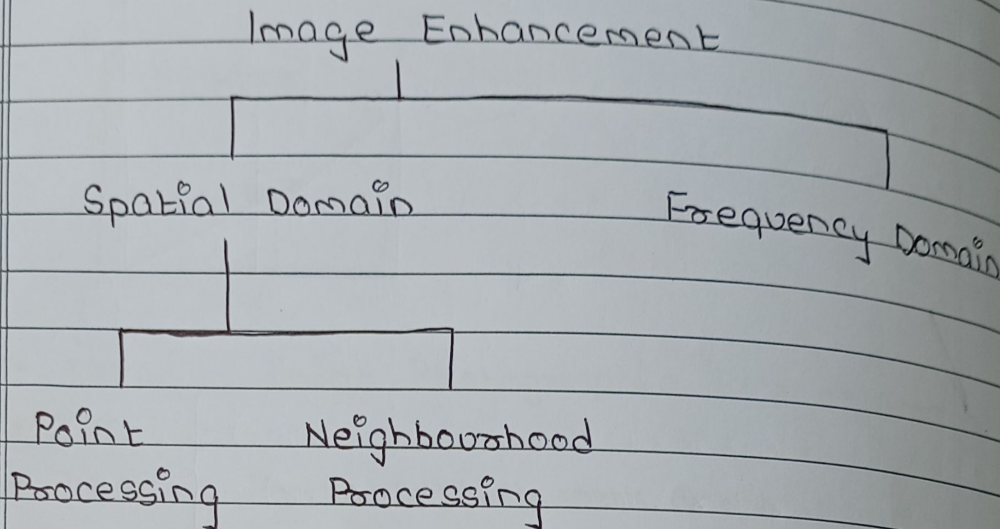
UNIT: 2 Date: / / Page no.:

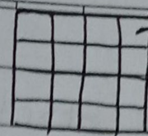
IMAGE ENHANCEMENT



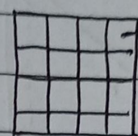
- Image Enhancement is subjective and cosmetic procedure ; no addition info. is added in image. It just improves quality of image.
- It is divided into 2 domains :
  - Spatial Domain
  - Frequency Domain

* Image Enhancement in Spatial Domain (Working with pixel value)

- Point Processing :
  - Works with ~~simple~~ single pixel
  - Eg: Digital Negative, Thresholding, Contrast Streching ; Grey-level, Bit-Plane slicing, Power law transformation, Dynamic Range Comp-ression

i/p Image : $F(x,y)$  'r' gray-level

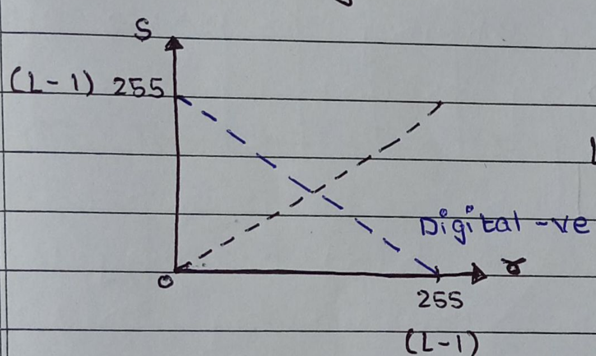
Operator (T)

o/p Image : $g(x,y)$  's' gray-level
(Modified)

$s = T(r)$

$\therefore g(x,y) = T[F(x,y)]$

$\therefore$ If $s = r$:
Identity Transformation



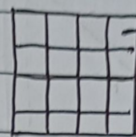
L : No. of grey-level

$\therefore$ If $s = (L-1) - r$:
Digital -ve

- Identity Transformation ($s = r$)
  (i) Digital Negative
  - ~~Image~~ Inverting the grey-level i.e. black in original will be now looking white.
  - Function : $s = (L-1) - r$; L = No. of grey-leve
  - If $r = 0$, $s = (L-1)$
    $r = (L-1)$, $s = 0$
  - Eg: X-ray image

i/p Image : $f(x,y)$  → 'r' gray-level

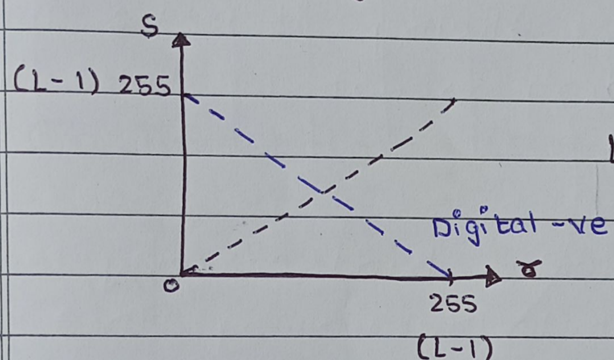
↓ Operator (T)

o/p Image : $g(x,y)$  → 's' gray-level
(Modified)

$s = T(r)$

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

$\therefore$ If $s = r$:
Identity Transformation



L: No. of grey-level

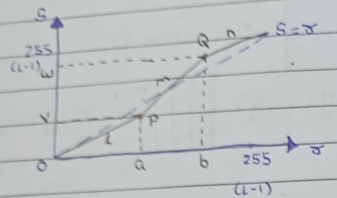
$\therefore$ If $s = (L-1) - r$:
Digital -ve

- Identity Transformation ($s = r$)

(i) Digital Negative
- ~~Image~~ Inverting the grey-level i.e. black in original will be now looking white.
- Function : $s = (L-1) - r$; L = No. of grey-lev
- If $r = 0$, $s = (L-1)$
  $r = (L-1)$, $s = 0$
- Eg: X-ray image

Page no.: ____/
Date: __/__/____/

(ii) Contrast Stretching



$s = T(r)$

$$\therefore s = \begin{cases} l.r & ; \quad 0 \le r \le a \\ m(r-a)+v & ; \quad a \le r \le b \\ n(r-b)+w & ; \quad b \le r \le (L-1) \end{cases}$$

$P(a, v)$

$\therefore$ ~~by~~ $l = \dfrac{v-0}{a-0}$

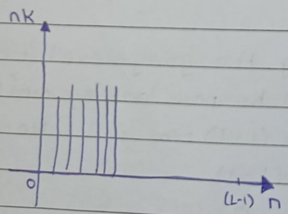
$$\boxed{\therefore v = la}$$

$Q(b, w)$

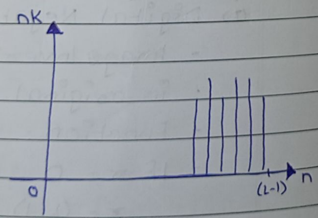
$m = \dfrac{w-v}{b-a}$

$\therefore w = m(b-a)+v$

$$\boxed{\therefore w = m(b-a)+la}$$



Towards left
Dark Histogram



Towards Right
Light Histogram

Page no.: ____/
Date: __/__/____/

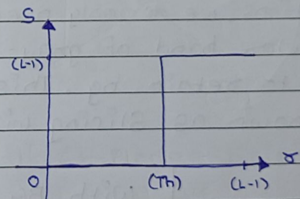
l, m, n are slopes and are assigned values such that,

$$l, n < 1$$
$$m > 1$$

so that, dynamic range of image can be increased.

We get low contrast images due to poor illumination ~~or~~ wrong setting of lens aperture. Contrast can be increased by making dark portion darker and light portion lighter, increasing dynamic range.

(iii) Thresholding (Extreme Contrast Stretching

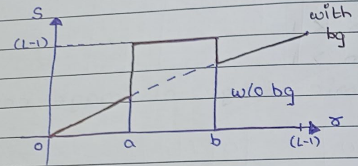


$$s = \begin{cases} 0 & ; \quad r \le (Th) \\ 1 & ; \quad \text{else} \end{cases}$$

$(l = n = 0, \ m = \infty)$

In thresholding, we make first and last slope of contrast stretching zero and center slope is increased (∞). Maximum contrast is achieved as it has only two gray level B/W.

(iv) Grey-Level Slicing (Intensity Slicing)



Application: X-ray and CT image

Thresholding splits into two parts, but sometimes, we need to highlight specific range of grey-levels. In such cases, we use transformation known as Grey-level slicing. It looks similar to Thresholding except we take band of grey levels.

In some applications, we ~~do~~ not only need to enhance particular band of gray-level but also want to retain bg. This technique is known as Slicing with bg.

Without BG:	With BG:
$s = \begin{cases} (L-1) & ; a \le r \le b \\ 0 & ; \text{else} \end{cases}$	$s = \begin{cases} (L-1) & ; a \le r \le b \\ r & ; \text{else} \end{cases}$

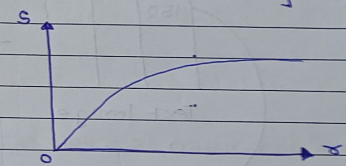
(v) Dynamic Range Compression (Log Transformat

Sometimes, dynamic range of image exceeds capacity of displayed devices. Technique to compress range, is called Dynamic

Log is excellent compressor. Dynamic Range Compression can be achieved by

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

↑ normalizing factor



(vi) Bit-Plane Slicing:

In this technique, we find out contri made by each bit in final image. MSB contains significant data. Bi Plane slicing can be used in comp ~~wit~~ where we use MSB and remove LSB.

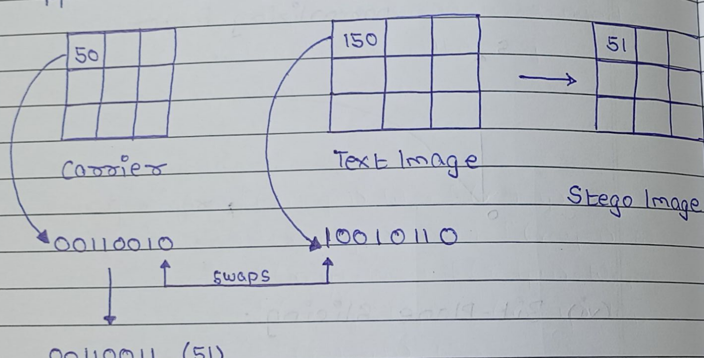
Eg:

$$\begin{bmatrix} 0 & 7 & 2 \\ 2 & 3 & 2 \\ 1 & 5 & 4 \end{bmatrix} \xrightarrow{\text{Binary}} \begin{bmatrix} 000 & 111 & 0010 \\ 0010 & 011 & 010 \\ 001 & 101 & 000 \end{bmatrix}$$

MSB / Middle / LSB

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

Application: Steganography



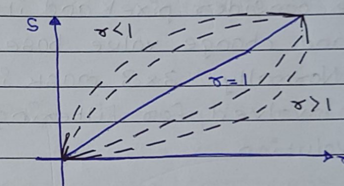
Bit-Plane Slicing can be used for steganography, which is art of hiding info., where secret data is hide behind carrier image.
LSB of carrier image is replaced with MSB of secret data.
Stego image looks visually same as carrier image. If we replace more than 2 MSBs and LSBs, we get super-imposed image called as Watermark.

(vii) Power Law Transformation

Non-linearity encountered during image capturing, can be corrected using Power-Law Transformation a.k.a. Gamma Correction.

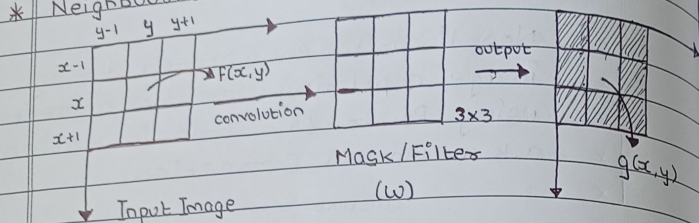
constant

$$s = c \cdot r^{\gamma}$$ ← Gamma (Correction Factor)



It can be used to increase dynamic range of image, By changing gamma values, we get series of ~~transform~~ curve

* Neighbourhood Processing

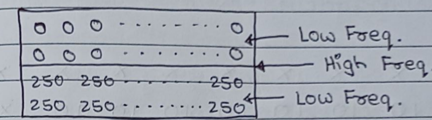


This is image enhancement in spatial domain. In this we consider pixel and its neighbour, and change value based on neighbours. Normally, 3x3 mask is used. Operations involved for filtering involve convolution.

Here,

$$g(x,y) = f(x-1,y-1).w_{11} + f(x-1,y).w_{12} + f(x-1,y+1).w_{13} + f(x,y-1).w_{21} + f(x,y).w_{22} + f(x,y+1).w_{23} + f(x+1,y-1).w_{31} + f(x+1,y).w_{32} + f(x+1,y+1).w_{33}$$

Many applications are possible using Neighbourhood Processing.
We can have mask 5x5, 7x7, etc. Multiply each component of mask with corresponding value of image, add them up and place result at center, shift mask to right and then down, so as to cover entire image.



High Freq. : Edge, Noise
Low Freq. : Background

Low Pass Filter (Smoothing) : HF get blurred
High — " — (Sharpening) : LF get blurred (or removed)

- Low Pass Filtering (Smoothing)

It removes HF component. Normally, used to blur the noise.
Consider image and consider 8x8 image

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50

Filter : $\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = w$ (Mask value

Output:

X	X	X	X	X	X	X	X	X	X
X	10	10	10	10	10	10	10	10	X
X	10	10	10	10	10	10	10	10	X
X	10	10	10	10	10	10	10	10	X
X	23.3	23.3	23.3	23.3	23.3	23.3	23.3	23.3	X
X	36.6	36.6	36.6	36.6	36.6	36.6	36.6	36.6	X
X	50	50	50	50	50	50	50	50	X
X	50	50	50	50	50	50	50	50	X
X	X	X	X	X	X	X	X	X	X

If image has Gaussian noise, it can be removed using LP averaging F.
Consider this eg. LF regions have remain unchanged. Sharp edge b/w 10 and 50 has become blur i.e. From 10 to 23.3 to 36.6 to 50. We can have LPAF of order 5x5. Bigger the mask, more is blurring.

$$\frac{1}{25}\begin{matrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{matrix}$$

- Low Pass Median Filtering
(Salt and Pepper Noise)

To remove salt and pepper noise, we work with non-linear filter, a.k.a. order statistic filter.
There is no mask. We consider empty mask, put on an image, arrange pixels in sorted order to calculate median, which is used in output image.

Example:

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	$\boxed{250}$	10	10	10	10	10	10
10	10	10	10	10	10	10	10
50	50	50	50	$\boxed{250}$	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50
50	50	50	50	50	50	50	50

$\big\downarrow$ Median Filtering

X	X	X	X	X	X	X	X
X	10	10	10	10	10	10	X
X	10	10	10	10	10	10	X
X	10	10	10	10	10	10	X
X	50	50	50	50	50	50	X
X	50	50	50	50	50	50	X
X	50	50	50	50	50	50	X
X	X	X	X	X	X	X	X

250 as Noise got removed, using median filter. If we have image with S and P noise and if we appling averaging filter, noise will spread on entire image, resulting in detoriation of image.

- High Pass Filtering (Sharpening)

It is used to enhance edges and removes background.
Consider 8x8 image.

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

Filter:

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

Addition of all values = 0

Output:

X	X	X	X	X	X	X	X
X	0	0	0	0	0	0	X
X	0	0	0	0	0	0	X
X	-270	-270	-270	-270	-270	-270	X
X	270	270	270	270	270	270	X
X	0	0	0	0	0	0	X
X	0	0	0	0	0	0	X
X	X	X	X	X	X	X	X

We can consider mask as

$$\frac{1}{9}\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

so that values are in range. It still can not be -ve, but mod is not possible. Hence -270 is dark as 0 and edge is enhanced in output.
1/9 is just scaling factor.

Page no.: ___ Date: ___

- High Boost Filtering

In some application, we want to enhance edge as well as retain bg. To pass some part of bg, we multiply original image with multiplicative factor A.

$$HP = O - LP$$

$$HB = A^{*}O - LP$$

$$\therefore HB - HP = AO - LP - O + LP$$
$$= (A-1)O$$

$$\boxed{\therefore HB = (A-1)O + HP}$$

-1	-1	-1
-1	X	-1
-1	-1	-1

When $A=1$,
$HB = HP$

$A = 1.1$
$X = 9A - 1$
$\therefore X = 8.9$

~~When~~ $\therefore A > 1$

Page no.: ___ Date: ___

10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
10	10	10	10	10	10	10	10
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100
100	100	100	100	100	100	100	100

X	X	X	X	X	X	X	X
X	9	9	9	9	9	9	X
X	9	9	9	9	9	9	X
X	-261						X
X	360						X
X	90	90	90	90	90	90	X
X	90	90	90	90	90	90	X
X	X	X	X	X	X	X	X

We can see that bg is not completely removed, some part of bg is retained as we get value as 9, 90.

Q missing.

Page no.: ____
Date: __/__/____

Q) Find digital negative of given image which is represented using 8 bits.

121	205	217	156	151
139	127	157	117	125
252	117	236	138	142
227	182	178	197	242

→ No. of bits = 8

$\therefore L = 2^8 = 256$

$\therefore S = T(r)$

$\therefore S = (L-1) - r$

$\therefore S = 255 - r$

$\therefore$ Digital Neg

134	50	38	99	104
116	128	98	138	130
3	138	19	117	113
28	73	77	58	13

Page no.: ____
Date: __/__/____

Q) Perform intensity slicing for 3 bit image where a = 3, b = 5. Draw modified image w/o bg.

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

→ $L = 2^3 = 8$

$\therefore$ Without bg.

0	0	0	0	0
0	7	7	7	0
0	0	0	0	0
0	0	0	7	0
0	7	0	0	0

$\therefore$ With bg.

2	1	2	2	1
2	7	7	7	2
6	2	7	6	0
2	6	6	7	1
0	7	2	2	1

graph missing.

Page no.: ____ Date: __/__/____

Q) What would happen to dynamic range if all slopes of contrast slopes are less than 1. (~~corder~~ Considering original image has level between 0 to 10 and $a=4,\ b=8$)($l=0.2,\ m=0.5,\ n=0.2$)

$a=4,\ b=8$

$s = lr \quad ; \quad 0 \le r \le 4$

$s = m(r-a) + v \quad ; \quad 5 \le r \le 8$

$s = n(r-b) + w \quad ; \quad 9 \le r \le 10$

r	s
0	0
1	0.2
2	0.4
3	0.6
4	0.8
5	1.3
6	1.8
7	2.3
8	2.8
9	3
10	3.2

$v = la$
$= 0.8$

$w = m(b-a) + v$
$= 2.8$

$\therefore r \Rightarrow 0 \rightarrow 10$

$\therefore s \Rightarrow 0 \rightarrow 3.2$

$\therefore$ Dynamic range has decreased.

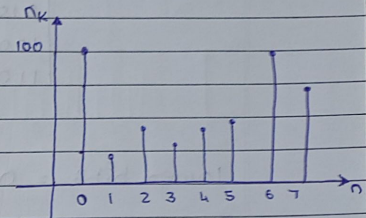
We notice that dynamic range becomes 0 to 3.2 for modified image.

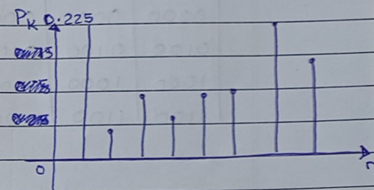
Page no.: ____ Date: __/__/____

* Histogram Modelling

n	n_k	P_k
0	100	0.225
1	20	0.045
2	40	0.090
3	30	0.067
4	40	0.090
5	45	0.101
6	100	0.225
7	70	0.157

$S = 445$





Q) What would be effect of setting zero lower order bits on histogram of image? What would be effect if we set higher order bits to 0.

0	1	2	3
4	5	6	7
8	9	10	11
12	13	14	15

→ Original :

0000	0001	0010	0011
0100	0101	0110	0111
1000	1001	1010	1011
1100	1101	1110	1111

Lower bits = 0

0000	0000	0000	0000
0100	0100	0100	0100
1000	1000	1000	1000
1100	1100	1100	1100

=

0	0	0	0
4	4	4	4
8	8	8	8
12	12	12	12

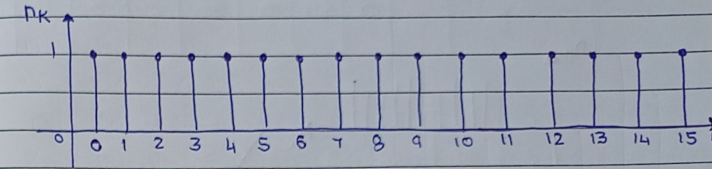
Higher bits = 0

0000	0001	0010	0011
0000	0001	0010	0011
0000	0001	0010	0011
0000	0001	0010	0011

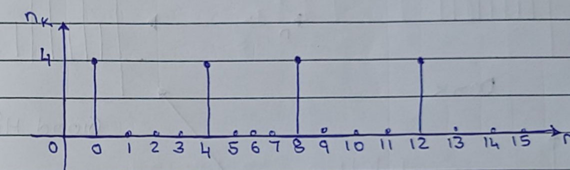
=

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

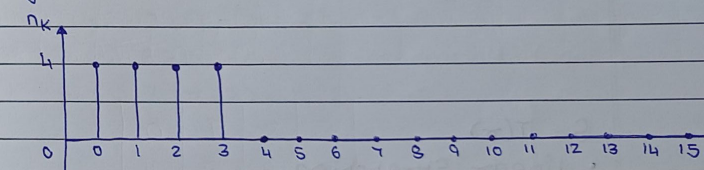
Input Histogram:



Lower bits:



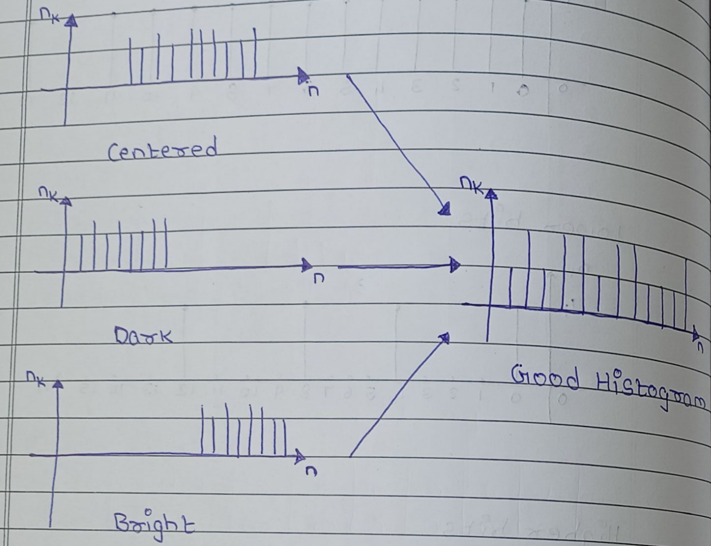
Higher bits



For Lower, variation in level is this. No. of levels are reduces, freq. is higher.

In Higher, it is darker compared to input histogram.

* Histogram Stretching



$S = T(r)$

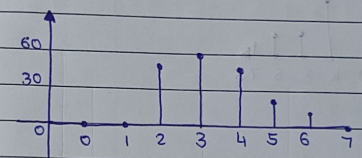
- Linear Stretching

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

Q) Performing histogram stretching for given image.

n	0	1	2	3	4	5	6	7
nk	0	0	50	60	50	20	10	0

→ Input Histogram:



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

$$= \frac{7 - 0}{6 - 2} (r - 2) + 0$$

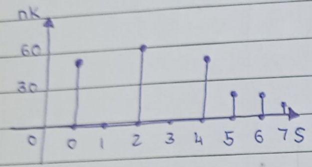
$$S = \frac{7}{4} (r - 2)$$

∴

r	s	s	nk
2	0	0	50
3	1.75	2	60
4	3.5	4	50
5	5.25	5	20
6	7	7	10

New s	0	1	2	3	4	5	6	7
nk	50	0	60	0	50	20	0	10

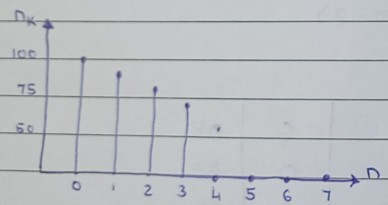
Output Histogram:



Q) Perform histogram stretching.

r	0	1	2	3	4	5	6	7
n_k	100	90	85	70	0	0	0	0

→ Input Histogram:



∴ Now,

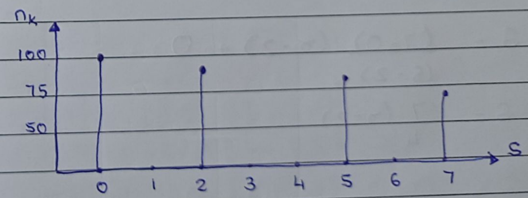
$$s = \frac{(s_{max} - s_{min})}{(r_{max} - r_{min})}(r - r_{min}) + s_{min}$$

$$= 7/3\,(r - 0) + 0$$

r	$s = 7/3 \cdot r$		$\approx s$	n_k
0	0	= 0	0	100
1	7/3	= 2.3	2	90
2	14/3	= 4.6	5	85
3	7	= 7	7	70
4	28/3	= 9.3		
5	35/3	= 11.6		
6	14	= 14		
7	49/3	= 16.3		

New s	0	1	2	3	4	5	6	7
n_k	100	0	90	0	0	85	0	70

Output Histogram:

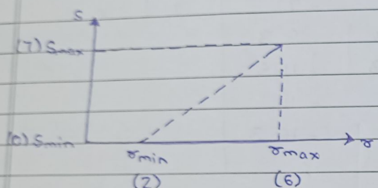


We notice that after linear stretching dark histogram is converted into evenly spaced histogram.

Page no.: ____/ Date: ___/___/_____

Q) For given slope formation and image, draw freq. table. Also sketch output histogram and modified image.

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



→ $r_{min} = 2$, $r_{max} = 6$

$s_{min} = 0$, $s_{max} = 7$

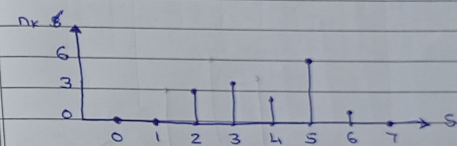
$\therefore s = \frac{(7-0)}{(6-2)}(r-2) + 0$

$s = \frac{7}{4}(r-2)$

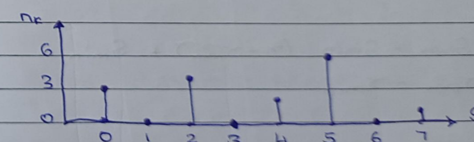
Page no.: ____/ Date: ___/___/_____

r	s	s'	n_k
2	0	0	3
3	1.75	2	4
4	3.5	4	2
5	5.25	5	6
6	7	7	1

s'	0	1	2	3	4	5	6	7
n_k	0	0	3	4	2	6	1	0



s'	0	1	2	3	4	5	6	7
n_k	3	0	4	0	2	6	0	1



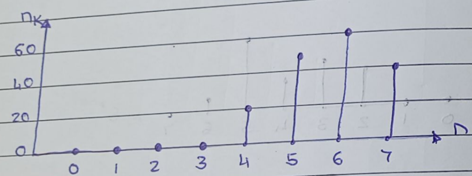
0	2	4	0
5	5	0	4
2	7	2	5
5	2	5	5

Q) Consider bright histogram, and show how histogram stretching helps to improve dynamic range.

→ Consider below freq. table for bright histogram f of ~~an~~ 3-bit image.

r	0	1	2	3	4	5	6	7	8
n_k	0	0	0	0	20	50	60	40	

∴ Input Histogram would look like:



Now, applying histogram streching,

$S_{min} = 0$, $S_{max} = 7$

$r_{min} = 4$, $r_{max} = 7$

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

∴ Then

$$S = \frac{7-0}{7-4} (r-4) + 0$$

$$\therefore S = \frac{7}{3} (r-4)$$

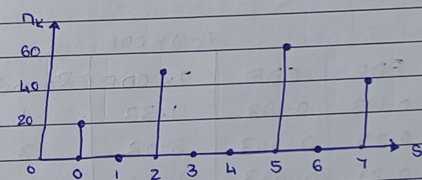
∴ Now,

n_k	r	s'	$\approx s'$
20	4	0	0
50	5	2.3	2
60	6	4.6	5
40	7	7	7

∴ Modified freq table is,

s'	0	1	2	3	4	5	6	7
n_k	20	0	50	0	0	60	0	40

∴ Thus, resulting output histogram would be,



Thus,
∴ ~~Since~~ dynamic range of input = 4-7 has increased to 0-7.

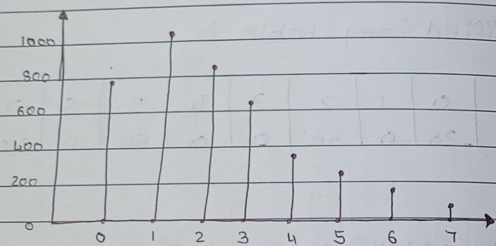
∴ Thus, bright histogram is converted into evenly-spaced histogram.

* Histogram Equalization

Q) Perform histogram equalization.

r	0	1	2	3	4	5	6	7
n_k	790	1023	850	656	329	245	122	81

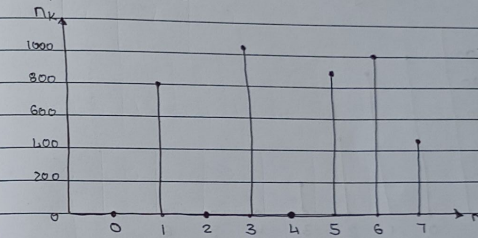
→ Input Histogram :



r	n_k	PDF	CDF	$(L-1) \times$ CDF $7 \times$ CDF	Round up (r')	n_k'
0	790	0.19	0.19	1.33	1	790
1	1023	0.25	0.44	3.08	3	1023
2	850	0.21	0.65	4.55	5	850
3	656	0.16	0.81	5.67	6	985
4	329	0.08	0.89	6.23	6	
5	245	0.06	0.95	6.65	7	448
6	122	0.03	0.98	6.86	7	
7	81	0.02	1	7	7	
	4096					

r'	0	1	2	3	4	5	6	7
n_k'	0	790	0	1023	0	850	985	448

Output Histogram :

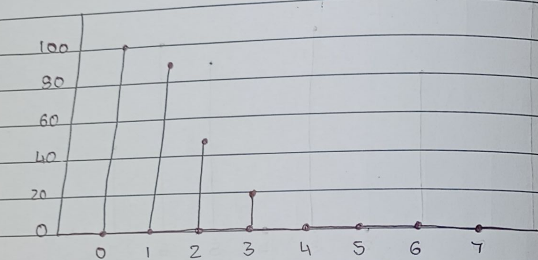


Dark histogram is converted into evenly-spaced histogram
We better distribution of levels in output histogram

Q) Perform histogram

n	0	1	2	3	4	5	6	7
n_k	100	90	50	20	0	0	0	0

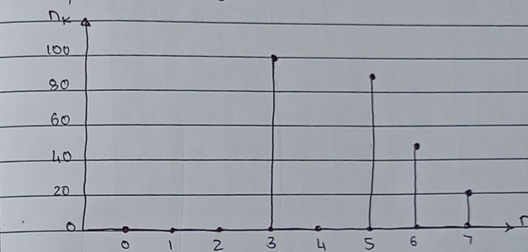
→ Input Histogram:



n	n_k	PDF	CDF	(L-1) × CDF	n'	n_k'
0	100	0.38	0.38	2.66	3	100
1	90	0.35	0.73	5.11	5	90
2	50	0.19	0.92	6.44	6	50
3	20	0.08	1	7	7	
4	0	0	1	7	7	
5	0	0	1	7	7	20
6	0	0	1	7	7	
7	0	0	1	7	7	
	260					

n'	0	1	2	3	4	5	6	7
n_k'	0	0	0	100	0	90	50	20

Output Histogram:

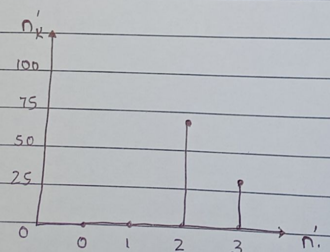


Q)

n	0	1	2	3
n_k	70	20	7	3

n	n_k	PDF	CDF	3×CDF	n'	n'_k
0	70	0.7	0.7	2.1	2	70
1	20	0.2	0.9	2.7	3	30
2	7	0.07	0.93	2.91	3	
3	3	0.03	0.03	3	3	
	100					

n'	0	1	2	3
n'_k	0	0	70	30



n'	n'_k	PDF	CDF	3× CDF	n''	n''_k
0	0	0	0	0	0	0
1	0	0	0	0	0	0
2	70	0.7	0.7	2.1	2	70
3	30	0.3	1	3	3	30
	100					

If we equalize an equalized histogram, it gives the same result

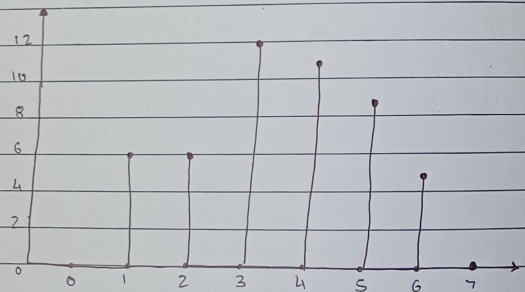
Page no.: ____ Date: __/__/____

Q) Plot histogram of given image. Perform histogram equalization ~~Plot~~ Draw histogram modified image

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

→ Freq. Table

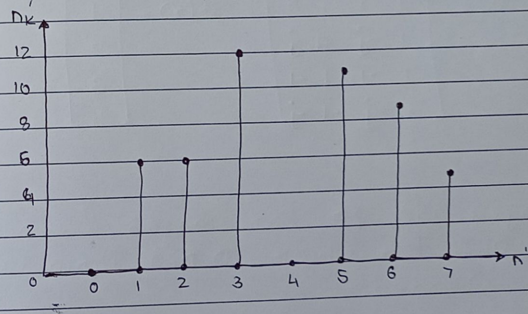
n	0	1	2	3	4	5	6	7
n_k	0	6	6	12	11	9	5	0



Page no.: ____ Date: __/__/____

n	n_k	PDF	CDF	$7\times$CDF	n'	n_k'
0	0	0	0	0	0	0
1	6	0.12	0.12	0.84	1	6
2	6	0.12	0.24	0.84	2	6
3	12	0.24	0.48	1.68	3	12
4	11	0.22	0.70	4.9	5	11
5	9	0.18	0.88	6.16	6	9
6	5	0.10	0.98	7	7	] 5
7	0	0	0	7	7	
	49					

n'	0	1	2	3	4	5	6	7
n_k'	0	6	6	12	0	11	9	5



Page no.: Date: / /

Modified Image :

1	1	3	7	5	3	1
6	7	3	5	6	6	3
3	5	3	2	5	3	6
6	6	5	1	3	2	3
1	3	5	6	7	6	5
5	7	5	1	2	2	3
2	5	7	3	2	5	6

IMAGE ENHANCEMENT IN FREQUENCY DOMAIN

* Fourier Transformation (FT):

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

↑ Discrete F.T.

$$\underset{\text{Time}}{x(n)} \xrightarrow{\text{DFT}} \underset{\text{Freq}}{X(k)}$$

$(u = 0 \rightarrow N-1)$

$$\therefore F(u) = \sum_{x=0}^{N-1} f(x) \cdot e^{-j2\pi ux/N} \rightarrow W$$

$\therefore F(u) = W \cdot f(x)$

$N \rightarrow$ Length of signal

$f(x) = \{1, 2, 3, 4\}$

$\rightarrow N = 4$

$\therefore$ For $N = 4$,

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}, \quad f(x) = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

Images are 2D. For images, we need 2D DFT, for which 1D DFT formula can be extended.

$$\underset{\substack{F(x) \\ \text{Time}}}{x(n)} \xrightarrow{\text{DFT}} \underset{\substack{F(x) \\ \text{Freq.}}}{X(k)}$$

To calculate 1D DFT, we can use

$$F(u) = W \cdot F(x)$$

where,

W is Twiddle Matrix

$F(x)$ is Signal

$F(u)$ is DFT of signal

$\therefore$ For $N = 4$,

$$W = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}, \quad f(x) = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

$\therefore F(u) = W \cdot F(x)$

$$\therefore F(u) = \begin{bmatrix} 10 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix}$$

$\therefore$ Now, for $\underset{\text{freq.}}{F(u)} \underset{\text{IDFT}}{\xrightarrow{\text{DFT}^{-1}}} \underset{\text{time}}{f(x)}$

$$f(x) = \frac{1}{N} \sum_{u=0}^{N-1} F(u) \cdot e^{+j2\pi xu/N}$$

$x : 0 \rightarrow N-1$

Page no.: ___ Date: ___

For 2D - DFT

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

$x \rightarrow 0 : M-1$

$y \rightarrow 0 : N-1$

$F(u,v)$ ↑ Freq.

$f(x,y)$ ↑ Time

Separability Property : 2D-DFT = 2× 1D-DFT

$$F(u,v) = \sum_{x=0}^{M-1} e^{-j2\pi ux/M} \cdot \underbrace{\sum_{y=0}^{N-1} F(x,y)\, e^{-j2\pi vy/N}}$$

$$= \underbrace{\sum_{x=0}^{M-1} e^{-j2\pi ux/M} \cdot F'(x,v)}$$

$\therefore F(u,v) = F(u,v)$

Page no.: ___ Date: ___

Q) For given image, find 2D - DFT.

0	1	2	1
1	2	3	2
2	3	4	3
1	2	3	2

→ Doing DFT row-wise,

$X[k] = W \cdot x(n)$

For $x(n) = \{0, 1, 2, 1\}$

$$\therefore X[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ -2 \\ 0 \\ -2 \end{bmatrix}$$

For $x(n) = \{1, 2, 3, 2\}$

$$X[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \\ 2 \end{bmatrix} = \begin{bmatrix} 8 \\ -2 \\ 0 \\ -2 \end{bmatrix}$$

$\therefore$ For $x(n) = \{2, 3, 4, 3\}$

$$X[k] = \begin{bmatrix} 12 \\ -2 \\ 0 \\ -2 \end{bmatrix}$$

$\therefore$ For $x(n) = \{1, 2, 3, 2\}$

$$X[k] = \begin{bmatrix} 8 \\ -2 \\ 0 \\ -2 \end{bmatrix}$$

∴ Modified image (Intermediate Image)

4	-2	0	-2
8	-2	0	-2
12	-2	0	-2
8	-2	0	-2

∴ Doing DFT column-wise,

$x(n) = \{4, 8, 12, 8\}$

$$X[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 4 \\ 8 \\ 12 \\ 8 \end{bmatrix} = \begin{bmatrix} 32 \\ -8 \\ 0 \\ -8 \end{bmatrix}$$

$x(n) = \{-2, -2, -2, -2\}$

$$X[k] = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} -2 \\ -2 \\ -2 \\ -2 \end{bmatrix} = \begin{bmatrix} -8 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$x(n) = \{0, 0, 0, 0\}$

$$X[k] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$x(n) = \{-2, -2, -2, -2\}$

$$X[k] = \begin{bmatrix} -8 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

∴ Final image:

32	-8	0	-8
-8	0	0	0
0	0	0	0
-8	0	0	0

* Translation Property (Shifting

- If $F(x,y)$ is multiplied by an exponentia

$$e^{j2\pi[u_0x+v_0y/N]}$$

then original F.T. $F(u,v)$, gets shifted b

$$F(u-u_0, v-v_0)$$

$$F'(u',v') = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) \cdot e^{-j2\pi(ux+vy/N)}$$

$$= \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) \cdot e^{j2\pi(u_0x+v_0y/N)} \cdot e^{-j2\pi(ux+v}$$

$$F'(u',v') = \sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y) \cdot e^{-j2\pi((u-u_0)x+(v-v_0)y/N)}$$

$$\boxed{F'(u',v) = F(u-u_0, v-v_0)}$$

$$\therefore u, v \Rightarrow u-u_0, v-v_0$$

For $u_0 = v_0 = \frac{N}{2}$,

$$e^{j2\pi(u_0x+v_0y/N)} = e^{j2\pi(Nx+Ny/2N)}$$

$$= e^{j\pi(x+y)}$$

$$= (e^{+j\pi})^{x+y}$$

$$e^{j2\pi(u_0x+v_0y/N)} = [\cos(\pi)+j\sin(\pi)]^{x+y} = (-1)^{x+y}$$

$$\therefore f(x,y)\cdot(-1)^{x+y} \Rightarrow F(u-N/2,\ v-N/2)$$

For filtering in freq. domain, we have to multiply image with $(-1)^{x+y}$ then take 2D-DFT.

In case of situation where dynamic range is out of range, we need to take log of value.

(Image) (Filter) (Output)

$$f(x,y) \circledast h(x,y) = g(x,y)$$

$\times(-1)^{x+y}$ DFT ↓ DFT ↓ 2D-IDFT ↑

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

$(-1)^{x+y}$: Enables us to gather low freq. at center and high freq. at border

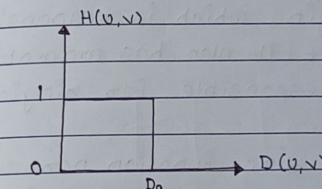
Steps for filtering in freq. domain:

$$f(x,y) \longrightarrow f(x,y)\times(-1)^{x+y} \xrightarrow{DFT} F(u,v)$$

$$\downarrow \times$$

$$g(x,y) \xleftarrow{IDFT} G(u,v) \longleftarrow H(u,v)$$

* Ideal Low Pass Filter (LPF)



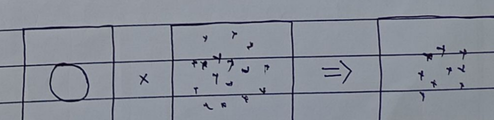
It is called Ideal as it removes anything beyond D_0

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

where,

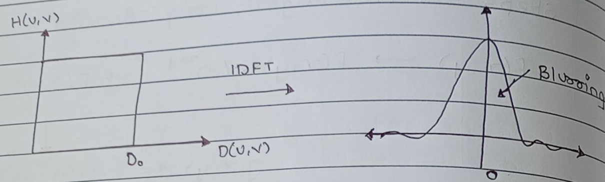
$$D(u,v) = \left[\left(u-\frac{M}{2}\right)^2 + \left(v-\frac{N}{2}\right)^2\right]^{1/2}$$

$D(u,v)$ gives concentric rings having fixed values, depending on value of D_0 white circle becomes smaller or larger.



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

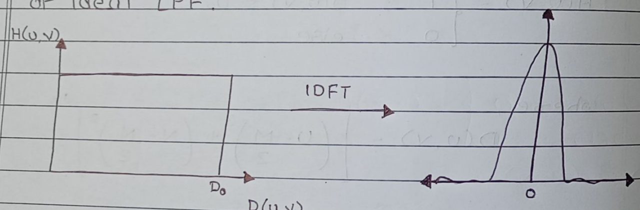
Page no.: Date:



H(u,v) h(x,y)

With respect to h(x,y), it has dominant component at origin, which is responsible for blurring effect. It also has concentric rings which are responsible for ringing effect.

If we increase D_0, sinc function narrows. Hence, we get more blurring and ringing effect in output image. This is disadvantage of Ideal LPF.



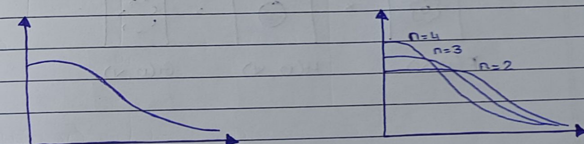
H(u,v) h(x,y)

Page no.: Date:

* Butterworth Low Pass Filter (LPF)

$$H(u,v) = \frac{1}{1+\left[\frac{D(u,v)}{D_0}\right]^{2n}} \quad ; \; n : \text{order}$$

Butterworth LPF does not have sharp cutoff, therefore ringing effects are less. As order of filter goes on increasing, small amount of ringing effect will be observed because Butterworth LPF tends to become an Ideal LPF.



$n \leq 5$

$\uparrow n$ BLPF $\rightarrow$ ILPF

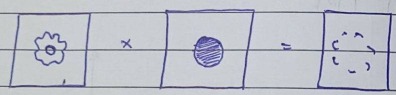
$\therefore$ To get good results, order should be less than or equal to 5 $(n \leq 5)$

* Gaussian Low Pass Filter

$$H(u,v) = e^{-D^2(u,v)/2D_0^2}$$

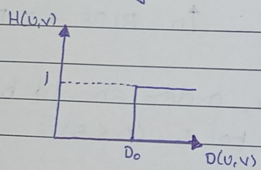
Response of this Gaussian LPF is similar to response of BLPF. There will be no ringing effects observed in GLPF. Gaussian Low Pass Filter is preferred over ILPF.

$$HPF = 1 - LPF$$



$F(u,v)$ $\quad$ $H(u,v)$ $\quad$ $G(u,v)$

* Ideal High Pass Filter



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

* Butterworth HPF:

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

* Gaussian HPF:

$$H(u,v) = 1 - e^{-D^2(u,v)/2D_0^2}$$

* Other ways to do 2D-DFT,

$$F = T.f.T'$$

$\therefore$ For $f = \begin{bmatrix} 0 & 1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 2 \end{bmatrix}$

$$\therefore F = T.f.T'$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} 0 & 1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 2 \end{bmatrix} . T'$$

$$= \begin{bmatrix} 4 & 8 & 12 & 8 \\ -2 & -2 & -2 & -2 \\ 0 & 0 & 0 & 0 \\ -2 & -2 & -2 & -2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

$$\therefore F = \begin{bmatrix} 32 & -8 & 0 & -8 \\ -8 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -8 & 0 & 0 & 0 \end{bmatrix}$$

Image transforms in addition to 2D-DFT, few more transformations prove to be useful. Image transform refers to class of unitary matrices used for representing images. Images can be extended as discrete set of basis arrays called as basis images.

* Walsh-Hadamard Transform:

This transform is based on Hadamard matrix, which is a square-array having entries 1, -1 only. Hadamard matrix of order 2 is given by,

$$H[2] = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} + & + \\ + & - \end{bmatrix}$$

Hadamard matrices of order 2^n can be derived using Kronecker product, which is given by,

$$H[2^n] = H(2) \cdot H[2^{n-1}]$$

$$\therefore H(4) = H(2) \cdot H(2) = H(2^2)$$

$$= \begin{array}{c} \\ \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \end{array} \begin{array}{c} \text{Sequency} \\ 0 \\ 3 \\ 1 \\ 2 \end{array}$$

(No. of sign change)

Q) Derive Hadamard matrix for order 8.

$$H(8) = H(2^3)$$
$$= H(2) \cdot H(2^2)$$
$$= H(2) \cdot H(4)$$

$$= \begin{array}{c} \\ \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \end{array} \begin{array}{c} \text{Sequency} \\ 0 \\ 7 \\ 3 \\ 4 \\ 1 \\ 6 \\ 2 \\ 5 \end{array}$$

Q) Compute Hadamard ~~matrix~~ transform for given sequence $x(n) = \{1, 2, 0, 3\}$

→ $x(n) = \{1, 2, 0, 3\}$

$\therefore$ Hadamard transform is given by,

$$\therefore X[K] = H(4) \cdot x(n)$$

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

Page no.: ____/ Date: ___/___/____

Q) For given image, find Hadamard transform

$$f = x(n) = \begin{bmatrix} 2 & 1 & 2 & 1 \\ 1 & 2 & 3 & 2 \\ 2 & 3 & 4 & 3 \\ 1 & 2 & 3 & 2 \end{bmatrix}$$

$$F = T.f.T'$$

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

$$= \begin{bmatrix} 6 & 8 & 12 & 8 \\ 2 & 0 & 0 & 0 \\ 0 & -2 & -2 & -2 \\ 0 & -2 & -2 & -2 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

$$\therefore F = \begin{bmatrix} 34 & 2 & -6 & -6 \\ 2 & 2 & 2 & 2 \\ -6 & 2 & 2 & 2 \\ -6 & 2 & 2 & 2 \end{bmatrix}$$

Page no.: ____/ Date: ___/___/____

* Walsh Transform

It is obtained for Hadamard matrix, from rearranging rows in increasing sign change order.

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

$$H(4) = \begin{array}{cc} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} & \begin{array}{c} 0 \\ 3 \\ 1 \\ 2 \end{array} \end{array}$$

$$\therefore W(4) = \begin{array}{cc} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} & \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} \end{array}$$

Q) Derive Walsh Transform for N = 8.

$$\therefore W(8) = \begin{array}{cc} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} & \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \end{array}$$

Page no.: ___ Date: __/__/____

Q) For given image, find Walsh transform

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

→ $F = T.F.T'$

$F = H(4).F.H(4)'$

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

$$= \begin{bmatrix} 6 & 8 & 12 & 8 \\ 0 & -2 & -2 & -2 \\ 0 & -2 & -2 & -2 \\ 2 & 0 & 0 & 0 \end{bmatrix} . W(4)'$$

$$F = \begin{bmatrix} 34 & -6 & -6 & 2 \\ -6 & 2 & 2 & 2 \\ -6 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}$$

Page no.: ___ Date: __/__/____

* Haar Transform

$$F = H.F.H'$$

The transformation H contains Haar basis functions $h_{pq}(x)$, which are defined over continuous closed intervals $x\,(0,1]$.

The Haar basis function are,

$$h_{00}(x) = \frac{1}{\sqrt{N}}\, x \quad \in [0,1]$$

and,

$$h_{pq}(x) = \frac{1}{\sqrt{N}} \begin{cases} 2^{p/2} & : \frac{q-1}{2p} \leq x < \frac{q-0.5}{2p} \\ -2^{p/2} & : \frac{q-0.5}{2p} \leq x < \frac{q}{2p} \end{cases}$$

where, $0 \leq p < \log_2 N$ and $1 \leq q < 2^p$

For $N = 2$,

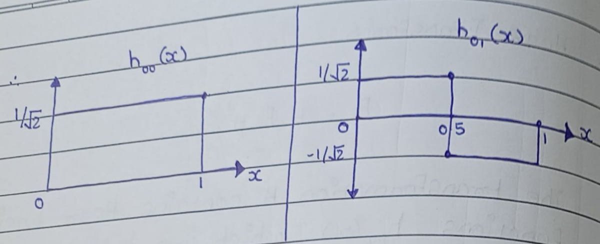
$\log_2 N = \log_2 (2) = 1$

$\therefore 0 \leq p < \log_2 N$	$1 \leq q < 2^p$
$\therefore 0 \leq p < 1$	$1 \leq q < 2^0$
$\therefore p = 0$	$\therefore q = 1$

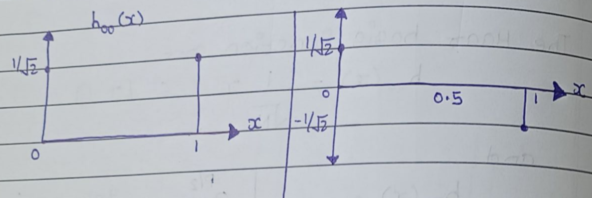
$$\therefore h_{00}(x) = \frac{1}{\sqrt{N}}\, x,\ x \in [0,1] = \frac{1}{\sqrt{2}}\, x$$

$$\therefore h_{01}(x) = \frac{1}{\sqrt{N}} \begin{cases} 1 & : 0 \leq x < 0.5 \\ -1 & : 0.5 \leq x < 1 \end{cases}$$

Page no.: ___ Date: __/__/____



∴ Discrete Haar Functions:



∴ $Haar(2) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ (Same as Hadamard(2))

For N = 4,

$\log_2 N = \log_2 4 = 2$

∴ $0 \leq p < \log_2 N$	$0 \leq q < 2^p$
∴ $0 \leq p < 2$	
∴ $p = 0, 1$	∴ $1 \leq q < 1 \rightarrow q = 1$
	∴ $1 \leq q < 2 \rightarrow q = 1, 2$

∴ $h_{00}(x) = \frac{1}{\sqrt{N}} = \frac{1}{\sqrt{4}} = 1/2 \quad ; \quad x \in [0, 1]$

Page no.: ___ Date: __/__/____

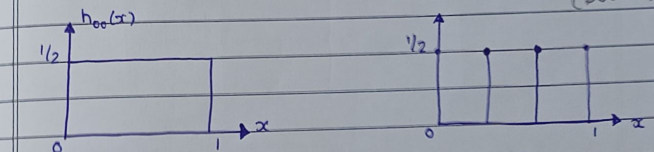
∴ $h_{pq}(x) = 1/2 \begin{cases} 2^{p/2} & ; \ {}^{q-1}/_{2^p} \leq x < {}^{q-0.5}/_{2^p} \\ -2^{p/2} & ; \ {}^{q-0.5}/_{2^p} \leq x < {}^{q}/_{2^p} \end{cases}$

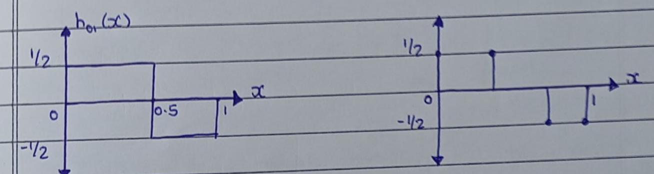
∴ $h_{01}(x) = 1/2 \begin{cases} 1 & ; \ 0 \leq x < 0.5 \\ -1 & ; \ 0.5 \leq x < 1 \end{cases}$

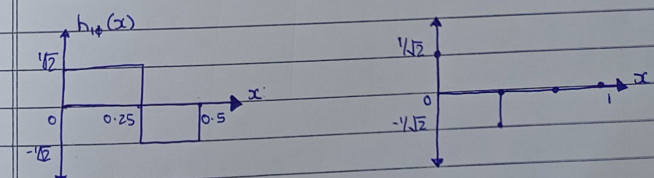
∴ $h_{11}(x) = 1/2 \begin{cases} \sqrt{2} & ; \ 0 \leq x < 0.25 \\ -\sqrt{2} & ; \ 0.25 \leq x < 0.5 \end{cases}$

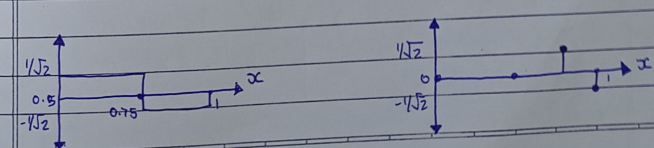
∴ $h_{12}(x) = 1/2 \begin{cases} \sqrt{2} & ; \ 0.5 \leq x < 0.75 \\ -\sqrt{2} & ; \ 0.75 \leq x < 1 \end{cases}$

(Doubt)









Date: __/__/__ Page no.: __

$$\therefore \text{Haar}(4) = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

$$\therefore \text{Modified Haar}(4) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

Date: __/__/__ Page no.: __

* Discrete Cosine Transform (DCT)

JPEG images used DCT as initial stage for compression.
F.T. uses sine and cosine waves
DCT uses only cosine waves
$\therefore$ DCT is purely real, unlike DFT

- 1D - DCT

For sequence $f(x)$, $0 \le x \le N-1$

$$F(u) = \sum_{x=0}^{N-1} f(x) \cdot \cos\left[\frac{\pi(2x+1)u}{2N}\right] \; ; \; 0 \le u \le N-1$$

- 2D - DCT

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

DCT Matrix is real, orthogonal but not symmetric.

$\therefore C = C^*$

$\therefore C^{-1} = C'$

$C \cdot C' = I$

$$\therefore \text{2D-DCT} : F = C \cdot f \cdot C' \; ; \; C = \begin{bmatrix} 0.5 & 0.5 & 0.5 & \ldots \\ 0.653 & 0.2705 & -0.2705 & \ldots \\ 0.5 & -0.5 & -0.5 & \ldots \\ 0.2705 & -0.653 & 0.653 & \ldots \end{bmatrix}$$

Date: / / Page no.:

Q) Find DCT. $f(x) = \{1, 2, 4, 7\}$

$$F = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.653 & 0.2705 & -0.2705 & -0.653 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.2705 & -0.653 & 0.653 & -0.2705 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 4 \\ 7 \end{bmatrix}$$

$$\therefore F = \begin{bmatrix} 7 \\ -4.459 \\ 1 \\ -0.3170 \end{bmatrix}$$

Page no.: ___ Date: ___

UNIT 3:

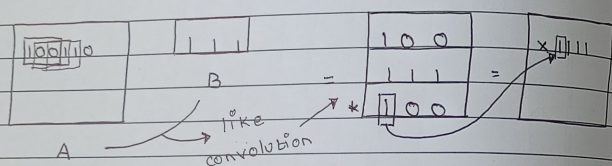
IMAGE MORPHOLOGY

* Dialation

Let $A(x,y)$ be a binary image, B as structuring element, then let $C(x,y)$ will be,

$$C(x,y) = \max\{A(x,y) * B\}$$

$$C = A \oplus B$$



* Erosion:

$$C(x,y) = \min\{A(x,y) * B\}$$

$$C = A \ominus B$$

Page no.: ___ Date: ___

Q) Consider 10x10 image, Perform dilation using structuring image given.

B =

1	1
1	1

A =

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

C =

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

Same as input or blank

Dialation adds pixels to the boundary and helps in enlarging / expanding the boundary. Size of black-box at center has reduced and white-boundary has increased in size.

Q) Perform erosion on 10x10 image with same structuring element, and image.

C =

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	0	0	_
0	0	1	0	0	0	1	0	0	_
0	0	1	0	0	0	1	0	0	_
0	0	1	0	0	0	1	0	0	_
0	0	1	1	1	1	1	0	0	_
0	0	0	0	0	0	0	0	0	_
0	0	0	0	0	0	0	0	0	_
_	_	_	_	_	_	_	_	_	_

Size of white-boundary has reduced and black-box at center has increased in size. Erosion shrinks the boundary and helps in enlarging / expanding the gaps.