NEW HORIZON COLLEGE OF ENGINEERING

Autonomous College Permanently Affiliated to VTU, Approved by AICTE & UGC
Accredited by NAAC with 'A' Grade.

"Digital Image Processing using Filters in MATLAB"

A MINIPROJECT REPORT

Submitted by

Bhanu Prakash Madala (1NH18EC064)

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

ELECTRONICS AND COMMUNICATION

NEW HORIZON COLLEGE OF ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

Certified that the mini project work entitled "**Digital Image Processing using Filters in MATLAB**" carried out by **Bhanu Prakash Madala (1NH18EC064)** bonafide student of Electronics and Communication Department, New Horizon College of Engineering, Bangalore.

The mini project report has been approved as it satisfies the academic requirements in respect of mini project work prescribed for the said degree.

Mini Project Guide:	HOD ECE
Ms. Monika Gupta	Dr. Sanjeev Sharma
Senior Assistant Professor	B.Tech, M.Tech, Ph.D
Department of Electronics and Communication	
-------------------------	-------------------------

External Viva

Name of Examiner	Signature with Date
1.	
2.	

ABSTRACT

Digital Image Processing using Filters in MATLAB

In today's Digital World, there are many digital applications that help us take pictures and videos of the beautiful nature, but these images are not always clear there is always certain noise in it, which is need to be treated so we can get the clear image. Noise is always present in digital images during acquisition of these image, coding or transmission, and processing steps.

When an image is taken from a camera or other imaging systems, often the vision system for which it is made is unable to use it directly. The image might be corrupted due to variation in intensity, illumination or poor contrast, these all are termed as noise.

It's very difficult to remove noise from the digital image without a proper knowledge of filtering technique. In this project report we are using certain types of Filtering techniques to remove the noise present in an image. Some of the Filtering technique used are Mean Filter, Median Filter and Adaptive Median Filter.

Further the result of these images is analyzed using certain parameters to understand which among the three Filtering techniques mentioned is best. The parameters used in the project are Peak Signal-to-Noise ratio (PSNR) and Mean Square Error (MSE).

ACKNOWLEDGEMENT

The satisfaction that accompany the successful completion of any task would be, but impossible without the mention of the people who made it possible, whose constant guidance and encouragement helped us succeed.

I thank **Dr. Mohan Manghnani**, Chairman of **New Horizon Educational Institution**, for providing necessary infrastructure and creating good environment.

I also record here the constant encouragement and facilities extended to us by **Dr.Manjunatha**, Principal, NHCE and **Dr. Sanjeev Sharma**, head of the department of Electronics and Communication Engineering. We extend sincere gratitude to them.

I sincerely acknowledge the encouragement, timely help and guidance to us by our beloved guide **Ms. Monika Gupta** to complete the project within stipulated time successfully.

Finally, a note of thanks to the teaching and non-teaching staff of electronics and communication department for their co-operation extended to us, who helped us directly or indirectly in this successful completion of mini project.

Bhanu Prakash Madala (1NH18EC064)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 01

INTRODUCTION

Digital image processing is used to create, communicate, process, and display digital images using certain computer algorithms. Digital image processing algorithms can be used to convert image sensor signals into digital images. Improve clarity, remove noise and other artifacts. Extract scale, size or the number of objects in a scene. It has many advantages over analog image processing and it allows a much wider range of algorithms to be applied to the data input and can avoid problems such as the build-up of noise and distortion during processing.

Image processing is often seen as arbitrarily manipulating an image to get an aesthetic standard or to support a reality. However it is more accurately defined as a means to translate between human visual system and digital imaging devices. The human eyes does not perceive the world in the same manner as digital detectors, with display devices imposing additional noise and bandwidth restrictions. An image is often filled with some sort of distortion and noise which is needed to be removed from the image to get a proper image it is achieved by using filtering techniques.

Filtering is a technique used for modification or enhancement of an image like highlight certain features or remove other features. Image filtering include sharpening of image, smoothing of the image and edge enhancement of the image.

In this project we are trying to implement certain filtering techniques for image processing to get a clear image without any kind of noise or distortion. Mainly 3 filters are used in this project namely Mean filter, Median filter, and Adaptive Median filter. The results of these filters is viewed and analyzed using two parameters namely Peak Signal-to-Noise Ratio(PSNR) and Mean Square Error(MSE) to find which among the 3 filters used is best for noise filtering.

CHAPTER 02

LITERATURE SURVEY

Title of the paper	Author & Year of Publication	Outcome	Limitation
Comparative Analysis of image filtering techniques	J Singh (2015)	various image processing techniques have been discussed n explained	Time consuming
Application of matlab image processing technology	H Cao (2009)	easy understanding of need and application of matlab features in image processing	Lack of qualified professional
Image enhancement using filtering techniques	Krishna Kant Albania (2012)	image enhancement techniques to improve the image quality has been understood	A bit complicated

Table 2.1 Literature Survey

CHAPTER 03

EXISTING SYSTEM AND PROBLEM STATEMENT

Existing Systems:

The project deals with the filtering of the image there already exists many applications and software in market which will perform filtering action to an image and give good results.

Some of the software are paid, and some are free, some example of these software and applications are Adobe Photoshop, Snapseed, PicsArt Photo Studio, etc.

Problem Statement:

To read an image and use certain types of filtering techniques to get a clear image and compare the results of filters using the PSNR and MSE as the parameter for image Quality check and find out which is the best filtering technique.

Objectives:

- To get the clear image without any noise.
- Compare the PSNR and MSE values for the different filters.
- Finding out which filtering technique gives much favorable results.

CHAPTER 04

PROJECT DESCRIPTION

Digital Image Processing using Filters in MATLAB

In this project we are using the concepts of Digital image processing to remove the noise and distortion from the image and get a proper, clear and enhanced image using certain filtering techniques. These filtering techniques play a vital role in image processing domain.

We have used MATLAB for implementing image filtration process as the main software tool.

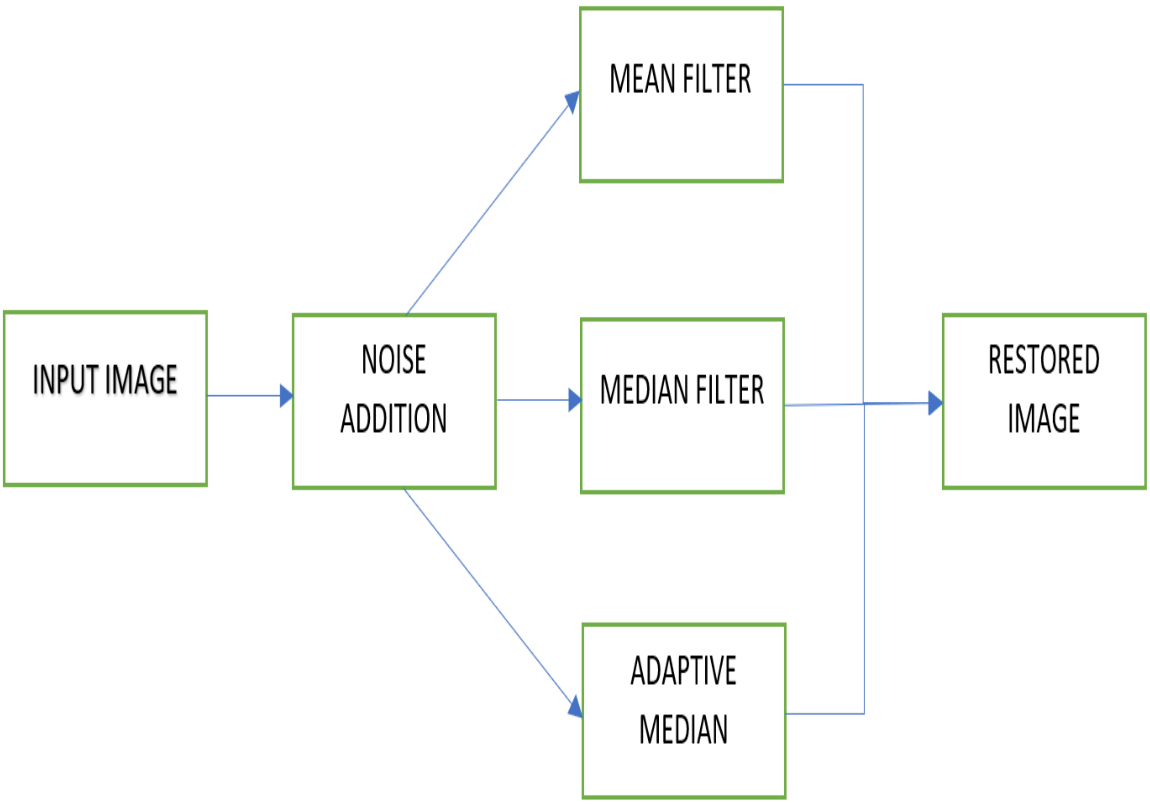

Fig 4.1: Basic Block Diagram of the project

The above given block diagram explains the working of the project in very brief manner. The project is built using the Software tool MATLAB. The block diagram contains few important details regarding the project.

The steps followed in the project are as follows:

- Input an Image
- Addition of Noise
- Applying the filtering process namely Mean, Median, Adaptive Median filters.
- Display of Restoring of the image.

4.1 Noise:

Digital images are always prone to various types of noises. Noises are the results of the errors in image acquisition process that results in the pixel values that do not reflect true intensities of the image. There are several types of noise and several ways that noise can be introduced into an image, depending on the way how image is created. For example:

- The image is scanned from a photograph made on film, the film grain is source of noise. Noise can also be the result of damage in the film or be introduced by the scanner itself.
- If the image is taken directly in digital format, the mechanism for gathering the data such as a CCD detector can introduce noise into the image.
- Electronic transmission of image can also introduce the noise in image.

Types of noise:

There are different types of noise in MATLAB that can be imposed into an image, they are:

- Gaussian noise
- Speckle noise
- Salt and pepper noise.

Gaussian noise:

Gaussian noise is named after Carl Friedrich Gauss, is a noise having a probability density function equal to that of normal distribution, which is known as gaussian distribution. The values that the noise can take will be the gaussian distributed values. Idealized for of white noise added to noise and normally distributed.

Fig 4.2 Gaussian noise

Speckle noise:

Pixel values multiplied with random noise.

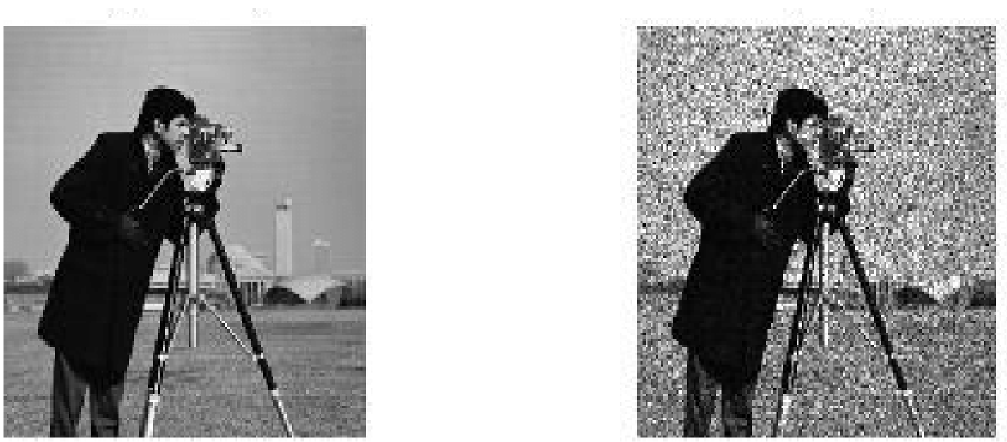

Fig 4.3 Speckle noise

Salt and pepper noise:

We use salt and pepper noise in the project, it is randomly scattered black and white pixels. Salt and pepper noise can be introduced into image using a MATLAB function

Imnoise(I,'salt & pepper',0.02);

Where 'imnoise' is the keyword, 'salt & pepper' is the type of noise that is supposed to be added. '0.02' is the value of noise that should be implemented.

Fig 4.4 Salt and Pepper noise

4.2 Filtering:

Digital images are prone to noises and these noises can be removed using certain techniques and certain algorithms, this technique of removal of noise from the image is known as Filtering. Filtering is a process for modifying or enhancing an image features, or remove other features. Image filtering include sharpening of image, smoothing of image, edge enhancement of the image.

There are different types of filtering techniques in Digital Image processing.

The types of Filters used in this project include:

- Mean Filter.
- Median Filter.
- Adaptive Median Filter.

Mean Filter:

Mean (or Average) filtering is one of the method of ‘smoothing’ images by reducing the amount of intensity variation between the neighbouring pixels. The average filter works by moving through the image pixel by pixel, replacing each pixel value with the average of the neighbouring pixels, including itself.

The major use of averaging filters is in the reduction of the irrelevant details in an image. A spatial averaing filter in which all the coefficients are equal is sometimes referred as Box filter. Also know as a Low Pass Filter. MxN mask will have a normalizing constant equal to 1/MN.

Mean filtering is simply replacing each pixel value in an image with the average value of its neighbors, including itself.

$$\frac{1}{9} \times \begin{array}{|c|c|c|} \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

$$R = \frac{1}{9} \times \sum_{i=1}^{9} z_1,$$

Fig 4.5 Mean Filter mathematic form

Fig 4.6 Mean filter Output of a Noised image

The image with the salt and pepper noise can be removed using a Mean Filtering technique which can be implemented in MATLAB with few lines of code,

h=fspecial('average',3);

filter_image_mean=imfilter(im,h);

here ‘fspecial’ is the function call to implement different filter, word ‘average’ is the argument given to this function call fspecial which will in turn implement ‘Average Filter or Mean Filter’.

Here the function ‘imfilter’ will add filter to the image and get the clear output to the variable.

Advatages of Mean Filter:

- Easy to Implement.
- Used to remove the impulse noise.

Disadvantages of Mean Filter:

- It does not preserve details of the image, some details are removed from the image when mean filter is used.
- A Single pixel value with a very unrepresentative can significantly affect the average value of all pixels in the neighborhood.

Median Filter:

The Median Filter is simple but powerful non-linear Digital filtering technique, often it is used to remove noise from an image or signal. Such noise reduction technique is a typical pre-processing step to improve the quality for later processing(edge detection or etc). This filter is widely used in digital image processing because, under certain constrains, it still preserves edges while removing noise, it also have many applocations in signal processing.

Median Filter is used for reducing the amount of intensity variations between one pixel to the other pixel. In this filter, we replace each pixel value with the median value of all the neighboring pixels. The median is calculated by reading all the pixel values and than sorting them in ascending order and then replace the pixel being calculated with the middle pixel value. It is most suitable to remove the 'Salt and Pepper noise'.

The pattern of the neighbour is called as 'window', which slide, pixel by pixel over entire image. When there is no entry preceding first value, the first value is repeated to obtain enough entires to fill the window.

There are other approaches that will have different properties that might be prefered in particular circumtances:

- Avoide the processing of the boundaries, with or without cropping of the signals or the image boundary afterwards.
- Fetching the entires from other places in the image signal, for example, entries from vertical or horizontal boundary might be selected.

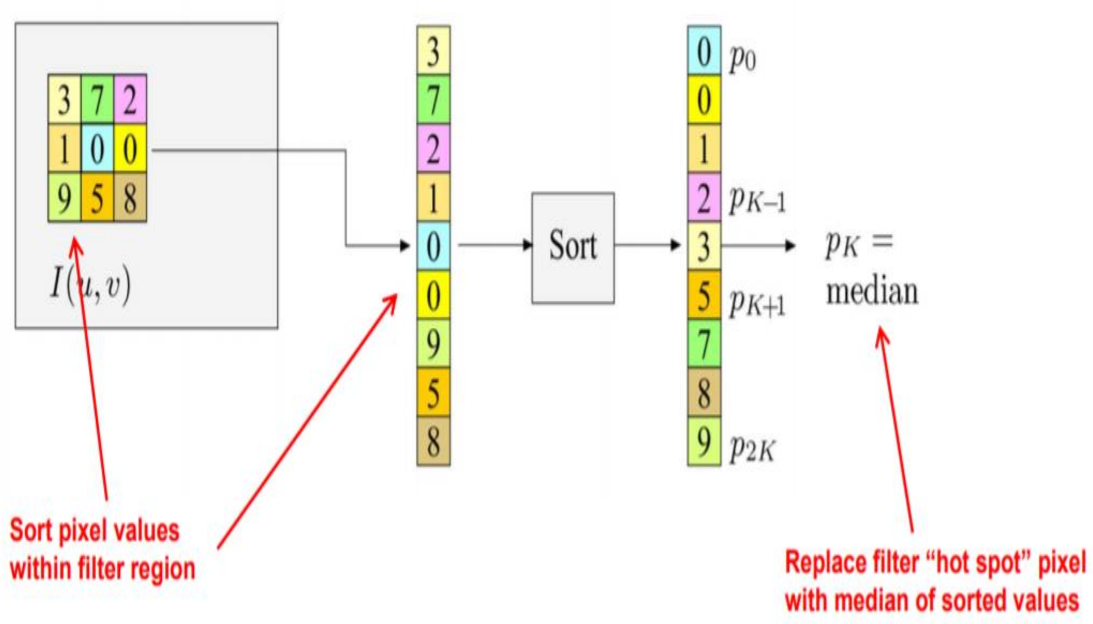

Fig 4.7 Structural representaion of Median Filter

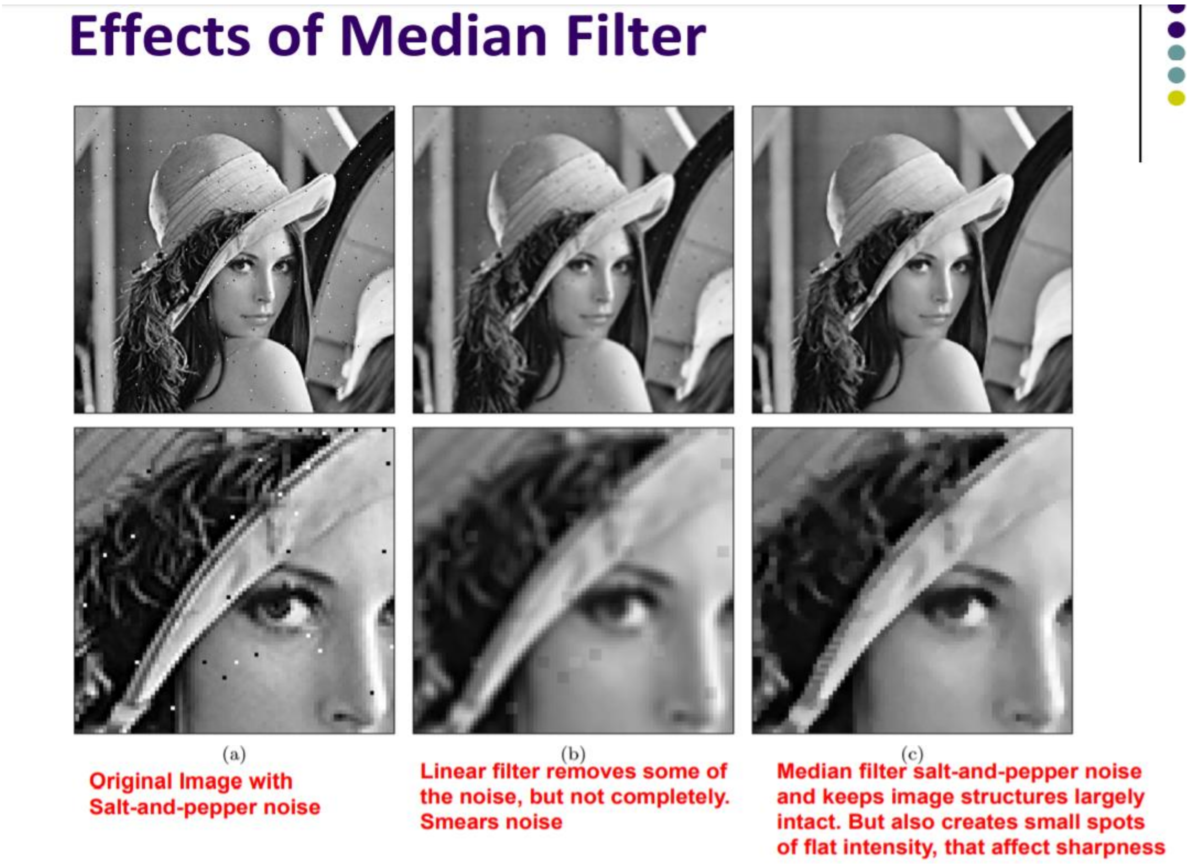

Fig 4.8 Effect of Median Filter

The image with the salt and pepper noise can be removed using a Median Filtering technique which can be implemented in MATLAB with a line of code,

filter_image_median=medfilt2(nim);

Here ‘medfilt2’ is the in-built function in the MATLAB that is used to implement Median Filter to the noised image ‘nim’. filter_image_median is the variable used to store the filtered image after implementing median filter.

Advantages of Median Filter:

- It is easy to implement.
- Used to de-noise different types of noises.
- The median values will be more robust than the mean value, the median value will not be affected by any one very unrepresentative pixel in neighborhood.

Disadvantages of Median Filter:

- Median filter will tend to remove image details when the impulse noise is more than 0.4%.
- Median filter will replace the pixel value of each pixels in the image regardless whether there is noise content or not.

Adaptive Median Filter:

Median Filtering is known that it is based on statistical operation of finding the median. It is clever, simple and very intuitive noise reduction method that reduces the noise by computing local stats of data and rejecting noise. For each pixel in the image, calculate the median value from the surrounding neighborhood of pixels. Replace the pixel with the median value.

This algorithm is simple and easy to implement. However, it affects all the image pixels regardless of noise content or not and that's why in image and videos with high noise content (low SNR), it blurs the image substantially. It only performs well on low noise densities.

So why not a **Adaptive median filter?** This question flashed and was brought to existence, it still computes the median over the surrounding neighborhood, however, it is smart enough to check is it necessary to replace the pixel value with the median or not.

Steps involved in Adaptive Median Filter:

- First for each pixel in the image, calculate the median value from the neighbourhood of the pixels.
- It will than compare the median value with the threshold and decides to either replace the pixel value, or keep the pixel, or increase the neighbourhood size and recalculate.
- Only affects image pixel that has noise content.
- Performs well with low and high noise densities.
- It is more computationally intensive.

Flowchart for Adaptive Median Filter:

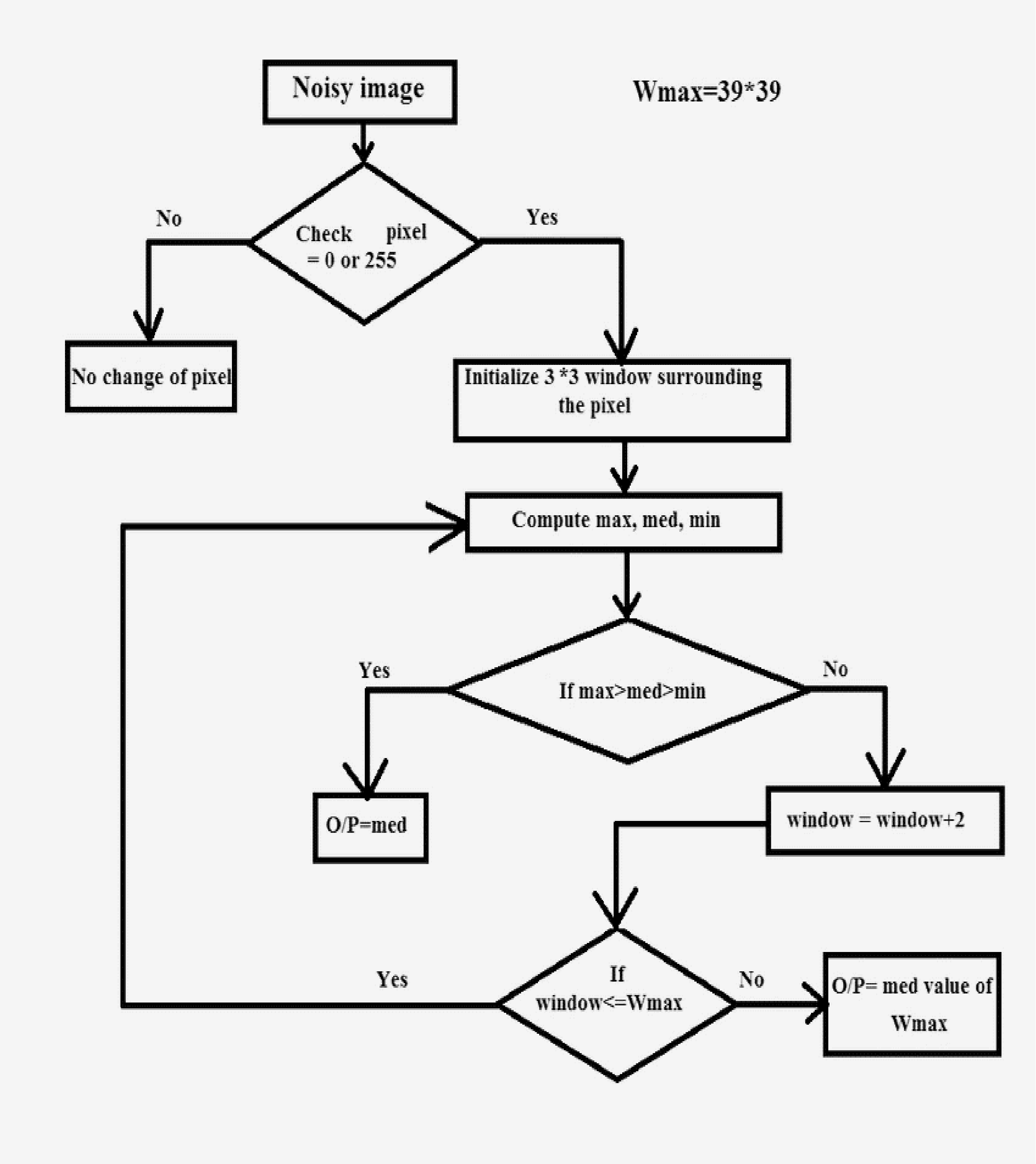

Fig 4.9 Flowchart of Adaptive Median Filter

Algorithm Used in Adaptive Median filter:

Step 1: Set window size, ws=3,   wsmax=21

Step 2: m,n- row and column index of the noised image.

Step 3: Read neighboring pixel values w.r.t m,n and ws.

Step 4: Find S1, S2, and S3.

Where S1=min(neighboring pixel)

S2=median(neighboring pixel)

S3=max(neighboring pixel)

Step 5: A1 = S2 - S1

A2 = S2 - S3

If A1 > 0 and A2 < 0, Go to Step 6

Else increase the window size by 2

If window size <= wsmax repeat Step 5

Else output S2

Step 6: B1 = pixel(m,n) - S1

B2 = pixel(m,n) - S3

If B1 > 0 and B2 < 0, output pixel(m,n)

Else output S2

Explanation of the Adaptive Median Filter Algorithm:

Input a noised image, check for the pixel values ranging from 0 to 255, now set the window size to be 3 at the beginning and window size max be 21. Let m,n be the index values of row and column of the image. Now taking window size 3 read the value of the neighboring pixels w.r.t to m,n and ws. Find the values of S1(min), S2(median), S3(max) values from the pixels.

Now calculate the value of A1 and A2 from the formula and if A1>0 and A2<0 than we can go to further step of finding B1 and B2 or else increase the window size by 2 every time. Continue till the window size becomes less than or equal to window size max.

After finding the value of B1 and B2, check whether the value of pixel(m,n) lies between B1 and B2, if yes the output pixel value is same as the earlier or else it is replace by the median value i.e, S2.

Fig 4.10 Output of Adaptive Median Filter

4.3 Image Quality Metric:

Image improving or enhancement of the visual quality of a digital image can be very subjective. Like there are many methods to improve an image so quality of image vary from person to person. For this reason; it is nessesary to make measures to compare the effects of image enhancement algorithms on image quality.

We in this project are using two main parameters to test the quality of an image, they are:

- Peak Signal-to-Noise ratio (PSNR)
- Mean Square error (MSE).

Mathematics:

For the implementation we take a standard 2D array of data or matrix. The dimentions of the filtered image and noised image must be same. The mathematical representation of PSNR is as follows,

$$PSNR = 20\log_{10}\left(\frac{MAX_f}{\sqrt{MSE}}\right)$$

Where MSE is,

$$MSE = \frac{1}{mn}\sum_{0}^{m-1}\sum_{0}^{n-1}\|f(i,j) - g(i,j)\|^2$$

The PSNR and MSE can be implemented in MATLAB using the formula as stated below,

*MSE = (1/(m*n))*sum(sum((f-g).^2))*

*PSNR = 20*log(max(max(f)))/((MSE)^0.5)*

Legend:

f represents the data array of the filtered image.

g represents the data array of the noised image.

m represents the number of rows of the pixels in an imageand **i** represents the index of the row.

n represents the number of columns of the pixels in an image and **j** represents the index of the column.

MAX_f is the maximum signal value that is present in the filtered image.

The PSNR and MSE are inverse in relation for an image, if the PSNR value of the images HIGH for a particular filter among other filters than that filter has the best filtering action on an image where as the image with low MSE value, is the best.

The values of PSNR and MSE will decide the Quality of the filters.

4.4 Working:

Initially the original image is read into the MATLAB workspace using inbuilt library function (imread). And displayed on the figure window using the command (imshow).

Fig 4.11 Original image

Then the percentage of noise, here we use salt and pepper noise, it is added to the original image using an inbuilt library function(imnoise). And the result is shown in figure window.

Fig 4.12 Noised image with 5% noise percentage

After the addition of noise the image need to be filtered, at the beginning we use Mean Filter to filtered the noised image. It is done using MATLAB inbuilt function (imfilter) and (fspecial).

h=fspecial('average',3);

filter_image_mean=imfilter(im,h);

Fig 4.13 Mean Filtered Image

After filtering the image with Mean Filter, we calculate the value of PSNR and MSE of the filtered image w.r.t the noised image. And these values are displayed on command window.

Now the Median Filter is implemented on the noised image using the MATLAB inbuilt function (medfilt2) and the filtered image is shown in the figure window.

filter_image_median=medfilt2(nim);

Fig 4.14 Median Filtered Image

Again the PSNR and MSE value is calculated for the Median Filtered image w.r.t noised image and these values are displayed on the command window.

Now the Adaptive Median Filter is implemented on the noised image using the algorithm mentioned above, the filtered image now is displayed on the figure window.

Fig 4.15 Adaptive Median Filtered Image

The PSNR and MSE value of the Adaptive Median Filtered image is calculated and displayed in the command window.

All the images, that is Original Image, Noised Image, Mean Filtered Image, Median Filtered Image, and Adaptive Median Filtered Image are shown together in the figure window with the help of the MATLAB inbuilt function (subplot) and (title)

CHAPTER 05

SOFTWARE SPECIFICATION

Software Tool used: MATLAB 16.0

The Software tool used in this project is MATLAB 16.0, MATLAB (matrix laboratory) is a multi-paradigm numerical computing environment and a programing language developed by MathWorks. MATLAB stands for Matrix Laboratory, it allows matrix manipulations, implementing algorithms, plotting of functions and data, and dealing with the programs written in other languages.

Designed By: Cleve Barry Moler.

Developer: MathWorks

Some of the functions used in this project are:

- **Im=imread(filename.fmt);**

  This function is used to read an image from the data into the workspace, here the 'filename' is the name of the file saved in the data and 'fmt' is the format in which the file is saved. Im is the variable used to store the value of the image.

- **Im=imresize(im,[512 512]);**

  This function is used to resize the image size to a certain value like here in example the image 'im' is resized to [512 512].

- **imshow(im);**

  This function is used to display the image 'im' in the figure window.

- **subplot(, ,);**

  This function is used to create a division in the figure window to display different images in the same figure window.

- **title(‘ ‘);**

This function is used to give a title to the image under display.

- **disp();**

This function is used to display some arguments in command window.

Advantages of MATLAB:

- Test and implement algorithms easily.
- Development of computational codes is easy.
- Debugging is easy.
- Use of large database with built in algorithms.
- Processing of image and simulating of videos is easy.
- We can call external libraries.

Disadvantages of MATLAB:

- MATLAB Compiler SDK, Matlab Compiler and packaging tools for add-on are not supported.
- xlsread and xlswrite will work only in basic mode.
- The cost of original licence for MATLAB software is very high and unaffordable for many.

CHAPTER 06

RESULT AND DISCUSSION

The Software Project, Digital Image Processing using Filters is working perfectly as per our objectives. It takes image input and filtering action is done on the noised image and Filtered output is show on the figure window.

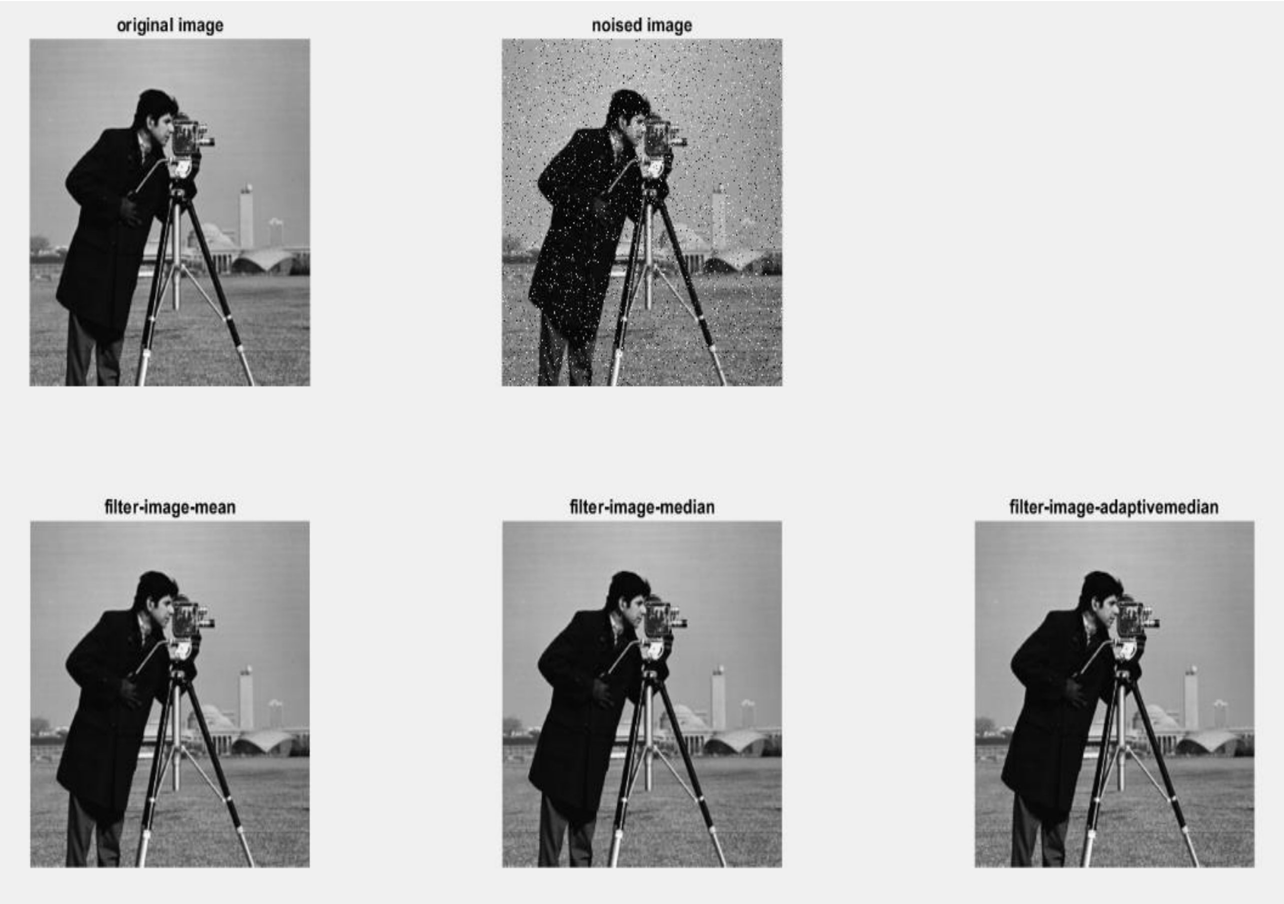

Fig 6.1 Output of the Final Project

This is the figure output when 5% “salt and Pepper” noise is applied to the original image

For 5% Salt and Pepper noise

SL.NO	Name of the Filter	PSNR	MSE
1	Mean Filter	45.2712	1.9318
2	Median Filter	48.1856	0.9875
3	Adaptive Median Filter	54.3200	0.2405

Table 6.1 Results for 5% noise

For 10% Salt and Pepper noise

SL.NO	Name of the Filter	PSNR	MSE
1	Mean Filter	45.2712	1.9318
2	Median Filter	46.3185	1.5179
3	Adaptive Median Filter	51.7501	0.4346

Table 6.2 Results for 10% noise

For 20% Salt and Pepper noise

SL.NO	Name of the Filter	PSNR	MSE
1	Mean Filter	45.2712	1.9318
2	Median Filter	43.2428	3.0818
3	Adaptive Median Filter	47.6034	1.1291

Table 6.3 Results for 20% noise

For 30% Salt and Pepper noise

SL.NO	Name of the Filter	PSNR	MSE
1	Mean Filter	45.2712	1.9318
2	Median Filter	40.6206	5.6367
3	Adaptive Median Filter	44.9995	2.0565

Table 6.4 Results for 30% noise

CHAPTER 07

ADVANTAGES AND APPLICATIONS

Advantages of the project :

- Remove impulse Noise.
- Smoothing other noise.
- Reduce distortion, like excessive thickening or thinning of object boundaries.
- The standard median filter do not perform well enough when the impluse noise is
  - **a)** Greater than 0.2, but the adaptive median filter performs better with these noises.
  - **b)** Adaptive Median filter will preserve the details and smooth non-impulsive noise, while the standard median filter does not.

Applications of the Project :

- Image sharpening and restoration.
- Image restoration in Medical fields.
- Remote sensing satellites images need to be filtered.
- Digital cameras.
- Mobile applications.

CHAPTER 08

CONCLUSION AND FUTURE SCOPE

Conclusion:

When the Filtered Image has high PSNR value and low MSE value w.r.t a Noised image, than that image is better in details, least noise in it. Seeing the results of our project the PSNR and MSE values for Mean Filter do not vary with increase or decrease in noise content of the image.

The Median Filter performs well only till certain noise content, once the noise content is increased it will have poor filtering effect on the image, where as the Adaptive Median Filter will work very well in maintaining the details of the image and removing the noise, even when the noise content is high.

So we can conclude that the Adaptive Median Filter among the other two filters works very well, it is one of the best Filtering techniques that can be implemented in Image restoration.

Future Scope:

- To overcome the disadvantages and looking for implementing it into a proper software Application.
- To implement this software into the hardware components like digital image sensors and make the image quality much better.
- To implement this project on colored images as well.
- To implement this and use in some mobile application to add some filtered effect to the image.

REFERENCES

[1] R.E. Twogood and F. Graham Sommer ,” **Digital Image Processing**”, IEEE, IEEE Transactions on Nuclear Science, Volume: 29, 1982.

[2] Jasvinder Singh and Navin Rajpal ,” **Comparative analysis of image filtering techniques**”, IEEE, 2015.

[3] G. Devarajan and C.S. Sridhar ,” **Analysis of Median Filter**”, IEEE, 1991

[4] H. Hwang and R.A. Haddad,”**Adaptive Median Filters: new algorithms and results**”, IEEE Transactions on Image processing, Volume: 4, 1995.

[5] R. Gonzalez and R.E. Woods,**”Digital Image Processing”**, Prentice Hall, 2002.

[6] W. K. Pratt, " **Digital Image Processing**", 2nd edition, CRC press, Boca Raton, 1995.

[7] S. Oudaya Coumar and P. Rajesh, **“Image restoration using filters and image quality assessment using reduced reference metrics”**, IEEE, 2013.

[8] Suman Shrestha, **“Image Denoising using New Adaptive Based Median Filters”**, Cornell University, 2014.

APPENDIX

Project Code: Digital Image Processing using Filters in MATLAB

```
clc;

clear all;

close all;

%Reading the image and displaying it

fname= 'cameraman.tif';

[im,map] = imread(fname);

im=imresize(im,[512 512]);

% im=rgb2gray(im);

subplot(2,3,1)

imshow(im);

title('original image');

[N,M]=size(im);


% percentage of noise (salt & pepper)  to add into the input image

noisep =20/100;

nim=imnoise(im,'salt & pepper',noisep);

subplot(2,3,2);

imshow(nim);

title('noised image');
```

```
%mean filter

h=fspecial('average',3);

filter_image_mean=imfilter(im,h);

subplot(2,3,4);

imshow(filter_image_mean);

title('filter-image-mean');


L=M*N;

%  mean square error between original and denoised  image

mse1=(1/L)* sum(sum((im-filter_image_mean).^2));

%  peak to signal ratio between original and denoised  image

psnr1=20.0* (log10(255/sqrt(mse1)));

disp('MEAN FILTER');

disp('PSNR='); disp(psnr1);

disp('MSE='); disp(mse1);


%median filter

filter_image_median=medfilt2(nim);

subplot(2,3,5);

imshow(filter_image_median);

title('filter-image-median');
```

```
% mean square error between original and denoised image

mse2=(1/L)* sum(sum((im-filter_image_median).^2));

% peak to signal ratio between original and denoised image

psnr2=20.0* (log10(255/sqrt(mse2)));

disp('MEDIAN FILTER');

disp('PSNR='); disp(psnr2);

disp('MSE='); disp(mse2);


%adaptive median filter


% initialize denoised image matrix with zeros elements

dim=zeros(M,N);


% set maximum median filter window size

wsmax=21;


for m=1 : M

  for n=1 : N


    % set minimum median filter window size

    ws =3;

    while ( ws < wsmax )
```

```
 %  neighbouring subscripts of centre pixel

hws = floor(ws/2);

[J,I]=meshgrid(n-hws:n+hws ,m-hws:m+hws);

I=I(:); J=J(:);


 % exclude subscripts out of border of the image

 if ( m<=hws || n<=hws || m>=M-hws || n>=N-hws)

       s =(I>=1 & I<=M);

       I=I(s); J=J(s);

       s =(J>=1 & J<=M);

       I=I(s); J=J(s);

 end


%  converts neighbouring subscripts index

 ind = sub2ind([M,N],I,J);


%  minimum of neighbouring pixels

 S1 = min ( nim(ind)  ) ;

%  median of neighbouring pixels

 S2 =  median ( nim(ind)  );

%  maximum of neighbouring pixels

 S3 =  max( nim(ind) );
```

```
    if ( S1<S2 && S2<S3 )


      if ( nim(m,n)>S1 && nim(m,n)<S3 )

          dim(m,n) = nim(m,n);

      else

         dim(m,n) = S2;

      end

      break;


    else

      % increment  median filter window size by 2

      ws = ws+2;


       %  if max window size reached, set S2 as denoised pixel and

        % break the while-loop

      if (  ws > wsmax )

          dim(m,n) = S2;

          break;

      end

    end


  end

 end
```

```
end

dim=uint8(dim);

% mean square error between original and denoised image

mse3=(1/L)* sum(sum((im-dim).^2));

% peak to signal ratio between original and denoised image

psnr3=20.0* (log10(255/sqrt(mse3)));

subplot(2,3,6);

imshow(dim);

title('filter-image-adaptivemedian');

disp('ADAPTIVE MEDIAN FILTER');

disp('PSNR='); disp(psnr3);

disp('MSE='); disp(mse3);
```