Al-Mustaqbal University

College of Engineering & Technology

Biomedical Engineering Department

Subject Name: Image processing

5 Class, Second Semester

Subject Code: [MO01015206]

Academic Year: 2025-2026

Lecturer: Assist lect. Hiba Diaa Alrubaie

Email: hiba.diaa.abdulameer@uomus.edu.iq

Lecture No.:-3

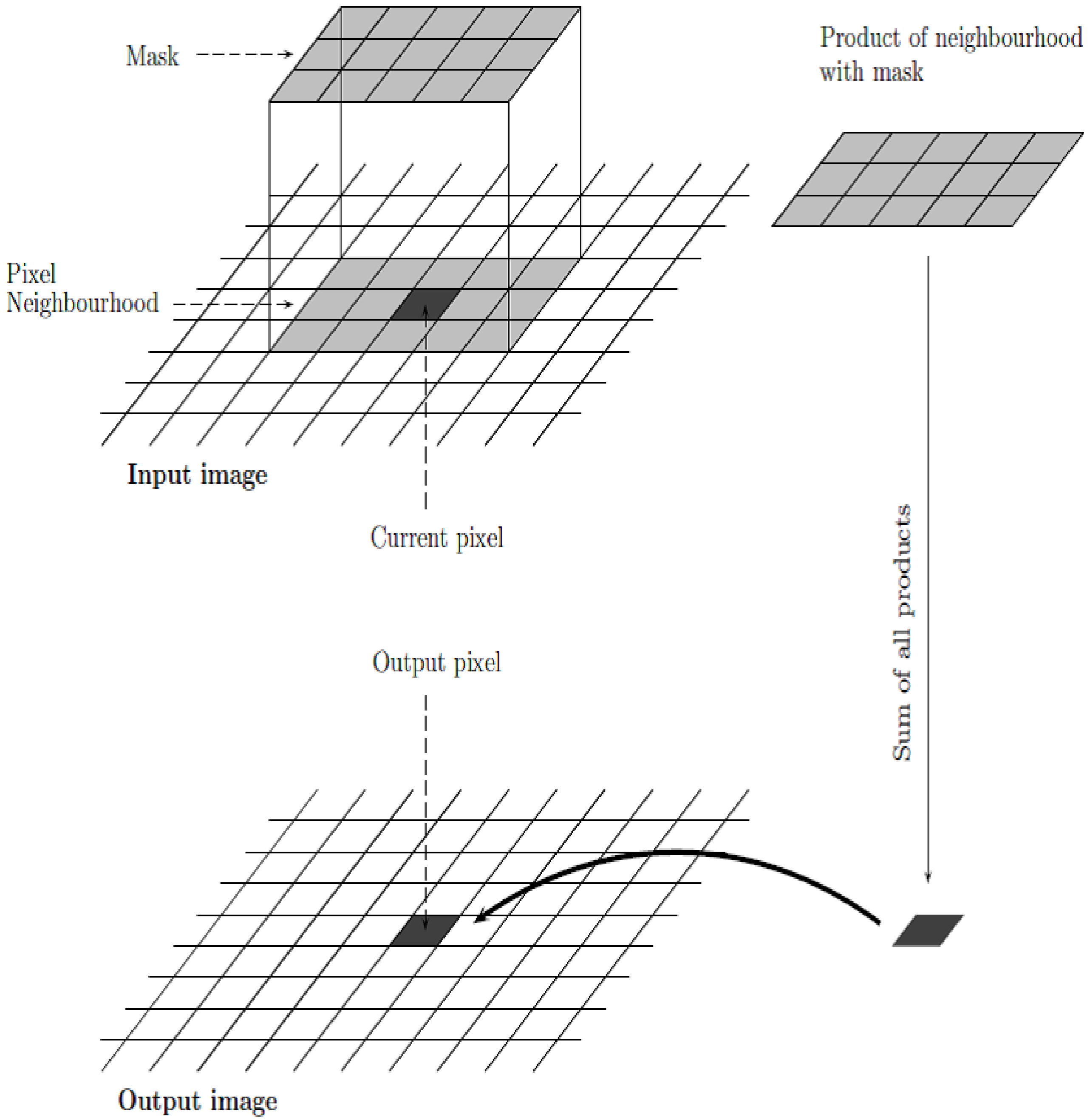
Lecture Title: [Spatial Filtering _Part2]

We see that spatial filtering requires three steps:

1. position the mask over the current pixel,
2. form all products of filter elements with the corresponding elements of the neighbourhood,
3. add up all the products.

This must be repeated for every pixel in the image.

An example: One important linear filter is to use a mask and take the average of all nine values within the mask. This value becomes the grey value of the corresponding pixel in the new



Median Filter The median filter is a non linear filter (order filter) used to remove noise from images.

The median filter is also used to preserve edge properties while reducing the noise

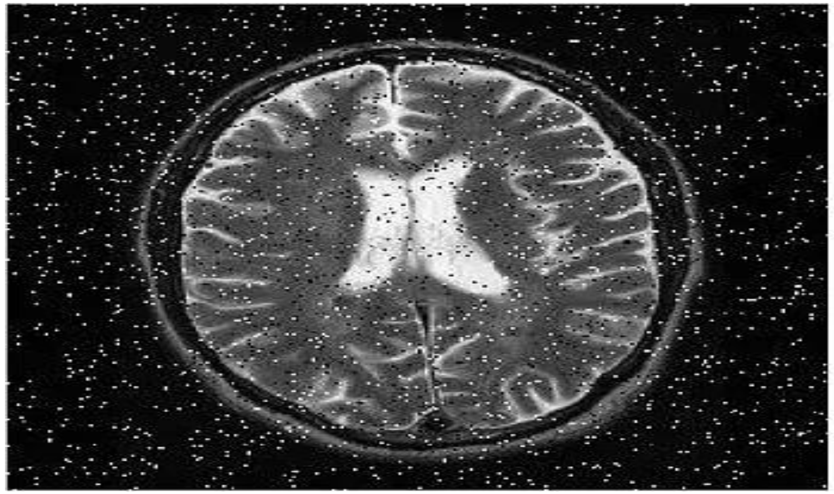
These filters are based on as specific type of image statistics called order statistics.

Typically, these filters operate on small sub image, “Window” , and replace the center pixel value (similar to the convolution process).

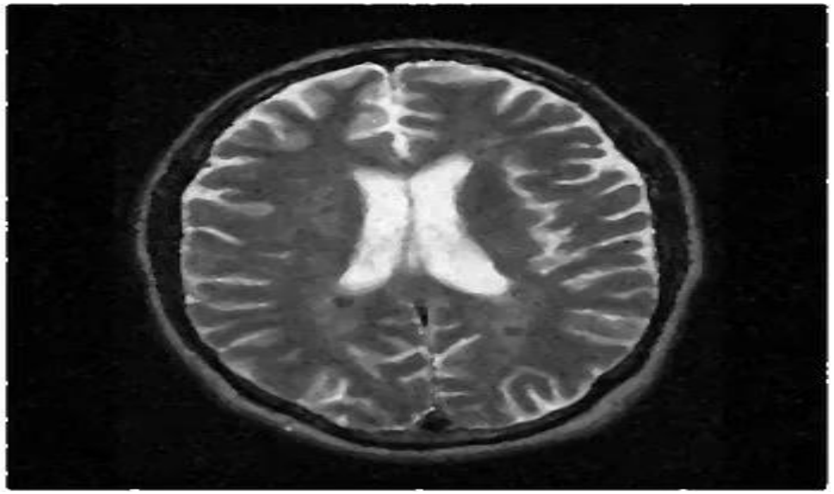
Order statistics is a technique that arranges the entire pixels in sequential order, given an N×N window (W), the pixel values can be ordered from smallest to the largest (ascending order).

$I1 < I2 < I3 < .... < IN\ 2$

Where I1 ,I2 , I3 ...., IN are the intensity values of pixels within (NxN) window



a. Salt and pepper noise



b. Median filtered image (3x3)

Example : Given the following 3×3 neighborhood (window) :

$$\begin{bmatrix} 5 & 5 & 6 \\ 3 & 4 & 5 \\ 3 & 4 & 7 \end{bmatrix}$$

- Sort the values in window (ascending order) (3,3,4,4,5,5,5,6,7)
- Then we select the **middle value** ,in this case it is 5.
- The middle value 5 is then placed in center location.

$$\begin{bmatrix} 5 & 5 & 6 \\ 3 & 5 & 5 \\ 3 & 4 & 7 \end{bmatrix}$$

The Procedure of median filter:

- ■ The window is overlaid on the upper left corner of the image, and the median value is determined.
- ■ This value (median) is put into the output image (buffer) corresponding to the center location of the window.
- ■ The window is then slide one pixel over, and the process is repeated
- ■ When the end of the row is reached, the window is slide back to the left side of the image and down one row, and the process is repeated.
- ■ This process continues until the entire image has been processed.

Not: the outer rows and columns are not replaced.. And these “wasted” rows and columns are often filled with zeros (or cropped off the image). For example, with 3X3 mask, we lose one outer row and column, a 5X5 mask we lose two rows and columns.

Example:

Consider the following 5×5 image:

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Apply a 3×3 median filter on the shaded pixels, and write the filtered image.

Solution

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Sort:
20, 25, 30, 30, **30**, 70, 80, 80, 255

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Sort
0, 20, 30, 70, **80**, 80, 100, 100, 255

20	30	50	80	100
30	20	80	100	110
25	255	70	0	120
30	30	80	100	130
40	50	90	125	140

Sort
0, 70, 80, 80, **100**, 100, 110, 120, 130

Filtered Image =

20	30	50	80	100
30	20	80	100	110
25	**30**	**80**	**100**	120
30	30	80	100	130
40	50	90	125	140

Figure below shows an example of applying the median filter on an image corrupted with salt-and-pepper noise.

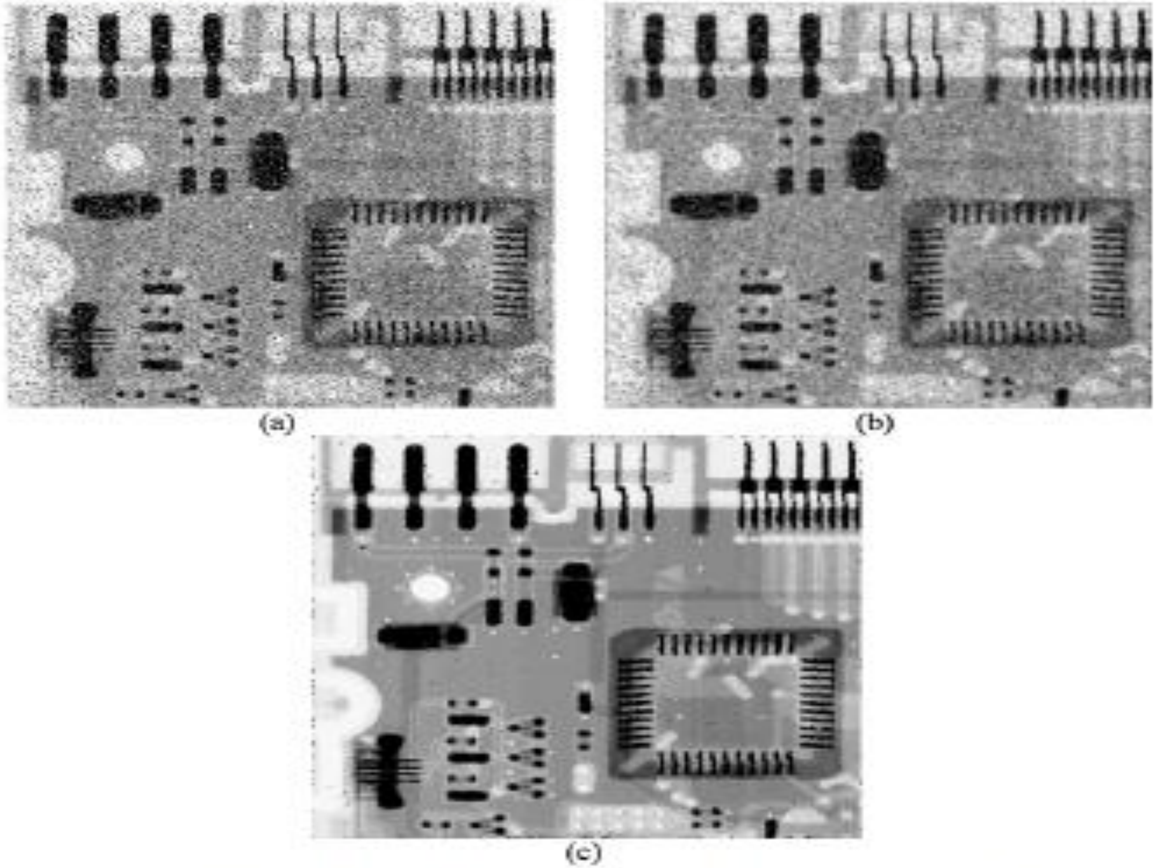


Figure 6.3 Effect of median filter. (a) Image corrupted by salt & pepper noise. (b) Result of applying 3×3 standard averaging filter on (a). (c) Result of applying 3×3 median filter on (a).

As shown in the figure, the effects of median filter are:

1. Noise reduction
2. Less blurring than averaging linear filter

Maximum and Minimum Filters

■ Maximum and Minimum Filters are two order filters that can be used for elimination of salt- and-pepper noise.

■ The maximum filter selects the largest value within an ordered window of pixels values and replaces the central pixel with the largest value (lightest one) .

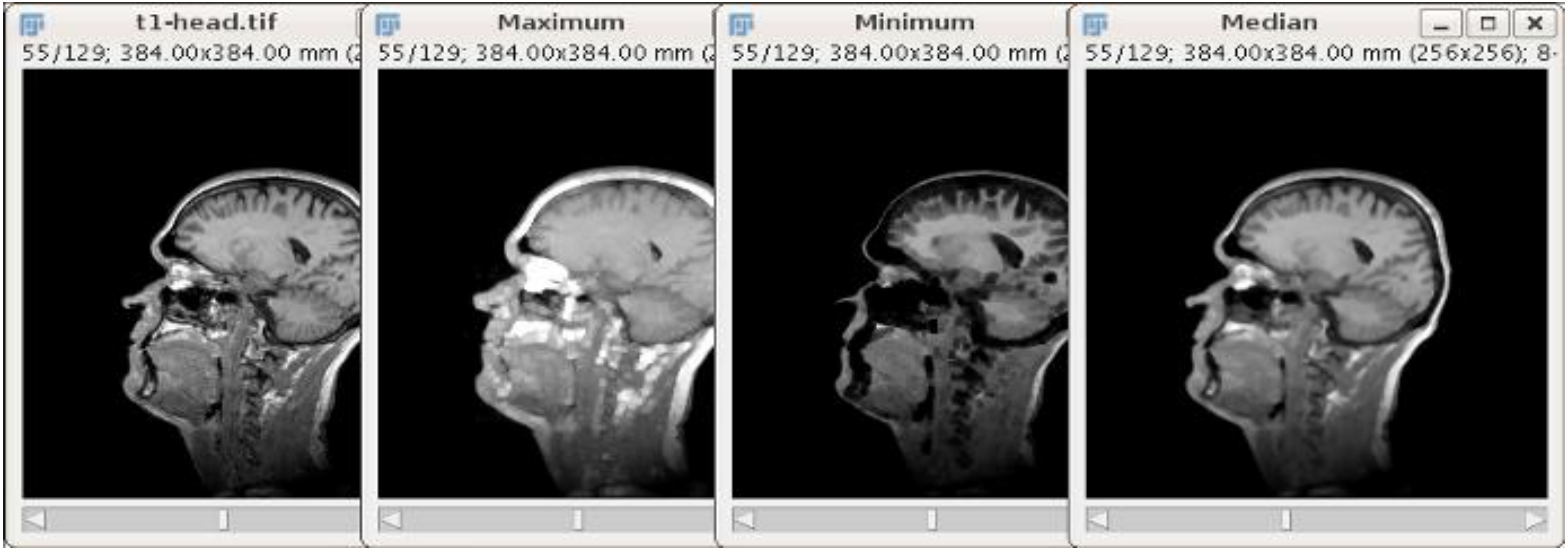
■ Used to find the brightest points in an image.

■ The maximum filters work best for removing pepper-type noise.

■ The minimum filter selects the smallest value within an ordered window of pixels values and replaces the central pixel with the smallest value (darkest one) in the ordered window .

■ The minimum filters work best for removing salt- type noise,

NOT: a minimum or low rank filter will tend to darken an image and a maximum or high rank filter will tend to brighten an image


t1-head.tif
55/129; 384.00x384.00 mm (2
Maximum
55/129; 384.00x384.00 mm (2
Minimum
55/129; 384.00x384.00 mm (2
Median
55/129; 384.00x384.00 mm (256x256); 8-

Example: Apply a maximum filter (Order Statistic) on the following assumed image, using (3×3) Window size

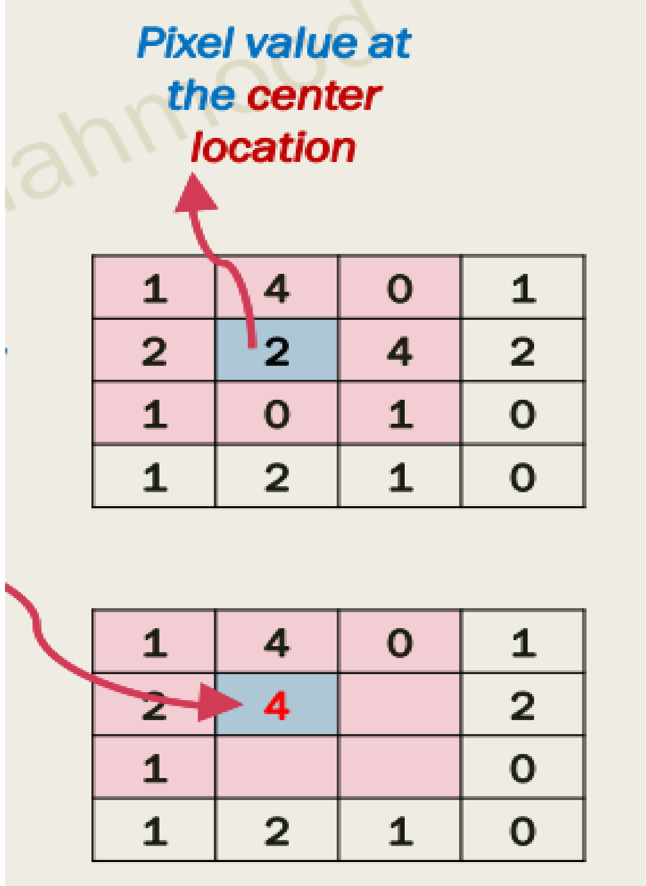
I =

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

Input Image

Solution:

1. Overlade (3x3) window at the upper left corner of the input image :
2. Sort the pixels values (ascending order) : 0, 0, 1, 1, 1, 2, 2, 4, 4
3. Determine the maximum value : 0, 0, 1, 1, 1, 2, 2, 4, 4
4. Replace the pixel at the center location by median value
5. The window is then slide one pixel over, and Repeat the steps (1- 4)



Sharpening Spatial Filters

Sharpening aims to highlight fine details (e.g. edges) in an image, or enhance detail that has been blurred through errors or imperfect capturing devices.

Image blurring can be achieved using averaging filters, and hence sharpening can be achieved by operators that invert averaging operators. In mathematics, averaging is equivalent to the concept of integration, and differentiation inverts integration. Thus, sharpening spatial filters can be represented by partial derivatives.

Partial derivatives of digital functions

The first order partial derivatives of the digital image *f(x,y)* are:

$$\frac{\partial f}{\partial x} = f(x+1, y) - f(x, y) \quad \text{and} \quad \frac{\partial f}{\partial y} = f(x, y+1) - f(x, y)$$

The first derivative must be:

1) zero along flat segments (i.e. constant gray values).
2) non-zero at the outset of gray level step or ramp (edges or noise)
3) non-zero along segments of continuing changes (i.e. ramps).

The second order partial derivatives of the digital image *f(x,y)* are:

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

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

The second derivative must be:

1) zero along flat segments.
2) nonzero at the outset and end of a gray-level step or ramp;

We conclude that:

- 1^{st} derivative detects thick edges while 2^{nd} derivative detects thin edges.
- 2^{nd} derivative has much stronger response at gray-level step than 1^{st} derivative.

Thus, we can expect a second-order derivative to enhance fine detail (thin lines, edges, including noise) much more than a first-order derivative.

High pass filter

if it "passes over" the high frequency components, and reduces or eliminates low frequency components,

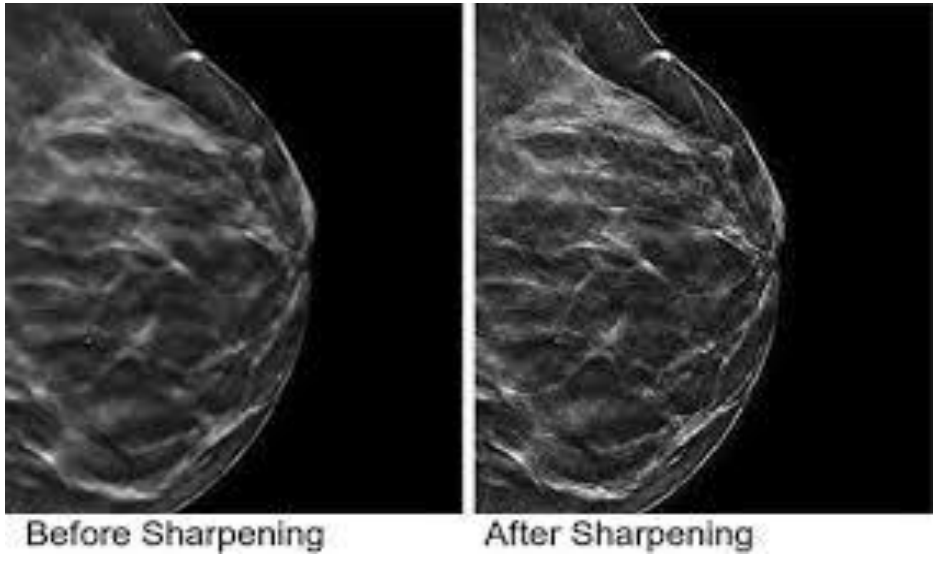
Low pass filter

if it "passes over" the low frequency components, and reduces or eliminates high frequency components,

For example, the averaging filter is low pass filter, as it tends to blur edges. The filter

is a high pass filter. We note that the sum of the coefficients (that is, the sum of all e elements in the matrix), in the

high pass filter is zero. This means that in a low frequency part of an image, where the grey values are similar, the result of using this filter is that the corresponding grey values in the new image will be close to zero. To see this, consider a block of similar values pixels, and apply the above high pass filter to the central four:



The Laplacian Filter

The Laplacian operator of an image *f(x,y)* is:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

This equation can be implemented using the 3×3 mask:

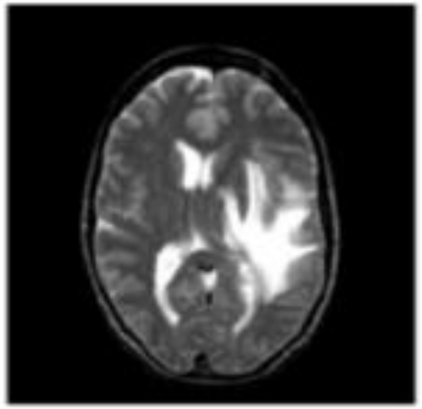
$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Since the Laplacian filter is a linear spatial filter, we can apply it using the same mechanism of the convolution process. This will produce a laplacian image that has grayish edge lines and other discontinuities, all superimposed on a dark, featureless background.

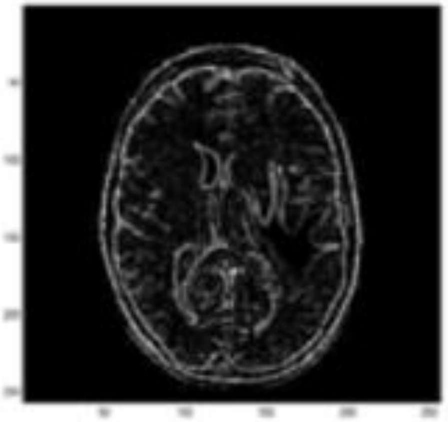
Background features can be "recovered" while still preserving the sharpening effect of the Laplacian operation simply by adding the original and Laplacian images.

$$g(x,y) = f(x,y) + \nabla^2 f(x,y)$$

The figure below shows an example of using Laplacian filter to sharpen an image.



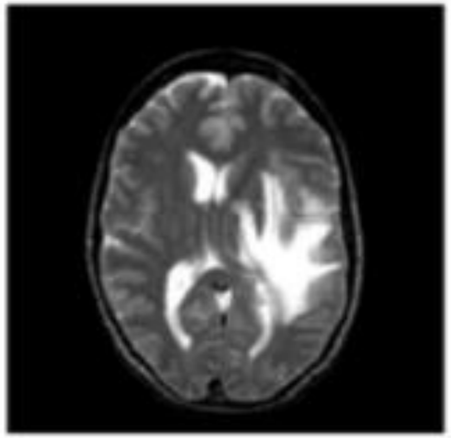
0	-1	0
-1	8	-1
0	-1	0



Before filtering

After filtering

Laplacian: Example 4



-1	-1	-1
-1	9	-1
-1	-1	-1

Before filtering

After filtering