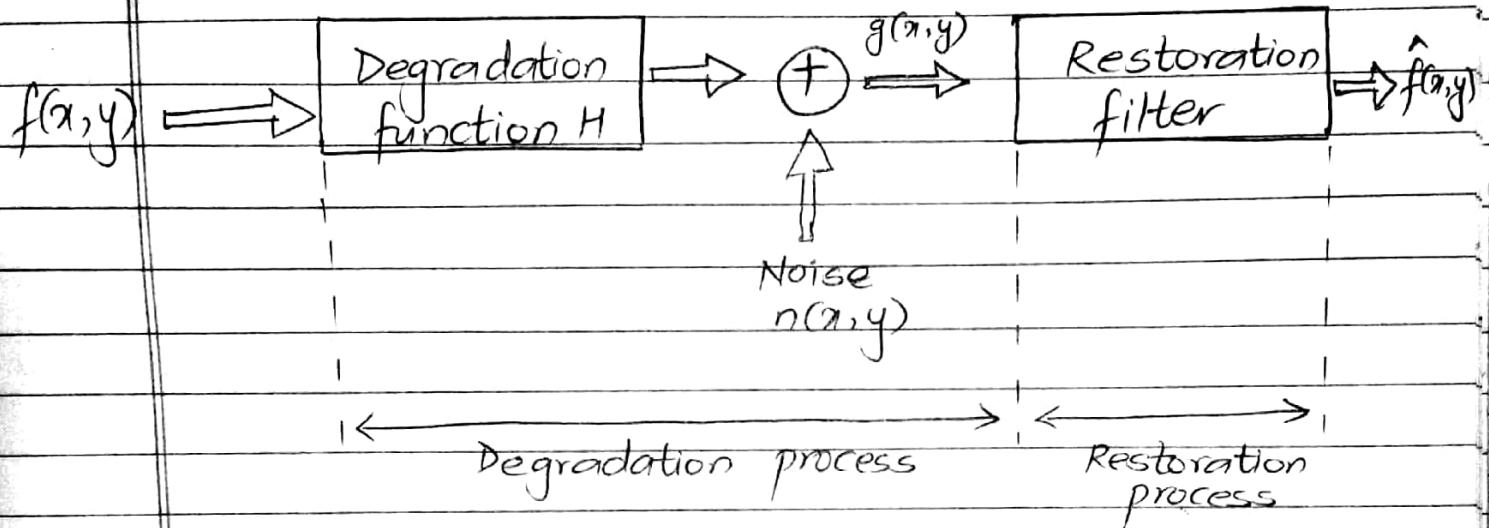
$$\text{Degraded Image} = \text{original image} + \text{Noise}$$

classmate

Date _____

Page _____

A model of the Image Degradation/ Restoration Process :



(fig: A model of Image Degradation/ Restoration process)

The Degraded image $g(x,y)$ can be produced as:

$$g(x,y) = H[f(x,y)] + n(x,y) \quad \dots \dots \dots \text{(i)}$$

Where,

$f(x,y)$ = Input Image

$n(x,y)$ = Additive Noise term

H = Degradation function

Now,

The degraded image can be defined in spatial Domain as:

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

Date _____
Page _____

In function analysis convolution is a ~~methe maticeous operation~~ ~~3rd function~~ ~~classmate~~ operation of two function ($f \& g$) it produce the 3rd function

Where,

$h(x, y)$ = spatial Representation of the degradation function.

Then,

the degraded image in spatial domain is equivalent to the convoluted frequency domain represented as:

$$G(u, v) = H(u, v) * F(u, v) + N(u, v) \dots \text{---(iii)}$$

\therefore The degradation process is also referred to as convolving the image with a point spread function (PSF).

Similarly,

Restoration process is also referred to as De-convolution.

Hence,

Restoration Process can be obtained as,

$$\hat{f}(x, y) = R[g(x, y)] \text{ in Linear spatial Invariant form}$$

$$\hat{f}(u, v) = R[G(u, v)] \text{ in Deconvolution form}$$

(2015 fall)

in image (Spatial Domain)

classmate

Date _____

Page _____



Noise Model Restoration

In the presence of noise only spatial Filtering:

The method of choice for reducing the noise is spatial filtering. After filtering the noise, there should have a restoration process to have or obtain original image.

For noise reduction, following summarize and implement two types of spatial filters:

- (1) Spatial Noise Filters
- (2) Adaptive spatial filters

(1) Spatial Noise Filters:

In spatial noise filters, image ^{is} independently off how image characteristics vary from one location to another location.

In this condition, the spatial noise filters can be used and explained as:

- (i) Arithmetic Mean,

$$A = \left[\frac{1}{n} + \sum_{i=1}^n m_i \right]$$

↑
number of individual values
↓
number of terms

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

(ii) Geometric Mean,

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

(iii) Harmonic Mean,

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

(iv) Median,

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

(v) Maximum,

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

(vi) Minimum,

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

Where,

$m \& n$ = Variables denotes image rows & columns spanned by filter respectively

s_{xy} = Denotes an $m \times n$ subImage of the input noisy image g .

$\hat{f}(x,y)$ = filter response at (x,y) co-ordinate

(x,y) = co-ordinates of sub-image.

(2) Adaptive Spatial Filter:

In some application, results can be improved by using filters (capable of adapting behaviour) based on the characteristics of the image in the region being filtered.

The adaptive median filtering algorithm uses two processing levels for adaptive spatial filter that can be denoted as:

$$\text{Level A : } \begin{cases} A_1 = Z_{\text{med}} - Z_{\text{min}} \\ A_2 = Z_{\text{med}} - Z_{\text{max}} \end{cases}$$

- Algorithm
- If $A_1 > 0$ AND $A_2 < 0$, go to level B
 - * If $Z_{\text{min}} < Z_{\text{med}} < Z_{\text{max}}$ go to level B
 - * Else Increase the window size
 - If window size $\leq S_{\text{max}}$, repeat level A
 - Else output Z_{med}

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

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

classmate

Date _____

Page _____

algorithm

Level B: (If $B_1 > 0$ AND $B_2 < 0$, output Z_{xy})

If $Z_{min} < Z_{xy} < Z_{max}$, output Z_{xy}
Else ~~else~~ output Z_{med}

Where,

Z_{min} = Min^m intensity value in say \rightarrow gray level

Z_{max} = Max^m intensity value in say

Z_{med} = Median of the intensity value in say

Z_{xy} = Intensity value at co-ordinates (x, y)

Say = Max^m allowed size of the adaptive filter window

(Frequency Domain)

Periodic Noise Reduction Using Frequency Domain Filtering:

Periodic Noise produces impulse like burst that often are visible in the Fourier Spectrum.

The principle approach for filtering these components is to use Notch reject filtering.

The general expression for Notch Reject Filtering having "Q" notch pair can be defined as :

$$H_{NR}(U, V) = \prod_{k=1}^Q H_k(U, V) H_{-k}(U, V)$$

$$\sqrt{(m_2 - m_1)^2 + (y_2 - y_1)^2}$$

classmate

Date _____

Page _____

Where,

$H_k(u, v)$ & $H_{-k}(u, v)$ = High pass filter with centers at (U_k, V_k) & $(-U_k, -V_k)$ respectively.

These translated centers are specified with respect to the center of the frequency rectangle $(M/2, N/2)$.

Therefore, the distance computation for the filters are given by following expression:

$$D_k(u, v) = \left[(u - M/2 - U_k)^2 + (v - N/2 - V_k)^2 \right]^{1/2}$$

&

$$D_{-k}(u, v) = \left[(u - M/2 + U_k)^2 + (v - N/2 + V_k)^2 \right]^{1/2}$$

→ In a video stream, periodic noise is typically caused by the presence of electrical or electromechanical interference during video acquisition or transmission.

→ periodic noise can be reduced with frequency domain filtering.

∴ Frequency domain filtering isolate the frequency occupied by the noise and suppresses them using ~~low~~-reject filter.

(Numerical side)

certain information in the image is relatively less important than other this information is said to be visually redundant

Mapper: reduce interpixel redundancy
Quantizer: reduce psychovisual redundancy
Symbol encoder: reduce coding redundancy

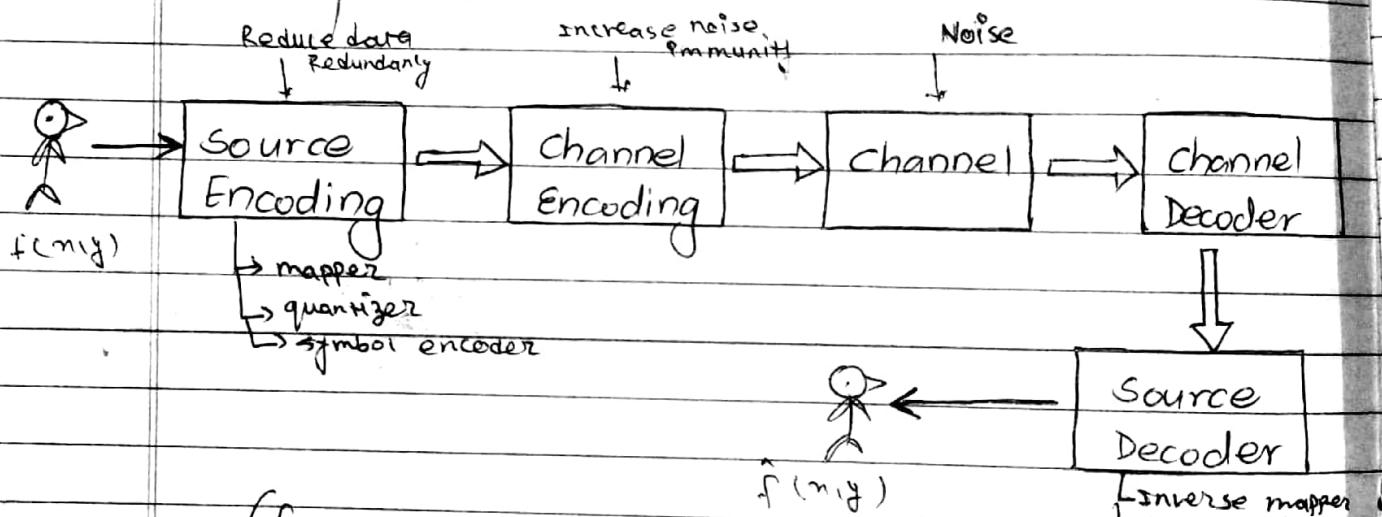
Chapter - 5

Image Compression

[6 hours]

#

Introduction To Image Coding And Compression:



(fig: Block Diagram of Image Compression)

Image compression addresses the problem of reducing the amount of data required to represent a digital image.

The underlined basis of the reduction process is removal of redundant data.

The compression image is decompressed to reconstruct the original image and approximation of it.

Thus, Image (data) ~~is~~ compression is

the reduction of number of bits required to transmit the image probably without any remarkable loss of information.

The image encoding process through the channel is called image compression and image decoding process is known as image decompression.

Application of Data Compression:

- (1) Broadcasting T.V
- (2) Remote Sensing through satellite
- (3) Military Communication
- (4) Facsimile Transmission & Tele-communication

Data Redundancy:

It contains data that either provide no relevant information or simply restate that which is already known, this is called Data Redundancy.

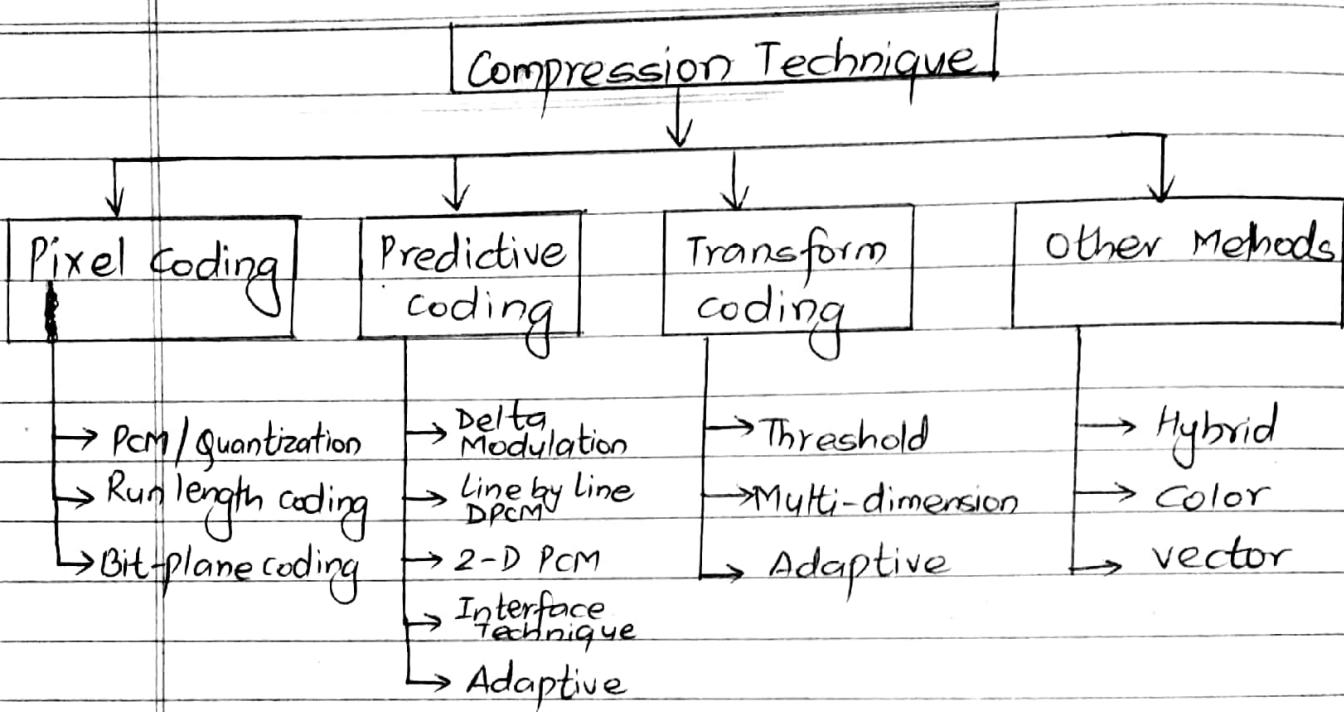


Image (Data) Compression Technique:

The image (data) compression techniques are:

- (1) Pixel coding Technique
- (2) Predictive Coding Technique
- (3) Transform coding Technique
- (4) And, other Methods.

(P.T.O)



(fig: Different Image compression Technique)

① Pixel Coding Technique:

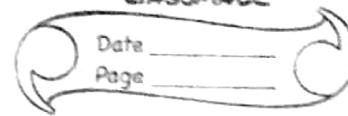
In this technique, each pixel is processed independently ignoring the inter-pixel dependencies.

This technique firstly remove the inter-dependencies between the neighbourhood pixel and then removal of redundant data.

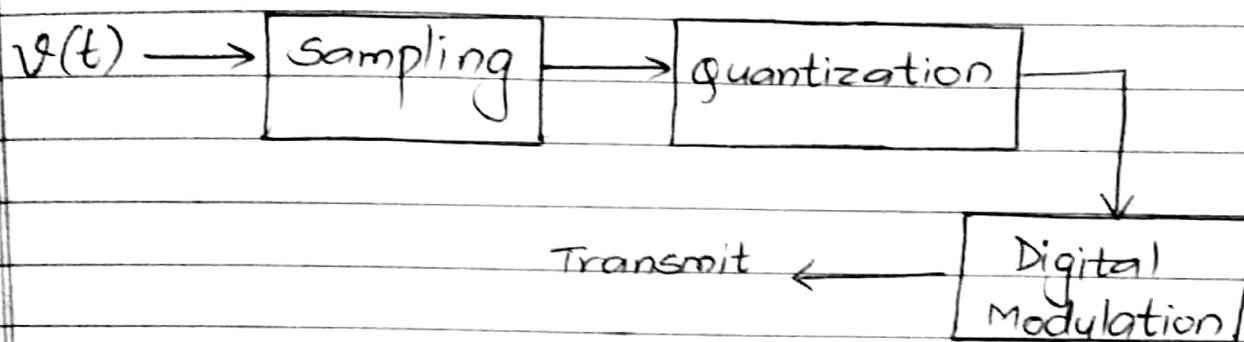
Pixel coding technique is further divided into following three types :

- PCM / Quantization
- Run-Length coding
- Bit-Plane coding

sampling: digitizing the spatial coordinates
quantization: Amplitude digitization (gray-level)
classmate



(a) Pulse Code Modulation (PCM):



(fig: PCM / quantization)

In PCM technique, the incoming video signal is sampled, quantized, and coded by the suitable code word before inputting it to a digital modulator for transmission.

The quantizer output is coded by a fixed length binary code word having usually 8-bit.

The formula for achieving rate of compression of PCM is given by the rate distortion formula.

i.e

$$R_{PCM} = \frac{1}{2} \log \frac{\sigma_{V^2}}{\sigma_{Q^2}}, \quad \sigma_{Q^2} < \sigma_{V^2}$$

Where,

σ_{V^2} = The variance of quantize i/p

σ_{Q^2} = The quantizer mean square Deviation

(b) Run Length Coding:

It is a very simple form of the data compression which represents each subsequence of identical symbols by a pair like (L, a) where, L is the subsequence and a is the recurring symbol.

Run-length coding describes each row of the image by a sequence of length that describes successive runs.

For example, $aaabbbbaaa$ is coded as $3a3b3a$.

The most common methods are as follows:

- (i) To specify the value of the first run of each row.
- (ii) To assume that each row begins with a white ~~black~~ run.

for example : 
black (1) white (2) black (3)

From above figure, the run length code can be written as : 1 black 2 white 3 black

In this technique, the numeric run length coding can be also defined as:

$$000002222111 = 0\ 1\ 5\ 2\ 4\ 1\ 3$$

(Imp)

Entropy Coding: (Replace input string by codeword)

encodes

Entropy ~~coders~~ encodes the given set of symbols with minimum number of bits required to represent them.

The most important entropy coding is Huffman Coding. The theoretical minimum average of bits that are required to transmit a particular source string is known as entropy of source and it can be computed by using following formula:

$$\text{Entropy } (H) = - \sum_{i=1}^N P_i \log_2 P_i$$

It has two major parts. They are:

- (i) Construction of Probability Tree
- (ii) Assigning code to each node of the constructed tree.

It gives the variable length code words and highest probability assigns ~~sub-qa~~ short path and lowest

probability assigns longest path i.e code length.

Huffman Coding Algorithm:

- (1) Arrange the symbol probabilities P_i in decreasing order and consider them as leaf node of a tree.
- (2) While there is more than one node;
 - (a) Merge the 2 nodes with smallest probability to form a new node.
 - (b) Assign '1' & '0' to each pair of branches merging into a node.
- (3) Read sequentially from root node to leaf node.

(Imp)

Source generates the symbol s_1, s_2, s_3, s_4 and s_5 randomly with probability $P_1 = 0.4, P_2 = 0.2, P_3 = 0.2, P_4 = 0.1$ and $P_5 = 0.1$ respectively.

Generate the code word for each symbol using Huffman coding. Also, calculate the compression ratio and efficiency of the system.

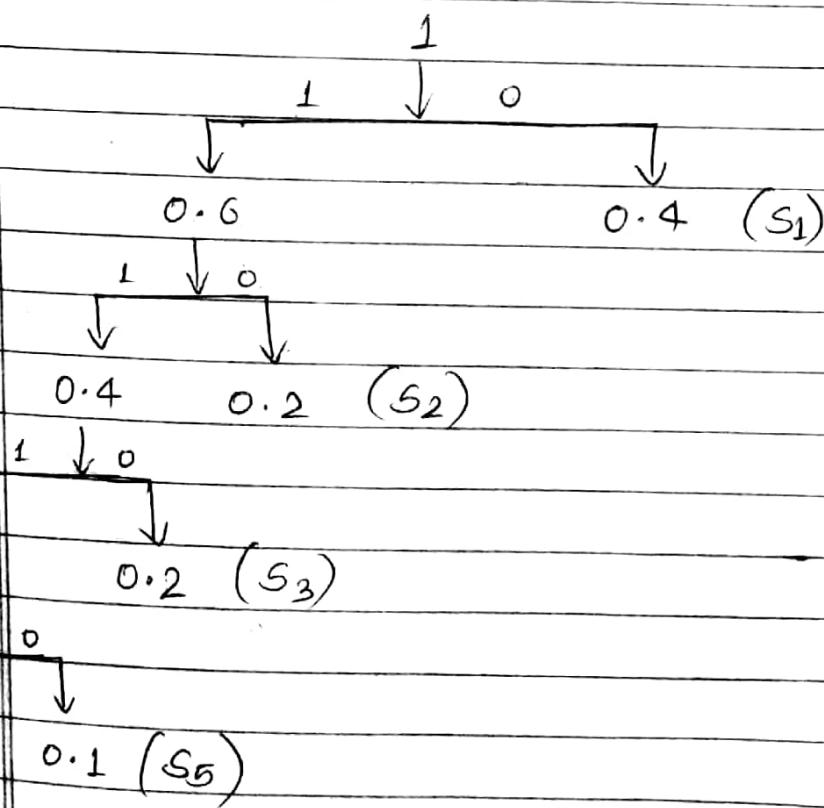
Step(1):

Step	Symbol	Original size Probability (P_i)	step
			1 2 3 4
	S_1	0.4	0.4
	S_2	0.2	0.2
	S_3	0.2	0.2
	S_4	0.1	0.2
	S_5	0.1	

Step(2):

Construct a Huffman Tree:

Assume, Highest probability $0.6 = 1$
 Lowest probability $0.4 = 0$



Step(3): Generate code Word length:

Symbol	Probability (P_i)	Code length (l_i)
s_1	0.4	0 = 1
s_2	0.2	10 = 2
s_3	0.2	110 = 3
s_4	0.1	1111 = 4
s_5	0.1	1110 = 4

Now,

$$\begin{aligned} \text{Compression Ratio (or } L_{\text{avg}}) &= \sum_{i=1}^N P_i \cdot l_i \\ &= 0.4 \times 1 + 0.2 \times 2 + 0.2 \times 3 + 0.1 \times 4 \\ &\quad + 0.1 \times 4 \\ &= 2.2 \text{ bits / symbol} \end{aligned}$$

$$\therefore \text{Entropy (H)} = - \sum_{i=1}^N P_i \log_2 P_i$$

$$= - \left[0.4 \times \log_2 0.4 + 0.2 \times \log_2 0.2 + 0.2 \times \log_2 0.2 + 0.1 \times \log_2 0.1 + 0.1 \times \log_2 0.1 \right]$$

$$= 2.12$$

$$\text{Hence, Efficiency} = \frac{\text{Entropy}}{L_{\text{avg}}} \times 100$$

$$= \frac{2.12}{2.2} \times 100 \%$$

$$= 96.36\%$$

Note:

$$\log_2 P_i = \frac{\log_{10} P_i}{\log_{10} 2}$$

\log_{10} □

classmate

Date _____

Page _____

Q

Calculate the compression ratio and efficiency from below image:

Gray level (r)	0	1	2	3	4	5	6	7
No. of Pixel (n _K)	400	1350	659	2034	816	2560	250	1500

Sol: Step(1): calculate the probability in each gray level

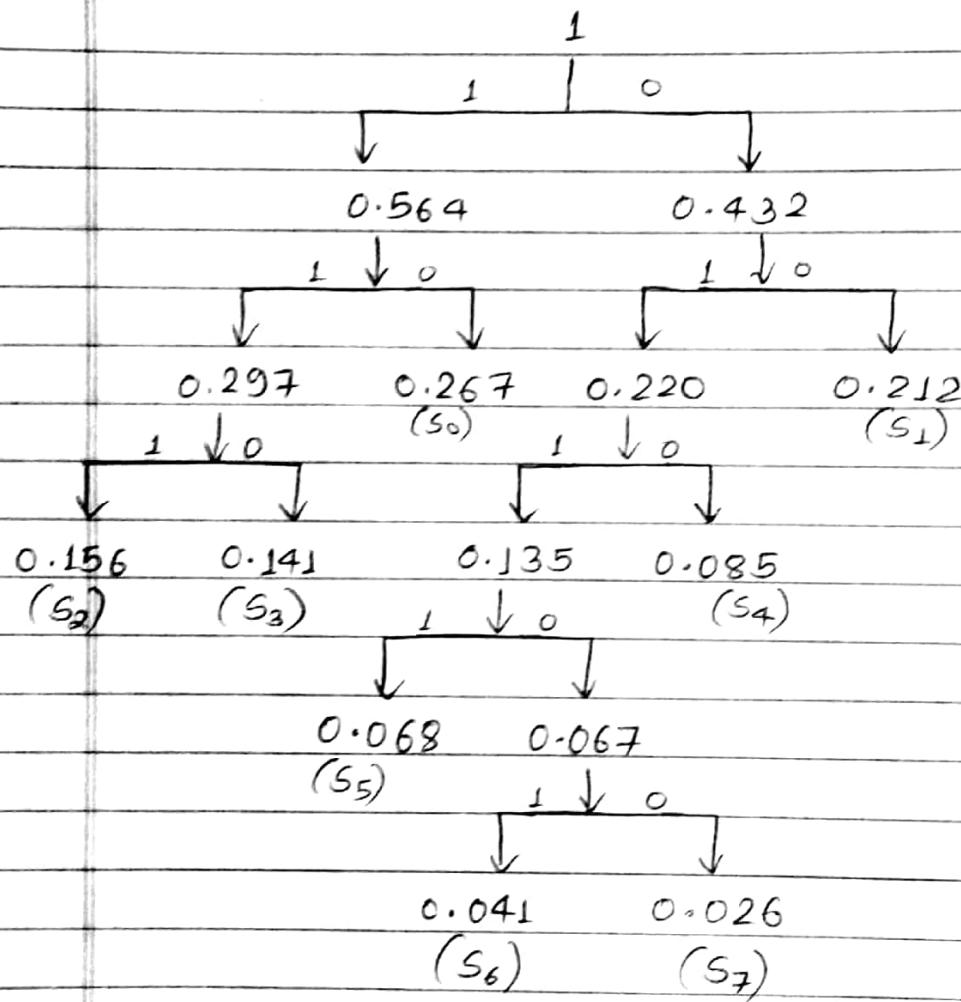
Gray level (r)	No. of Pixel (n _K)	Probability (P _i) = $\frac{n_k}{n}$
0	400	0.041
1	1350	0.141
2	659	0.068
3	2034	0.212
4	816	0.085
5	2560	0.267
6	250	0.026
7	1500	0.156
$n = 9569$		

step(2): Assume symbol for each probability in descending order.

Symbol	P _i	1	Step 2	3	4	5	6	7
S ₀	0.267	0.267	0.267	0.267	0.297	0.432	0.564	1
S ₁	0.212	0.212	0.212	0.220	0.267	0.297	0.432	
S ₂	0.156	0.156	0.156	0.212	0.220	0.267		
S ₃	0.141	0.141	0.141	0.156	0.212			
S ₄	0.085	0.085	0.135	0.141				
S ₅	0.068	0.068	0.085					
S ₆	0.041	0.067						
S ₇	0.026							

Step(3): Construct Huffman Tree

Assume, Highest Probability $0.564 = 1$
 lowest probability $0.026 = 0$



Step(4): Generate code Word length

Symbol	Probability (P_i)	Code Word Length (l_i)
S_0	0.267	10 • = 2
S_1	0.212	00 = 2
S_2	0.156	111 = 3
S_3	0.141	110 = 3
S_4	0.085	010 = 3
S_5	0.068	0111 = 4
S_6	0.041	01101 = 5
S_7	0.026	01100 = 5

Now,

$$\therefore \text{Compression Ratio (Lavg)} = \sum_{i=1}^N P_i l_i$$

$$= (0.267 \times 2) + (0.212 \times 2) + (0.156 \times 3) + (0.141 \times 3) \\ + (0.085 \times 3) + (0.068 \times 4) + (0.041 \times 5) + \\ (0.026 \times 5)$$

$$= 2.711$$

A Log E → calc

$$\therefore \text{Entropy (H)} = - \sum_{i=1}^N P_i \log_2 P_i$$

$$= - [0.267 \log_2 (0.267) + 0.212 \log_2 (0.212) + 0.156 \\ \log_2 (0.156) + 0.141 \log_2 (0.141) + 0.085 \log_2 (0.085) \\ + 0.068 \log_2 (0.068) + 0.041 \log_2 (0.041) \\ + 0.026 \log_2 (0.026)]$$

$$= - [-0.508 - 0.474 - 0.418 - 0.398 - 0.302 \\ - 0.263 - 0.188 - 0.136]$$

$$= 2.687$$

$$\therefore \text{Efficiency (\eta)} = \frac{\text{Entropy (H)}}{\text{Lavg}} \times 100$$

$$= \frac{2.687}{2.711} \times 100 \%$$

$$\boxed{\eta = 99.11 \%}$$

→ Based on the concept of decomposing a multilevel (monochrome or color) image into a series of binary images and compressing each ^{classmate} binary image via one of several well-known binary compression methods.

(C) Bit Plane Coding:

It breaks the image into bit planes and apply run length coding to each plane.

Let, 'I' be an image where every pixel value is 'n'-bit long. We can express every pixel in binary using n-bit from out of an image and n-binary matrix (called Bit plane) Where, I^i th matrix consist of i th bit of the pixel of 'I'.

It can be illustrated as:

$$I = \begin{bmatrix} 101 & 110 \\ 111 & 011 \end{bmatrix}, \text{ corresponding } 3\text{-bit plane and } 2 \times 2 \text{ matrix}$$

Now,

$$I' = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$$

Bit plane coding uses the run length coding but in case of m-bit gray level scale image at polynomial base 2 is defined as:

$$a_{m-1} \cdot 2^{m-1} + a_{m-2} \cdot 2^{m-2} + \dots + a_1 \cdot 2^1 + a_0 \cdot 2^0$$

The intensities of m-bit monochrome image can be represented in the form of base-2 polynomials

- Based on eliminating the redundancies of closely spaced pixels - in space / time
- by extracting & coding only the new classmate information in each pixel.

Date _____
Page _____

→ New Information of pixel
is defined as the difference
between the actual &
predicted value of pixel.

(2) Predictive Coding:

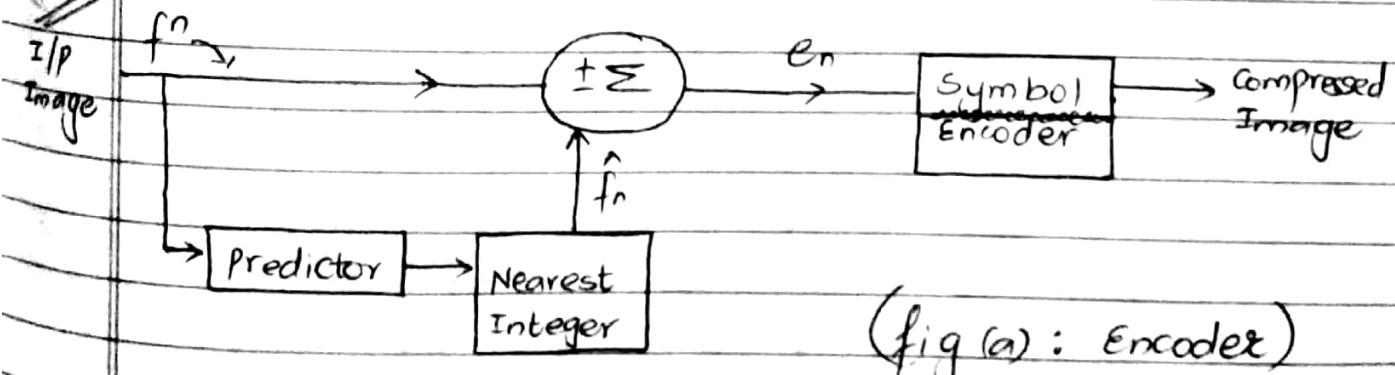
Future value will be predicted on the basis of past values.

If we know ~~that~~ the past behaviour of a signal upto a certain point in time then, it is ~~impossible~~ to make some prediction about its future values then such process is called Prediction.

It is of following two types :

- lossless predictive coding
- lossy Predictive coding

(a) Lossless Predictive Coding: (2015 fall)



(fig(a) : Encoder)

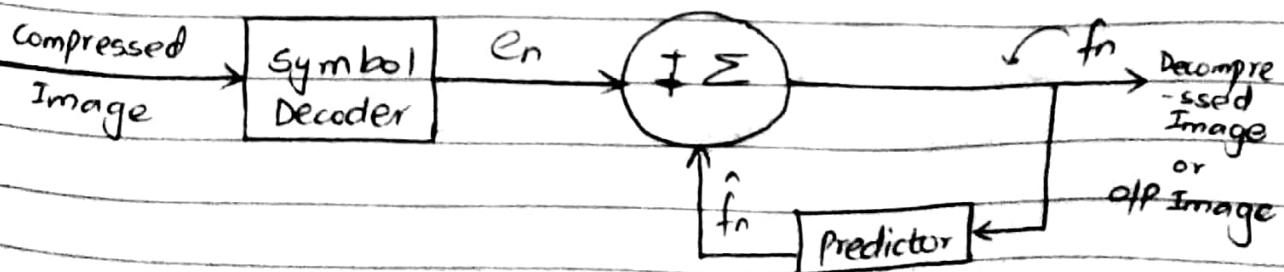


fig: lossless Predictive coding]

(fig(b) : Decoder)

There is no information loss and image can be reconstructed exactly the same as the original.

So, compression process is reversible.

The system consist of an encoder and a Decoder each containing an identical predictor.

The [predictor] generates anticipated value of that pixel based on some number of past values. The output is then rounded to the nearest integer.

Then, the differences or prediction error is calculated as :

$$e_n = f_n + \hat{f}_n$$

Where,

e_n = Prediction Error

f_n = i/p Image symbol

\hat{f}_n = prediction function

In case of encoding, symbol encoder uses variable length coding technique to encode symbol.

In case of decoding, symbol decoder uses decoding function as:

$$\hat{f}_n = e_n + f_n$$

Where,

\hat{f}_n = Decompressed Image or o/p Image

In most cases, the predictor is formed by a linear combination of 'm' previous pixel.

i.e

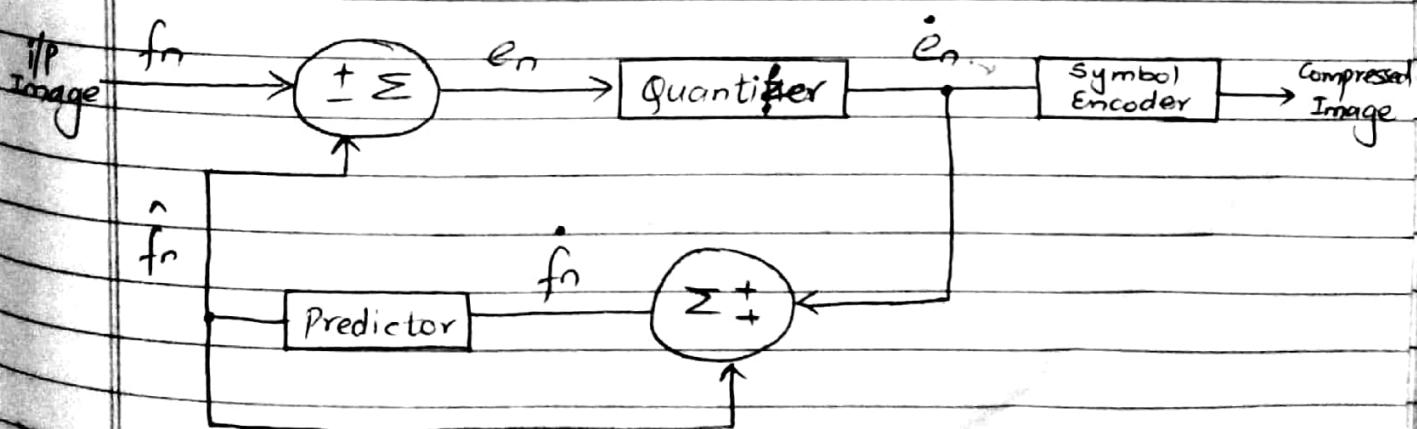
$$\hat{f}_n = \text{round} \left[\sum_{i=1}^m \alpha_i f_{n-i} \right]$$

Where,

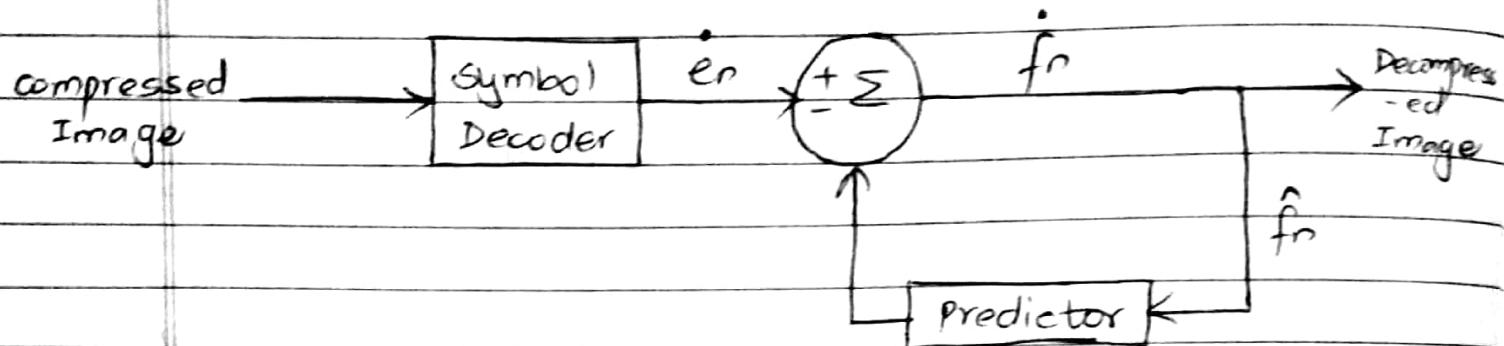
α = Prediction co-efficients

m = Order of linear prediction

(b) Lossy Predictive Coding:



[fig: encoder]



[fig: Decoder]

The major difference in this technique is that a quantizer is added between the symbol encoder and the point where the prediction error is calculated.

The quantizer absorbs the nearest value of a prediction error and maps it into a limited range of output denoted by \hat{e}_n at encoding & \hat{f}_n at decoding, which is defined as :

\hat{e}_n is given by :

$$\hat{e}_n = \hat{f}_n - f_n$$

And,

\hat{f}_n is given by :

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

~~compressing~~ ~~number of frame~~
~~into~~ ~~together~~

~~compressing~~ ~~stage~~

Interframe & Intraframe Coding:

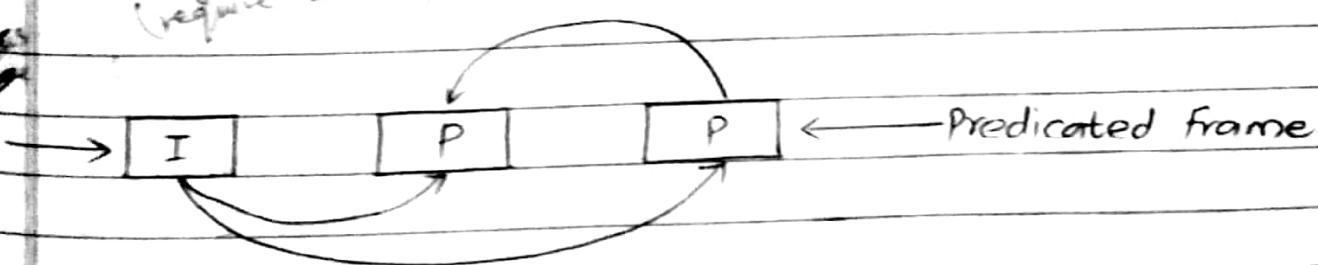
It is used for video compression.
The Intraframe coding is also known as I-frame. Intraframe compression uses only current frame for encoding.

Interframe coding is encoding and compression technique between the video signal. (use number of frame)

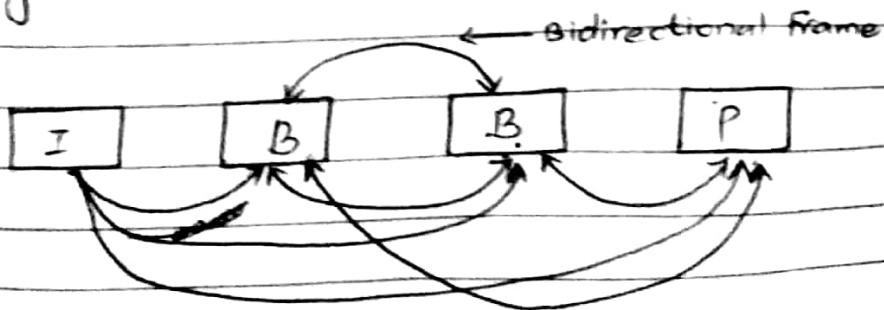
Interframe & Intraframe coding is categorized into three types on the basis of predicted frame.

- (1) Predictive Frame (P-Frame)
- (2) Intraframe or Independent Frame (I-Frame)
- (3) Bi-directional Frame (B-Frame)

(refers to previous & next frame for decod.)



[fig(a): I and P-frame]



[fig(b) : I, P and B-Frame]

- (10)
- + morphology is a broad set of image processing operation that process image based on shape.
 - * A binary image is digital image ~~that has~~ ^{classmate} only two possible values. (black ~~white~~ ^{top})
 - * each pixel is stored as a single bit i.e. 0 or 1

Chapter - 6

Introduction To Morphological Image Processing

[4 hours]

Logic Operation Involving Binary Images:

The language and theory of mathematical morphology often present a dual (but equivalent) view of binary images.

Thus,

We have considered a binary image to be a bitvalued function of spatial co-ordinates (x and y).

Morphological Theory views a binary image as a set of foreground (one valued) pixel and elements of which are in \mathbb{Z}^2 (cartesian product) i.e pair of elements.

Set operation such as Union and Intersection can be applied directly to binary image sets.

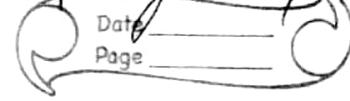
For Example,

If 'A' & 'B' are binary images then,

$C = A \cup B$ is a binary image.
where, a pixel in C is a foreground pixel

* It is a type of image processing in which structure of the object within an image are modified.

* dilation, erosion are fundamental morphological operation



if either or both the corresponding pixel in A and B are foreground pixel!

The function c is given by two view.
They are :

(1) First View :

$$c(x,y) = \begin{cases} 1 & , \text{ if either } A(x,y) \text{ or } B(x,y) \text{ is 1} \\ & \text{or} \\ & \text{if both are 1} \\ 0 & , \text{ otherwise} \end{cases}$$

(2) Second View :

$$c = \{ (x,y) : (x,y) \in A \text{ or } (x,y) \in B \text{ or } (x,y) \in (A \text{ and } B) \}$$

Where,

A & B are Binary Images where the elements of A and B are 1-valued.

Thus,

We see that the function point of view deals with the ^{both} foreground (1) and background (0) pixel simultaneously.

The set point of view deals only with foreground pixel and it is understood that all pixel are not foreground pixel

constitute the ~~diagram~~ background.

The set operation can be performed on binary images using following logical operations:

Set operation	Logical operation for Binary Images	Name
(1) $A \cap B$	$A \& B$	AND
(2) $A \cup B$	$A \mid B$	OR
(3) A^c	$\sim A$	NOT
(4) $A - B$	$A \& \sim B$	DIFFERENCE

~~N.I.M.P~~ # Dilation & Erosion:

~~(P.R)~~ Dilation:

Dilation is an operation that grows or thickens objects in an image.

The specific manner and extent of this thickening is controlled by a set referred to as a structuring element.

Graphically, structuring elements can be represented either by a matrix of 0's and 1's or as a set of foreground (1-valued) pixels.

So, the origin of the structuring element must be clearly identified by

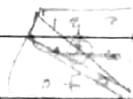
using the both representation.

Dilation operation can be done by using following steps:

- ① Representation of original Image with Rectangular object.

let,

0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0	0



- ② Structuring Element with 5-pixels arranged in a diagonal line.

1
1
1 ← origin or center of the structuring element as shown with a dark border.
1

- ③ Structuring element translated to several location in the image.

overlap 1-valued pixel
in the original Image

*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*
1	1	1	1	1	1	1	*	*	*
*	1	1	1	1	1	1	*	*	*
*	*	1	1	1	1	1	1	1	*
*	*	*	*	*	*	*	*	*	*
*	*	*	*	*	*	*	*	*	*

Donot overlap 1-valued pixel
in the original Image

*	*	1	1	1	1	1	1	*	*
*	*	*	1	1	1	1	1	1	*
*	*	*	*	1	1	1	1	1	*
*	*	*	*	*	1	1	1	1	*
*	*	*	*	*	*	1	1	1	*

- ④ Output Image ie shaded region will be replaced by the 1's in the original Image.

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	1	1	1	1	1	1

Now,

the dilation of A by B denoted as $A \oplus B$ is defined as the set operation as :

reflected &
 translated B
 overlaps
 A

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

classmate

Date _____

Page _____

Where,

\emptyset = Empty set

B = structuring element

The dilation of A by B is the set consisting of all the structuring element origin location where the reflected and translated B overlaps at least one element of A .

It is a convention in image processing to the first operand of $A \oplus B$ be the image and second operand be the structuring elements which usually is much smaller than the image.

Dilation is of following two types:

→ Dilation is associative

$$A \oplus (B \oplus C) = (A \oplus B) \oplus C$$

→ Dilation is commutative

$$A \oplus B = B \oplus A$$

~~(pre)~~ Erosion:

~~(S/N)~~ Erosion is an operation that thins or sinks objects in a binary image.

As in dilation, the manner and extent of erosion is controlled by a structuring element.

The erosion process can be done by using following steps:

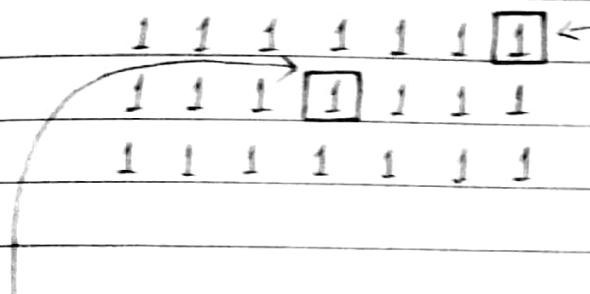
- ① Representation of Original Image with Rectangular Object.

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

- ② Structuring element with 3-pixel arranged in vertical line.

1
1 ← Origin of the structuring element is shown with dark border.
1

- ③ Structuring element translated into several location in the image.



The result is 0 at this location in the output image because all or part of the structuring element overlaps the background.

The result is 1 at this location in the output image because the structuring element fits entirely within the foreground.

- ④ Output Image: The shaded region shows the location of 1's in the original image:

0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

Now,

the erosion of A by B denoted as:
 $A \ominus B$ is defined as:

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

where,

$(B)_z \subseteq A$ means that $(B)_z$ is a subset of A .

This equation says that, the erosion

of A by B is the set of all points
 z such that B translated by z is
contained in A:

Thus, B is contained in A, is equivalent to B not sharing any elements with the background of A.

The equivalent expression as the definition of erosion is defined as:

$$A \ominus B = \{ z \mid (B)_z \cap A^c = \emptyset \}$$

Properties: $A \ominus B \neq B \ominus A$,

~~#~~ Opening & closing:

Opening:

The morphological opening of A by B is denoted as $A \circ B$ and defined as the erosion of A by B and followed by a dilation of the result by B.
 i.e

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

Now,
 an equivalent formulation of opening
 is :

$$A \circ B = \cup \{ (B)_z \mid (B)_z \subseteq A \}$$

Where,

$\cup \{ \}$ denotes the union of all sets inside the braces.

$A \circ B$ is the union of all transactions of B that fits entirely within A .

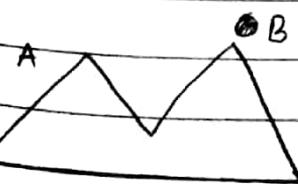
Closing:

The morphological closing of A by B is denoted as $A \bullet B$ and defined as the dilation of A by B and followed by a erosion of result by B .
i.e

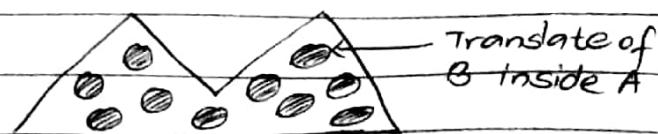
$$A \bullet B = (A \oplus B) \ominus B$$

Geometrically, $A \bullet B$ is the complement of the union of all the translation of B that donot overlap A .

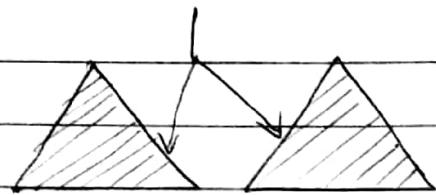
The opening & closing can be illustrated as:



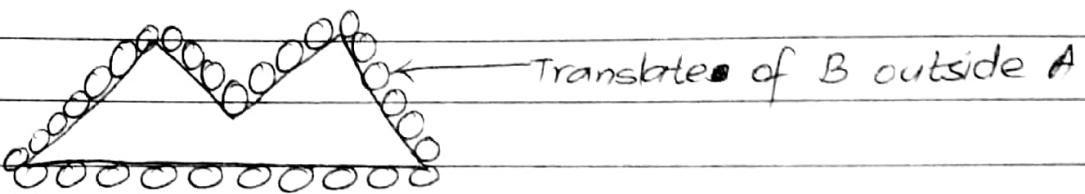
fig(a): set A and structuring element B



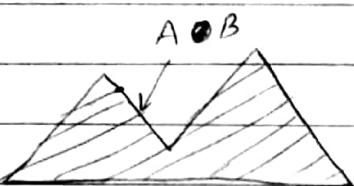
fig(b): Translation of B that fit entirely within set A .

$A \ominus B$ 

(fig(c)) : The complete opening)



(fig(d)) : Translate of B outside the border of A)



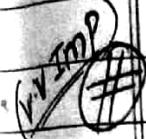
(fig(e)) : The complete closing)

Fig: Opening & closing as onions and translated structuring element.

Chapter- 5

Image Compression

[Remaining topic]



Redundancy:

In digital image processing, three basic data redundancy can be identified and ~~exploit~~ exploited as follows:

- (1) Coding Redundancy
- (2) Interpixel Redundancy
- (3) Psychovisual Redundancy

CIP

① Coding Redundancy:

We know that, average number of bits required to represent ~~each pixel~~ is given by:

$$L_{avg} = \sum_{k=0}^{L-1} L(k) \cdot P(k)$$

$$L_{avg} = \frac{1}{n} \sum_{i=0}^n L_i$$

Hence, the number of bits required to represent the image is $n \times L_{avg}$, where n is the total number of pixels in the given image.

Maximum compression ratio is

achieved when L_{avg} is minimized.
But,

Coding the gray level in such a way that the L_{avg} is not minimized then resulting image is said containing coding redundancy.

(2) Interpixel Redundancy:

It is related to interpixel correlation within an image. Usually, the value of certain pixel in the image can be reasonably predicted from the values of its neighbours in the image.

Thus, the values of the individual pixel carries relatively small amount of information and much more information about pixel value that can be inferred on the basis of its neighbours' ^{concluded} value.

These dependencies between pixel value in the image is called interpixel redundancy.

③ Psychovisual Redundancy:

The eye does not respond with equal sensitivity to all visual information. Human perception searches for important features (edges, texture) and does not perform quantitative analysis of every pixel in the image.

So, Psychovisual Redundancy takes into perception of the human visual system.

Chapter - 7

Image Segmentation [7 hours]

Introduction:

Image segmentation is a process of partitioning a digital image into multiple regions (set of pixels).

The goal of segmentation is to simplify and change the representation of an image into something that is more meaningful and easier to analyze.

It is typically used to locate objects and boundaries (line and curves) in an image.

The result of an image segmentation is a set of regions that collectively cover the entire image.

Each of the pixels in a region are similar with respect to the some characteristics such as colour intensity or texture.

Segmentation accuracy determines the success or failure of computerized analysis procedure.

✓ Some practical application of image segmentation are as follows:

(1) Medical Imaging:

- Measurement of tissue volumes
- Diagnosis
- Treatment Planning
- Study of anatomical structure

(2) Locate Object in satellite Image

(3) Face Recognition

(4) Finger print Recognition

(5) Machine vision

(6) Iris Recognition

Image segmentation algorithm are generally based on two properties:

- (a) Discontinuity
- (b) Similarity (continuity)

Discontinuity: (pre-university)

This approach is to partition an image based on abrupt changes in

intensity such as edges in an image

There are three basic types of gray level discontinuities in an image i.e points or spots, lines and edges.

Continuity: (Me-university)

This approach is based on partitioning image into regions that are similar according to pre-defined criteria such as thresholding, region growing, region merging, and splitting.

V. IMP
#

Types Of Image Segmentation On the property of similarity:
[Segmentation Technique]

- (1) Region Based Segmentation
- (2) Segmentation Using Region Growing
- (3) Using Region splitting & Merging
- (4) Segmentation By Thresholding
- (5) Basic Global Thresholding

① Region Based Segmentation:

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

This technique is based on finding the regions directly. To apply this technique following basic formulation condition must be applied:

Let, R represent the entire image region, segmentation partition R into n -sub regions as $R_1, R_2, R_3, R_4 \dots R_n$ such that:

$$(a) \bigcup_{i=1}^n R_i = R$$

It indicates that every pixel must be in a region.

$$(b) R_i \cap R_j = \emptyset$$

It indicates that the multiple regions must be disjoint.

$$(c) P(R_i) = \text{TRUE}$$

It indicates that properties must be satisfied by ^{all} the pixels in a segmented regions.

$$(d) P(R_i \cup R_j) = \text{FALSE}$$

It indicates that ~~reset~~ region R_i and R_j are different.

(2) Segmentation Using Region Growing:

Region growing is a technique that groups pixels or sub-regions into larger regions based on pre-defined criteria.

let us, pick up an arbitrary pixel (r, c) from the domain of an image to be segmented. This pixel is called seed Pixel.

The basic approach is to start with a set of seed points. Then, examine the nearest neighbour (i.e 4 or 8 neighbour) of (r, c) one by one and neighbourhood pixel accepted belongs to the same region as (r, c) .

Once, a new pixel is accepted due to the homogeneity property of a region as a member of the current region. Then,

Now, the nearest neighbour of this new pixel are examined. This

- (ii) If the formula or the condition is satisfied then merge the all sub-regions

This technique starts somewhere at the middle level. Suppose we start with a rectangular region of size $m \times n$ pixels.

To each region homogeneity property is tested. If the test fails, the region is split into four quadrants each of size $m/2 \times n/2$.

Now, If the region satisfies the homogeneity property then merging process is followed to form a size $2m \times 2n$.

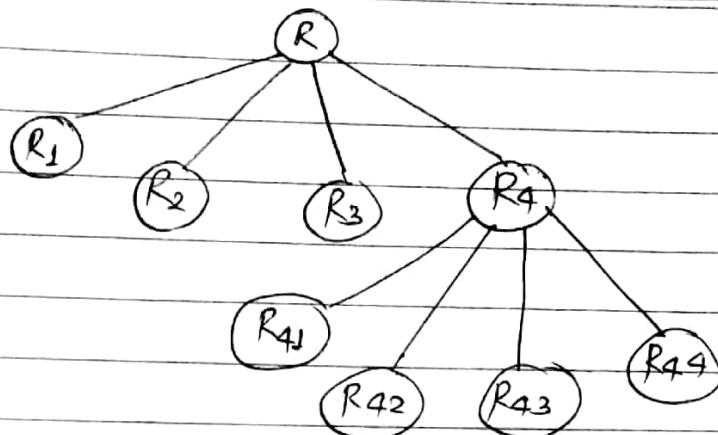
So, the procedure is summarized as:

the region R_i :

- (i) Split R_i into 4 disjoint quadrant any region R_j for which $P(R_j) = \text{FALSE}$
- (ii) Merge any adjacent regions R_j and R_k for which $P(R_j \cup R_k) = \text{TRUE}$
- (iii) Stop when no further splitting or merging is possible.

Example:

R_1	R_2
R_3	R_{41} R_{42}
	R_{42} R_{44}



fig(a): Partitioned Image

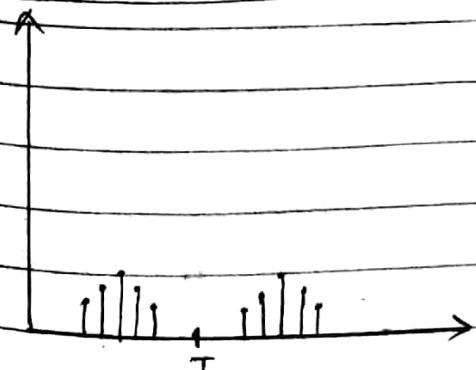
fig (b) : Corresponding quadtree

(imp)

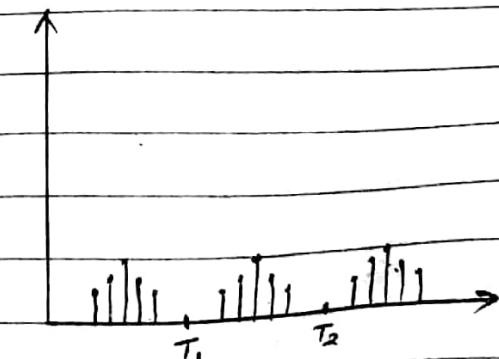
(4) Segmentation By Thresholding : (2012 fall) (2015 fall)

An image contains various regions corresponding to different objects. The pixels comprising a region and receive information from point of corresponding object.

Hence, ~~Thresholding~~ is the simplest method for image segmentation. It plays central ~~position~~ role in application of an image segmentation.

Foundation:

(a) A Single Threshold



(b) Multiple Threshold

fig: Gray level Histogram that can be Partitioned

Suppose that, the gray level histogram of an image $f(x, y)$ composed of light objects on a dark background is shown in figure above.

The figure(a), it shows to extract the object from the background by selecting threshold T that separate the object from background. Then any point (x, y) for which $f(x, y) > T$ is called an object point. Otherwise, background point.

In figure(b), it shows a slightly more general case of this approach (multiple thresholding). There are two types of light objects on the dark background.

Here, multi-level thresholding classifies a point (x, y) as belonging to one object class. It can be expressed as :

If,

$T_1 < f(x, y) \leq T_2$, to the other object.

Class if,

$f(x, y) > T_2$, to the background.

If,

$f(x, y) \leq T$, to the background.

Threshold can be viewed as operation that involves test against a function T of the form,

$$\therefore T = T[(x,y), P(x,y), f(x,y)]$$

Where,

$P(x,y)$ = Some local property of the point

$f(x,y)$ = Gray level of point (x,y)

A threshold image can be defined as,

$$g(x,y) = \begin{cases} 1, & \text{if } f(x,y) > T \\ 0, & \text{if } f(x,y) \leq T \end{cases}$$

Thus, if T depends on both $f(x,y)$ & $P(x,y)$, the threshold is called as local and if T depends on spatial co-ordinates, then, it is called Dynamic or Adaptive.

(5) Basic Global Thresholding: Automatic thresholding (2015 fall) notes

In this approach, first partition the image histogram by using a single global threshold T .

Segmentation is then accomplished by scanning the image pixel by pixel and labeling each pixel as object or background depending whether the gray level of the pixel is greater

- ① Initial value of threshold = T_0 (select)
 ② Once two group of pixel u_1 & u_2 ← may have similar info
 ③ mean \bar{u}_i for group G_1 , \bar{u}_2 for G_2
 repeat 4.

$$T = \frac{u_1 + u_2}{2}$$

$$\Rightarrow |T_i - T_{i+1}|^2 < \epsilon$$

or less than the value of T .

The pixel are assigned and labeled are also assigned. This technique is specially used in the industrial areas where control of illumination is possible.

How to obtain T using global thresholding as follows:

- Select an initial estimate for T .
- Segment the image using T . This will provide two group of pixels ie $G_1 \leq T$ and $G_2 > T$
- Compute the average gray level values u_1 and u_2 for the pixel in the regions G_1 and G_2 .

The average gray level threshold value is given by:

$$T = \frac{1}{2} (u_1 + u_2)$$

- Repeat the step (ii) through step (iii) until the difference in T in successive iteration is smaller than predefined parameter.

$$i.e |T_i - T_{i+1}| < \epsilon \leftarrow \begin{array}{l} \text{predefined} \\ \text{threshold} \end{array}$$

Therefore in i^{th} iteration
| $T_i - T_{i+1}| < \epsilon$ → T_{i+1} increase

- create a small sub image (template) of a object to be found
- do pixel by pixel matching of the template with the image by placing centre of the template at every possible pixel of main image

with correlation
Date _____
Page _____

Template Matching : (2012 fall) (10)

Template Matching is a technique in DIP for finding small parts of an image that match a template image.

It can be used in manufacturing as a part of quality control as a way to detect edges in an image.

Here,

We have a template $g(i, j)$ and we wish to detect its instances in an image.

The incoming is compared directly to copies (templates) stored in the memory. So, it has various different applications such as: Face Recognition, Finger Print Detection & Medical Image System.

The logic behind this technique is,

In this technique, the behind logic is that comparing between pixels or pattern. This method is also called as Linear

Spatial Filtering:

This is done only through the use of correlation theorem.

(P.T.O)

using correlation for the matching let template pixel of original image itself,
of correct recognition

Correlation Theorem:

It is the measure of degree to which two variables agree or not in a actual value.

Here,

two variables are the corresponding pixel values in two images i.e template and source image.
So, it is used to detect location of a certain object inside an image.

Therefore, Correlation coefficient is defined as :

$$\text{correlation} = \frac{\sum_{i=0}^{N-1} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=0}^N (x_i - \bar{x})^2} \sqrt{\sum_{i=0}^N (y_i - \bar{y})^2}}$$

Where,

x_i = Template Gray level Image

y_i = Source Gray level Image

\bar{x} = Average Gray level Template Image

\bar{y} = Average Gray level source Image

Edge Linking & Boundary Detection

① Edge or Boundary Linking & Detection:

① Using Template Matching:

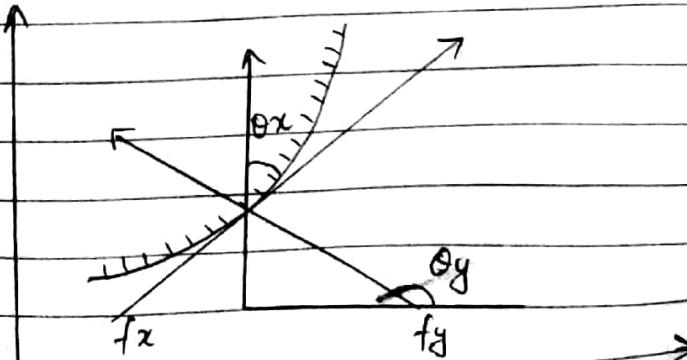
Edge detection extract edges of object from an image. Edge characterize object boundaries and therefore it is useful for segmentation, registration and identification of an object in the scenes.

The variation of an image features usually brightness to edges. So, the edges are the representation of discontinuities of an image intensity function.

So, edge detection algorithm is essentially a process of detection of discontinuities in an image.

Hence, the change in brightness level indicates edges.

→ Edge Detection by using template matching can be derived as follows:



(fig: Gradient of $f(x,y)$ along r-direction using Template matching)

gradient: direction of change in image
intensity.

classmate

Date _____

Page _____

For the continuous image $f(x, y)$, its derivatives assumes a local maximum in the direction of edge.

Therefore, measuring the gradient of $f(x, y)$ can be used for edge detection.

It can derived as :

∴ Gradient of Horizontal direction :

$$\Delta f = \left[\frac{df}{dx}, 0 \right]$$

∴ Gradient of vertical direction :

$$\Delta f = \left[0, \frac{df}{dy} \right]$$

∴ Gradient in θ (any) direction :

$$\Delta f = \left[\frac{df}{dx}, \frac{df}{dy} \right] \quad (1)$$

Now,

$$\frac{df}{dr} = \left[\frac{df}{dx} \cdot \frac{dx}{dr} + \frac{df}{dy} \cdot \frac{dy}{dr} \right]$$

$$\therefore \frac{df}{dr} = f_x \cos\theta + f_y \sin\theta$$

Then,

The Max^m value of $\frac{df}{dr}$ is obtained when,

$$\frac{d}{d\theta} \left(\frac{df}{dr} \right) = 0$$

Hence,

$$f_x \sin \theta_g + f_y \cos \theta_g = 0$$

$$\theta_g = \tan^{-1} \left(\frac{f_y}{f_x} \right)$$

$$\left(\frac{df}{dr} \right)_{\max} = \sqrt{f_x^2 + f_y^2}$$

Where,

θ_g is the direction of edge & $\left(\frac{df}{dr} \right)_{\max}$ is the gradient value that indicates the edges.

(Top)
②

Using Gradient Model (operator):

(A) First Derivative :

- (i) Roberts operator
- (ii) Prewitt operator
- (iii) Sobel operator

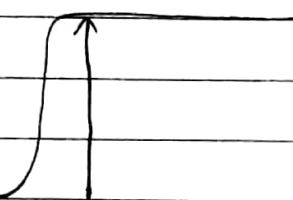
(B) Second Order Derivative

~~Edge detector~~

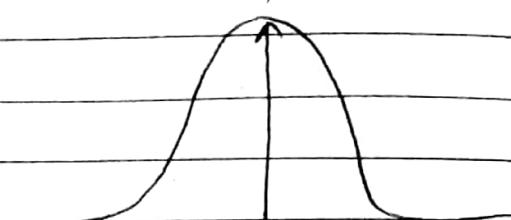
A) First Derivative:

[ie Difference operator]

$$f'(x,y)$$



edge $[f(x,y)]$



fig(b): cross section of
an image edge
using first derivative

The derivatives which yield high values at places where gray level changes rapidly are used to find gradient of an image as shown in the figure above.

Above figure shows that, first derivative produces thicker edge because the first derivative is positive at the point of transition into and out of the ramp as we move from left to right.

If $(\frac{df}{dx})$ and $(\frac{df}{dy})$ are the rates of changes of 2D-function $f(x,y)$ along x and y axis then the direction in which rate of change has the

greatest magnitude is defined as :

$$\theta = \tan^{-1} \left[\frac{(df/dy)}{(df/dx)} \right]$$

And,

$$\text{Magnitude} = \sqrt{\left(\frac{df}{dx}\right)^2 + \left(\frac{df}{dy}\right)^2}$$

The ~~vector~~ having this magnitude and direction is called the gradient of $f(x,y)$ and is denoted by $f'(x,y)$ for digital image.

First derivative is also called as Difference Operator which is used to find the differences like :

$$f(r,c) - f(r-1, c) = d_1$$

$$f(r,c) - f(r, c-1) = d_2$$

$$\therefore \text{The magnitude } f'(r,c) = \sqrt{d_1^2 + d_2^2}$$

And,

The direction of the greatest step as,

$$\theta(r,c) = \tan^{-1} \left(\frac{d_2}{d_1} \right)$$

There are various gradient operator only using first derivative are as follows:

- (i) Roberts Operators
- (ii) Prewitt Operators
- (iii) Sobel Operators

RPS

Let the 3×3 area shown in the figure below represents the gray level in a neighbourhood of an image.

$$\begin{bmatrix} z_1 & z_2 & z_3 \\ z_4 & z_5 & z_6 \\ z_7 & z_8 & z_9 \end{bmatrix}$$

(fig : 3×3 Image)

(i) Roberts Operators :

It is defined as :

$$d_1 = G_x = \frac{df}{dx} = z_9 - z_5$$

$$d_2 = G_y = \frac{df}{dy} = z_8 - z_6$$

The derivative can be implemented for the entire image by using mask which is given as :

$$\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \quad \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

d_1 d_2

$$\begin{bmatrix} z_1 & z_2 & z_3 \\ z_4 & -1 & 0 \\ z_7 & 0 & 0 \end{bmatrix}$$

Here,

the mask of 2×2 are upward because they do not have clear center.

(ii) Prewitt Operators :

It is defined as :

$$d_1 = Gx = [(z_7 + z_8 + z_9) - (z_1 + z_2 + z_3)]$$

$$d_2 = Gy = [(z_3 + z_6 + z_9) - (z_1 + z_4 + z_7)]$$

Then, The prewitt operator mask will be :

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

for d_1

for d_2

Here,

the difference between the first & third row of an 3×3 image region approximates the derivatives in the x -direction and the difference between the 3rd & 1st column approximates the derivatives in y -direction.

(iii) Sobel operators :

In Sobel operators higher weights are assigned to the pixels, close to the candidate pixel.

So, it is defined as :

$$d_1 = G_x = [(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)]$$

$$d_2 = G_y = [(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)]$$

Then,

the corresponding 3×3 mask for sobel operator will be :

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

for d_1

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

for d_2

(B) Second Order Derivative : (pre-university)



fig (a) : Cross Section
of ramp edge

fig (b) : Second derivative
of ramp edge

The second derivative is positive at the transition into ramp associated with the dark side of the edge and negative at the transition associated with the light side of the edge i.e. out of the ramp. And, zero along the ramp in area of constant gray level.

✓ The sign of second derivatives can be used to determine whether an edge pixel lies on the dark or light side of the edge.

So, there are additional properties of the second derivatives around the edge.

✓ For second order derivatives following two point is mostly important:

- It produces two values of every edge in an image.
- An imaginary straight line joining the extreme positive and negative value of the second derivative would cost zero near the mid-point of the edge.

Second order derivatives is derived by the Laplacian filter or operator.

The Laplacian of a 2D-function (x, y) is a second order derivative which is defined as :

$$\Delta^2 f = \frac{d^2 f}{dx^2} + \frac{d^2 f}{dy^2}$$

i.e

$$\Delta^2 f(x, y) = \frac{d^2 f(x, y)}{dx^2} + \frac{d^2 f(x, y)}{dy^2}$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

For a 3×3 region, one of the two features encountered most frequently as :

$$(i) \Delta^2 f = 4z_5 - (z_2 + z_4 + z_6 + z_8)$$

The Laplacian mask is :

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

(ii) A digital approximation including the diagonal neighbour is given by :

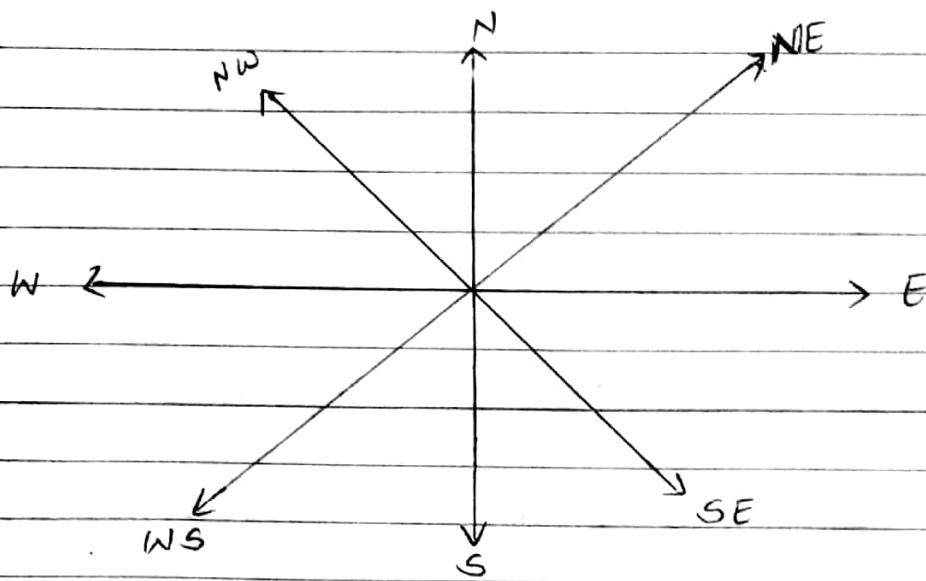
$$\Delta^2 f = 8z_5 - (z_1 + z_2 + z_3 + z_4 + z_6 + z_7 + z_8 + z_9)$$

Then,

The Laplacian mask will be :

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

③ Compass Operator:



(fig : compass operator)

Compass Operators measures the gradient in a selected number of direction.

The ~~matrix~~ representation for different direction is :

(i) \uparrow
$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

 N → +ve
 S → -ve

(ii) \leftarrow
$$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$
 ← W
 W → +ve
 E → -ve

(iii) \downarrow
$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

 G

(iv) \rightarrow
$$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$
 → E

(v) $\begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}$

NW
↑

(vi)

$$\begin{bmatrix} 0 & 1 & 1 \\ -1 & 0 & 1 \\ -1 & -1 & 0 \end{bmatrix} \rightarrow NE$$

(vii) $\begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix} \leftarrow SW$

(viii) $\begin{bmatrix} -1 & -1 & 0 \\ -1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \rightarrow SE$

(fig : Direction wise Matrix Representation of compass operator)

Line & Spot Detection:

① Line Detection / Curve Detection:

It is an important step in image processing and analysis. Lines are features in any scene taken by sensing devices.

We can detect the line by using following compass gradient i.e 4 line detection kernel are as below :

(i) $\begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$

→ Horizontal

(ii) $\begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix}$

→ +45°

-45°

(iii)

$$\begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}$$

→ vertical

(iv)

$$\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

→ -45°

② Spot Detection:

These are most easily detected by comparing the value of a pixel with an average or median of the neighbourhood pixel.

The point detection mask or spot is given by:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & (8) & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

(fig: Spot Detection kernel)

The point ^(spot) has been detected at the location on which the mask is centered if $|R| \geq T$.

where, T = Non-Negative Threshold

R = Response of mask at any point in image

The response of mask R is given by:

$$R = \sum_{i=1}^n w_i z_i$$

Where,

w_i = window or mask coefficient

z_i = gray level of the i^{th} pixel.

The idea is that, a spot will be detected easily from its surrounding by this method.

Chapter-8

Representation & Description

[3 hours]

Introduction to Representation:

The segmentation technique is a decomposition of an image generally referred to as raw data in the form of pixels along a boundary contained in a region.

Although these data sometimes are used directly to obtain descriptors that compact the data into representation that are considerably more useful in the computation of descriptors.

So, Representation refers to the dealing of naming for segmented image dedicated to the object point.

Representation can be classified into following two types:

(1) Chain Codes

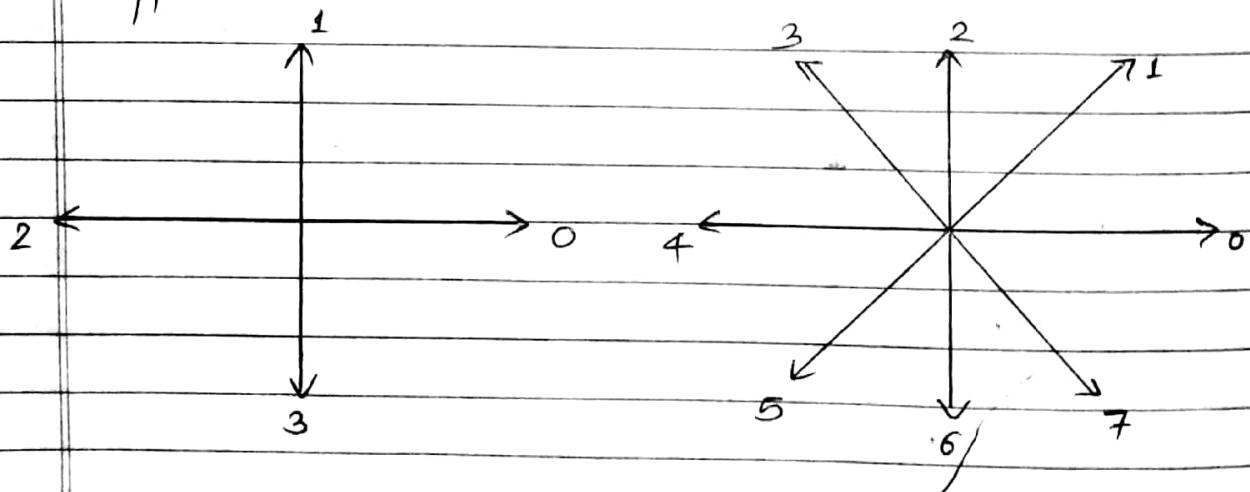
(2) Signatures

① Chain Codes :

(2015 fall)
 chain codes are used to represent a boundary by a connected sequence of straight line segments of specified length and direction.

Typically, this representation is based on 4 or 8 connectivity of the segments. The direction of each segment is coded by using a number.

Chain code on this scheme are referred as :



fig(a): Direction Number
for 4-directional
chain code

fig(b): Direction Number
for 8-directional
chain code

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

treating it as a circular sequence of direction numbers and redefining the starting point so that the resulting sequence of numbers is an integer magnitude.

We can normalize by using rotation i.e. 90° or 45° for the codes using the difference of the chain code.

Above figure shows the best example of 4 and 8-directional chain codes.

② Signatures :

(pre-university)

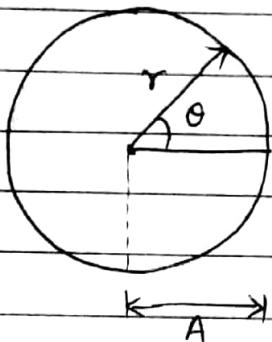
Signature is a 1-dimensional functional representation of a boundary and may be generated in various ways.

One of the simplest methods is plot the distance from a centroid point to the boundary as a function of angle.

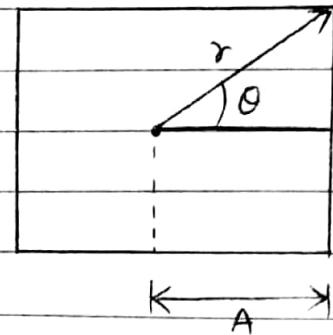
Signature generated by the approach just described are invariant to translation but they depends on rotation and scaling.

Normalization with respect to the rotation can be achieved by finding a way to select the same starting point to generate the signature.

The signature can be defined by using the following object:

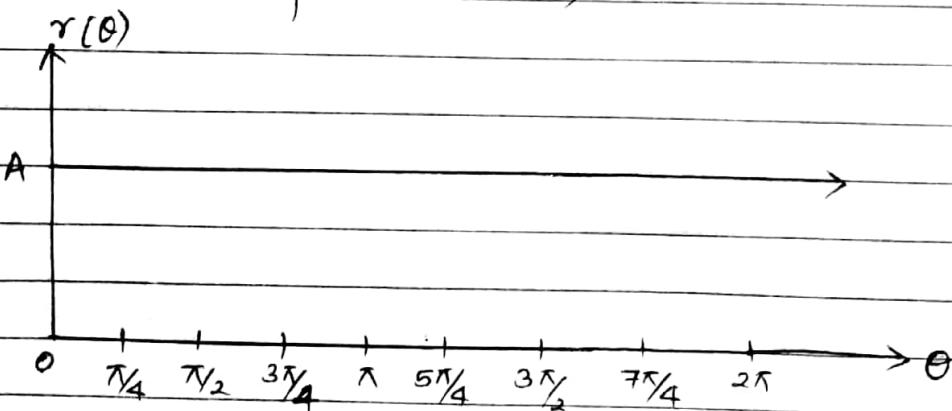


fig(a): circular object

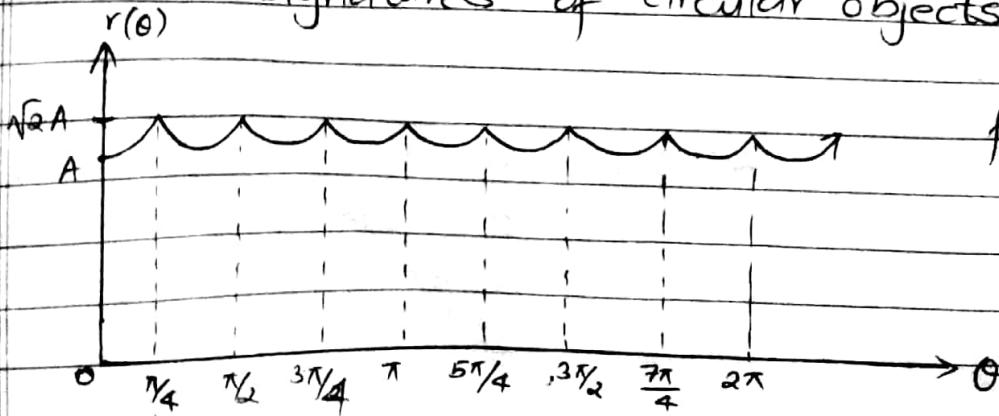


fig(b): square object

The corresponding angle signature can be defined as,

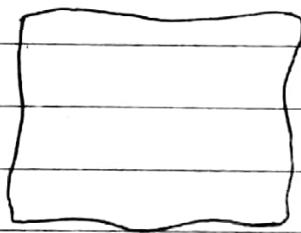


fig(c): Corresponding distance vs Angle signatures of circular objects

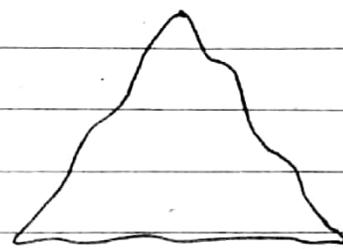


fig(d): Corresponding distance Vs Angle signature of square object

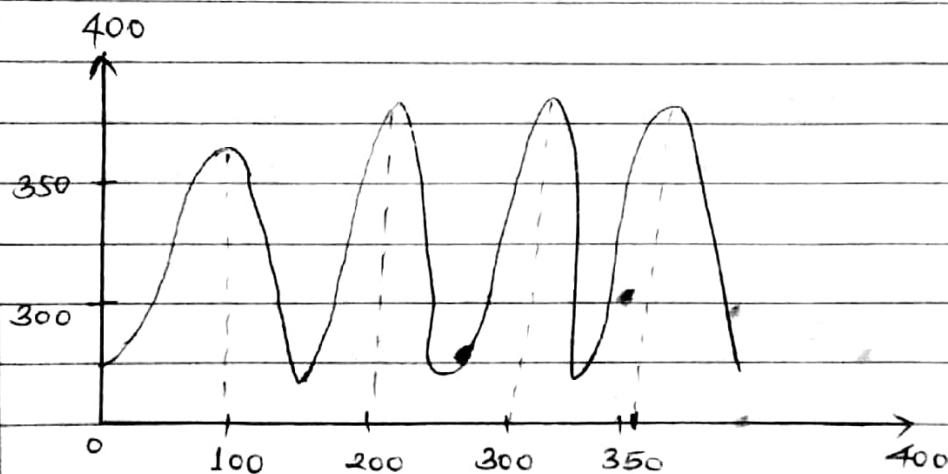
(A similar set of commands yeilded the plot.) simply counting the number of prominent peaks in the signature is sufficient to differentiate between the fundamental shape of the boundaries as shown in the figure below:



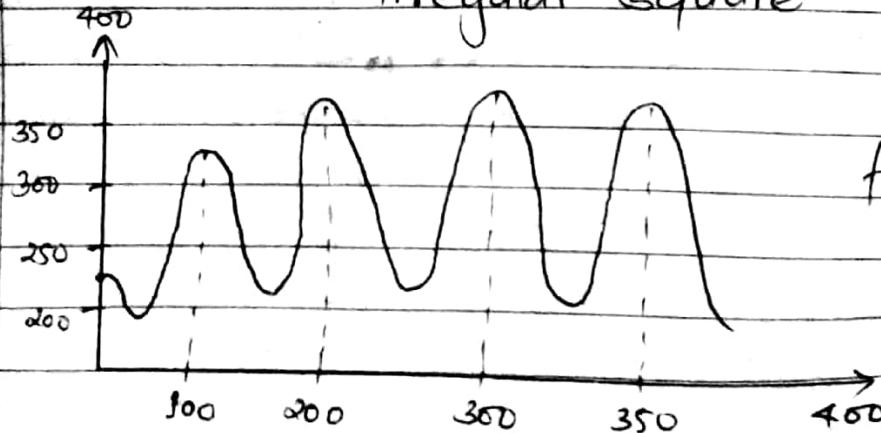
fig(a): Boundaries of an irregular square



fig(b): Boundaries of irregular triangle



fig(c): corresponding signature of an irregular square



fig(d): corresponding signature of irregular triangle

Introduction To descriptors:

Descriptors are also called features of an image that are useful when working with region boundaries.

Many of the descriptors are applicable to region and grouping of descriptors in the tool box. The length of a boundary is one of its simplest descriptors.

The length of a four connected boundary is defined as the number of pixels - 1.

Descriptor is usually categorized into two types based on the boundary:-

- (1) Shape Number
- (2) Fourier Descriptor

① Shape Number:

The shape number of a boundary generally based on ~~four directional chain codes~~ is defined as the first difference of smallest magnitude.

The ⁿ order of a shape number is defined as the number of digits in its representation.

Moreover, 'n' is even for a closed boundary and its value limits the number of possible different shapes.

classmate

Date _____

Page _____

Thus, the shape number of a boundary is given by the parameter and order of shape number (length).

The four directional chain codes can be made insensitive to the starting point by using the integer of minimum magnitude and made rotation that are multiples of 90° by using the first difference of the code.

Thus, shape numbers are insensitive to the starting point and rotation that are multiple of 90° .

Shape number is illustrated step by step as :



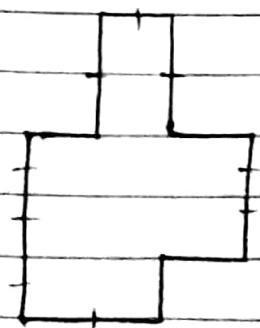
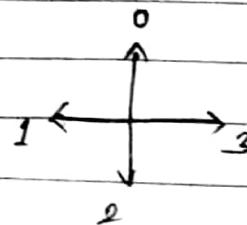
①



②



③



④

\therefore Chain code = 000030032232221211

Difference = 30003131131100310310

Shape no. = 000310330130031303

(fig: steps in the generation of shape number)

② Fourier Descriptor:

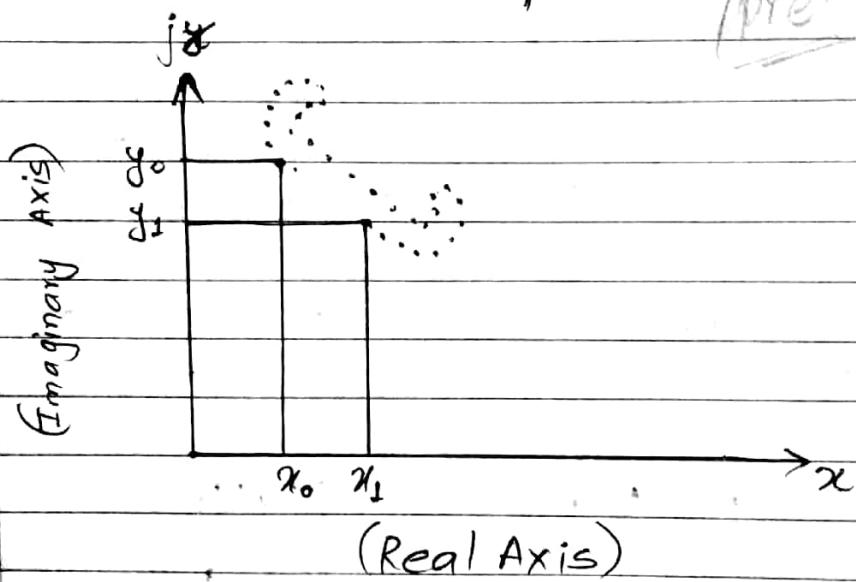


fig: A digital boundary and its representation as a complex sequence. point (x_0, y_0) is selected arbitrary or starting point. Point (x_1, y_1) is the next counter clockwise point in a sequence.

Fourier descriptors can be explained by using the following steps:

(i) Define the arbitrary point (x_0, y_0) and the co-ordinate point or pairs $(x_0, y_0), (x_1, y_1), (x_2, y_2) \dots (x_{k-1}, y_{k-1})$ are encountered in traversing the boundary in counter-clockwise direction.

(ii) These co-ordinates can be expressed in the form $x(k) = x_k$ and $y(k) = y_k$. The boundary can be represented as the sequence of co-ordinates as:

$$S(k) = [x(k) \cdot y(k)] \quad \text{for } k = 0, 1, 2, \dots, k-1$$

The each co-ordinate pair can be treated as a complex number so that $S(k) = x(k) + jy(k)$.

(iii) The discrete fourier transform of 1D sequence, $S(k)$ can be written as:

$$a(u) = \frac{1}{k} \sum_{k=0}^{k-1} S(k) \cdot e^{-j2\pi uk/k}$$

Where,

$$u = 0, 1, 2, \dots, k-1$$

(iv) The complex coefficient $a(u)$ are called the "Fourier Descriptors" of the boundary.

The inverse fourier transform of these coefficient is :

$$S(k) = \sum_{u=0}^{k-1} a(u) \cdot e^{j2\pi uk}$$

(V) Therefore,

for boundary descriptors, the Fourier Descriptors can be explained as Fourier coefficient which is defined as :

$$\hat{S}(k) = \sum_{u=0}^{P-1} a(u) \cdot e^{j2\pi uk/P}$$

Where,

P = Preceeding equation for $a(u)$

where $P = 0, 1, \dots, P-1$

$\hat{S}(k)$ = Approximation to $S(k)$

Chapter - 9

Object Recognition [3 hours]

classification or learning of an Image
or Decision Theoretic Methods:

Object Recognition
of classification

↓
(1) Unsupervised
classification

(No target or teacher
or sample signal)

↓
(2) supervised
classification
(Presence of target
or teacher or sample
signal)

↓
(1) statistical supervised
classification
(Bayesian classifier)

↓
(2) Distribution free
(Neighbourhood
classification)

(fig: Decision Theoretic Methods)

① Unsupervised classification:

A very common task concerning in image processing is classification which is done in order to use the image for mapping or further analysis.

In unsupervised classification learning, there is no teacher or target or sample signal in the same way, it is also computerized method without direction of analysis.

It means that the output are based on software without providing user signal or sample classes.

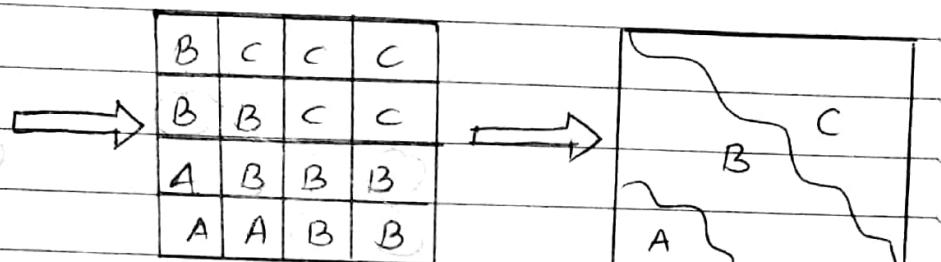
The computer uses the technique to determine which pixel is detected and what classes belongs together.

In this technique, the user or human does not interface to the unsupervised classification.

Cluster is one of the unpervised learning technique or classification.

For unsupervised classification, the figure below shows the technique of object recognition:

101	105	105	105
101	101	105	105
103	101	101	101
103	103	101	101



i/p image

class classification

fig: Example of unsupervised learning or classification using clustering technique

where,

A = Water

B = Land

C = Rock

② Supervised classification:

In supervised classification, it does not use the computer software to create the classes.

Here, the

the analysis identify the several areas in an image which represents different features. These known areas are referred to as Training site. The computer only do the assignment of pixel to the classes.

It is based on the ideal that, a user can select sample pixel or target signal in an image i.e. representative of specific classification.

It means that, the provided user signal or teacher signal can help to make the classification without using computer tool.

Supervised classification is categorized into following two types:

- (i) Statistical supervised classification
- (ii) Distribution free

(i) Statistical Supervised classification:

It is a type of supervised classification on which the hypothetical or probability of that classification is determined using presence of user signal or sample signal or teacher signal or target signal.

This still can be extended to classification of features or pattern where similar type of pattern are grouped into one class that same as the characteristics of user signal.

The term similar is in the containing objects is closeness to the features between sample signal and segmented region.

~~We~~ To justify this classification, we use Bayesian classifier.

- * one conditional probability
 - * two unconditional probability
- $$P(A|B) = \frac{P(AB)}{P(B)}$$
- $$P(B|A) = \frac{P(AB)}{P(A)}$$
- $$\Rightarrow P(B|A) \text{ and } P(A) = P(A|B) \cdot P(B)$$
- classmate
Page

(IMP)

Bayesian classifier:

A Bayesian classifier is simple and probabilistic classifier based on the Bayesian Theorem.

The Bayesian classifier classified the features to contribute the probability of hypothetical evidence.
i.e

$$P(H_i/E) = \frac{P(E/H_i) \cdot P(H_i)}{\sum_{n=1}^K P(E/H_n) \cdot P(H_n)}$$

Where,

$P(E/H_i)$ = The probability that we will observe evidence "E" given that Hypothesis H_i is true.

$P(H_i)$ = A prior probability that the Hypothesis H_i is true.

$P(H_i/E)$ = The probability that we will observe hypothesis H_i given that evidence "E" is true.

K = The no. of possible Hypothesis.

(ii) Distribution Free:

It is a type of supervised classification and donot require knowledge of any density function or p.d.f based on the heuristic information or algorithm.

The example of distribution free is nearest neighbourhood classification so that it is also called as Neighbourhood classification.

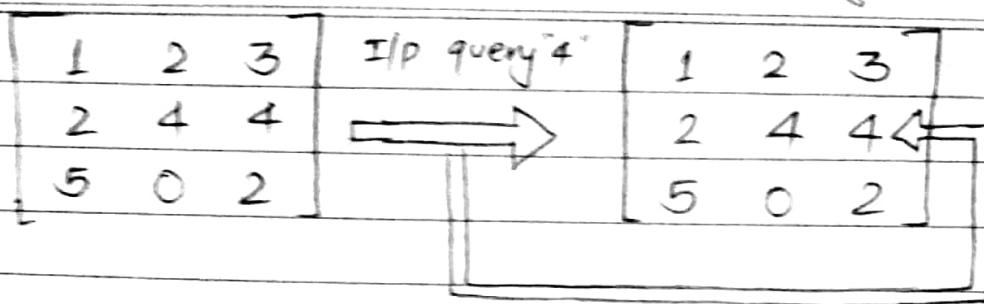
A very commonly used classification method is called Nearest Neighbourhood classification.

Therefore, it works as follows:

- (1) We create a database of any object for which we already known that the correct classification should be known.

When the system is given a query i.e new object classify the system. It finds the nearest neighbourhood of the query in the database.

- (2) Then, the system classify the query as belongs to the same classes as its nearest neighbours i.e database images.

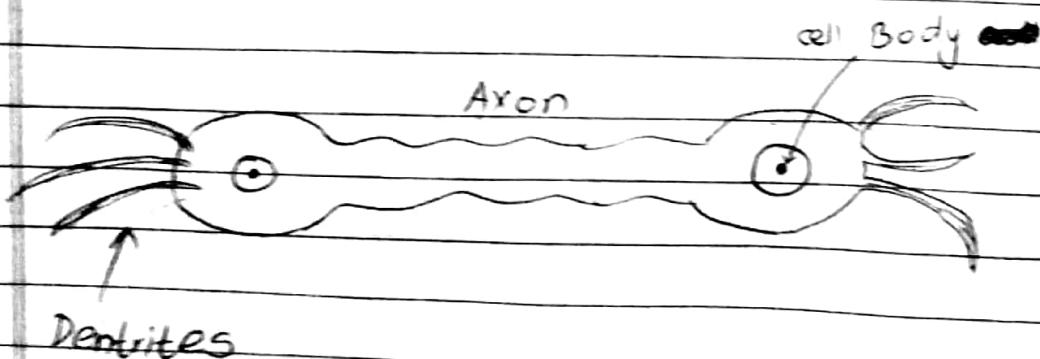


(fig: ~~recording~~ Recognizing digit using Neighbourhood Classifier)

- (3) Then, the system classify the query as an image of 4.
- (4) Now, the image is identified and classify by the distribution free supervised classification.

Here, query 4 is the target signal by the user.

Neuron, Neural Network (NN) & Artificial Neural Network (ANN) :



(fig: Neural components)

Neuron:

A neuron is a cell in the brain whose principle function is the collection, processing and dissemination of electrical signal.

The cell body is a part of the cell containing the nucleus and maintaining protein synthesis.

A neuron may have many dendrites which branch out tree like structure and receive signals from other neurons.

The axon conducts electric signal generated at the axon hillock.

Neural Network (NN) & Artificial Neural Network (ANN) :

Neural network is the branch of the field known as AI. A neural network can be considered as black-box i.e able to prediction of output pattern when it recognizes the given input pattern.

Artificial Neural Network (ANN) is the type of Neural Network which is inspired as artificially to show the human behaviour.

ANN shows the brain process information system.

Why use NN ?

NN is only used for following:

- (1) Self organization
- (2) Real time operation
- (3) Information is distributed over the entire network
- (4) Adaptive learning

Applications of Neural Network (NN):

- (i) Automotive, Automobile and Automatic Guidance system
- (ii) Aerospace and Aircraft component simulation.
- (iii) Banking cheque reader
- (iv) ~~Deep~~ Different facial recognition
- (v) Manufacturing product design (CAD)
- (vi) Telecommunication automated information
- (vii) Robotics control system

Disadvantages of Neural Network (NN):

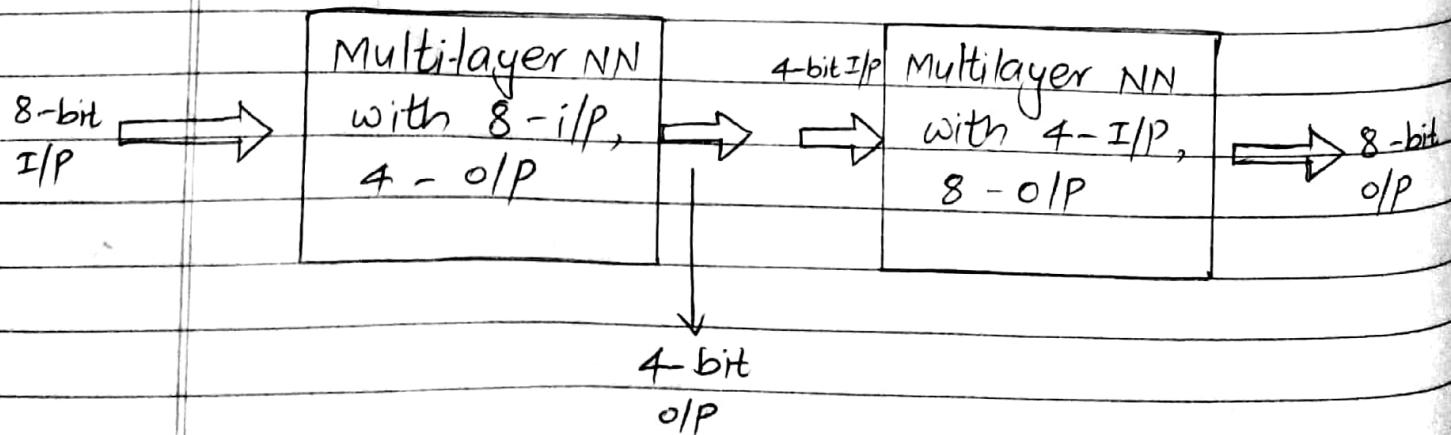
- (i) NN needs training to operate.
- (ii) The architecture of NN is different from the architecture of micro-processor.
- (iii) Requires high processing time for large neural network.
- (iv) Hard to setup programming.
- (v) Lack of well manpower.

~~(V. IMP)~~

Applications Of NN In Image Processing:

- (1) NN for Image compression
- (2) NN for Pattern Recognition
- (3) NN for Perceptron (Perception)

① NN for Image Compression:



(fig: Block Diagram of Image compression)

NN model have received much attention in many fields where high compression ratio is required.

Many NN approaches for image compression gives superior performance than the discrete traditional approach.

Digital Image generated with 8-bit may be reduced in size by feeding it to 8×4 multi-layer NN.

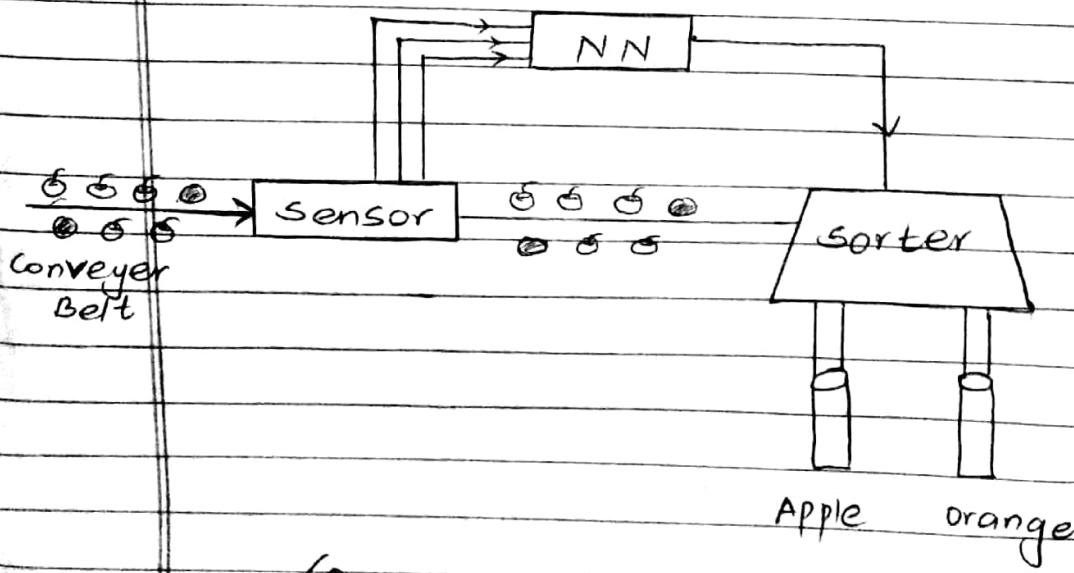
Neural network(NN) on the other side then process the compressed image by 4×8 multilayer NN for reconstruction.

(Imp)

②

NN for Pattern Recognition:

(2012 fall)



(fig: Block Diagram of Pattern Recognition (PR))

The NN is ideal tool for pattern recognition. Any recognition system needs to be trained to recognize different patterns.

NN is also simplification of human neuron network system. It is more likely to adapt the human way of solving the recognition problem than other techniques.

The design of NN system for PR starts from collecting data on each of objects that is to be recognized by the system.

for example: A dealer has a warehouse that stores variety of fruits and vegetables that are mixed together.

The dealer wants a machine that will sort the fruits according to their type.

The fruit is loaded on the conveyor belt and the fruit passes through the sensor which measures the shape, texture, color and weight properties.

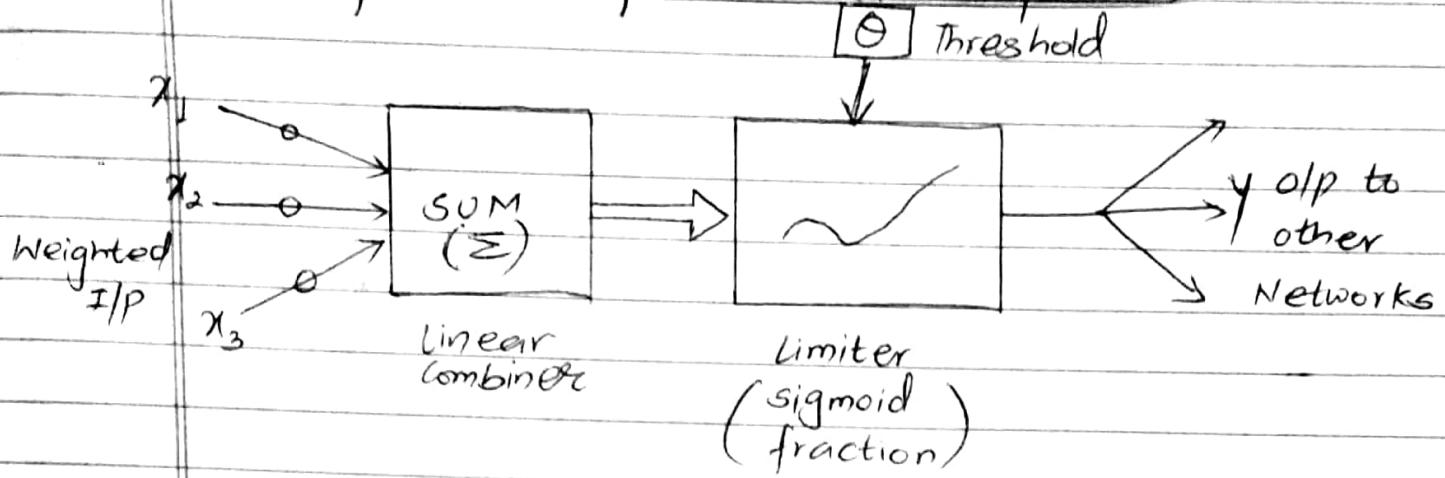
The output will be input to the purpose of NN to decided what ^{kind} of fruit is on the conveyor belt. So, fruit can be directed to the correct storeroom.

(2015 fall) L.9

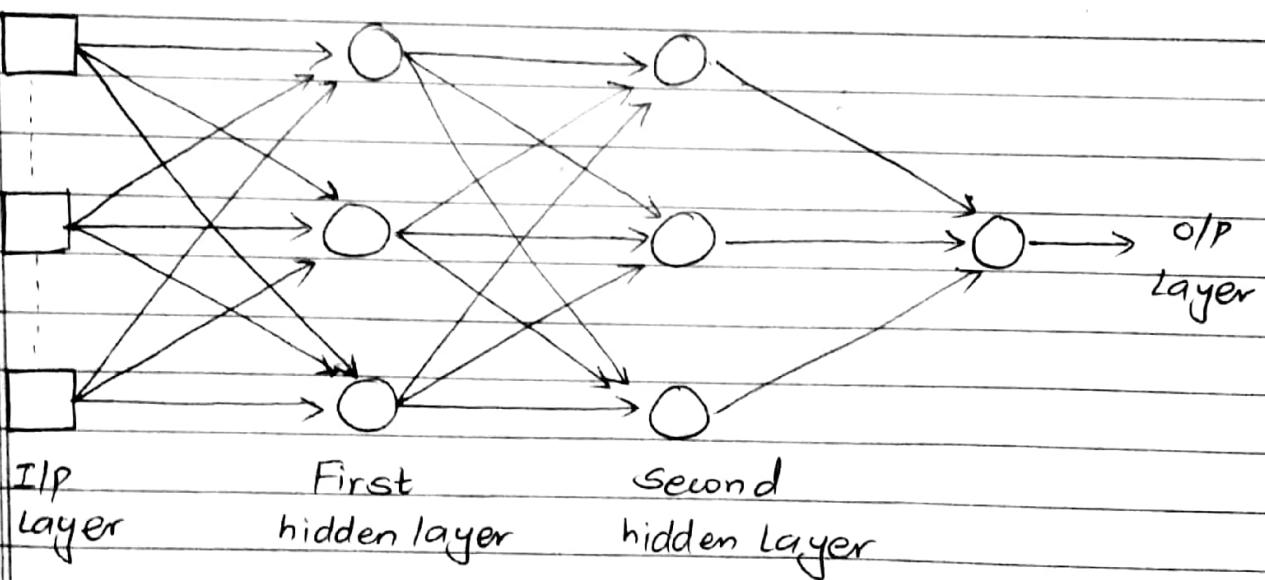
classmate

Date _____
Page _____

③ NN for Perceptron (Perception):



(fig: Block Diagram of Perceptron)



(fig: Formation of o/p by Perceptron)

It is one of the earliest neural network models.

Here, i/p's from one or more previous neurons are individually weighted and then summed.

The result is non-linearly

scaled between 0 and +1. And then, the result is passed to the neurons in the next layer for output result.

Several perceptions can be combined to form multi-layer perception (MLP). So MLP is a development from the simple perception in which extra hidden layers are added.

Here, more than one layer can be used.

Generally, connections are allowed from input layer, hidden layer and so on. At first i/p are feeded into the i/p layer and get multiplied by inter-connection weights as they are passed from the i/p layer to the first hidden layer.

After first layer then go to the second layer and finally data is multiplied by inter-connection weight and so on.

Back propagation never take the perception methods but at the time of sensing accuracy it is used.

(Assignment - I)

FFT (Fast Fourier Transform on the perspective of Image Enhancement)

pramodparanjuli@gmail.com

8m / D.C / 005E / E.S / NET Tech / Project
A / A- / A / A- / A / A

(3.82)

6th sem

Perception

- simplest kind of feed forward neural n/w
- the Model consist of linear combiner followed by an activation function
- The weighted sum of i/p's is applied to the activation function, which produces an o/p equal to +1, if its i/p is +ve & -1 if its i/p is -ve.

