Image Processing and Analysis

Unit - 2

Image Enhancement

Spatial Domain: Gray level transformations - Histogram processing - basics of spatial filtering - smoothing and sharpening spatial filla; Frequency domain: Introduction to Fourier transform - The 2D convolutional theorem - Smoothing and Sharpening Frequency Domain Filters : Ideal Butterworth - Gaussian filters ; Homomorphic filtering

Case study: medical image enhancement

* Spatial vs. Transform Domain

Spatial Domain - refers to the image plane itself

→ image processing methods are based on direct manipulation of pixels in an image.

Transform Domain -

→ Take image to transform, process

→ Obtain the inverse transform to bring results back into spatial domain

* Categories of Spatial Processing

(i) Intencity Transformation (point processing)

→ operate on single pixels of an image

→ called point processing

→ used for contrast manipulation & image thresholding

(ii) Spatial Filtering

→ performs operations on the neighborhood of every pixel in an image

→ used for image smoothing and sharpening

* Expression for Spatial domain processing

→ The spatial domain processes are based on the expression:

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

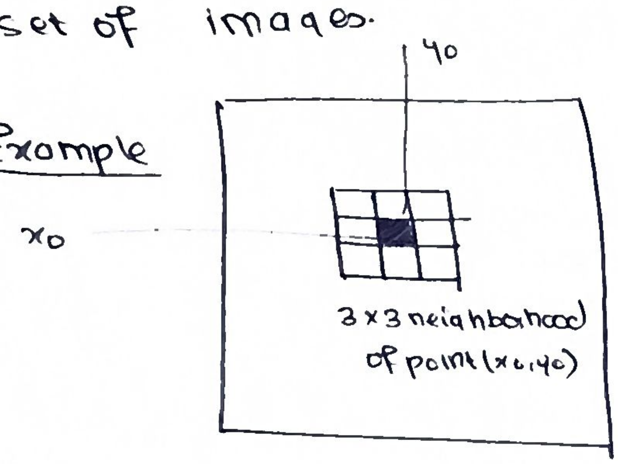
$f(x,y)$ = input image

$g(x,y)$ = output image

T is an operator defined over a neighborhood of point (x,y)

→ The operator can be applied to pixels of a single image, or to a set of images.

Example



→ The value of the output image g at the coordinates is equal to the result of applying T to the neighborhood with the origin at (x_0,y_0) in f.

→ T may be any operation - for eg. it may be the average intensity of pixels in the neighborhood.

* Terminology

A. Spatial Mask - A small matrix / grid of numbers used to perform operations on an image.

→ Typically a square matrix with odd dimensions (3×3, 5×5 etc)

B. Kernel - same as a spatial mask.

→ Kernels are applied to the input image by sliding them over the image's pixels and performing mathematical operations on them.

∴ Template, Window also refer to the same as A & B

* Intensity Transformation - 1×1

→ The smallest possible neighborhood is of size 1×1. In this case, g depends only on the value of f at a single point (x,y)

→ T in $g(x,y) = T[f(x,y)]$ becomes an intensity

→ This is called a gray-level or mapping transformation. It is of the form:

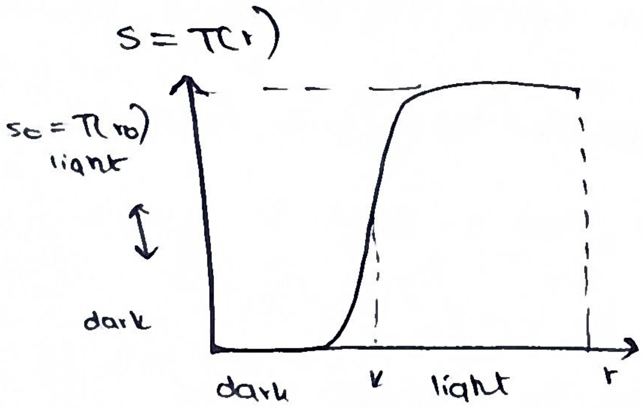
$$s = T(r)$$

s and r denote intensity of g and f at any point.

A. Intensity Transformation for Contrast Stretching

→ The resulting of applying the transformation to every pixel in f to generate the corresponding pixels in g would produce an image of higher contrast than the original.

→ There would be darkening of intensity levels ~~above~~ below k & brightening the levels above k.



B. Intensity Transformation for Thresholding

→ In the limiting case, rather than a range of values, anything below k is mapped to 0, and above k to 1.

→ It produces a 2-level binary image.

→ A mapping of this form is called contrast stretching?

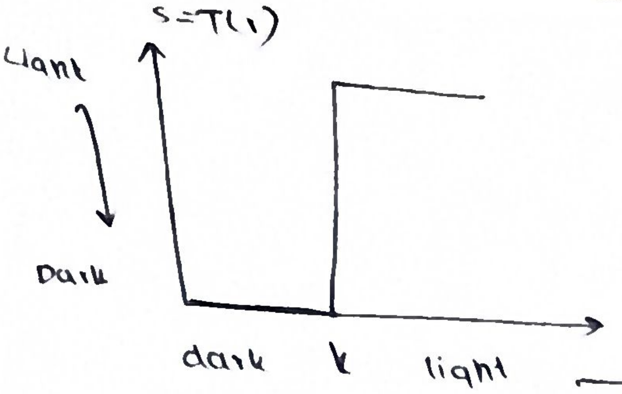


Image Enhancement in Spatial Domain

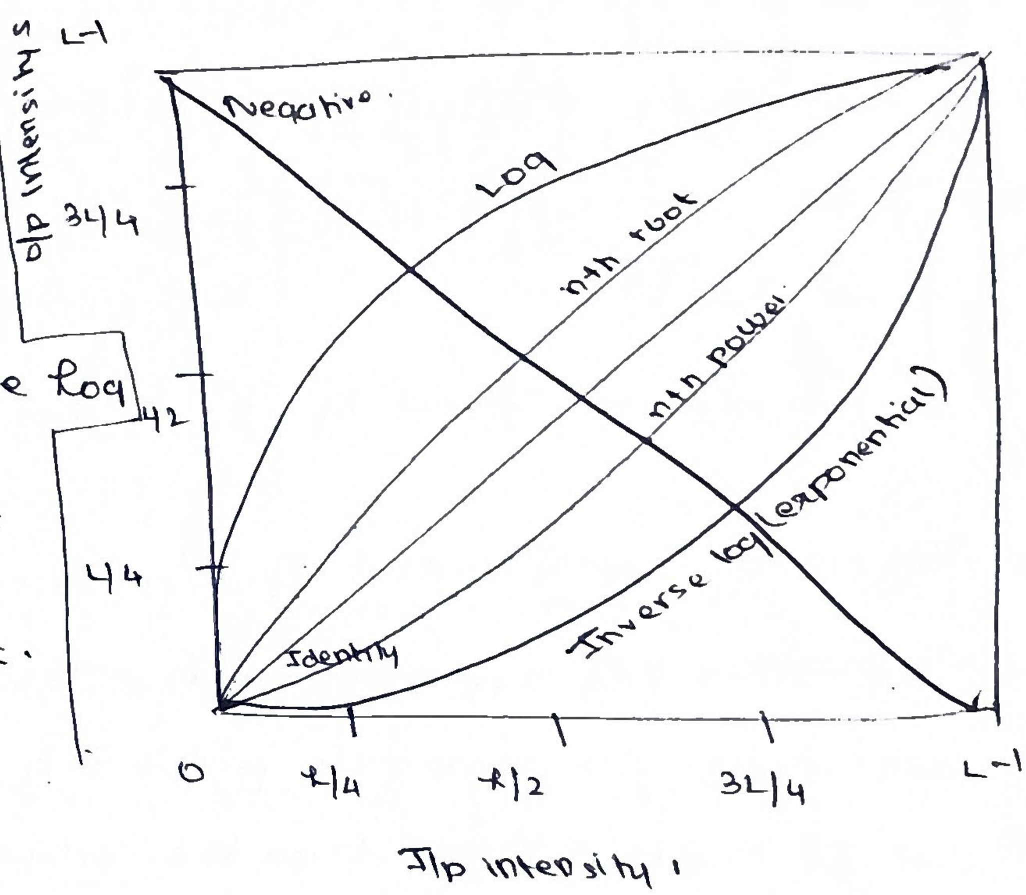
* Intensity Transformation Functions (Gray Level Transformations)

Types of functions:

A. Linear — Identity
— Negative

B. Logarithmic — Log
— Inverse Log

C. Power-law — nth power
— n^{th} root.



A. Linear Transformations

(i) Identity - Input and output intensity levels remain the same.

(ii) Negative - The negative of an image with intensity levels in the range $[0, L-1]$ is obtained using the negative transformation function : $\boxed{s = L-1-r}$

→ Reversing the intensity levels produces the equivalent of a photographic negative.

→ Used to enhance white or gray detail embedded in dark regions of an image.

B. Log Transformations

→ The general form of the log transformation is

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

c is a constant

$r \geq 0$

→ This transformation maps a narrow range of low intensity values in the input to a wider range of output levels

→ eg. input levels $[0, L/4]$ map to $[0, 3L/4]$

→ Conversely, higher values of input levels are mapped to a narrower range in the output.

→ This kind of transformation is used to expand the values of dark pixels in an image, while compressing the higher-level values.

→ The opposite is true of the inverse log (exponential) transformation.

C. Power-Law (Gamma) Transformations

→ Power-law transformations have the form

$$s = cr^{\gamma}$$

where c and γ are positive constants

→ Sometimes, the equation is $s = c(r+\varepsilon)^{\gamma}$ to account for offsets.

→ As in the case of log transformations, power-law curves w/ fractional values of γ map a narrow range of dark input values to a wider range of output values

→ Curves generated with $\gamma > 1$ have the opp. effect than when $\gamma < 1$.

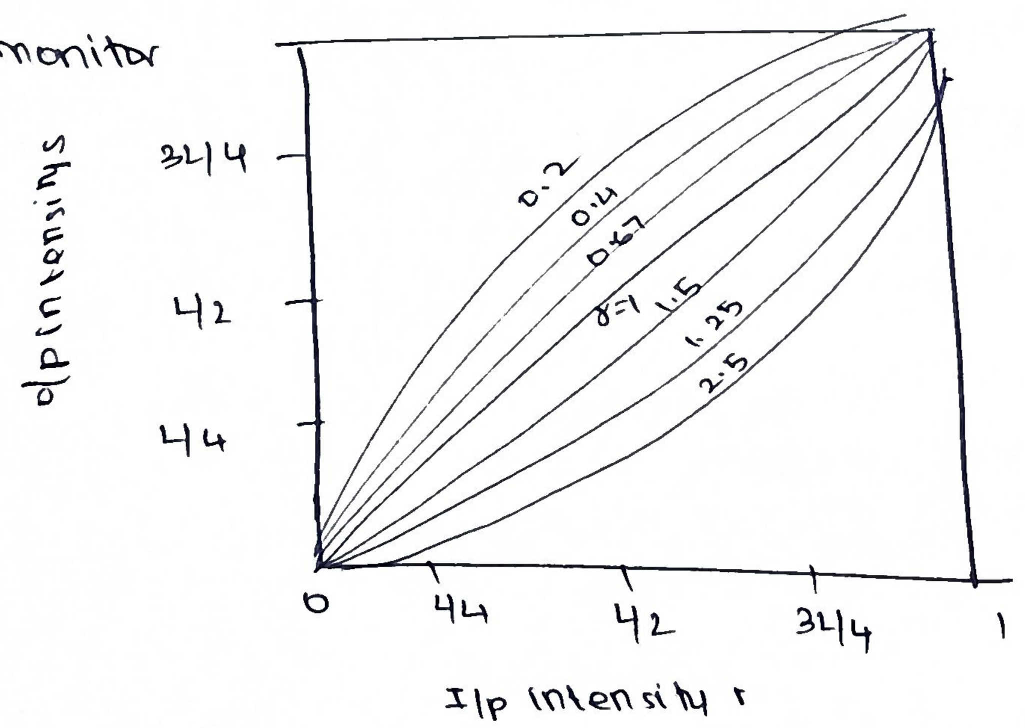
→ When $c = \gamma = 1$, it reduces to the identity transform.

→ By convention, the exponent in a power-law equation is referred to as gamma.

Gamma Correction / Gamma Encoding?

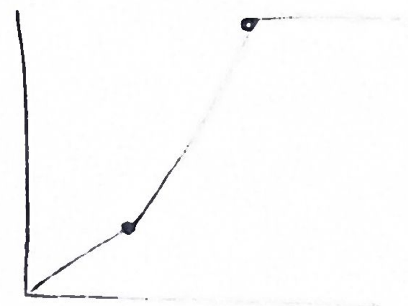
→ The process used to correct the power-law response phenomena is called gamma correction / encoding.

→ eg. Cathode ray tube devices have an intensity to voltage response that is a power function. Such display systems would tend to produce images that are darker than intended. Gamma correction is used to preprocess the image, before inputting it into the monitor



* Piecewise Linear Transformation Functions

can be : (i) contrast stretching



(ii) intensity-level slicing - used to highlight specific portions of an image.

(iii) bit-plane slicing - highlight the contribution made to image appearance by specific bits. An 8-bit image may be considered as being composed of eight 1-bit planes

plane1 - lowest order bit

plan 8 - highest order bit

Lower order bits contribute more to subtleties

* Histogram Equalization

Steps* : 1. Form the cumulative histogram

2. Normalize the value by dividing it by the total no. of pixels

3. Multiply the value by the maximum gray level value & round off the result.

4. Map the original value of the result of step3 by 1-1 correspondence.

Example1

Consider the input image

4	4	4	4	4
3	4	5	4	3
3	5	5	5	3
3	4	5	4	3
4	4	4	4	4.

Perform histogram equalization.

Ans

Gray level	no. of pixels	PDF	CDF	CDF * (k-1)	rounded
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	6	0.24	0.24	1.68	2
4	14	0.56	0.80	5.6	6
5	5	0.20	1.0	7	7
6	0	0	1.0	7	7
7	0	0	1.0	7	7
	n= 25				

new pixel mapping

rk	pk
2	6
6	14
7	5

output image

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

Example 2 : Perform histogram equalization to the given 8×8 level image

r_k	0	1	2	3	4	5	6	7
p_k	8	10	10	2	12	16	4	2

r_k	p_k	PDF	CDF	CDF $\times (L-1)$	r_k rounded
0	8	0.125	0.125	0.875	1
1	10	0.15625	0.28125	1.96	2
2	10	0.15625	0.4375	3.06	3
3	2	0.03125	0.46875	3.28	3
4	12	0.1875	0.65625	4.593	5
5	16	0.25	0.90625	6.343	6
6	4	0.0625	0.96875	6.78	7
7	2	0.03125	1	7	7

n = 64

r_k'	p_k'
1	8
2	10
3	12
5	12
6	16
7	6.

Example 3 : Given the following input & reference images, perform histogram specification on it.

Input Image

rk	pk
0	790
1	1023
2	850
3	656
4	329
5	245
6	122
7	81
	4096

Reference image

rk	pk
0	0
1	0
2	0
3	614
4	819
5	1230
6	819
7	614
	4096

Histogram equalization on the i/p image.

rk	pk	PDF	CDF	CDF × (L-1)	rk'
0	790	0.19287	0.1929	1.35	1
1	1023	0.2498	0.4427	3.098	3
2	850	0.2075	0.6502	4.55	5
3	656	0.1602	0.8104	5.672	6
4	329	0.0803	0.8907	6.23	6
5	245	0.060	0.9507	6.65	7
6	122	0.030	0.9807	6.86	7
7	81	0.020	1.0	7	7
	4096				

Histogram Equalization of reference image

rk	pk	PDF	CDF	CDF × (L-1)	rk'
0	0	0	0	0	0
1	0	0	0	0	0
2	0	0	0	0	0
3	614	0.1500	0.15	1.05	1
4	819	0.200	0.35	2.45	3
5	1230	0.3003	0.65	4.55	5
6	819	0.200	0.85	5.95	6
7	614	0.180	1	7	7
	4096				

Reference image		Input image	
0 – 0		1	→ 790
1 – 0		3	→ 1023
2 – 0		5	→ 850
3 – 1		6, 6	→ 985
4 – 3		7, 7, 7	→ 448
5 – 5			
6 – 6			
7 – 7			

Output image

rk	0	1	2	3	4	5	6	7
pk	0	0	0	790	1023	850	985	448

Classification of Filters

(i) low-pass – preserve low frequencies

(ii) high pass – preserve high frequencies

(iii) band-pass – preserve frequencies within a band

(iv) band-reject – reject frequencies within a band

Spatial Filtering – Correlation & Convolution

are LINEAR SPATIAL FILTERING METHODS

* Correlation and Convolution

Correlation consists of moving the center of a kernel over an image & computing the sum of products at each location

It is given by:

$$g(x,y) = \sum_{s=-a}^{a} \sum_{t=-b}^{b} w(s,t)\, f(x+s, y+t)$$

Convolution - same principle, but the correlation kernel is rotated by 180°.

Dimensions of the resultant image.

- no. of rows of $x(m,n)$ + no. of rows of $h(m,n)$ − 1

×

no. of cols of $x(m,n)$ + no. of cols of $h(m,n)$ − 1

Example 1 : Apply correlation on the following image

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

new dimensions $= (5+3)-1 \times (5+3)-1 = 7 \times 7$

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

Example 2: Convolve the given mask

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

Example 3 : Find the correlated image

Image : $\begin{bmatrix} 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix} \quad \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$

o/p dimensions :

$(2+3)-1 \times (3+1)-1 = 4 \times 3$

$$\begin{bmatrix} 0 & 0 & 0 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \\ 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 11 & 13 & 15 \\ 11 & 13 & 15 \end{bmatrix}$$

Example 4: Perform linear convolution on the given images.

(1) $x(m,n) = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ mask $= \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}$

correlation / convolution:

dims: $\begin{matrix} (3+3)-1 \\ \times \\ (3+2)-1 \end{matrix} = 5 \times 4$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 3 \\ 0 & 4 & 5. & 6 \\ 0 & 7 & 8 & 9 \\ 0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} 5 & 12. & 16 \\ 12 & 27 & 33 \\ 11 & 24 & 28 \end{bmatrix}$$

(2) $x(m,n) = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ mask $= [3\ 4\ 5]$

dims: $\begin{matrix} (3+1)-1 \\ \times \\ (3+3)-1 \end{matrix} = 3 \times 5$

convolved mask

$[5\ 4\ 3]$

$$\begin{bmatrix} 0 & 1 & 2 & 3 & 0 \\ 0 & 4 & 5 & 6 & 0 \\ 0 & 7 & 8 & 9 & 0 \end{bmatrix} = \begin{bmatrix} \cancel{10}\ 10 & 22 & 22 \\ 31 & 58 & 49 \\ 52 & 94 & 76 \end{bmatrix}$$

3×3

$5+8+9$

$10+12$

$16+15$

$20+20+18$

$25+24$

$28+24$

$40+36$

$35+32+27$

(3) $x(m,n) = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$ mask $= \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$

flip mask : $\Rightarrow \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$

dims : $(5+3)-1 \times (5+3)-1 = 7 \times 7$

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

* Smoothing and Sharpening Filters

Smoothing Filters

A. Linear / Box Filters

→ replace by average of subimage

or

→ find the weighted average

* Smoothening and Sharpening Spatial Filters

A Smoothing Filters → Low pass Filter

1. Linear / Box Filters

→ replace by average of subimage

or

→ Find the weighted average

2. Order Statistics Filters

(a) Median Filter

→ replace center value by the median of values in the mask

→ good for impulse / salt & pepper noise

(b) Max Filter

→ To detect the brightest points

(c) Min Filter

3. Gaussian Filter.

→ denoted by the Gaussian Kernel:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

→ removes high-frequency components from the image

→ convolution with self gives another Gaussian

B. Sharpening Filters → High pass filters

→ highlights the fine details of the image like lines, edges and boundaries.

1. Spatial Differentiation

→ can help find if it is a step or a ramp

$$\frac{\partial f}{\partial x} = f(x+1) - f(x)$$

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x)$$

→ The first derivative (Gradient)

(i) must be zero in the areas of constant intensity

(ii) must be non-zero at the intensity ramp or step

(iii) must be non-zero along ramps

→ The second derivative (also called Laplacian ?)

(i) must be zero in the areas of constant intensity

(ii) must be non-zero at the onset & end of a ramp or step

(iii) must be zero along ramps of constant slope.

2. Laplacian Filter

$\frac{\partial f}{\partial x} =$

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

$\frac{\partial f}{\partial y} =$

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

note that if the center is +ve, the surrounding should be -ve & vice versa.

→ Convolve the Laplacian mask on the input image to sharpen the image.

→ To sharpen edge along the x-axis, y-axis, or in all directions, apply appropriate masks.

3. $\boxed{\text{Unsharp Masking and Highboost Filtering}}$

- Blur (unsharp) the original image
- Subtract the blurred image from the original to get a mask
- add the mask to the original

Specifically,

(i) $f(x,y)$ is to be made $g(x,y)$

(ii) smooth or blur image

(iii) smoothed image $= f'(x,y)$

(iv) $g_{mask} = f(x,y) - \bar{f}(x,y)$

(v) $g(x,y) = f(x,y) + g_{mask}$

sharpened image

$$g(x,y) = k * g_{mask} + f(x,y)$$

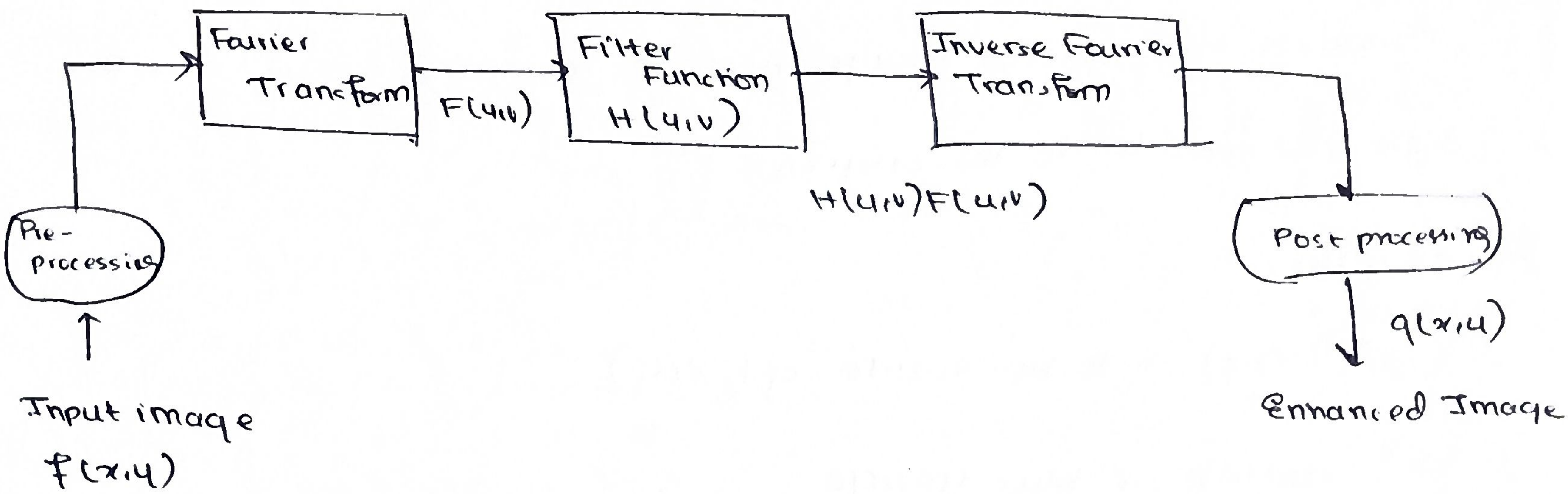
when $k=1 \Rightarrow$ unsharp masking

$k>1 \Rightarrow$ highboost filtering

* Frequency Domain

*** Fourier Transform** – Fourier Transform transforms one function to another domain – called the frequency domain representation of the original function

*** Filtering in the Frequency Domain.**



*** Properties of Fourier Transform**

→ As one moves away from the origin in F(u,v), the lower frequencies correspond to slow gray level changes

→ Higher frequencies correspond to the fast changes in gray levels - smaller details such as edges of objects and noise.

*** Filtering using Fourier Transforms**

→ H(u,v) – the filter function determines how the enhanced image looks like

→ A lowpass H(u,v) results in a blurring effect

A highpass H(u,v) results in a sharpening effect

High Frequency: edges and sharp details in an image.

Low Frequency: slowly varying characteristics of an image - eg. overall contrast and average intensity.

★ Filters in the Frequency Domain

A. Low Pass Filters (Smoothing Filters)

- Ideal low pass filters
- Butterworth low pass filters
- Gaussian low pass filters.

B. High pass Filters (Sharpening Filters)

- Ideal high pass filters
- Butterworth high pass filters
- Gaussian high pass filters
- Laplacian in frequency domain
- Unsharp masking, high boost filtering and high frequency emphasis filters

C. Homomorphic Filters

* Low Pass vs. High Pass vs. Band Pass Filters

A. Low Pass Filters - A low pass filter attenuates high frequencies and retains low frequencies unchanged

→ equivalent of a smoothing filters

→ blocks high frequencies corresponding to sharp intensity changes i.e the fine - scale details and noise in the spatial domain image.

B. High Pass Filters - yields edge enhancement or edge detection in the spatial domain. (since edges contain many high frequencies)

→ Areas of constant gray level consisting of mainly low frequencies, are therefore suppressed.

C. Band Pass Filtering - Attenuates very low & very high frequencies but retains a middle band of frequencies.

→ used to enhance edges, while reducing noise at the same time

→ a combination of both lowpass and bandpass filters

→ attenuate all frequencies $< D_0$ and $> D_1$

Low Pass Filters (smoothening)

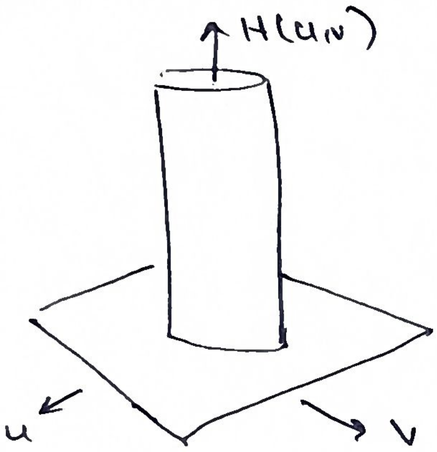
1 Ideal Low Pass Filter

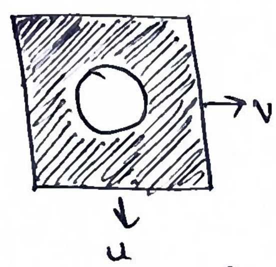
$$H(u,v) = \begin{cases} 1, & D(u,v) \le D_0 \\ 0, & D(u,v) > D_0 \end{cases}$$

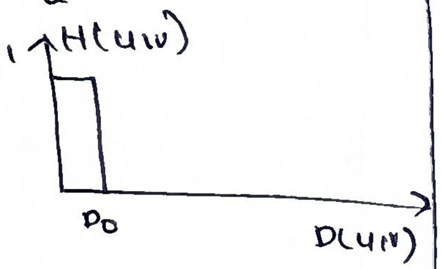
$$D(u,v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$

D_0 = cutoff frequency

$D(u,v)$ = distance between a point (u,v) in the frequency domain & the center of the frequency rectangle.





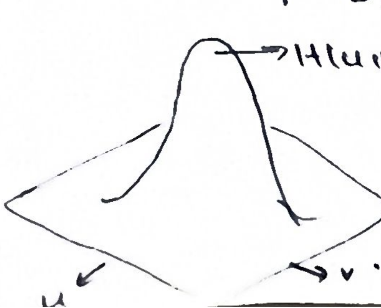


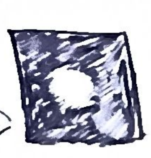
2 Butterworth Low Pass Filters

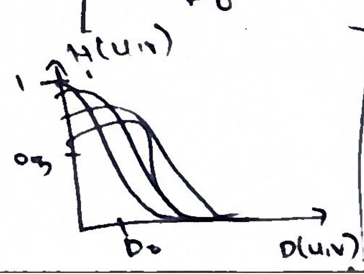
→ provides a smooth transition between low & high frequencies

→ reduce the ringing effect caused by the Ideal Low Pass Filter (ILPF)

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

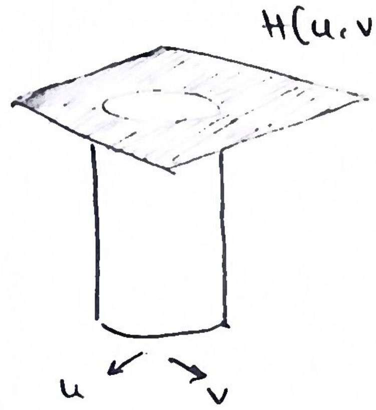


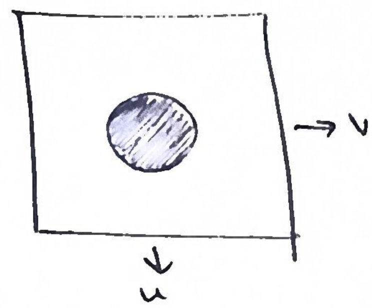




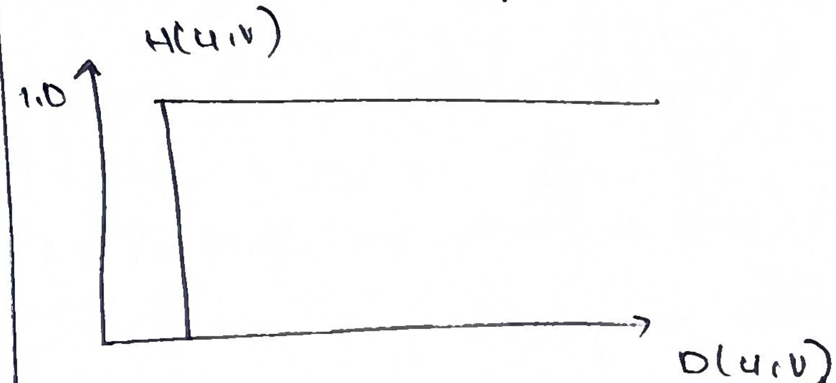
High Pass Filters (sharpening)

1 Ideal High Pass Filter



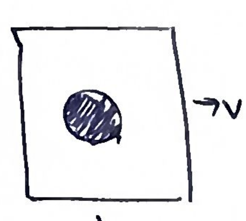


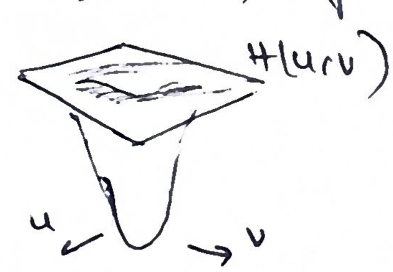
$$H(u,v) = 1 - H_{lp}(u,v)$$

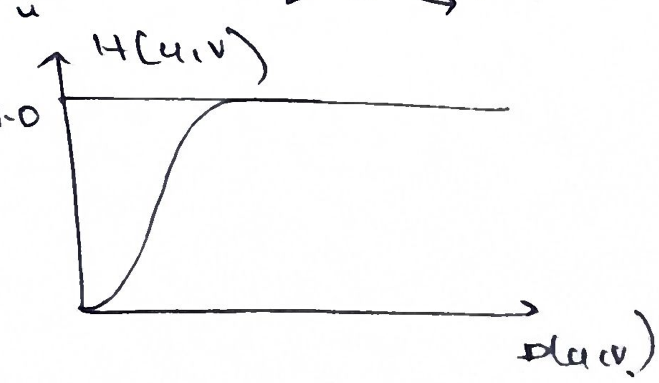


2 Butterworth High Pass Filters

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







Smoothening

Low Pass Filter

③ Gaussian Low Pass Filter

→ given by:

$$H(u,v) = e^{\frac{-D^2(u,v)}{2\sigma^2}} \xrightarrow{\sigma = D_0}$$

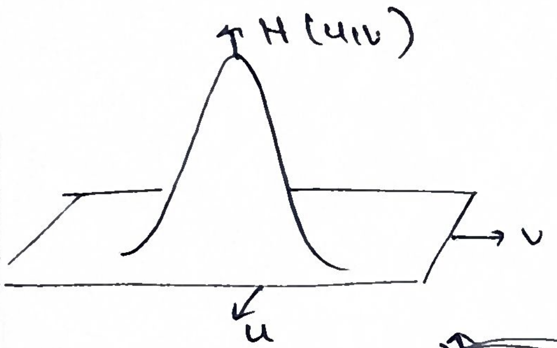
$$H(u,v) = e^{\frac{-D^2(u,v)}{2D_0^2}}$$

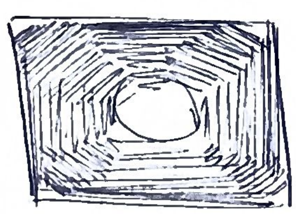
where : $D^2(u,v) = (u - M/2)^2 + v(-N/2)^2$

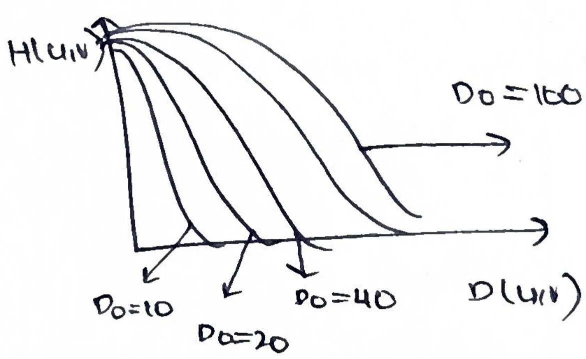
here : $H(D(u,v) = 0) = 1$

$H(D(u,v) = D_0) = e^{-0.5} = 0.607$

→ The inverse Fourier transform of GLPF is also Gaussian.





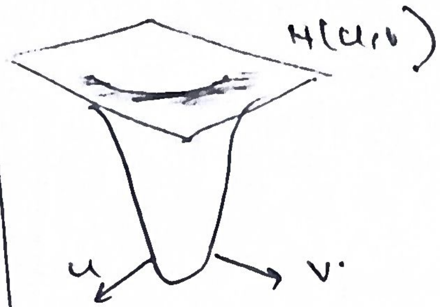


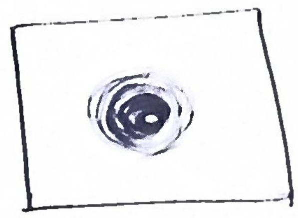
Sharpening

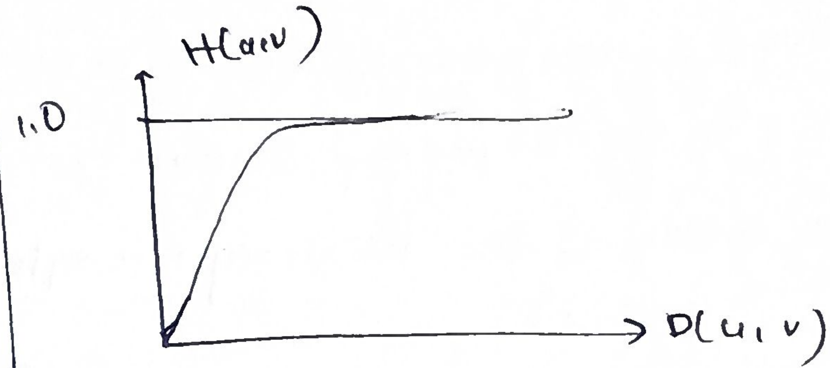
High Pass Filter.

③ Gaussian High Pass Filter :

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







* Laplacian in the Frequency Domain

→ It is equivalent to applying the Fourier domain filter:

$$H(u,v) = -(u^2+v^2)$$

which is centered and given by:

$$H(u,v) = -\left[\left(u - M/2\right)^2 + \left(v - N/2\right)^2\right]$$

$$\therefore \nabla^2 f(x,y) = -\left[\left(u-M/2\right)^2 + \left(v-N/2\right)^2\right] F(u,v)$$

* Unsharp Masking & High Boost Filtering in the Frequency Domain

A. Unsharp Masking

$f_{hp}(x,y) = f(x,y) - f_{ep}(x,y)$ → in spatial domain

→ in the frequency domain:

$$\boxed{H_{hp}(u,v) = 1 - H_{ep}(u,v)}$$

B. High Boost Filtering

In the spatial domain, it is

$$f_{hp}(x,y) = (A-1)f(x,y) + f(x,y) - f_{ep}(x,y)$$

In the frequency domain, it is:

$$\boxed{H_{hp}(u,v) = (A-1) + H_{hp}(u,v)}$$

* Homomorphic Filtering:

Goal: simultaneous brightness range compression and contrast enhancement

→ The illumination reflectance model is given by:

$$f(x,y) = i(x,y) \cdot r(x,y)$$

$i(x,y)$ = illumination → characterized by slow spatial variations - slow frequency components

$r(x,y)$ = reflectance → tends to vary abruptly, particularly at the junctions of dissimilar objects - high frequency components.

Steps in Homomorphic Filtering

$$f(x,y) \rightarrow \boxed{\ln} \rightarrow \boxed{DFT} \rightarrow \boxed{H(u,v)} \rightarrow \boxed{(DFT)^{-1}} \rightarrow \boxed{\exp} \rightarrow g(x,y)$$

Consider $f(x,y) = i(x,y) \cdot r(x,y)$

Step 1: Apply logarithm

$$\underbrace{\ln(f(x,y))}_{z(x,y)} = \ln(i(x,y)) + \ln(r(x,y))$$

Step 2: Apply DFT

$$\mathcal{F}[\ln(f(x,y))] = \mathcal{F}[\ln(i(x,y))] + \mathcal{F}[\ln(r(x,y))]$$

This can be written as:

$$Z(u,v) = F_i(u,v) + F_r(u,v)$$

where $F_i(u,v)$ = Fourier transform of $\ln[i(x,y)]$

$F_r(u,v)$ = Fourier transform of $\ln[r(x,y)]$

Step 3 : Apply a filter.

$$S(u,v) = Z(u,v)\,H(u,v) = F_i(u,v)\,H(u,v) + F_r(u,v)\,H(u,v)$$

Step 4: Apply inverse fourier transform

$$s(x,y) = \mathcal{F}^{-1}\left[Z(u,v)\right] = \mathcal{F}^{-1}\left[F_i(u,v)\,H(u,v)\right] + \mathcal{F}^{-1}\left[F_r(u,v)\,H(u,v)\right]$$

$$s(x,y) = i'(x,y) + r'(x,y)$$

where $i' = \mathcal{F}^{-1}\left[F_i(u,v)\,H(u,v)\right]$

$r' = \mathcal{F}^{-1}\left[F_r(u,v)\,H(u,v)\right]$.

Step 5: Exponentiate to get result

$$g(x,y) = \exp(i'(x,y)) + \exp(r'(x,y))$$

$$\therefore \boxed{g(x,y) = i_o(x,y) \cdot r_o(x,y)}$$

Where $i_o = \exp(i'(x,y))$

$r_o = \exp(r'(x,y))$

Miscellaneous Filters

* $\boxed{\text{Notch Filter}}$: A constant with a hole in the center

$$H(u,v) = \begin{cases} 0, & \text{if } (u,v) = (M/2, N/2) \\ 1, & \text{otherwise.} \end{cases}$$

→ This filter takes out the average gray level of the image.

Unit-2 Numericals

① Correlation, Convolution - see pg. 11

② Histogram Equalization & Specification - see pg. 8

③ Intensity Transformations

CAT-1

Q. Obtain the digital negative of the following 3×3 gray scale image. Apply log, square root & exponential functions for the resultant image.

122	150	200
225	225	225
250	250	240

$L = 256$

Negative

negative $= L - 1 - r$

$= 255 - r$.

=

133	105	55
30	30	30
5	5	15

Logarithmic Transformation (on the resultant image, i.e on the digital negative)

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

let $c = 1$

$$s = \log(1+r)$$

⇒ The resultant image is:

2	2	2
1	1	1
1	1	1

Square Root Transformation

$$S = cr^{\gamma}$$

here $\gamma = 1/2$, let $c = 1$

⇒ The resultant image is:

12	10	7
5	5	5
2	2	4

Exponential Function Transformation (Inverse log transformation)

$$s = c \log^{-1}(1+r)$$

let $c = 1$

⇒ The resultant image is:

(4) Piecewise Linear Transformations

Bit Plane Slicing?

1. Apply bit plane slicing to the given image.

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

Here, $L = 8$

$n = 3$

convert each pixel into its binary equivalent -

then make the images corresponding to the MSB, middle & LSB planes.

100	011	101	010
011	110	100	110
010	010	110	101
111	110	100	001

MSB

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

middle

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

LSB

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

2. Apply bit plane slicing for the following image.

159	180	45
57	91	166
181	30	143

Ans. In binary form:

100 1111	10110100	00101101
001 1101	01011011	10100110
10110101	00110010	10001111

1	1	0
0	0	1
1	0	1

0	0	0
0	1	0
0	0	0

- - - - .

1	0	1
1	1	0
1	0	1

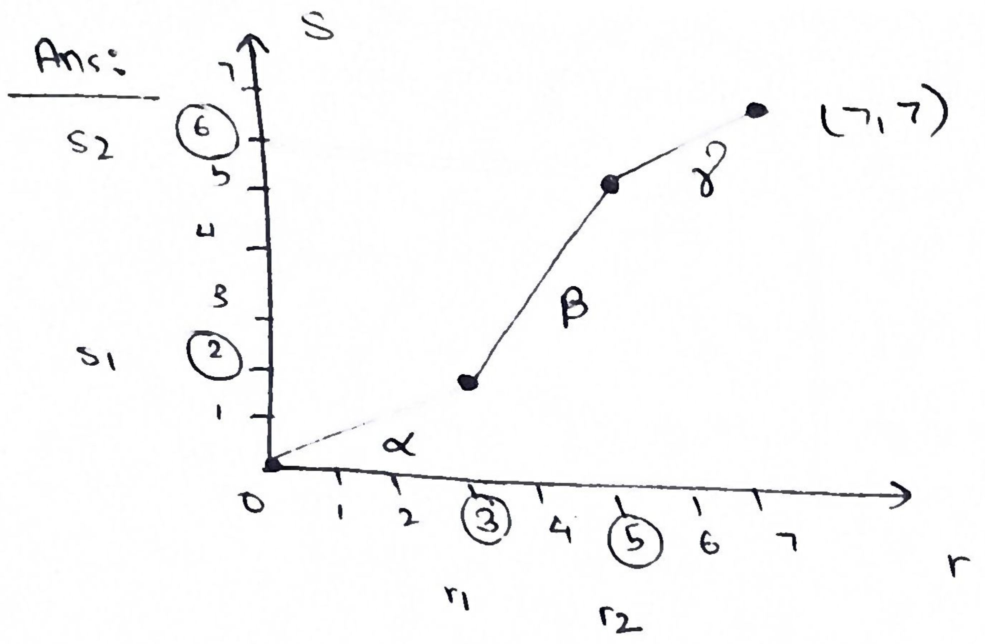
Contrast Stretching & Intensity Level Slicing

Contrast Stretching

1. Apply contrast stretching to the following image, given that $r_1 = 3$ and $s_1 = 2$ and $r_2 = 5$ and $s_2 = 6$

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

Ans:



Find the slope values for α, β and γ.

$$\alpha = \frac{2-1}{3-0} = 0.66$$

$$\beta = \frac{6-2}{5-3} = \frac{4}{2} = 2$$

$$\gamma = \frac{7-6}{7-5} = \frac{1}{2} = 0.5$$

The ranges are given as:

$$\boxed{\begin{array}{l} r_1 = 3 \\ s_1 = 2 \end{array}}$$

$$s = \begin{cases} \alpha \cdot r & 0 \leq r \leq 3 \\ \beta(r - r_1) + s_1 & 3 \leq r \leq 5 \\ \gamma(r - r_2) + s_2 & 5 \leq r \leq 7 \end{cases}$$

0, 1, 2

3, 4

5, 6

$$\boxed{\begin{array}{l} r_2 = 5 \\ s_2 = 6 \end{array}}$$

r	s
0	$\alpha \cdot (0) = 0$
1	$\alpha(1) = 0.66 = 1$
2	$\alpha(2) = 0.66 \times 2 = 1.32 = 1$
3	$\beta(3-3) + 2 = 2$
4	$\beta(4-3) + 2 = 2(1) + 2 = 4$
5	$\gamma(5-5) + 6 = 6$
6	$\gamma(6-5) + 6 = 0.5(1) + 6 = 6.5 = 7$
7	$\gamma(7-5) + 6 = 0.5(2) + 6 = 7$

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

∴ The output image would be:

4	2	6	1
2	7	4	7
1	1	7	6
7	7	4	1

2. Apply contrast stretching to the following image (Here, the r_i and s_i values are not given)

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

Ans. When the r_i and s_i values are not given; use the formula.

$$l = \frac{(l_{max} - l_{min})(m - m_{min})}{(m_{max} - m_{min})} + l_{min}$$

Here: $l_{max} = 7$ $\quad$ $m_{max} = 6$

$l_{min} = 0$ $\quad$ $m_{min} = 1$

$$\therefore \; l = \frac{(7) \times (m-1)}{5} = \frac{7}{5} \times (m-1)$$

$m = 0 \Rightarrow \frac{7}{5}(-1) = -1.4 = 0$

$m = 1 \Rightarrow 0$

$m = 2 \Rightarrow \frac{7}{5} \times 1 = 1$

$m = 3 \Rightarrow \frac{7}{5} \times 2 = 2.8 = 3$

$m = 4 \Rightarrow \frac{7}{5} \times 3 = 4.2 = 4$

$m = 5 \Rightarrow \frac{7}{5} \times 4 = 5.6 = 6$

$m = 6 \Rightarrow \frac{7}{5} \times 5 = 7$

$m = 7 \Rightarrow \frac{7}{5} \times 6 = 8.4 = 8 = 7$ (max gray-level value)

Find the frequency of occurrence of each intensity value in og image.

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

populate the new frequency distribution based on the calculated m values.

(check $m = \square$ value, look up that value in og table)

i	0	1	2	3	4	5	6	7
ni	4	4	0	~~3~~ 1	~~4~~ 2	~~6~~ 0	~~7~~ 2	~~8~~ ~~0~~ 3

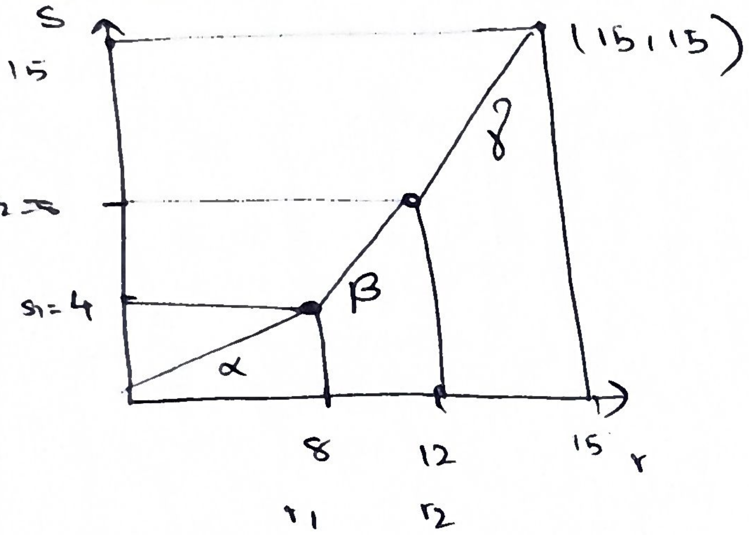
3. Apply ~~bitplane slicing~~ contrast stretching to the following image.

$$\begin{bmatrix} 7 & 12 & 2 & 3 \\ 10 & 15 & 1 & 1 \\ 12 & 4 & 6 & 15 \\ 8 & 2 & 7 & 15 \end{bmatrix}$$

given that $r_1 = 8$, $r_2 = 12$

$s_1 = 4$ $s_2 = 8$

Solution



calculation of slope values

$$\alpha = \frac{4-0}{8-0} = \frac{1}{2} = 0.5$$

$$\beta = \frac{8-4}{12-8} = \frac{4}{4} = 1$$

$$\gamma = \frac{15-8}{15-12} = \frac{7}{3} = 2.33$$

The ranges would be:

$$s = \begin{cases} \alpha r & 0 \le r \le 8 \\ \beta(r-r_1)+s_1, & 8 \le r \le 12 \\ \gamma(r-r_2)+s_2, & 12 \le r \le 15 \end{cases} = \begin{cases} 0.5r & 0 \le r \le 8 \\ (r-8)+4 & 8 \le r < 12 \\ 2.33(r-12)+8 & 12 \le r \le 15 \end{cases}$$

r	s
0	0.5×0 = 0
1	0.5×1 = 0.5 ≈ 1
2	0.5×2 = 1 = 1
3	0.5×3 = 1.5 = 2
4	0.5×4 = 2 = 2
5	0.5×5 = 2.5 = 3
6	0.5×6 = 3
7	0.5×7 = 3.5 = 4
8	(8−8)+4 = 4
9	(9−8)+4 = 5
10	(10−8)+4 = 6
11	(11−8)+4 = 7
12	2.33(0)+8 = 8
13	2.33(1)+8 = 10
	2.33(2)+8 = 13
	2.33×3+8 = 15

Output Image

4	8	1	2
6	15	1	1
8	2	3	15
4	1	4	15

4. Given the gray values and new range from 10-60 & 120-180 respectively. Find $f(x,y) = 15$

$$G(x,y) = \frac{\max(new) - \min(new) \times (f(x,y) - \min(old))}{\max(old) - \min(old)} + \min(new)$$

new = l

old = m

&

$$\frac{(l_{max} - l_{min})(m - m_{min})}{(m_{max} - m_{min})} + l_{min}$$

$$= \frac{(180-120)}{(60-10)} \times (15-10) + 120$$

$$= \underline{\underline{126}}$$

Intensity Level Slicing

1. Perform intensity level slicing on a 3bit image with A = 3 and B = 5, with background and without background

I =

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

Ans.

without background

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

with background

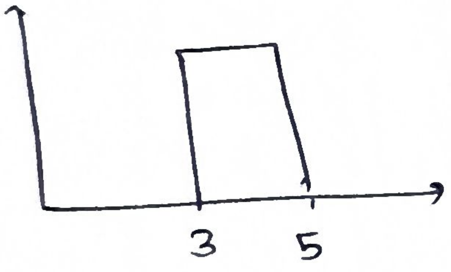
2	1	2	2	1
2	7	7	7	2
6	2	7	6	0
2	6	6	7	1
0	7	2	2	1

2. Perform intensity level slicing with $r_1 = 3$ and $r_2 = 5$ without background and with background

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

Ans: **without background** (also called clipping)

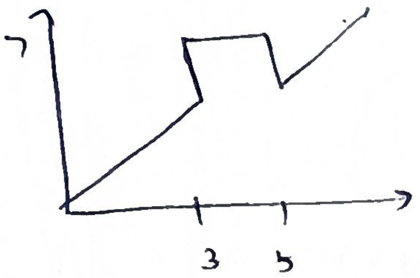
7	7	7	0
7	0	7	0
0	0	0	7
0	0	7	0



$$s = \begin{cases} L-1 = 7, & 3 \le r \le 5 \\ 0, & \text{otherwise.} \end{cases}$$

with background

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



$$s = \begin{cases} L-1 = 7 & 3 \le r \le 5 \\ r, & \text{otherwise} \end{cases}$$

Smoothing Filters

1. If an image is given as

1	5	7
2	4	8
3	6	9

, what would be the output of the box filter.

Ans: new value of center pixel $= \frac{1}{9}(1+5+7+2+4+8+3+6+9)$

$$= \frac{45}{9} = 5$$

output image =

1	5	7
2	5	8
3	6	9

2. Consider the following image. What would be the new value of the pixel (2,2) if smoothing is done using a 3x3 neighborhood

(a) mean filter

(b) weighted average filter

(c) median filter

(d) min filter

(e) max filter

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

Consider a 3x3 mask to be:

1	1	1
1	2	1
1	1	1

Ans 3x3 neighborhood is:

7	7	4
6	4	3
1	0	7

(i) mean filter $= \frac{1}{9}(7+7+4+6+4+3+1+0+7)$

$= \frac{39}{9} = 4.33 = \underline{4}$

(ii) Weighted average filter $= \frac{1}{9}\{7+7+4+\cancel{4+4+6}^{6}+8+3+1+0+7\}$

$= \frac{1}{9} \times 43 = 4.77 = \underline{5}$

(iii) Median Filter

arranging in ascending order

0 1 3 4 4 6 7 7

new image =

7	7	4
6	4	3
1	0	7

(iv) **max Filter** : sub middle with 7

(v) **min Filter** : sub middle with 0

Laplacian Filter.

1. What would be the effect of applying the Laplacian Filter (a) on the image (b)?

(a)

1	1	1
1	-8	1
1	1	1

(b)

50	50	50	50	50	50
50	50	50	50	50	50
50	50	50	50	50	50
100	100	100	100	100	100
100	100	100	100	100	100
100	100	100	100	100	100

Ans. Pad the boundary of the image.

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
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

sq.1 = 50 + 50 + 50 + 50 +
50 + 50 + 50 + 50 +
- 400
= 0

move mask to the right

→ The resultant image is:

0	0	0	0	0	0
0	0	0	0	0	0
150	150	150	150	150	150
-150	-150	-150	-150	-150	-150
0	0	0	0	0	0
0	0	0	0	0	0

2. Apply the given Laplacian Filter with zero padding

0	-1	0
-1	5	-1
0	-1	0

23	43	102	234	201
112	167	189	199	204
156	188	167	187	178
138	49	85	45	56

Ans. Pad 0s around the original image

0	0	0	0	0	0	0
0	23	43	102	234	201	0
0	112	167	189	199	204	0
0	156	188	167	187	178	0
0	138	49	85	45	56	0
0	0	0	0	0	0	0

The resultant image would be:

-40	77	44	668	567
214	303	310	191	442
442	401	186	346	443
-15	-66	164	-103	57

Sharpening?

1. Appy the sharpening filter on the following image sequence

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

Ans

original img:	5	5	4	3	2	1	0	0	0	6	0	0	0	0	1	3	1	0	0	0	0	7	7	7	7
1st order derivative		-1	-1	-1	-1	-1	-1	0	0	6	-6	0	0	0	1	2	-2	-1	0	0	0	7	0	0	0
2nd order derivative			0																						

2nd derivative = $f(x+1) + f(x-1) - 2f(x)$

$-1 + -1 - 2(-1)$

$-2 \quad -2$