Image Enhancement Using Spatial Filtering

Kiran TALELE

☎ 99 870 30 881

Kiran TALELE

@ Bharatiya Vidya Bhavans' Sardar Patel Institute of Technology Andheri(w) Mumbai

- **Associate Professor,** Electronics Engineering Department (1997)
- **Dean,** Students, Alumni & External Relations (2022)

@ Sardar Patel Technology Business Incubator(SP-TBI), Funded by Department of Science & Technology(DST), Govt. of India

- **Head**, Academic Relations (2015)

@ IEEE Bombay Section

- **Treasurer** (2020)
- Executive Committee Member (2015)

Image Enhancement by Neighbourhood Processing : **Spatial Filtering**

(i) In Neighbourhood opération, Tx. function works on a **group of input pixel values**

(ii) The Output pixel value at (x,y) position is obtained by **masking operation**

(iii) The use of spatial mask for image processing is called spatial filtering and the mask is called spatial filter

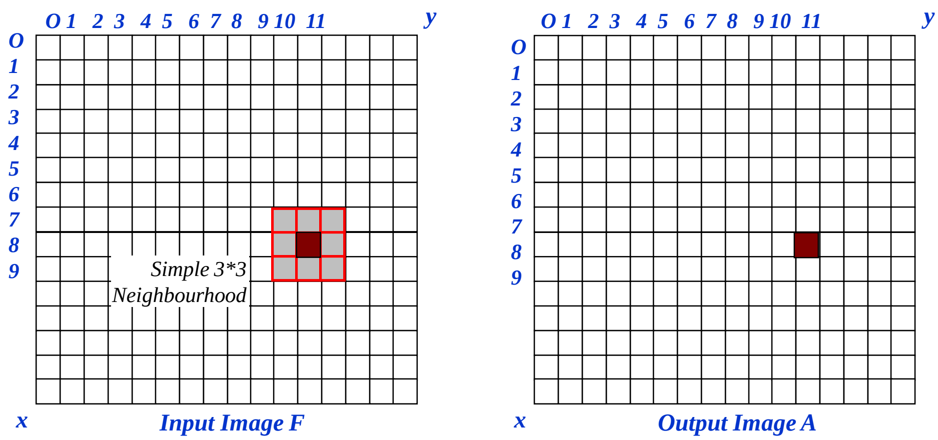
Masking Operation: Consider an image F and filter mask W

$$\mathbf{F} = \begin{array}{|c|c|c|}\hline Z_1 & Z_2 & Z_3 \\ \hline Z_4 & Z_5 & Z_6 \\ \hline Z_7 & Z_8 & Z_9 \\ \hline \end{array} \ast \quad w = \begin{bmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \\ w_7 & w_8 & w_9 \end{bmatrix} \qquad \mathbf{A} = \begin{array}{|c|c|c|}\hline & & \\ \hline & R & \\ \hline & & \\ \hline \end{array}$$

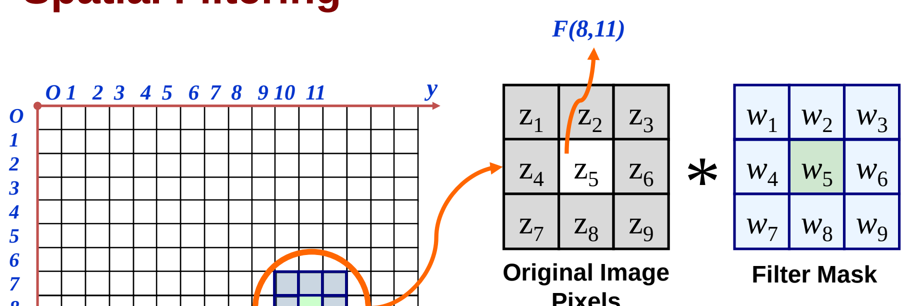
The response of mask is given by,

$$\mathbf{R = z_1}\, w_1 + \mathbf{z_2}\, w_2 + \mathbf{z_3}\, w_3 + \ldots + \mathbf{z_9}\, w_9$$

Spatial Filtering



Spatial Filtering



The response of Filter at A(8,11) position is given by,

$$A(8,11) = \mathbf{z_1} w_1 + \mathbf{z_2} w_2 + \mathbf{z_3} w_3 + \ldots + \mathbf{z_9} w_9$$

The above is repeated for every pixel in the original image to generate the smoothed image

List of Spatial Filters

[I] Smoothing Linear Filters (3)

Eg. Low Pass Averaging Filter, Weighted Average Filter, Trimmed Averaging Filter and Geometric Mean Filter

[II] Smoothing Non-Linear Filters (3)

Eg. Median, Max and Min Filter.

[III] Sharpening : First Order Derivative Filters (4)

Eg. Robert, Prewit, Sobel and Fri-Chen filter

[IV] Sharpening Second Order Derivative Filters (3)

Eg. Laplacian Filter, HPF and High Boost Filter

[I] Smoothing Linear Filters

(1) Low Pass Averaging Filter

(i) LPF attenuates High frequency Components and allows to pass LOW frequency components of the image.

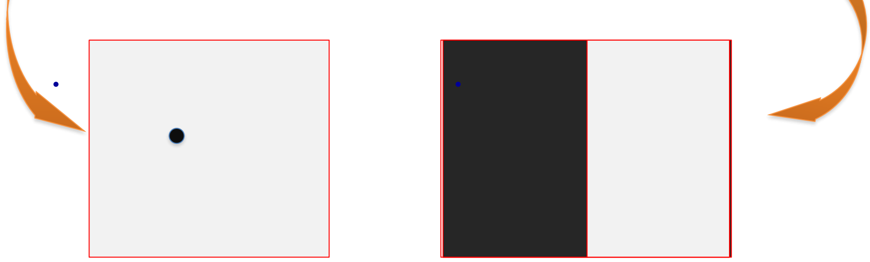
(ii) LPF reduces the difference between neighbouring pixels by distributing its' energy using averaging technique.

(iii) LPF removes sharp transitions in the image

- Sharp transitions are due to :

(1) High frequency random Impulse Noise.

(2) High frequency Edges of an object.



Eg. Consider a Digital Image F with impulse noise

$F =$

0	0	0
0	200	0
0	0	0

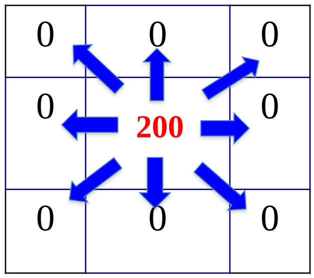
L P F

$A =$

(IV) LPF reduces the difference between the neighbouring pixels by distributing its energy in all directions by averaging techniques

(v) The net effect of LP filtering is Image Blurring

Eg. Digital Image with impulse noise



LPF

22	22	22
22	22	22
22	22	22

LPF reduces the difference between the neighbouring pixels by distributing its energy in all directions by averaging techniques

Eg-1 : Digital Image with impulse noise

$F =$

0	1	2
7	255	3
6	5	4

LPF

Case-1 : Assume virtual Rows and Columns with Zero pixel Values

$F =$

0	0	0	0	0
0	0	1	2	0
0	7	255	3	0
0	6	5	4	0
0	0	0	0	0

LPF

$A =$

Eg-2 : Consider a Digital Image with impulse noise

$F =$

0	1	2
7	255	3
6	5	4

L P F

$F =$

0	0	0	0	0
0	0	1	2	0
0	7	255	3	0
0	6	5	4	0
0	0	0	0	0

L P F

$A =$

29	30	30
30	31	30
30	33	30

$F =$

0	1	2
7	255	3
6	5	4

L P F

$A =$

29	30	30
30	31	30
30	33	30

All the pixel in the neighourhood of impulse noise have almost same value.
Thus high frequency impulse noise is completely suppressed in the output image.

Digital Image with impulse noise

$F =$

0	1	2
7	255	3
6	5	4

L P F

30	30	30
32	31	31
34	33	32

Case-2 : Assume Virtual Rows and Columns with repeated border pixel Values

$F =$

0	**0**	**1**	**2**	**2**
0	0	1	2	**2**
7	7	255	3	**3**
6	6	5	4	**4**
6	**6**	**5**	**4**	**4**

L P F

$A =$

30	30	30
32	31	31
34	33	32

Eg-2. Digital Image with High Frequency Edge

10	10	200	200
10	10	200	200
10	10	200	200

L P F

10	**73**	**137**	**200**
10	**73**	**137**	**200**
10	**73**	**137**	**200**

10	10	10	200	200	200
10	**10**	**10**	**200**	**200**	200
10	**10**	**10**	**200**	**200**	200
10	**10**	**10**	**200**	**200**	200
10	10	10	200	200	200

	10	**73**	**137**	**200**	
	10	**73**	**137**	**200**	
	10	**73**	**137**	**200**	

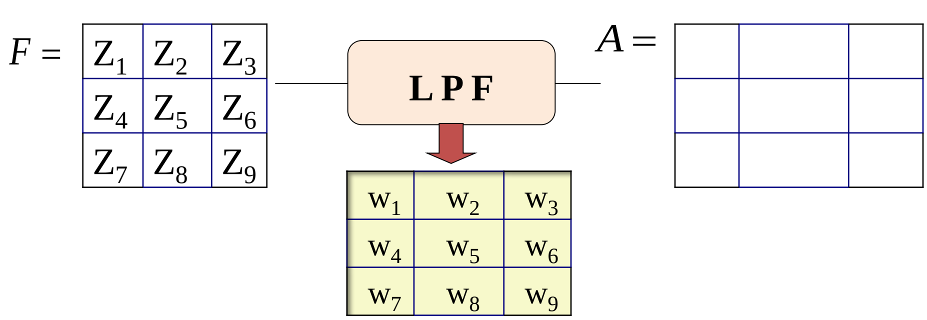
Summary :

- All the pixels in the output image have almost same value.
- Averaging of gray values in the neighbourhood reduces the sharp transitions in the images.
- Thus High frequency Impulse noise is completely suppressed. This is desirable effect.
- However high frequency edges are blurred.

  This is undesired effect.

Derivation of LPF Mask

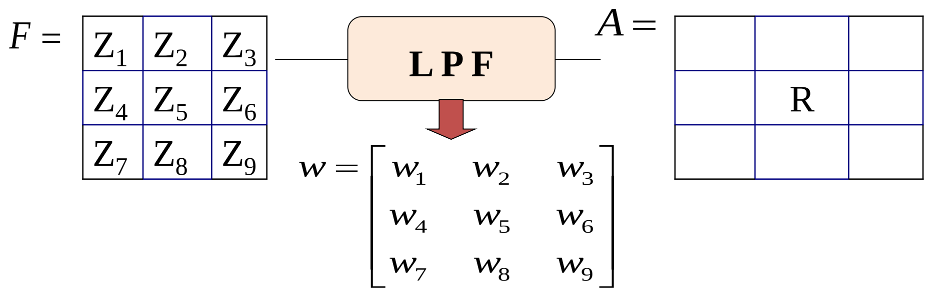
Consider a Digital Image F and LPF Mask w as shown in figure below,



The response of LPF mask at A(x,y) position is given by,

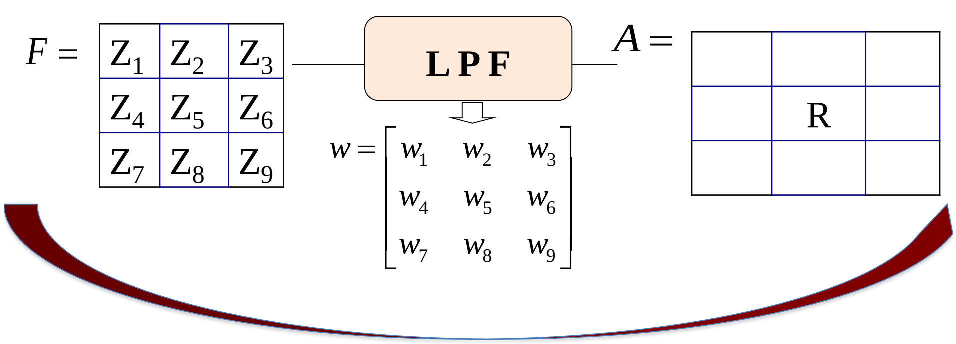
$$\mathbf{R = z_1\,w_1 + z_2\,w_2 + z_3\,w_3 + \ldots + z_9\,w_9 \quad \ldots\ldots(I)}$$

Consider a Digital Image F and LPF Mask w as shown in figure below,



The response of LPF mask at A(x,y) position is given by,

$$\mathbf{R = z_1\, w_1 + z_2\, w_2 + z_3\, w_3 + \ldots + z_9\, w_9 \quad \ldots\ldots(I)}$$



By averaging output pixel value at A(x,y) position is given by,

$$\mathbf{R = \frac{1}{9}(z_1 + z_2 + z_3 + \ldots + z_9)}$$

$$\mathbf{R = \frac{1}{9} z_1 + \frac{1}{9} z_2 + + \ldots + \frac{1}{9} z_9 \ ) \ldots\ldots (II)}$$

$$R = z_1 w_1 + z_2 w_2 + z_3 w_3 + \ldots + z_9 w_9 \quad \ldots\ldots(I)$$

$$R = \frac{1}{9} z_1\, z_1 + \frac{1}{9} \quad z_2 + + \ldots + \frac{1}{9} \quad z_9 \quad \ldots\ldots (II)$$

By equating equation (I) and (II) we get,

w =

$W_1 = \frac{1}{9}$	$W_2 = \frac{1}{9}$	$W_3 = \frac{1}{9}$
$W_4 = \frac{1}{9}$	$W_5 = \frac{1}{9}$	$W_6 = \frac{1}{9}$
$W_7 = \frac{1}{9}$	$W_8 = \frac{1}{9}$	$W_9 = \frac{1}{9}$

w = $\frac{1}{9}$

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

(2) Low Pass Weighted Averaging Filters

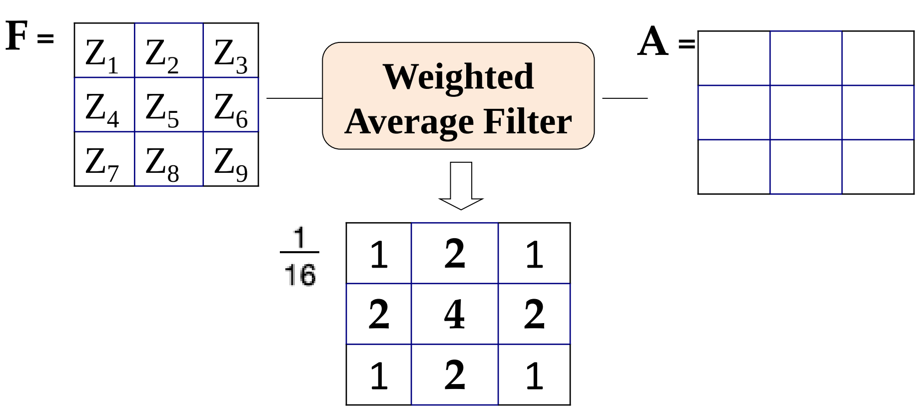
(i) In this filter, output pixel value at each (x,y) position is obtained by averaging the weighted neighbouring pixel values.

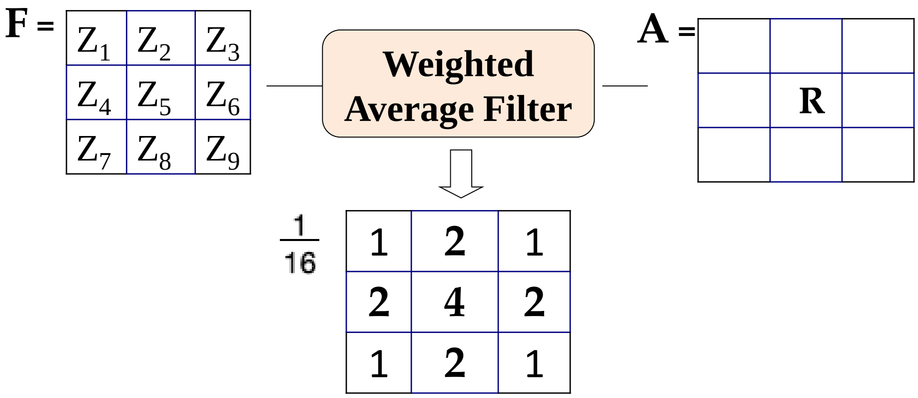
(ii) The **Weight** (i.e. Scaling factor) depends on the position of the neighbouring pixel with reference to the center pixel.

	Pixel Position	Weight
1	Center Pixel	4
2	4 Directional Neighbour	2
3	Diagonal Direction Neighbour	1

$$F = \begin{array}{|c|c|c|} \hline Z_1 & Z_2 & Z_3 \\ \hline Z_4 & Z_5 & Z_6 \\ \hline Z_7 & Z_8 & Z_9 \\ \hline \end{array}$$

Consider a Digital Image F and filter mask w as shown in fig below,





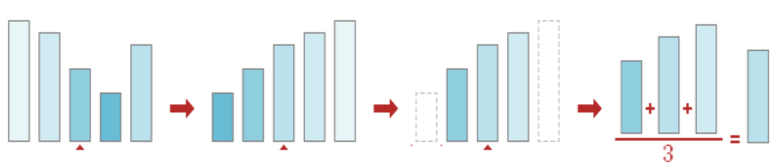
The response of mask at A(x,y) position is given by,

$$R = \frac{1}{16}\left[(z_1 + 2z_2 + z_3) + (2z_4 + 4z_5 + 2z_6) + (z_7 + 2z_8 + z_9)\right]$$

(3) Trimmed Averaging Filter

A non-linear smoothing filter.

This filter deals well with images containing spiky noise, but it is less effective at attenuating random noise



Trimmed Mean Filter ALGORITHM :

1. Place a window over element;
2. Pick up elements;
3. Order elements;

4.Discard elements at the beginning and at the end of the got
5.ordered set;
5.Take an average — sum up the remaining elements and
divide the sum by their number.

Since the filter applies a mean to the untrimmed data, it is not edge preserving

(4) Alpha (α)-Trimmed Averaging Filter

A non-linear smoothing filter.

For ex. Consider below window size

F =

20	20	18
21	19	12
19	22	10

A =

	A(x,y)	

The ordered list is :
F = { 8, 10, 12, 19, 19, 20, 20, 21, 22}

To find A(x,y)

For different values of α, the output is as follows:

Case-1 : When α = 0

F = { 8, 10, 12, 19, 19, 20, 20, 21, 22}

A(x,y) = 18.8

Case-2 : When α = 1

F = {$\cancel{8}$, 10, 12, 19, 19, 20, 20, 21, $\cancel{22}$}

A(x,y) = 17.3

Case-3 : When α = 2

F = {$\cancel{8}$, $\cancel{10}$, 12, 19, 19, 20, 20, $\cancel{21}$, $\cancel{22}$}

A(x,y) = 18.0

Case-4 : When α = 3

F = {$\cancel{8}$, $\cancel{10}$, $\cancel{12}$, 19, 19, 20, $\cancel{20}$, $\cancel{21}$, $\cancel{22}$}

A(x,y) = 19.3

Case-5 : When α = 4

F = {$\cancel{8}$, $\cancel{10}$, $\cancel{12}$, $\cancel{19}$, 19, $\cancel{20}$, $\cancel{20}$, $\cancel{21}$, $\cancel{22}$}

A(x,y) = 19.0

This filter is a combination of mean filter and median filter as you said.

This is an algorithmic approach that tries to combine properties of the mean filter with properties of the median filter.

(5) Geometric Mean Filter

The geometric mean filter is an image filtering process meant to smooth and reduce noise of an image.

It is based on the mathematic geometric mean.

The output image G(x,y) of a geometric mean is given by

$$G(x,y) = \left[\prod F(i,j) \right]^{\frac{1}{mn}}$$

Eg. Consider a digital subimage F

F =

5	16	22
6	3	18
12	3	15

→ **Geometric Mean Filter** →

A =

	A(x,y)	

The output pixel value at A(x,y) position is given by,

A(x,y) = (5 * 16 * 22 * 6 * 3 * 18 * 12 * 3 * 15)^(1/9)

A(x,y) = 8.77

[II] Smoothing Non-Linear Filters
[also called as ordered statistic filters]

(1) Median Filter

(i) Medial filter is a Non-Linear ordered statistic filter.

(ii) Output pixel value at (x,y) position is obtained by selecting the median of neighbourhood of input pixel value.

* Median Filter ALGORITHM

(I) Arrange the pixels in the window either in Increasing order OR in Decreasing order.

(II) Select the middle value

If number of pixels in the window are even,

Then find average of the middle two values.

(III) Replace the input pixel value by the selected median value in the output image.

Eg. Consider a Digital Image F

F =

41	**200**	60
52	**45**	**72**
10	**58**	81

Median Filter

A =

	A(x,y)	

1. Arrange All the pixel values in asceding order

10	**41**	45	52	58	60	72	81	200

2. Select the middle value

A(x,y) = 58

Ex of Digital Image with Salt and Pepper Noise :

F =

41	200	60
52	45	42
0	58	51

Salt Noise Value = 200 (White)

Pepper Noise Value = 0 (Black)

Most of the pixels in the image are in the range [40 to 60]

There are two extreme values in the image :
i.e. 200 (Salt Noise) and 0 (Pepper Noise)

F =

41	200	60
52	45	42
0	58	51

Median Filter

A =

To remove noisy pixels, assume virtual rows and columns with repeated border pixel values

F =

41	41	200	60	60
41	41	200	60	60
52	52	45	42	42
0	0	58	51	51
0	0	58	51	51

Median Filter

A =

45	52	60
45	51	51
45	51	51

F =

41	200	60
52	45	42
0	58	51

Median Filter

A =

45	52	60
45	51	51
45	51	51

Input image with Salt and Pepper Noise

Median Filtered output image with eliminated Salt and Pepper Noise values

F =

41	200	60
52	45	42
0	58	51

A =

45	52	60
45	51	51
45	51	51

- **Median Filter forces the points with distinct gray levels to be more like their neighbours**
- **When all the pixel values are arranged in order, distinct values occupy extreme positions and Median filter selects median i.e. middle vales of ordered pixel values.**
- **Therefore, noisy pixel values are eliminated median filter**

F =

41	200	60
52	45	42
0	58	51

A =

45	52	60
45	51	51
45	51	51

- **Eliminated noisy pixel values are replaced by one of the neighbouring pixel values.**
- **Therefore, Median Filter is the most suitable to remove Salt and Pepper Noise in the image.**

(2) Max Filter

(i) Max filter is a Non-Linear ordered statistic filter.

(ii) Output pixel value at (x,y) position is obtained by selecting the Maximum of neighbourhood of input pixel value.

(3) Min Filter

(i) Min filter is a Non-Linear ordered statistic filter.

(ii) Output pixel value at (x,y) position is obtained by selecting the Minimum of neighbourhood of input pixel value.

Eg.

F =

41	200	60
52	45	42
0	58	51

Max Filter

A =

	200	

F =

41	200	60
52	45	42
0	58	51

Min Filter

A =

	0	

Sharpening Derivative Filters

Concept :

Smoothing Filter **Ex. LPF**	**SUM**	**Integration**
Sharpening Filter **Ex. HPF**	**Difference**	**Differentiation**

[III] Sharpening First Order Derivative filters [also called as Gradient Filters]

Eg: Robert, Prewit, Sobel and Fri-chen filters

- **Method of differentiation is Gradient**
- **The Gradient of F at (x,y) position is defined as,**

$$\nabla f = \left(\frac{\partial f}{\partial x} \; \frac{\partial f}{\partial y} \right) = \left[G_x \, , G_y \right]$$

Where,

$$G_x = \frac{\partial f}{\partial x} = \lim_{h \to 1} \frac{f(x+h, y) - f(x,y)}{h}$$

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

$$G_y = \frac{\partial f}{\partial y} = \lim_{h \to 1} \frac{f(x, y+h) - f(x,y)}{h}$$

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

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

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

F =

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

1. Robert Filter

(1^{st} Order Derivative Sharpening filter)

F =

	y		
x	z_1	z_2	z_3
	z_4	z_5	z_6
	z_7	z_8	z_9

- To find Gx

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

$$G_x = z_8 - z_5$$

G_x =

	-1	0
	1	0

- To find Gy

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

$$G_y = z_6 - z_5$$

G_y =

	-1	1
	0	0

2. Prewit Filter

(1st Order Derivative - Sharpening filter)

x \ y			
	z_1	z_2	z_3
x	z_4	z_5	z_6
	z_7	z_8	z_9

- **To find Gx**

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

$Gx =$

-1	-1	-1
0	0	0
1	1	1

- **To find Gy**

$Gx = (z_3 - z_1) + (z_6 - z_4) + (z_9 - z_7)$

$Gy =$

-1	0	1
-1	0	1
-1	0	1

Kiran TALELE 49

3. Sobel Filter

(1st Order Derivative - Sharpening filter)

x \ y			
	z_1	z_2	z_3
x	z_4	z_5	z_6
	z_7	z_8	z_9

- **To find Gx**

$Gx = (z_7 - z_1) + \mathbf{2}\,(z_8 - z_2) + (z_9 - z_3)$

$Gx =$

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

- **To find Gy**

$Gy = (z_3 - z_1) + \mathbf{2}\,(z_6 - z_4) + (z_9 - z_7)$

$Gy =$

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

Kiran TALELE 50

4. Fri-Chen Filter

(1st Order Derivative - Sharpening filter)

x \ y			
	z_1	z_2	z_3
x	z_4	z_5	z_6
	z_7	z_8	z_9

- **To find Gx**

$$Gx = (z_7 - z_1) + \sqrt{2}\,(z_8 - z_2) + (z_9 - z_3)$$

Gx =

-1	$-\sqrt{2}$	1
0	0	0
1	$\sqrt{2}$	1

- **To find Gy**

$$Gy = (z_3 - z_1) + \sqrt{2}\,(z_6 - z_4) + (z_9 - z_7)$$

Gy =

-1	0	1
$-\sqrt{2}$	0	$\sqrt{2}$
-1	0	1

1. The Magnitude of Gradient at (x,y) is given by,

$$|\nabla f| = \sqrt{Gx^2 + Gy^2}$$

$$|\nabla f| = |Gx| + |Gy|$$

2. Orientation at (x,y) is given by,

$$\alpha = \tan^{-1}\left(\frac{Gy}{Gx}\right)$$

3. To find response of the Gradient filter,

$$F = \begin{bmatrix} z_1 & z_2 & z_3 \\ z_4 & z_5 & z_6 \\ z_7 & z_8 & z_9 \end{bmatrix} \longrightarrow \boxed{\text{Gradient Filter}} \longrightarrow A = \begin{bmatrix} & & \\ & R & \\ & & \end{bmatrix}$$

where,

$$R = |Gx| + |Gy|$$

Example-1 :

$$F = \begin{bmatrix} 41 & 200 & 60 \\ 52 & 45 & 42 \\ 0 & 58 & 51 \end{bmatrix} \longrightarrow \boxed{\text{Prewit Gradient Filter}} \longrightarrow A = \begin{bmatrix} & & \\ & R & \\ & & \end{bmatrix}$$

Prewit Vertical Mask is given by,

$$Gx = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

(i) To find Gx

Gx = (-41) + (-142) + (-9)

Gx = -192

Similarly,

F =

41	200	60
52	45	42
0	58	51

Prewit Gradient Filter

A =

	R	

Prewit Horizontal Mask is given by,

-1	0	1
-1	0	1
-1	0	1

(ii) To find Gy

Gy = (19) + (-10) + (51)

Gx = 60

Example...

F =

41	200	60
52	45	42
0	58	51

Prewit Gradient Filter

A =

	R	

(iii) To find R

$$R = |Gx| + |Gy|$$

$$R = (192) + (60)$$

$$R = (252)$$

[IV] Second Order Derivative filters

Eg. Laplacian Filter, HPF and High Boost Filter

(2) Laplacian Filter

The Laplacian filter is Second Order Derivative filter

The Laplacian of Digital image f at (x,y) position is given by,

$$\nabla^2 f = G_x^2 + G_y^2$$

(i) Find G^2x

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

Gx = f(x+1,y) - f(x,y)

G^2x = [f(x+1,y) - f(x,y)] – [f(x,y) - f(x-1,y)]

G^2x = f(x+1,y) – 2 f(x,y) + f(x-1,y)

G^2x =

0	1	0
0	-2	0
0	1	0

$\xrightarrow{\text{Scale by (-1)}}$

G^2x =

0	- 1	0
0	2	0
0	- 1	0

(i) Find G^2y

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

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

$$G^2y = [\, f(x, y+1) - f(x,y)] - [\, f(x,y) - f(x, y-1)\,]$$

$$G^2y = f(x, y+1) - 2\, f(x,y) + f(x, y-1)$$

$G^2y =$

0	0	0
1	-2	1
0	0	0

Scale by (-1)

$G^2y =$

0	0	0
-1	2	-1
0	0	0

(iii) Find $\nabla^2 f$

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

$$\nabla^2 f = G_x^2 + G_y^2$$

=

0	1	0
0	2	0
0	-1	0

+

0	0	0
-1	2	-1
0	0	0

$\nabla^2 f =$

0	-1	0
-1	4	-1
0	-1	0

Laplacian 4 directional Mask

(iv) Find $\nabla^2 f$

$$\nabla^2 f = \underset{(\text{4 Directional})}{\nabla^2 f} + \underset{(\text{Diagonal Directional})}{\nabla^2 f}$$

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

$$\nabla^2 f = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Laplacian 8 directional Mask

(2) High Pass Filter

(i) HPF attenuates LOW frequency Components and allows to pass HIGH frequency components of the image.

(ii) HPF image can be obtained by subtracting LPF image from the original image.

(iii) HPF Mask is given by,

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

Eg. Consider a Digital Image F as given below

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

H P F

$$W = \frac{1}{9}\begin{array}{|c|c|c|}\hline -1 & -1 & -1 \\ \hline -1 & 8 & -1 \\ \hline -1 & -1 & -1 \\ \hline\end{array}$$

0	30	30	0
0	30	30	0
0	30	30	0

10	10	10	100	100	100
10	**10**	**10**	**100**	**100**	100
10	**10**	**10**	**100**	**100**	100
10	**10**	**10**	**100**	**100**	100
10	10	10	100	100	100

	0	30	30	0	
	0	30	30	0	
	0	30	30	0	

Derivation of HPF Mask

Consider a Digital Image F and HPF Mask w as shown in figure below,

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

L P F

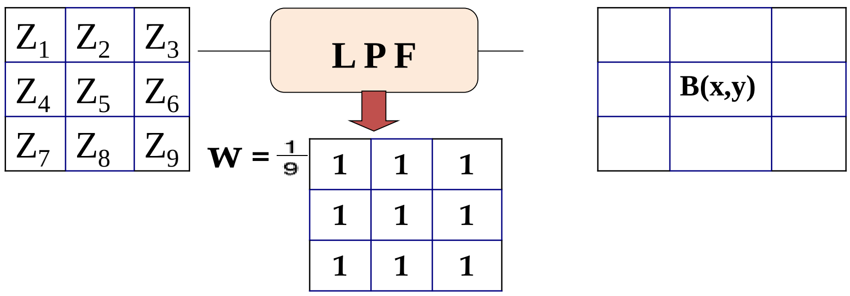
$$w = \begin{bmatrix} w_1 & w_2 & w_3 \\ w_4 & w_5 & w_6 \\ w_7 & w_8 & w_9 \end{bmatrix}$$

	A(x,y)	

The response of HPF mask at A(x,y) position is given by,

$$\mathbf{A(x,y)} = \mathbf{z_1}\, w_1 + \mathbf{z_2}\, w_2 + \mathbf{z_3}\, w_3 + \ldots + \mathbf{z_9}\, w_9 \quad \ldots\ldots(\mathrm{I})$$

Consider a Digital Image F and LPF Mask w as shown in figure below,

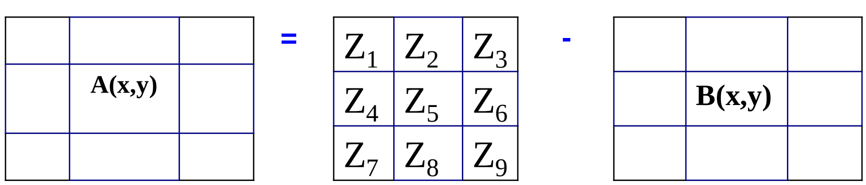


The response of LPF mask at B(x,y) position is given by,

$$\mathbf{B(x,y)} = \frac{1}{9} z_1 + \frac{1}{9} z_2 + + \dots \frac{1}{9} z_9 \quad \dots\dots \text{(II)}$$

Now, **Original Image = LPF Image + HPF Image**

HPF Image = Original Image - LPF Image



At (x,y) position,

A(x,y) = Z_5 - **B(x,y)**

Now, $A(x,y) = Z_5 - B(x,y)$

Where, $B(x,y) = \frac{1}{9} z_1 \; - \frac{1}{9} z_2 + + \ldots \frac{1}{9} z_9$

$$A(x,y) = Z_5 - \left(\frac{1}{9} z_1 + \frac{1}{9} z_2 + + \ldots + \frac{1}{9} z_9 \right)$$

$$= Z_5 - \frac{1}{9} z_1 - \frac{1}{9} z_2 - \frac{1}{9} z_3 - \frac{1}{9} z_4 - \frac{1}{9} z_5 \ldots \frac{1}{9} z_9 \,)$$

$$A(x,y) = -\frac{1}{9} z_1 - \frac{1}{9} - \frac{1}{9} - \frac{1}{9} + \frac{8}{9} \ldots - z \frac{1}{9} \text{ ----(III)}$$

$$A(x,y) = z_1 w_1 + z_2 w_2 + z_3 w_3 + \ldots + z_9 w_9 \text{ ......(I)}$$

$$A(x,y) = -\frac{1}{9} z_1 - \frac{1}{9} - \frac{1}{9} - \frac{1}{9} + \frac{8}{9} \ldots - z \frac{1}{9} \text{ ----(III)}$$

By equating equation (I) and (III) we get,

W =

$W_1 = \frac{-1}{9}$	$W_2 = \frac{-1}{9}$	$W_3 = \frac{-1}{9}$
$W_4 = \frac{-1}{9}$	$W_5 = \frac{8}{9}$	$W_6 = \frac{-1}{9}$
$W_7 = \frac{-1}{9}$	$W_8 = \frac{-1}{9}$	$W_9 = \frac{-1}{9}$

W = $\frac{1}{9}$

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

(3) High Boost Filter

High Boost Filter (HBF) Image is obtained by adding HPF image with the original Image.

HBF Image = Original Image + HPF Image

$$\begin{array}{|c|c|c|}\hline & & \\ \hline & A(x,y) & \\ \hline & & \\ \hline\end{array} = \begin{array}{|c|c|c|}\hline Z_1 & Z_2 & Z_3 \\ \hline Z_4 & Z_5 & Z_6 \\ \hline Z_7 & Z_8 & Z_9 \\ \hline\end{array} + \begin{array}{|c|c|c|}\hline & & \\ \hline & B(x,y) & \\ \hline & & \\ \hline\end{array}$$

At (x,y) position,

$$\mathbf{A(x,y)} = Z_5 + \mathbf{B(x,y)}$$

Now, $\mathbf{A(x,y)} = Z_5 + \mathbf{B(x,y)}$

Where, $\mathbf{B(x,y)} = -\frac{1}{9}z_1 - \frac{1}{9}z_2 - \frac{1}{9}z_3 - \frac{1}{9}z_4 - \frac{1}{9}z_5 \dots - \frac{1}{9}z_9\,)$

By Substituting,

$$A(x,y) = Z_5 - \frac{1}{9}z_1 - \frac{1}{9}z_2 - \frac{1}{9}z_3 - \frac{1}{9}z_4 + \frac{8}{9}z_5 \dots - \frac{1}{9}z_9\,)$$

$$A(x,y) = -\frac{1}{9}z_1 - \frac{1}{9}z_2 - \frac{1}{9}z_3 - \frac{1}{9}z_4 + \frac{17}{9}z_5 \dots - \frac{1}{9}z_9\,)$$

$$A(x,y) = -\frac{1}{9} z_1 - \frac{1}{9} z_2 - \frac{1}{9} z_3 - \frac{1}{9} z_4 + \frac{17}{9} z_5 \ldots - \frac{1}{9} z_9)$$

$$\text{Let } A(x,y) = z_1 w_1 + z_2 w_2 + z_3 w_3 + \ldots + z_9 w_9$$

By equating, HBF Mask is then given by,

W =

$W_1 = \frac{-1}{9}$	$W_2 = \frac{-1}{9}$	$W_3 = \frac{-1}{9}$
$W_4 = \frac{-1}{9}$	$W_5 = \frac{17}{9}$	$W_6 = \frac{-1}{9}$
$W_7 = \frac{-1}{9}$	$W_8 = \frac{-1}{9}$	$W_9 = \frac{-1}{9}$

W = $\frac{1}{9}$

-1	-1	-1
-1	17	-1
-1	-1	-1

Dr. Kiran TALELE

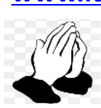
- Academic : PhD
- Professional :
  - Dean-Students, Alumni & External Relations @ Bharatiya Vidya Bhavans' Sardar Patel Institute of Technology (SP-IT), Mumbai
  - Head-Academic Relation @ Sardar Patel Technology Business Incubator (SP-TBI), Mumbai
  - Treasurer-IEEE Bombay Section

091-9987030881

kiran.talele@spit.ac.in / ktvtalele@gmail.com

https://www.linkedin.com/in/k-t-v-talele/

www.facebook.com/Kiran-Talele-1711929555720263



Stay Connected

SARDAR PATEL TECHNOLOGY BUSINESS INCUBATOR

- **Dr. Kiran TALELE** is an Associate Professor in Electronics & Telecommunication Engineering Department of Bharatiya Vidya Bhavans' Sardar Patel Institute of Technology, Mumbai with 33+ years experience in Academics.

- He is a Dean of Students, Alumni and External Relations at Sardar Patel Institute of Technology, Andheri Mumbai.
  He is also a Head of Sardar Patel Technology Business Incubator, Mumbai.

- His area of research is Digital Signal & Image Processing, Computer Vision, Machine Learning and Multimedia System Design.

- **He has published 85+ research papers at various national & international refereed conferences and journals. He has filed published 12+ patents at Indian Patent Office. One patent is granted in 2021.**

- He is a Treasurer of IEEE Bombay Section and Mentor for Startup Incubation & Intellectual Asset Creation.

- He received incentives for excellent performance in academics and research from Management of S.P.I.T. in 2008-09. He is a recipient of P.R. Bapat IEEE Bombay Section Outstanding Volunteer Award 2019.