Sudan University of Science and Technology
College of Graduate Studies

A thesis submitted in partial fulfillmentof the requirements for the Degree of Master of Computer Engineering

Design of Digital Image Enhancement System Using Noise Filtering Techniques

تصميم نظام لتحسين الصور الرقمية باستخدام تقنيات ترشيح الضوضاء

By :
Hiba Ahmed Awad

Supervised By:
D. Rania A. Mokhtar

Aug.2015

الآية

قال تعالى:

بسم الله الرحمن الرحيم

(اللَّهُ نُورُ السَّمَاوَاتِ وَالْأَرْضِ مَثَلُ نُورِهِ كَمِشْكَاةٍ فِيهَا مِصْبَاحٌ الْمِصْبَاحُ فِي زُجَاجَةٍ
الزُّجَاجَةُ كَأَنَّهَا كَوْكَبٌ دُرِّيٌّ يُوقَدُ مِنْ شَجَرَةٍ مُبَارَكَةٍ زَيْتُونَةٍ لَا شَرْقِيَّةٍ وَلَا غَرْبِيَّةٍ
يَكَادُ زَيْتُهَا يُضِيءُ وَلَوْ لَمْ تَمْسَسْهُ نَارٌ نُورٌ عَلَى نُورٍ يَهْدِي اللَّهُ لِنُورِهِ مَنْ يَشَاءُ
وَيَضْرِبُ اللَّهُ الْأَمْثَالَ لِلنَّاسِ وَاللَّهُ بِكُلِّ شَيْءٍ عَلِيمٌ)

[سورة النور:35]

Dedication

To all I love

Hiba

Acknowledgement

I Would like to acknowledge D.Rania who kindly supervised this thesis and provided detailed guidance.

Grateful to Eng. Babekir with his assistancethis research became true.

Thanks to the family of Electronics Department.

My colleges who made the journey possible.

Table of Contents

الآية i
Dedication ii
Acknowledgement iii
ABSTRACT ix
خلاصة x
1.Introduction 2
1.1. Overview..2
1.2. Digital Images 2
1.3. Digital Image Quality 3
1.4. Sources of Noise..3
1.5. Importance of Noise Reduction..4
1.6. Problem definition: 4
1.7. Aim and Objectives: 5
1.8. Methodologies and tools: 5
1.9. The Thesis Outlines: 5
1.1 Previous work : 6
2Literature Review..12
2.1 Digital Image Processing Definition 12
2.2 Digital Image: 12
2.3 Digital Image Creation Methods 12
2.4 Fundamental Steps of Digital Image Processing 13
2.5 Digital image concept 16
2.6 Types of Digital Images 17
2.7 Digital Image Noise Concept 21
2.7.1 Types of noise 21
2.7.2Noise Modeling..23
2.7.3The concept of noise reduction: 24

2.7.4 Noise Removal....25
2.7.5 Image filtering....25
2.7.6 Basics steps of Spatial Domain Filtering 26
3 Methodology 32
3.1 Experimental setup....32
3.2 Performance Metrics 32
3.2.1Filtering in Matlab 36
3.3 Graphic User Interface Block Diagram: 38
4 Results 40
4.1 Qualitative Analysis:....40
4.2Impact (PSNR), (MAE) and (MSE)43
4.3 Histogram of noisy and filtered image45
5 Conclusion....50
5.1 Future Work 50
References 51
Matlab Code 57

LIST OF FIGURES

Figure (2-1) Fundamental Steps of Digital Image Processing 13
Figure (2-2) Binary Image..18
Figure (2-3) Grayscale image...18
Figure (2-4) Color image (RGB)...19
Figure (2-5) Indexed Image..20
Figure (2-6) De-noising Concept 24
Figure (2-7) Frequency Domain Filtering 26
Figure)2-8) Filtering process in spatial domain 27
Figure (2-9) Filtering approach of mean filter 28
Figure (2-10) Filtering approach of Median Filter...30
Figure (3-1) Noisy Image 37
Figure (3-2) Filtered Image 38
Figure (3-3) GUI Block Diagram 39
Figure (4-1) Gaussian Noise 41
Figure (4-2) Salt & Pepper Noise 42
Figure (4-3) Effect of Speckle Noise..43
Figure (4-4) Gaussian Noise 46
Figure (4-5) Salt & Pepper 47
Figure (4-6) Speckle Noise 48

List of Tables

Table (3-1) Experimental Setup 32
Table (3-2) Noise Types and Description..34
Table (5-1) 128*128 PSNR, MAE, and MSE...44
Table (5-2) 256*256 PSNR, MAE, and MSE...44
Table(5-3) 512*512 PSNR, MAE, and MSE..44

LIST OF ABBREVIATIONS

MATLAB: Matrix Laboratory

MSE : Mean Square Error

MAE : Mean Absolute Error

PSNR : Peak Signal to Noise Ratio

RGB : Red Green Blue

ABSTRACT

Noise Reduction in digital image is one of the most important and difficult techniques in image research. The aim of *Noise Reduction* in digital image is to improve the visual appearance of an image, or to provide a better transform representation for another automated image processing. Many images like medical images, satellite images, aerial images and even real life photographs suffer from viewing, removing blurring and noise, increasing contrast, etc.[34].

Reducing noise from the digital image is a challenge for the researchers. Several approaches are existed in the literature for noise reduction,generally there are common types of noises found in most digital image [35]. This project proposes filtering techniques for the removal of Gaussian,Salt-pepper and Speckle noise from the digital image. ***MATLAB*** software is used to simulate and implement the filtering techniques under study.

The efficiency of each filter is evaluated based on image metrics *PSNR* and *MSE*,*MAE* and *Histogram* analysis.

Results suggests that*Wiener* Filter performance is suitable for all Gaussian and Speckle noisy image and the *Median*shows better results with Salt & Pepper noisy images.

خلاصة

يعد الحد من الضوضاء في الصور الرقمية أحدالتقنيات الأكثر أهمية وصعوبة و يهدف الى تحسين المظهر المرئي للصورة، أو لتوفير تمثيل أفضل للمعالجة الآليةالأخرى للصور الرقمية، حيثتعاني العديد من الصور مثل الصور الطبية وصور الأقمار الصناعية والصور الجوية من قلة جودة العرض، الطمس والضوضاء وزيادة التباين، إلخ.

الحد من الضوضاء في الصورة الرقمية يمثل تحديا للباحثين، توجد عدة أساليب متبعة في هذا الفرع الهام لمعالجة الصور الرقمية،وقد تعرضنا في هذا المشروعلدراسة الأنواع الأكثر شيوعاً من تقنيات معالجة الضوضاء مثل التقنيات المتبعة لتقليل التشويش الجاوسي (Gaussian Noise)، الملح والفلفل(Salt& Pepper Noise) ،(Speckle Noise)من الصور الرقمية.

يتم استخدام برنامج *MATLAB* لمحاكاة وتنفيذ تقنيات التصفية قيد الدراسة، ويتم تقييم كفاءة كل مرشح استناداً إلى مقاييس الصورة PSNR and MSE, MAEوالرسم البياني.

أشارت النتائج إلى أن أداء "*Weiner Filter*" مناسبة لجميع الصور ذات التشويش (Gaussian & Speckle)، ويبين "*Median*" نتائج أفضل مع الصور ذات التشويش الملح و الفلفل.

CHAPTER 1

1. Introduction

1.1. Overview

Digital images often get corrupted with impulsive noise during their acquisition, transmission, or storage; hence there is a need to recover an original image from the degraded observations making the image restoration an important field of concern [6].

Getting an efficient method of removing noise from the images, before processing them for further analysis is a great challenge for the researchers. Noise can degrade the image at the time of capturing or transmission of the image.

Before applying image processing tools to an image, noise removal from the images is done at highest priority. The effectiveness of the noise removal algorithms to enhance an image depends on the type of noise present in it. Best results are obtained if testing image model follows the assumptions and fail otherwise [28].

Noise removal algorithm is the process of removing or reducing the noise from the image. It reduces or removes the visibility of noise by smoothing the entire image leaving areas near contrast boundaries. But these methods can obscure fine, low contrast details [23].

1.2. Digital Images

In the past decade, the quality of digital images has dramatically increased, while the cost of the hardware required to produce digital images has dramatically decreased. In its current state of development, digital imaging offers a flexibility and economy unmatched by film-based imaging. As a result, digital imaging has almost completely supplanted film-based imaging as the preferred method for capturing images. Consistent with this trend, camera manufacturers have all but abandoned the development of film cameras [38] .

1.3.Digital Image Quality

Key to the current dominance of digital imaging is the fact that digital image quality now rivals the quality of film-based images, and continues to increase. Like film-based images, the quality of digital images is affected by many factors, including the conditions under which the image was captured, the resolution of the optics used to capture the image and other physical characteristics of the image capture system. Unlike film-based images, digital image quality is also affected by the characteristics of the solid state device used to capture the image and the methods used to digitize the electrical signals generated by the solid state device. One of the more important factors affecting digital image quality is image noise, which, if present in any significant quantity, can noticeably degrade image quality [37].

1.4.Sources of Noise

The solid state device used to capture digital images is typically a charge coupled device, or *CCD*, consisting of an array of microscopic light sensing elements. *CCDs* and other similar image sensors are sensitive to more than just light from the image being captured, however. They are also sensitive to factors unrelated to the true image being captured, such as heat, and thus at the time of image capture generate spurious signals, or noise, that becomes part of the resulting image. Additionally, the amplification circuitry found in most digital imaging devices amplifies this noise and introduces additional noise of its own.

Finally, when amplified signals from the *CCD* undergo digitization, quantization error is introduced into the image, which can be considered yet another form of noise [37].

1.5.Importance of Noise Reduction

For the reasons above, noise is always present in digital images it is captured at the same time the image is captured and is part of the image data recorded by the imaging device. To the extent noise is present in an image, the true image is obscured and image detail is lost. In artistic, documentary and casual photography, noise has a negative impact on the aesthetics of digital images (they simply look worse), and in scientific and commercial applications of digital imaging (such as medical imaging, computer vision, and astronomy), noise can obscure important image details. In any case, it is worthwhile to attempt to reduce the noise inherent in digital images. Noise reduction can be accomplished at both the hardware level, by improving the design of image capturing hardware, and the software level, by attempting to detect and remove the noise from existing images. This research focuses on noise reduction in digital image use full **MATLAB**software and the interface design have been developed for this purpose [29].

1.6.Problem definition:

The major problem facing Digital Image is low quality caused by noise affection. Noise is an important factor that influences image quality which is mainly produced in the processes of image acquirement and transmission. Noise reduction is necessary for us to do image processing and image interpretation so as to acquire useful information that we want.

1.7.Aim andObjectives:

This research aims:To design a method to distinguish between certain noise types present in noisy grey- scale digital images (Gaussian – Salt & Pepper and Speckle). The objectives considered here are:

To present a comparative analysis of various noise suppression algorithms.

To demonstrate the results of applying different noise types to a grey- scale image model and investigates the results of applying various noise reduction techniques which produce satisfactory results.

To analyze noise reduction results in grey scale digital image.

To construct an easy interface to demonstrate the effect of the noise types under study, and their corresponding de-noising techniques.

Propose the suitable filter to improve image quality.

1.8.Methodologies and tools:

Three grey-scale original digital images of different sizes are prone to the three types of noise under study, getting nine samples.

Apply selected noise reduction tool to noisy images.

Design and implementation full MATLAB code package.

Results analysis.

1.9.The Thesis Outlines:

Chapter 1, discuss introduction of the project.

Chapter 2, illustrates literature review.

Chapter 3, discusses the methodology of the research, it introduces the design and implementation of the de-noising system.

Comment [U1]: This is outline of thesis contents e.g: chapter one discuss introduction of the project…
Chapter 2, illustrate literature review…
Chapter 3 discusses the methodology of the research, it introduces th design and implementation of ….
Chapter 4 illustrates the obtained results
Chapter 5draw a conclusion of the work and provide some future recommendations

Chapter 4, illustrates the obtained results.

Chapter 5, draw a conclusion of the work and provide some future recommendations.

1.1 Previous work :

Comment [U2]: You need to have more previous work here at least 10 papers. No need to put all of them in presentation.

The work done by various researchers for Image enhancement are discussed as follows:

- **Image De-noising by Various Filters for Different Noise:**Madhusuggested that the Adaptive histogram equalization produced a better result, but the image is still not free from washed out appearance. The sharpness is poor and the background information as well as the plane is still fogged and poor in contrast. Alpha rooting rendered the entire image in a dark tone. Even the outline of the clouds which was visible in case of histogram equalization is lost [17].
- **Image de-noising by sparse 3D transform-domain collaborative filtering:** Agaian suggested that the common no transform –based enhancement technique is global histogram equalization, which attempts to alter the spatial histogram of an image to closely match a uniform distribution. Histogram equalization suffers from the problem of being poorly suited for retaining local detail due to its global treatment of the image. It is also common that the equalization will over enhance the image, resulting in an undesired loss of visual data, of quality, and of intensity scale[13].
- **Noise removal in compound image using median filter:** Tang suggested global histogram equalization, which adjusts the intensity histogram to approximate uniform distribution. The global histogram equalization is that the global image properties may not be appropriately applied in a local context. In fact, global histogram modification treats all regions of the

image equally and, thus, often yields poor local performance in terms of detail preservation. Therefore, several local image enhancement algorithms have been introduced to improve enhancement[4].

- **Reduction of Speckle Noise Using Filtering Technique**:In this research, introduced a novel method which reduces speckle noise in ultrasound images and SAR images, retaining the original content of these images. This method enhancesthe Signal to Noise ratio and perceives the original features of the images. Introduced model automatically collect the information about the noise variance. Performance of the Speckle noise reduction model for Synthetic Aperture Radar (SAR) imagery is well as compared to other filters. Histogram results shows very closed equivalency in between SAR original images and SAR de-noised i.e. enhanced images [21].
- **Noise removal by using new spatial filters:** In this research, two novel image filters are presented. These filters, named as Far Distance Filter (FDF) and Near Distance Filter (NDF), are actually based on calculating the distance between image pixels and their neighbors in order to construct arbitrary values used to enhance abnormal pixels values [22].
- **Digital Image De-noising Using Histogram and Dynamic Filters:** In this research show implemented method for de-noising using two major steps as histogram and then filter it to the different levels [8].
- **Efficient Removal of Impulse Noise in Digital Images:** In this research such schemes of impulsive noise detection and filtering thereof are proposed. Here we presents a comparative study on six methods such as median filter, Progressive switching median filter, Fuzzy switching median filter, Adaptive median filter, Simple adaptive median filter and its modified version i.e. Modified Simple Adaptive median filter. Objective

evaluation parameters i.e. mean square error; peak signal- to noise ratio is calculated to quantify the performance of these filters [9].

- **Noise Reduction Using Multi-resolution Edge Analysis:** In this research find edges in the noisy image by multi-resolution analysis and use connectivity analysis to direct a MF to suppress the noise while preserving the edge information [19].
- **A Comparative Study of Image De-noising Techniques:** The comparative study of various de-noising techniques for digital images shows that wavelet filters outperforms the other standard spatial domain filters. Although all the spatial filters perform well on digital images but they have some constraints regarding resolution degradation. These filters operate by smoothing over a fixed window and it produces artifacts around the object and sometimes causes over smoothing thus causing blurring of image. Wavelet transform is best suited for performance because of its properties like sparsity, multi resolution and multi scale nature. Thresholding techniques used with discrete wavelet are simplest to implement [15].
- **Face Recognition Implementation for Client Server Mobile Application using PCA –[***The International Arab Conference on Information Technology (ACIT'2013)***]:** For the past few years in the face recognition research area are made very progressive improvements. This is because of the high level and versatile technologies in use nowadays, and high level of processors running on our machines and mobile phones. Available technologies provide mechanisms which use face recognition for security identification (user face) and authentication purposes. The aim of this paper is to present and propose client – server model and to compare it with the most recent client – server models for face recognition with a GPG infrastructure which uses security private key

(symmetric encryption) with main purpose to securely transmit image (user face) over the network. Moreover in the face recognition algorithm is implemented Principle Component Analysis (PCA) algorithm for face recognition. Proposed system has been tested on the mobile phone with Android OS platform, using previous research experiences where system was initially developed for DROID emulator. The implementation of the PCA is done on the **MATLAB** side.

- **The Implementation of Face Security for Authentication Implemented on Mobile Phone –***[The International Arab Conference on Information Technology (ACIT'2013]*: In this paper we are presenting the face recognition security for mobile phones. The model which has been applied for face recognition is Eigenface. The implementation consists from two parts: **MATLAB** and Droid Emulator for ANDROID mobile phones. The proposed implementation model has come as an idea, since today's mobile phones are computers in medium. We run our e-mails, agendas, storing data, using it for financial applications for viewing stock markets etc., and we would like to provide the approach of security model which will be based on face recognition as a biometric approach for authentication on mobile phones. Due the PIN vulnerability, as the most used mobile phone authentication mechanism we are presenting the approach which will enable a new level of mobile phone user's security.

  This has been tested with the database which consists from many images of facial expression. The algorithm which was implemented for mobile face recognition on **MATLAB** side is PCA. Limited with hardware capabilities we made of substitution between accuracy and computation complexity on the application. Proliferation of application and data has aim to increase the user need to protect the data which exist in mobile devices.

- **Biometry: Face Recognition Applying Logistic -** *The International Arab Conference on Information Technology (ACIT'2013)***:** Artificial methods as pattern recognition, machine learning, and artificial neural network are used for facial recognition. In this paper we present a LMT where we have used fallowing methods: extract skin colour, then converted into histogram values for performing LMT - logistic model tree. LMT is standard decision tree. The nodes of terminal are replaced with logic regression function. The accuracy of face recognition is 91%. We used histogram numerical values for performing supervised leering tasks for the prediction of nominal classes and numerical values. Here is presented LMT which adopts the idea of classification problems using logistic regression.

CHAPTER2

2 Literature Review

2.1 Digital Image Processing Definition

An image may be defined as a two-dimensional function, f(x, y), where (x) and (y) are *spatial* (plane) coordinates, and the amplitude of (f) at any pair of coordinates (x, y) is called the *intensity* or *gray level* of the image at that point. When (x, y, and the amplitude values of (f)) are all finite, discrete quantities, we call the image a *digital image* [1].

2.2 Digital Image:

Digital imaging or digital image acquisition is the creation of digital images, typically from a physical scene. The term is often assumed to imply or include the processing, compression, storage, printing, and display of such images. The most usual method is by digital photography with a digital camera but other methods are also employed.[1]

Everyday personal laptops, family desktops, and company computers are able to handle photographic software. Our computers are more powerful machines with increasing capacities for running programs of any kind—especially digital imaging software. And that software is quickly becoming both smarter and simpler. Although functions on today's programs reach the level of precise editing and even rendering 3-D images, user interfaces are designed to be friendly to advanced users as well as first-time fans.[1]

2.3 Digital Image Creation Methods

A digital photograph may be created directly from a physical scene by a *camera* or similar device. Alternatively, a digital image may be obtained from another image in an *analog* medium, such as photographs, photographic film, or printed paper, by an image scanner or similar device. Many technical images—such as those acquired with tomographic equipment, side-scan sonar, or radio telescopes—are actually obtained by complex processing of non-image data. Weather radar maps as seen on television news are a commonplace example. The digitalization of analog real-world data is known as digitizing, and involves sampling (discretization) and quantization[38].

Finally, a digital image can also be computed from a geometric model or mathematical formula. In this case the name image synthesis is more appropriate, and it is more often known as *rendering*.

Digital image authentication is an issue for the providers and producers of digital images such as health care organizations, law enforcement agencies and insurance companies. There are methods emerging in forensic photography to analyze a digital image and determine if it has been altered [38].

2.4 Fundamental Steps of Digital Image Processing [38]:

There are some fundamental steps but as they are fundamental, all these steps may have sub-steps. The fundamental steps are described in Figure (2-1) .

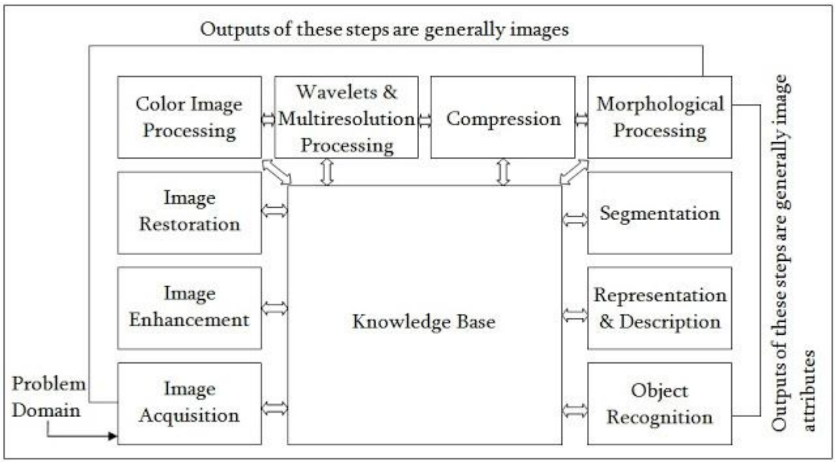


Figure (2-1) Fundamental Steps of Digital Image Processing

- **Image Acquisition:** This is the first step or process of the fundamental steps of digital image processing. Image acquisition could be as simple as being given an image that is already in digital form. Generally, the image acquisition stage involves preprocessing, such as scaling etc.
- **Image Enhancement:**Image enhancement is among the simplest and most appealing areas of digital image processing. Basically, the idea behind enhancement techniques is to bring out detail that is obscured, or simply to highlight certain features of interest in an image. Such as, changing brightness & contrast etc.
- **Image Restoration:**Image restoration is an area that also deals with improving the appearance of an image. However, unlike enhancement, which is subjective, image restoration is objective, in the sense that restoration techniques tend to be based on mathematical or probabilistic models of image degradation.
- **Color Image Processing:**Color image processing is an area that has been gaining its importance because of the significant increase in the use of digital images over the Internet. This may include color modeling and processing in a digital domain etc.
- **Wavelets and Multi-resolution Processing:**Wavelets are the foundation for representing images in various degrees of resolution. Images subdivision successively into smaller regions for data compression and for pyramidal representation.
- **Compression:**Compression deals with techniques for reducing the storage required to save an image or the bandwidth to transmit it. Particularly in the uses of internet it is very much necessary to compress data.

- **Morphological Processing:**Morphological processing deals with tools for extracting image components that are useful in the representation and description of shape.
- **Segmentation:**Segmentation procedures partition an image into its constituent parts or objects. In general, autonomous segmentation is one of the most difficult tasks in digital image processing. A rugged segmentation procedure brings the process a long way toward successful solution of imaging problems that require objects to be identified individually.
- **Representation and Description:**Representation and description almost always follow the output of a segmentation stage, which usually is raw pixel data, constituting either the boundary of a region or all the points in the region itself. Choosing a representation is only part of the solution for transforming raw data into a form suitable for subsequent computer processing. Description deals with extracting attributes that result in some quantitative information of interest or are basic for differentiating one class of objects from another.
- **Object recognition:**Recognition is the process that assigns a label, such as, "vehicle" to an object based on its descriptors.
- **Knowledge Base:** Knowledge may be as simple as detailing regions of an image where the information of interest is known to be located, thus limiting the search that has to be conducted in seeking that information. The knowledge base also can be quite complex, such as an interrelated list of all major possible defects in a materials inspection problem or an image database containing high-resolution satellite images of a region in connection with change-detection applications.[32]

2.5 Digital image concept[38]:

- **The pixel:** A Digital image is composed of an array of picture elements or pixels. Each pixel represents a single color and value. The computer arrange the pixels to create the illusion of a continuous image, in manner similar to that of a television screen or a Pointillist painting. Every image has an absolute width and height in pixels.
- **Image resolution:** The number of pixels packed into a unit of measure [e.g inch] that determines the quality of the image. This value is the image resolution. Image resolution most commonly refers to the number of pixels per inch. This is called "dots per inch", or dpi. In most cases, higher resolution [higher dpi] results in better image quality. Remember, however, that final image quality is limited by the quality of your image source. While image resolution can always be reduced, increasing resolution will not improve image quality.
- **Image Size:** Image size refers to the real-word dimensions of an image, usually measured in inches. The dimensions of an image are independent of its file size: e.g., a 6" × 8" image at 100 dpi will print out at the same size as a 6" × 8" image at 300 dpi (although the 300 pixel image will have a higher resolution); conversely, a 6" × 4" 100 dpi will print out at half the size of a 6" × 8" 100 dpi image, even though the images have the same resolution.
- **File Size:** File size refers to the amount of memory needed to store a given image document. File size is directly proportional to the number of pixels in an image, the more pixels, the greater the file size.

Since resolution measures dots per square inch, file size is proportional to the square of image resolution. For instance, the file size of a 300 dpi image is 9 times that of a 100 dpi image.

File size also depends on the kind of pixels that comprise the image; e.g., since a full-color pixel needs more memory than a black and white pixel, a 100 dpi color image will consume more memory than a 100 dpi grayscale images. The file format of an image document can also affect its file size.

- **Sampling Image:**Any time the resolution of an image is changed while keeping the image size constant, the image is being sampled. If the resolution is decreased, then the image has been down-sampled.
- **Scaling Image:**In order to scale an image without losing image quality, it is important to understand to relationship between image size and image resolution. When scaling in image, remember this formula: (pixels) = (image size) × (resolution).

When scanning a very small image, such as a color slide or film negative, it may be necessary to greatly increase the scale of an image after scanning. Accordingly, you will notice that a slide scanner will typically allow you to scan at resolutions as high as 2700 dpi (printers and flatbed scanners rarely go above 600 dpi). Such a high resolution may seem excessive, but is necessary to capture all the information contained in a 1" × 1.5" slide. Also the small image size would make for a relatively manageable file size (see above formula). A 2700 dpi 1" × 1.5" image can be scaled into a 300 dpi 6" × 9" image without a change in file size or image quality.[32]

2.6 Types of Digital Images

- **Binary Image**

Each pixel is just black or white. Since there are only two possible values for each pixel (0,1), we only need one bit per pixel , binary image is illustrated in Fig (2-2).

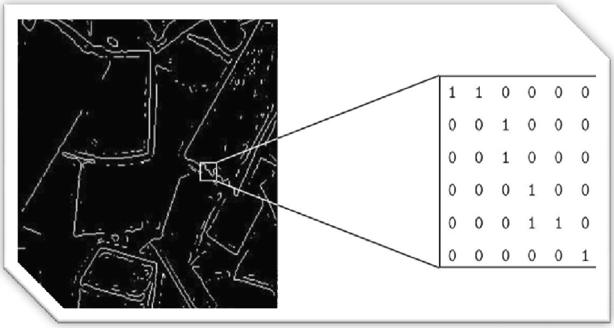
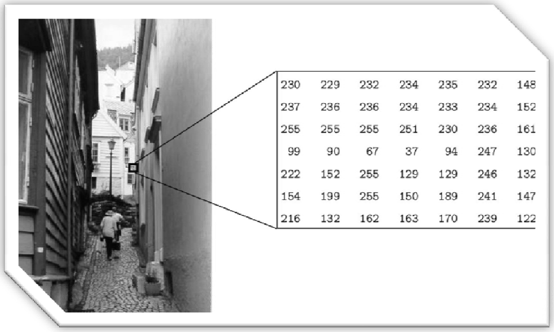


Figure 2-2) Binary Image

- **Grayscale image**

Each pixel is a shade of gray, normally from 0 (black) to 255 (white). This range means that each pixel can be represented by eight bits, or exactly one byte. Other grayscale ranges are used, but generally they are a power of 2, Greyscale image is illustrated in Fig (2-3).



Figure(2-3) Grayscale image

- **True Color image, or RGB**

Each pixel has a particular color; that color is described by the amount of red, green and blue in it. If each of these components has a range 0–255, this gives a total of 2563 different possible colors. Such an image is a "stack" of three matrices; representing the red, green and blue values for each pixel. This means that for every pixel there correspond 3 values, color image is illustrated in Fig (2-4).

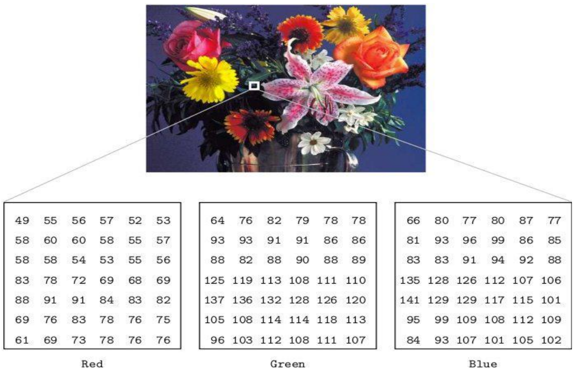


Figure 2-4) Color image (RGB)

- **Indexed image**

Most color images only have a small subset of the more than sixteen million possible colors. For convenience of storage and handling, the image has an associated color map, or color palette, which is simply a list of all the colors used in that image. Each pixel has a value which does not give its color (as for an RGB image), but an index to the color in the map. It is convenient if an image has (256) colors or less, for then the index values will only require one byte each to store. Some image file formats (for example, CompuServe

GIF), allow only (256) colors or fewer in each image, for precisely this reason. Figure.4. shows an example. In this image the indices, rather than being the grey values of the pixels, are simply indices into the color map. Without the color map, the image would be very dark and colorless. In the figure, for example, pixels labeled 5 correspond to 0.2627 0.2588 0.2549, which is a dark grayish color. Indexed image is illustrated in Fig (2-5).[32]

4	5	5	5	5	5
5	4	5	5	5	
5					
5	5	5	0	5	5
5	5	5	5	11	
11					
5	5	5	8	16	20
8	11	11	26	33	20
1	20	33	33	58	37

Indices

0.1211	0.1211	0.1416
0.1807	0.2549	0.1729
0.2197	0.3447	0.1807
0.1611	0.1768	0.1924
0.2432	0.2471	0.1924
0.2119	0.1963	0.2002
0.2627	0.2588	0.2549
0.2197	0.2432	0.2588

Color map

Figure (2-5) Indexed Image

2.7 Digital Image Noise Concept [39]

Image noise is random (not present in the object imaged) variation of brightness or color information in images, and is usually an aspect of electronic noise. It can be produced by the sensor and circuitry of a scanner or digital camera. Image noise can also originate in film grain and in the unavoidable shot noise of an ideal photon detector. Image noise is an undesirable by-product of image capture that adds spurious and extraneous information.

The magnitude of image noise can range from almost imperceptible specks on a digital photograph taken in good light, to optical and radio astronomical images that are almost entirely noise, from which a small amount of information can be derived by sophisticated processing (a noise level that would be totally unacceptable in a photograph since it would be impossible to determine even what the subject was).

2.7.1 Types of noise [39]

Various types of noise have their own characteristics and are inherent in images in different ways.

- **Gaussian noise [39]**

Principal sources of Gaussian noise in digital images arise during acquisition e.g. sensor noise caused by poor illumination and/or high temperature, and/or transmission e.g. electronic circuit noise.

A typical model of image noise is Gaussian, additive, independent at each pixel, and independent of the signal intensity, caused primarily by Johnson–Nyquist noise (thermal noise), including that which comes from the reset noise of capacitors ("kTC noise").Amplifier noise is a major part of the "read noise" of an image sensor, that is, of the constant noise level in dark areas of the image. In color cameras where more amplification is used in the blue color channel than in the green or red channel, there can be more noise in the blue channel. At higher exposures, however, image sensor noise is dominated by shot noise, which is not Gaussian and not independent of signal intensity.

- **Salt-and-pepper noise[39]**

Image with salt and pepper noise Fat-tail distributed or "impulsive" noise is sometimes called salt-and-pepper noise or spike noise. An image containing salt-and-pepper noise will have dark pixels in bright regions and bright pixels in dark regions. This type of noise can be caused by analog-to-digital converter errors, bit errors in transmission, etc. It can be mostly eliminated by using dark frame subtraction and interpolating around dark/bright pixels.Dead pixels in an LCD monitor produce a similar, but non-random, display.

- **Shot noise[39]**

The dominant noise in the lighter parts of an image from an image sensor is typically that caused by statistical quantum fluctuations, that is, variation in the number of photons sensed at a given exposure level. This noise is known as photon shot noise. Shot noise has a root-mean-square value proportional to the square root of the image intensity, and the noises at different pixels are independent of one another. Shot noise follows a Poisson distribution, which is usually not very different from Gaussian.

- **Quantization noise (uniform noise)**

The noise caused by quantizing the pixels of a sensed image to a number of discrete levels is known as quantization noise. It has an approximately uniform distribution.

- **Film grain**

The grain of photographic film is a signal-dependent noise, with similar statistical distribution to shot noise. If film grains are uniformly distributed (equal number per area), and if each grain has an equal and independent probability of developing to a dark silver grain after absorbing photons, then the number of such dark grains in an area will be random with a binomial distribution. In areas where the probability is low, this distribution will be close to the classic Poisson distribution of shot noise. A simple Gaussian distribution is often used as an adequately accurate model.

Film grain is usually regarded as a nearly isotropic (non-oriented) noise source. Its effect is made worse by the distribution of silver halide grains in the film also being random. [1]

- **Speckle Noise (Multiplicative Noise)**

While Gaussian noise can be modeled by random values added to an image, speckle noise can be modeled by random values multiplied by pixel values hence it is also called multiplicative noise. Speckle noise is a major problem in some radar applications.

2.7.2 Noise Modeling [1]

Noise is present in image either in additive or multiplicative form.

- **Additive Noise Model**

Noise signal that is additive in nature – Equation (1) gets added to the original signal to produce a corrupted noisy signal such as salt-pepper noise and follows the following model:

$$w(x,y) = s(x,y) + n(x,y)$$ ***Equation (2-1) Additive Noise***

- **Multiplicative Noise Model**

In this model, noise signal gets multiplied to the original signal Equation (2). The multiplicative noise model such as Gaussian noise and follows the following rule:

$$w(x,y) = s(x,y) \times n(x,y)$$ ***Equation 2-2) Multiplicative Noise***

Where, s(x,y) is the original image intensity and n(x,y) denotes the noise introduced to produce the corrupted signal w(x,y) at (x,y) pixel location.

2.7.3 The concept of noise reduction:

We will describe how a simple noise reduction algorithm works. This is definitely not the best, but the concept is very clear and most of the implementations in digital cameras are based on this idea:

- divide the signal in noise and image content
- reduce the noise
- put the image together again
- We start with an image that shows noise we want to reduce (Original Image). Make a low-pass filtering for noisy image. De-noising concept is illustrated in Fig(2-6).

Formatted: Space After: 5.7 pt

Figure (2-6) De-noising Concept

2.7.4 Noise Removal[40]

Image de-noising is still a challenging problem for researchers as image de-noising causes blurring and introduces artifacts. Different types of images inherit different types of noise and different noise models are used to present different noise types. De-noising method tends to be problem specific and depends upon the type of image and noise model. Various noise types, noise models and de-noising methods are discussed in this research.

2.7.5 Image filtering

The factors that degrade the quality of images result in blurred and noisy images with poor resolution. One of the most important factors that greatly affect the quality of images is image filtering. Image filtering is a mathematical processing for noise removal and resolution recovery. The goal of the filtering is to compensate for loss of detail in an image while reducing noise. Filters suppressed noise as well as de-blurred and sharpened the image. In this way, filters can greatly improve the image resolution and limit the degradation of the image. An image can be filtered either in the frequency or in the spatial domain.

Basics steps of filtering in the frequency domain Fig (2-7):

- The initial data is Fourier transformed.
- Multiplied with the appropriate filter.

- Taking the inverse Fourier transform,
- Re-transformed into the spatial domain.

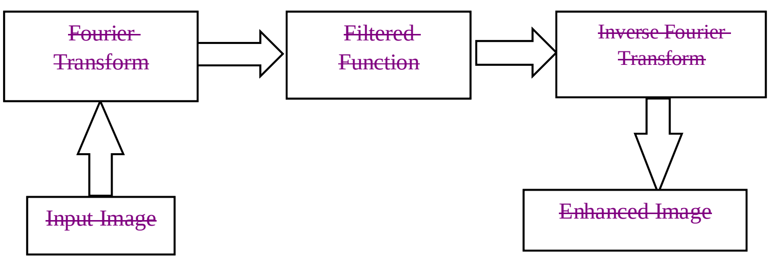
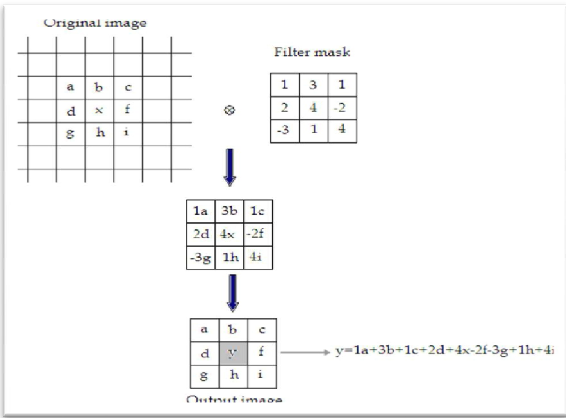


Figure (2-7) Frequen

Formatted: Space After: 5.7 pt

2.7.6 Basics steps of Spatial Domain Filtering

The filtering in the spatial domain demands a filter mask (it is also referred as kernel or convolution filter). The filter mask is a matrix of odd usually size which is applied directly on the original data of the image. The mask is centered on each pixel of the initial image. For each position of the mask the pixel values of the image is multiplied by the corresponding values of the mask. The products of these multiplications are then added and the value of the central pixel of the original image is replaced by the sum. This must be repeated for every pixel in the image. The procedure is described schematically in Fig. 2. If the filter, by which the new pixel value was calculated, is a linear function of the entire pixel values in the filter mask (e.g. the sum of products), then the filter is called linear. If the output pixel is not a linear weighted combination of the input pixel of the image then the filtered is called non-linear. According to the range of frequencies they allow to pass through filters can be classified as low pass or high pass. Low pass filters allow the low frequencies to be retained unaltered and block the high frequencies. Low pass filtering removes noise and smooth the image but at the same time blur the image as it does not preserve the edges. High pass filters sharpness the edges of the image (areas in an image where the signal changes rapidly) and enhance object edge information. A severe disadvantage of high pass filtering is the amplification of statistical noise present in the measured counts. The next section is referred to three of the most common filters used by **MATLAB**: the mean, median and Gaussian filter.Filtering process in spatial domain is illustrated in Fig (2-8).



Figure(2-8) Filtering process in spatial domain

- **Linear Filters**

Linear filtering is filtering in which the value of an output pixel is a linear combination of the values of the pixels in the input pixel'sneighborhood.

Linear smoothing filters

One method to remove noise is by convolving the original image with a mask that represents a low-pass filter or smoothing operation. For example, the Gaussian mask comprises elements determined by a Gaussian function. This convolution brings the value of each pixel into closer harmony with the values of its neighbors. In general, a smoothing filter sets each pixel to the average value, or a weighted average, of itself and its nearby neighbors; the Gaussian filter is just one possible set of weights. Smoothing filters tend to blur an image, because pixel intensity values that are significantly higher or lower than the surrounding neighborhood would "smear" across the area. Because of this blurring, linear filters are seldom used in practice for noise reduction; they are, however, often used as the basis for nonlinear noise reduction filters. Mean filter approach is illustrated in Fig (2-9).

a	b	c
d	e	f
g	h	i

$$\longrightarrow \frac{1}{9}(a+b+c+d+e+f+g+h+i)$$

Figure 2-9) Filtering approach of mean filter

The Fig.(10) depicts that by using the mean filter, the central pixel value would be changed from “e” to ‘’ (a+b+c+d+e+f+g+h+i) 1/9’’.

- **Adaptive Filter**

The wiener function applies a Wiener filter (a type of linear filter) to an image adaptively, tailoring itself to the local image variance. If the variance is large, wiener performs little smoothing. If it is small, wiener performs more smoothing. This approach often produces better results than linear filtering. The adaptive filter is more selective than a comparable linear filter, preserving edges and other high-frequency parts of an image. In addition, there are no design tasks; the *wiener2* function handles all preliminary computations and implements the filter for an input image. $wiener_2$, however, does require more computation time than linear filtering. Wiener works best when the noise is constant-power ("white") additive noise, such as Gaussian noise. Another method for removing noise is to evolve the image under a smoothing partial differential equation similar to the heat equation which is called anisotropic diffusion.[26]

- **Non-Linear Filters**

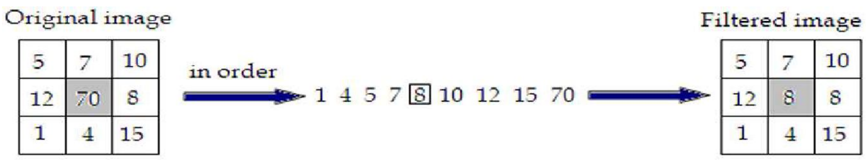
In recent years, a variety of nonlinear median type filters such as weighted median, rank conditioned rank selection, and relaxed median have been developed to overcome this shortcoming.

- **Median Filter**

A median filter is an example of a non-linear filter and, if properly designed, is very good at preserving image detail. To run a median filter:

1. Consider each pixel in the image.
2. Sort the neighboring pixels into order based upon their intensities
3. Replace the original value of the pixel with the median value from the list.

A median filter is a rank-selection (RS) filter, a particularly harsh member of the family of rank-conditioned rank-selection *(RCRS)* filters, a much milder member of that family, for example one that selects the closest of the neighboring values when a pixel's value is external in its neighborhood, and leaves it unchanged otherwise, is sometimes preferred, especially in photographic applications. *Median* and other *RCRS* filters are good at removing salt and pepper noise from an image, and also cause relatively little blurring of edges, and hence are often used in computer vision applications. *Median* filtering is similar to using an averaging filter, in that each output pixel is set to an average of the pixel values in the neighborhood of the



corresponding input pixel. However, with *median* filtering, the value of an output pixel is determined by the *median* of the neighborhood pixels, rather than the mean. The median is much less sensitive than the mean to extreme values (called outliers*)*. *Median* filtering is therefore better able to remove these outliers without reducing the sharpness of the image, Median filtering approach is illustrated in Fig (2-10).

Figure 2-10) Filtering approach of Median Filter

- **Fuzzy Filter**

  Fuzzy filters provide promising result in image-processing tasks that cope with some drawbacks of classical filters. Fuzzy filter is capable of dealing with vague and uncertain information . Sometimes, it is required to recover a heavily noise corrupted image where a lot of uncertainties are present and in this case fuzzy set theory is very useful. Each pixel in the image is represented by a membershipfunction and different types of fuzzy rules that considers the neighborhood information or other information to eliminate filter removes the noise with blurry edges but fuzzy filters perform both the edge preservation and smoothing.

CHAPTER 3

3 Methodology

In order to achieve our objectives, we will usedigital image processing tools inMATLAB.

3.1 Experimental setup

Table (3-1) Experimental Setup

Component	Parameter	
Image	Image Size	128x128 pixel
		256x256 pixel
		512x512 pixel
	Type	Grey-Scale
Noise	Gaussian + Speckle+ Salt & Pepper	
Hardware	Processor :i5-64 bit	
	Ram : 4 GB	
Software	MATLAB 7.12.0	

3.2 Performance Metrics

- Peak Signal to Noise Ratio -Equation (3-1):is the measure of peak error. It is an expression used to depict the ratio [4, 5] of maximum possible power of image (signal) and the power of the corrupting noise that affects the quality of its representation. It is represented in terms of mean square error as:

$$PSNR = 10\ log_{10}\left(\frac{MAX_I^2}{MSE}\right)$$ *Equation 3-1) Peak Signal to Noise Ratio*

MAX_I is the maximum possible pixel value of the image. It is equal to 255 for 8 bit represented image.

- Mean square error-Equation (3-2) is the cumulative squared error between the final, de-noised image and the original image before introduction of noise. This enables us to compare mathematically as to which method provides better results under same conditions like image size noise, etc. It is mathematically stated as:

$$MSE = \frac{1}{mn}\sum_{y=1}^{n}\sum_{x=1}^{m}\left[I(x,y) - I\ (\acute{x},y)\right]^2$$ ***Equation 3-2) Mean Square Error***

Where the image size is m x n. I(x,y) is the intensity of original image in spatial coordinates , I '(x,y) is the intensity of de-noised image in spatial coodinates.

- Mean absolute error- Equation (3-3) is the absolute error between the original image and the de-noised image obtained after applying one of the filters. It is used to measure the closeness to the true or original value of the pixel with respect to the de-noised pixel. It is given by:

$$MAE = \frac{1}{mn}\sum_{y=1}^{n}\sum_{x=1}^{m}\left[I(x,y) - I\ (\acute{x},y)\right]$$ ***Equation 3-3) Mean absolute error***

Where the image size is m x n. I(x,y) is the intensity of original image in spatial coordinates , I '(x,y) is the intensity of de-noised image in spatial coodinates.

- Image Quality- The original image and the de-noised image were placed side by side to compare the variance in degradation of image quality. This was tested on the images of varying sizes.
- Time Complexity- is used to define the time taken by each method under varying parametric conditions like image size, noise density, *etc*. Time

complexity defines the complexity of each algorithm and is hence used to define the algorithm with least and maximum computational cost.

Simulation Setup [41]

Algorithms were developed in **MATLAB** to simulate the methods for noise generation, creating a noisy image and finally perform filtering, Table(3-2).

Table (3-2) Noise Types and Description

Value	Description
'Gaussian'	Gaussian white noise
'Salt & pepper'	On and off pixels
'Speckle'	Multiplicative noise

Formatted Table

First a noisy image was generatedusing (**Imnoise) function;**

Syntax:

J = imnoise(I,type)

J = imnoise(I,type,parameters)

Description

J = imnoise(I,type) adds noise of a given type to the intensity image I. type is a string that can have one of these values.

J = imnoise(I,type,parameters) accepts an algorithm type plus additional modifying parameters particular to the type of algorithm chosen. If you omit these arguments, imnoise uses default values for the parameters. Here are examples of the noise types and their parameters:

J = imnoise(I,'gaussian',m,v) adds Gaussian white noise of mean m and variance v to the image I. The default is zero mean noise with 0.01 variance. J = imnoise(I,'localvar',V) adds zero-mean, Gaussian white noise of local variance V to the image I. V is an array of the same size as I.

J = imnoise(I,'localvar',image_intensity,var) adds zero-mean, Gaussian noise to an image I, where the local variance of the noise, var, is a function of the image intensity values in I. The image_intensity and var arguments are vectors of the same size, and , and plot(image_intensity,var) plots the functional relationship between noise variance and image intensity. The image_intensity vector must contain normalized intensity values ranging from 0 to 1.

J = imnoise(I,'poisson') generates Poisson noise from the data instead of adding artificial noise to the data. In order to respect Poisson statistics, the intensities of unit8 and uint16 images must correspond to the number of photons (or any other quanta of information). Double-precision images are used when the number of photons per pixel can be much larger than 65535 (but less than 10^12); the intensity values vary between 0 and 1 and correspond to the number of photons divided by 10^12.

J = imnoise(I,'salt&pepper',d) adds salt and pepper noise to the image I, where d is the noise density. This affects approximately d*prod(size(I)) pixels. The default is 0.05 noise density.

J = imnoise(I,'speckle',v) adds multiplicative noise to the image I, using the equation J = I+n*I, where n is uniformly distributed random noise with mean 0 and variance v. The default for v is 0.04.

The mean and variance parameters for 'gaussian', 'localvar', and 'speckle' noise types are always specified as if the image were of class double in the range [0, 1]. If the input image is of class uint8 or uint16, the imnoise function converts the image to double, adds noise according to the specified type and parameters, and then converts the noisy image back to the same class as the input can be of class uint8, uint16, or double. The output image J is of the same class as I. If I has more than two dimensions it is treated as a multidimensional intensity image and not as an RGB image.

Formatted: Space After: 5.7 pt

Formatted: Space After: 5.7 pt

3.2.1 Filtering in Matlab: [41]

In **MATLAB**, using Image Processing Toolbox we can design and implemented filters for image data. For linear filtering, **MATLAB** provides the fspecial command to generate some predefined common 2D filters.

h= fspecial(filtername, parameters).

The filter name is one of the average, disk, gaussian, laplacian, log, motion, prewitt, sobel and un sharp filters; that is the parameters related to the specific filters that are used each time. Filters are applied to 2D images using the function filter2 with the syntax:

Y = filter2(h,X)

The function filter2 filters the data in matrix X with the filter h. For multidimensional images the function imfilter is used.

B = imfilter(A,h)

This function filters the multidimensional array A with the multidimensional filter h. imfilter function is more general than filter2 function. For nonlinear filtering in **MATLAB** the function nlfilter is applied, requiring three arguments: the input image, the size of the filter and the function to be used.

To simulate the effects of some of the problems, the toolbox provides the imnoise function, which you can use to add various types of noise to an image. The examples in this section use this function.

For example, an averaging filter is useful for removing grain noise from a photograph. Because each pixel gets set to the average of the pixels in its neighborhood, local variations caused by grain are reduced

This section discusses linear filtering in **MATLAB** and the Image Processing Toolbox. It includes:

- A description of how to perform filtering using the imfilter function
- A discussion about using predefined filter types

The following example compares using an averaging filter and medfilt2 to remove salt and pepper noise. This type of noise consists of random pixels' being set to black or white (the extremes of the data range). In both cases the size of the neighborhood used for filtering is 3-by-3:

1- Filter the noisy image Fig(3-1) with an averaging filter and display the results.

```
K = filter2(fspecial('average',3),J)/255;
figure, imshow(K)
```

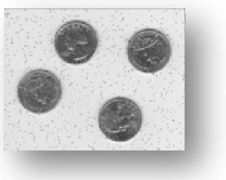


Figure (3-1) Noisy Image

Formatted: Space After: 5.7 pt

2. Now use a median filter to filter the noisy image Fig(3-2) and display the results. Notice that medfilt2 does a better job of removing noise, with less blurring of edges.

```
L = medfilt2(J,[3 3]);
figure, imshow(K)
figure, imshow(L)
```

Figure (3-2) Filtered Image

3.3 Graphic User Interface Block Diagram:

The user interaction (GUI) Block diagram is shown in Fig (3-3) below.

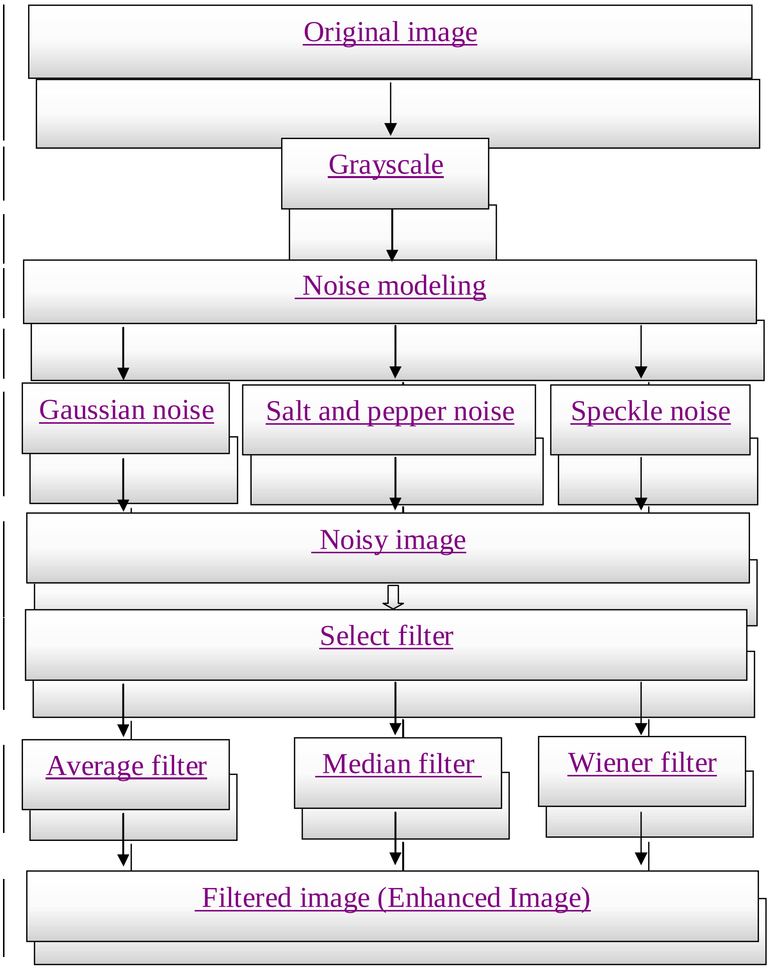


Figure (3-3) GUI Block Diagram

CHAPTER 4

4 Results

Consider three grayscale images with different sizes, three types of analysis was performed:

1. Qualitative Analysis.
2. Peak Signal to Noise Ration (PSNR) , Mean Square Error (MSE), Mean Absolute Error (MAE).
3. Histogram Analysis.

4.1 Qualitative Analysis:

In this analysis noise and filter effects were pictorially examined on each image size.

1. **Gaussian Noise Effect :**

From the following Figures, we notice that the image Quality of de-noised image increases with the increase in image size for a constant impulse noise density. Best results filtering with *Weiner* Filter for restoring a Gaussian noisy image , fig (4-1).

Figure 4-1)Gaussian Noise

1. **Slat & Pepper Noise Effect :**

From the following tables we notice that the image Quality of de-noised image increases with the increase in image size for a constant impulse noise density.

Best results filtering with *Median* Filter for a salt & Pepper noisy image , Fig (4-2).

Figure 4-2) Salt & Pepper Noise

1. **Speckle Noise Effect :**

From the following tables we notice that the image Quality of de-noised image increases with the increase in image size for a constant impulse noise density.

Best results filtering with Weiner Filter for speckle noisy image.

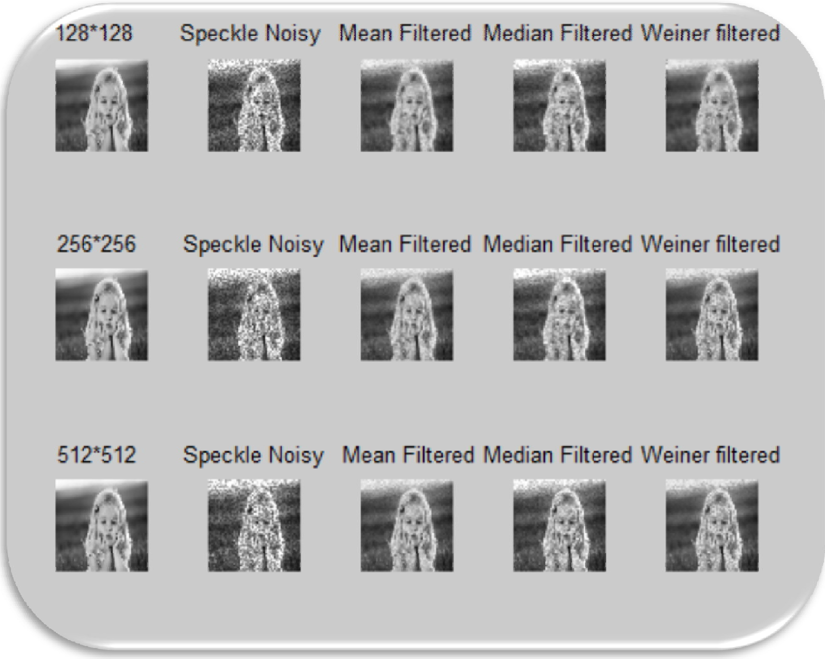


Figure (4-3) Effect of Speckle Noise

4.2 Impact on Peak Signal to Noise Ratio(PSNR), Mean Absolut Error (MAE) and Mean Square Error (MSE) :

Image quality measurement plays an important role in various image processing application. A great deal of effort has been made in recent years to develop objective image quality metrics.

MSE and PSNR are very simple, easy to implement and have low computational complexities. But these methods do not show good results. MSE and PSNR are acceptable for image similarity measure only when the images differ by simply increasing distortion of a certain type.

The Table below show the overall values of *PSNR, MAE,* and *MSE* for different image sizes .

PSNR is the primary measure ,then MSE and MAE.

Table (4-1) 128*128 PSNR, MAE, and MSE

128*128									
	Gaussian			**Salt &Pepper**			**Speckle**		
Filter	PSNR dB	MSE	MAE	PSNR dB	MSE	MAE	PSNR dB	MSE	MAE
Mean	80.44	24336	156	80.44	24336	156	80.44	24336	156
Median	79.19	64516	254	86.92	64516	254	80.10	64516	254
Weiner	81.66	900	30	81.66	900	30	82.02	529	23

Formatted Table

Table (4-2) 256*256 PSNR, MAE, and MSE

256*256									
	Gaussian			**Salt &Pepper**			**Speckle**		
Filter	PSNR dB	MSE	MAE	PSNR dB	MSE	MAE	PSNR dB	MSE	MAE
Mean	80.69	21316	146	85.52	19881	141	81.78	22801	151
Median	79.67	64516	254	79.67	64516	254	80.49	64516	254
Weiner	82.91	225	15	82.91	225	15	82.94	196	14

Formatted Table

Table(4-3) 512*512 PSNR, MAE, and MSE

512*512

Formatted Table

	Gaussian			Salt &Pepper			Speckle		
Filter	PSNR dB	MSE	MAE	PSNR dB	MSE	MAE	PSNR dB	MSE	MAE
Mean	81.15	21025	145	81.15	21025	145	81.15	21025	125
Median	80.02	64516	254	80.02	64516	254	80.71	64516	254
Weiner	83.86	100	10	83.86	100	10	82.78	1444	38

- Peak signal-to-noise ratio (PSNR) was employed with first priority to illustrate the quantitative quality of the reconstructed image for various methods. On increasing image size with constant impulse noise density, PSNR increases, MSE and MAE decreases. This is because the ratio of image size to noise density increases with increase in image size and constant noise, therefore the output image is recovered.

4.3 Histogram of noisy and filtered image, using different filters:

The histogram of an image (i.e., a plot of the gray levelfrequencies) provides important information regarding thecontrast of an image.

- **Gaussian Noise**

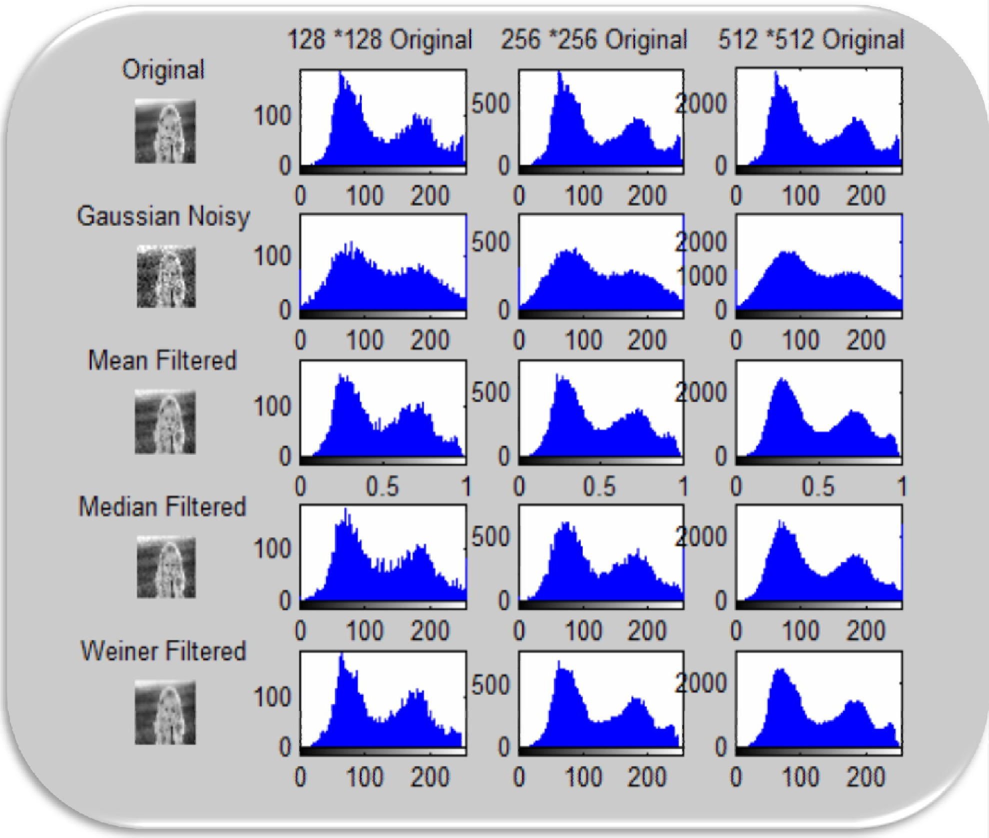


Figure (4-4) Gaussian Noise

- **Salt & Pepper Noise**

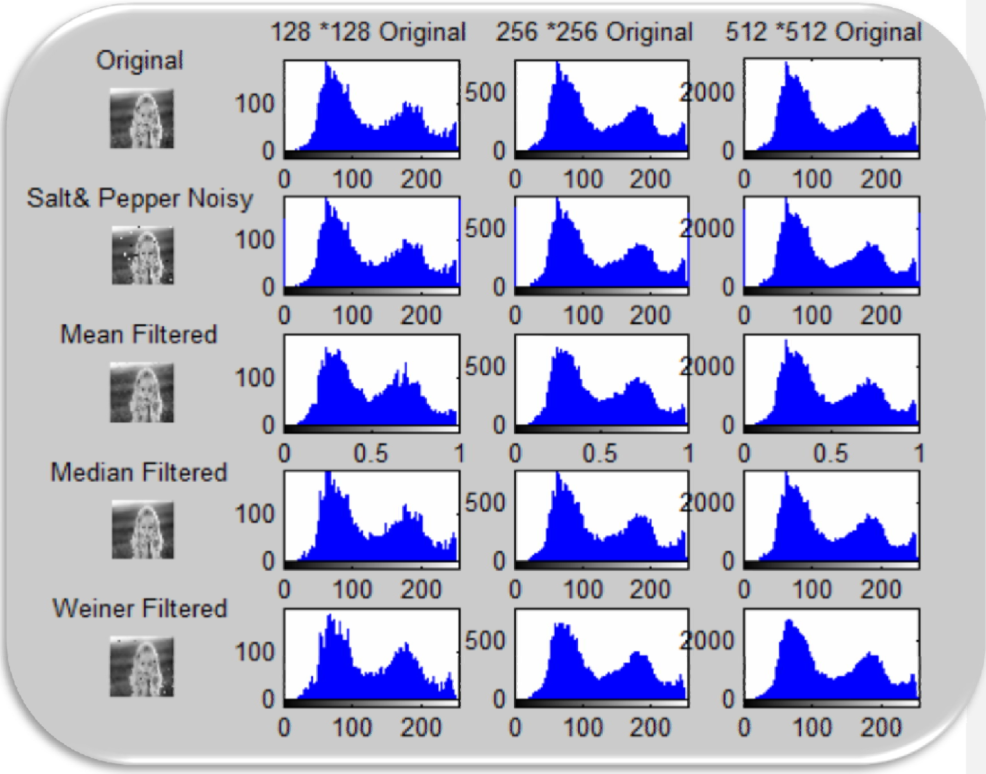


Figure 4-5) Salt & Pepper

- **Speckle Noise**

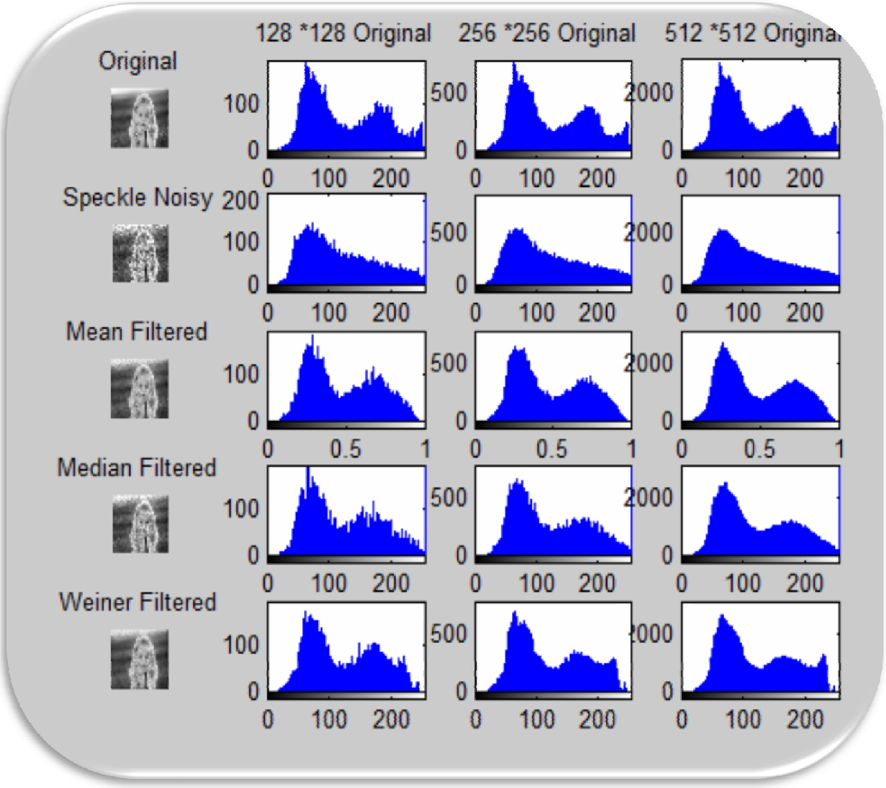


Figure (4-6) Speckle Noise

- Histogram analysis shows that noise severity is less when increasing image size.

CHAPTER 5

5 Conclusion

A Grey scale Image(Confused) in “JPG” format was used, adding three noises (Gaussian, Salt & Pepper and Speckle) to original image with default deviation (figure to figure) ,De-noised all noisy images by (Mean, Median and Wiener) filters and conclude from the results that:

The performance of the Wiener Filter after de-noising for all Speckle and Gaussian noise is better than Mean filter and Median filter.

The performance of the Median filter after de-noising for all Salt & Pepper noise is better than Mean filter and Wiener filter.

The image quality of de-noised image increases with the increase in image size for a constant impulse noise density.

5.1 Future Work:

1. Further study will be in enhancing the user interaction interface by including more image types (True color (RGB) and Indexed digital image) , more filters especially fuzzy filters.
2. Additional performance metric: Using metricadditional to Peak Signal to Noise Ratio(PSNR) ,Mean Absolute Error (MAE), Mean Squre Error and Histogram analysis that students and researchers can more benefit from the designed system.

References

1. A Comparative Study of Various Types of Image Noise and Efficient Noise Removal Techniques -Mr. RohitVerma Dr. Jahid Ali -School of Information Technology SSCIMT, Badhani, Pathankot, APJIMTC, Jalandhar, India India.
2. Bhateja, V., Devi, S.: An Improved Non-Linear Transformation Function for Enhancement of Mammographic Breast Masses. In: Proc. of (IEEE) 3rd International Conference on Electronics & Computer Technology (ICECT 2011), Kanyakumari (India), vol. 5, pp. 341–346 (2011)
3. Chen, L., Lu, G., Zhang, D.: Effects of Different Gabor Filter Parameters on Image Retrieval by Texture. In: Proc. of IEEE 10th International Conference on Multi-Media Modelling, Brisbane, Australia, pp. 273–278 (January 2004)
4. D. Maheswariet. al. Noise removal in compound image using median filter. (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 2010, 1359-1362 Department of Computer Science and Engineering & IT - SIPNA's College of Engineering Amravati(M.S) India Vol 2, No 1 (January 2011) ©IJoAT
5. Emir Kremić*, AbdulhamitSubaşi*- *The International Arab Conference on Information Technology (ACIT'2013)**Faculty of Engineering and Information Technology, International Burch University, Francuskerevolucije bb., 71210 Ilidža, Bosnia and Herzegovina .k.emir@acm.org, asubasi@ibu.edu.ba.
6. Impulse Noise Suppression and Edge Preservation of Digital Images Using Image Fusion R.C.Gonzalez, R.E. Wood: Digital Image Processing, Prentice Hall, India Second Edition, 2007. [2] R.H.Laskar, B. Bhowmick,

R. Biswas and S. Kar: Removal of impulse noise from color images, 2009 IEEE. James C Church, Yixin Chen and Stephen V Rice:A Spatial median filter for noise removal in digital images, 2008 IEEE.

7. International Journal of Computer & Organization Trends – Volume 5 – February 2014- Kamlesh Dixit#1, MohitKhandelwal*2 ,M.Tech Scholar#1, Associate -Professor*2 -IET, Alwar, India.
8. International Journal of Scientific and Research Publications, Volume 2, Issue 10, October 2012 1 ISSN 2250-3153 www.ijsrp.org.
9. International Journal of Signal Processing, Image Processing and Pattern Recognition Vol.7, No.1 (2014) ,Copyright ⓒ 2014 SERSC 409
10. Jain, A., Bhateja, V.: A Novel Image Denoising Algorithm for Suppressing Mixture of Speckle and Impulse Noise in Spatial Domain. In: Proc. of IEEE 3rd International Conference on Electronics and Computer Technology, Kanyakumari, India, pp. 207–211 (April 2013).
11. Jain, A., Singh, S., Bhateja, V.: A Robust Approach for Denoising and Enhancement of Mammographic Breast Masses. International Journal on Convergence Computing 1(1), 38–49 (2013) .
12. K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
13. Kaur, A., Singh, K.: Speckle Noise Reduction by Using Wavelets. In: National Conference on Computational Instrumentation, Chandigarh, India, pp. 198–203 (March 2010).
14. Kenneth, "Digital Image Processing", Prentice Hall, New Jersey, R. 1979
15. Mohideen, S.K., Perumal, S.A., Sathik, M.M.: Image Denoising Using Discrete Wavelet Transform. International Journal of Computer Science and Network Security 8(1), 213–216 (2008).

16. Mr. PawanPatidar and et al. Image De-noising by Various Filters for Different Noise in International Journal of Computer Applications (0975 – 8887) Volume 9– No.4, November 2010
17. Mr. PawanPatidar and et al.: Image De-noising by Various Filters for Different Noise in International Journal of Computer Applications (0975 – 8887) Volume 9 November 2010 K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080–2095, Aug. 2007. D. Maheswariet. al. Noise Removal In Compound Image Using Median Filter. (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 2010, 1359-1362 .
Mr. RohitVerma Dr. Jahid Ali -School of Information Technology SSCIMT, Badhani, Pathankot, -APJIMTC, Jalandhar, India India
18. Old Dominion University, Norfolk, VA 23529
19. Perreault, S., Hébert, P.: Median Filtering in Constant Time. IEEE Trans. on Image Processing 16(6), 2389–2394 (2007) .
20. Prashant R. Deshmukh and Milindkumar V. Sarode
21. RaghadJawadAhmed . PhD student, Department of Computer Science and Engineering, University Politehnica of Bucharest, Romania.Vol. 30, No. 7, 1997, pp. 1183-1189.
22. Salt and Pepper Noise Detection and Removal in Gray Scale Images: An Experimental Analysis -E.JebamalarLeavline, D.Asir Antony GnanaSinghBharathidasan Institute of Technology, Anna University Chennai, Regional Centre Tiruchirappalli- 620 024 jebi.lee@gmail.com, asirantony@gnail.com.
23. Singh, S., Jain, A., Bhateja, V.: A Comparative Evaluation of Various Despeckling Algorithms for Medical Images. In: Proc. of (ACMICPS)

CUBE International Information Technology Conference & Exhibition, Pune, India, pp. 32–37 (September 2012).

24. Solbo, S., Eltoft, T.: A Stationary Wavelet Domain Wiener Filter for Correlated Speckle. IEEE Trans. on Geoscience and Remote Sensing 46(4), 1219–1230 (2008).
25. Solbo, S., Eltoft, T.: Homomorphic Wavelet Based Statistical Despeckling of SAR Images. IEEE Trans. on Geoscience and Remote Sensing 42(4), 711–721 (2004) .
26. Strela, V.: Denoising via Block Wiener Filtering in Wavelet Domain. In: 3rd European Congress of Mathematics, Barcelona, Spain (July 2000)
27. Tong, Z., Chen, X., Akram, M.N., Aksnes, A.: Compound Speckle Characterization Method and Reduction by Optical Design. Journal of Display Technology 8(3), 132–137 (2012).
28. Mr. RohitVerma Dr. Jahid Ali -School of Information Technology SSCIMT, Badhani, Pathankot, -APJIMTC, Jalandhar, India India.
29. Salt and Pepper Noise Detection and Removal in Gray Scale Images: An Experimental Analysis- Bharathidasan Institute of Technology, Anna University Chennai, Regional Centre Tiruchirappalli- 620 024 jebi.lee@gmail.com, asirantony@gmail.com.
30. Mr. PawanPatidar and et al. Image De-noising by Various Filters for Different Noise in International Journal of Computer Applications (0975 – 8887) Volume 9– No.4, November 2010
31. 1 K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, “Image denoising by sparse 3D transform-domain collaborative filtering,” IEEE Trans. Image Process., vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
32. A Brief Study Of Various Noise Model And Filtering Techniques ,Priyanka Kamboj1 and Versha Rani*21(Assistant Prof.) Department of Computer Science and Application, Seth Jai ParkashMukandLal Institute

of Engineering & Technology College, Radaur, Yamuna Nagar, Haryana, India .

33. Digital ImageProcessingUsing MATLABSecond Edition Rafael C.Gonzalez University of TennesseeRichard E. WoodsMedData InteractiveSteven L. ddins. The MathWorks, Inc.
34. Various Image Enhancement Tecniques International- S.S. Bedi ,RatiKhandelwal- Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue3 ,March 2013 , RatiKhandelwal-Assistant Professor.
35. Brief Review On Image Denoising Techniques- Rahul Singh , Preeti Singh , FarzanaParveen
36. Noise Reduction in Digital Imaging—An Exploration of the State of The Art- James Ellenberger for CS525—Multimedia Computing and Communications Professor—C. Edward Chow Spring 2010 University of Colorado Colorado Springs.
37. Noise Reduction in Digital Imaging- by J Ellenberger - 2010
38. Digital Image Processing- Second Edition ,Rafael C. Gonzalez University of Tennessee Richard E. Woods
39. Image noise - From Wikipedia, the free encyclopedia.
40. The National Programme on Technology Enhanced Learning (NPTEL), Vedios of digital image processing.
41. Mathork tutorials.

APPENDIX

MATLAB CODE:

```
% loading original image (I)
I1=imread('C:\Users\Ali\Desktop\Research\Images\confused.jpg');
imshow (I1)

% Original resize 128*128
C1 = imresize(I1, [128 128]);
imshow (C1)
% Convert Original to Greay Scale
GC1=rgb2gray(C1);
imshow (GC1)

% add Gaussian noise with 1%
NGA1 = imnoise(GC1,'Gaussian',0,.01);
imshow (NGA1)



% filter image using mean filter & Display results
FMEAN1=filter2(fspecial('average',3),NGA1)/255;
imshow (FMEAN1)
% filter image using median filter & Display results
FMED1=medfilt2(NGA1);
imshow (FMED1)
% filter image using weiner filter & Display results
FWEIN1 = wiener2(NGA1,[5 5]);
imshow (FWEIN1)

% Original resize 256*256
C2 = imresize(I1, [256 256]);
imshow (C2)
% Convert Original to Greay Scale
GC2=rgb2gray(C2);
imshow (GC2)

% add Gaussian noise with 1%
NGA2 = imnoise(GC2,'Gaussian',0,.01);
imshow (NGA2)


% filter image using mean filter & Display results
FMEAN2=filter2(fspecial('average',3),NGA2)/255;
imshow (FMEAN1)
% filter image using median filter & Display results
FMED2=medfilt2(NGA2);
imshow (FMED1)
% filter image using weiner filter & Display results
FWEIN2 = wiener2(NGA2,[5 5]);
imshow (FWEIN2)
```

```
% Original resize 512*512
C3 = imresize(I1, [512 512]);
imshow (C3)
% Convert Original to Greay Scale
GC3=rgb2gray(C3);
imshow (GC3)

NGA3 = imnoise(GC3,'Gaussian',0,.01);
imshow (NGA3)


% filter image using mean filter & Display results
FMEAN3=filter2(fspecial('average',3),NGA3)/255;
imshow (FMEAN3)
% filter image using median filter & Display results
FMED3=medfilt2(NGA3);
imshow (FMED3)
% filter image using weiner filter & Display results
FWEIN3 = wiener2(NGA3,[5 5]);
imshow (FWEIN3)

subplot(3,5,1)
imshow(GC1)
title('128*128 Original ')

subplot(3,5,2)
imshow(NGA1)
title('Gaussian Noisy ')

subplot(3,5,3)
imshow(FMEAN1)
title('Mean Filtered')

subplot(3,5,4)
imshow(FMED1)
title('Median Filtered')

subplot(3,5,5)
imshow(FWEIN1)
title('Weiner filtered')

subplot(3,5,6)
imshow(GC2)
title('256*256 Original')

subplot(3,5,7)
imshow(NGA2)
title('Gaussian Noisy')

subplot(3,5,8)
imshow(FMEAN2)
title('Mean Filtered')

subplot(3,5,9)
imshow(FMED1)
title('Median Filtered')
```

```
subplot(3,5,10)
imshow(FWEIN3)
title('Weiner filtered')

subplot(3,5,11)
imshow(GC1)
title('512*512 Original')

subplot(3,5,12)
imshow(NGA3)
title('Gaussian Noisy')

subplot(3,5,13)
imshow(FMEAN3)
title(' Mean Filtered')

subplot(3,5,14)
imshow(FMED3)
title('Median Filtered')

subplot(3,5,15)
imshow(FWEIN3)
title('Weiner filtered')

% Effect of Salt and Pepper Noise
% loading original image (I)
I1=imread('C:\Users\Ali\Desktop\Research\Images\confused.jpg');
imshow (I1)

% Original resize 128*128
C1 = imresize(I1, [128 128]);
imshow (C1)
% Convert Original to Greay Scale
GC1=rgb2gray(C1);
imshow (GC1)

% add Salt & Pepper noise with 2%
NGA1 = imnoise(GC1,'salt & pepper',.02);
imshow (NGA1)



% filter image using mean filter & Display results
FMEAN1=filter2(fspecial('average',3),NGA1)/255;
imshow (FMEAN1)
% filter image using median filter & Display results
FMED1=medfilt2(NGA1);
imshow (FMED1)
% filter image using weiner filter & Display results
FWEIN1 = wiener2(NGA1,[5 5]);
imshow (FWEIN1)

% Original resize 256*256
C2 = imresize(I1, [256 256]);
imshow (C2)
% Convert Original to Greay Scale
GC2=rgb2gray(C2);
```

```
imshow (GC2)

% add Gaussian noise with 2%
NGA2 = imnoise(GC2,'salt & pepper',.02);
imshow (NGA2)


% filter image using mean filter & Display results
FMEAN2=filter2(fspecial('average',3),NGA2)/255;
imshow (FMEAN1)
% filter image using median filter & Display results
FMED2=medfilt2(NGA2);
imshow (FMED1)
% filter image using weiner filter & Display results
FWEIN2 = wiener2(NGA2,[5 5]);
imshow (FWEIN2)

% Original resize 512*512
C3 = imresize(I1, [512 512]);
imshow (C3)
% Convert Original to Greay Scale
GC3=rgb2gray(C3);
imshow (GC3)

NGA3 = imnoise(GC3,'salt & pepper',.02);
imshow (NGA3)


% filter image using mean filter & Display results
FMEAN3=filter2(fspecial('average',3),NGA3)/255;
imshow (FMEAN3)
% filter image using median filter & Display results
FMED3=medfilt2(NGA3);
imshow (FMED3)
% filter image using weiner filter & Display results
FWEIN3 = wiener2(NGA3,[5 5]);
imshow (FWEIN3)

subplot(3,5,1)
imshow(GC1)
title('128*128  ')

subplot(3,5,2)
imshow(NGA1)
title('Salt % Pepper Noisy ')

subplot(3,5,3)
imshow(FMEAN1)
title('Mean Filtered')

subplot(3,5,4)
imshow(FMED1)
title('Median Filtered')

subplot(3,5,5)
imshow(FWEIN1)
title('Weiner filtered')
```

```
subplot(3,5,6)
imshow(GC2)
title('256*256 ')

subplot(3,5,7)
imshow(NGA2)
title('Salt % Pepper Noisy')

subplot(3,5,8)
imshow(FMEAN2)
title('Mean Filtered')

subplot(3,5,9)
imshow(FMED1)
title('Median Filtered')

subplot(3,5,10)
imshow(FWEIN3)
title('Weiner filtered')

subplot(3,5,11)
imshow(GC3)
title('512*512 ')

subplot(3,5,12)
imshow(NGA3)
title('Salt % Pepper Noisy')

subplot(3,5,13)
imshow(FMEAN3)
title(' Mean Filtered')

subplot(3,5,14)
imshow(FMED3)
title('Median Filtered')

subplot(3,5,15)
imshow(FWEIN3)
title('Weiner filtered')

% Effect of Speckle Noise
% loading original image (I)
I1=imread('C:\Users\Ali\Desktop\Research\Images\confused.jpg');
imshow (I1)

% Original resize 128*128
C1 = imresize(I1, [128 128]);
imshow (C1)
% Convert Original to Greay Scale
GC1=rgb2gray(C1);
imshow (GC1)

% add Speckle noise
NGA1 = imnoise(GC1,'speckle');
imshow (NGA1)
```

```
% filter image using mean filter & Display results
FMEAN1=filter2(fspecial('average',3),NGA1)/255;
imshow (FMEAN1)
% filter image using median filter & Display results
FMED1=medfilt2(NGA1);
imshow (FMED1)
% filter image using weiner filter & Display results
FWEIN1 = wiener2(NGA1,[5 5]);
imshow (FWEIN1)

% Original resize 256*256
C2 = imresize(I1, [256 256]);
imshow (C2)
% Convert Original to Greay Scale
GC2=rgb2gray(C2);
imshow (GC2)

% add Spekle noise
NGA2 = imnoise(GC2,'speckle');
imshow (NGA2)


% filter image using mean filter & Display results
FMEAN2=filter2(fspecial('average',3),NGA2)/255;
imshow (FMEAN1)
% filter image using median filter & Display results
FMED2=medfilt2(NGA2);
imshow (FMED1)
% filter image using weiner filter & Display results
FWEIN2 = wiener2(NGA2,[5 5]);
imshow (FWEIN2)

% Original resize 512*512
C3 = imresize(I1, [512 512]);
imshow (C3)
% Convert Original to Greay Scale
GC3=rgb2gray(C3);
imshow (GC3)

NGA3 = imnoise(GC3,'speckle');
imshow (NGA3)


% filter image using mean filter & Display results
FMEAN3=filter2(fspecial('average',3),NGA3)/255;
imshow (FMEAN3)
% filter image using median filter & Display results
FMED3=medfilt2(NGA3);
imshow (FMED3)
% filter image using weiner filter & Display results
FWEIN3 = wiener2(NGA3,[5 5]);
imshow (FWEIN3)

subplot(3,5,1)
imshow(GC1)
```

```
title('128*128  ')

subplot(3,5,2)
imshow(NGA1)
title('Speckle Noisy ')

subplot(3,5,3)
imshow(FMEAN1)
title('Mean Filtered')

subplot(3,5,4)
imshow(FMED1)
title('Median Filtered')

subplot(3,5,5)
imshow(FWEIN1)
title('Weiner filtered')

subplot(3,5,6)
imshow(GC2)
title('256*256 ')

subplot(3,5,7)
imshow(NGA2)
title('Speckle Noisy')

subplot(3,5,8)
imshow(FMEAN2)
title('Mean Filtered')

subplot(3,5,9)
imshow(FMED1)
title('Median Filtered')

subplot(3,5,10)
imshow(FWEIN3)
title('Weiner filtered')

subplot(3,5,11)
imshow(GC3)
title('512*512 ')

subplot(3,5,12)
imshow(NGA3)
title('Speckle Noisy')

subplot(3,5,13)
imshow(FMEAN3)
title(' Mean Filtered')

subplot(3,5,14)
imshow(FMED3)
title('Median Filtered')

subplot(3,5,15)
imshow(FWEIN3)
title('Weiner filtered')
```

```
%Histogram Plot

subplot(5,4,1)
imshow(GC1)
title('Original' )
subplot(5,4,2)
imhist(GC1)
title('128 *128 Original')

subplot(5,4,3)
imhist(GC2)
title('256 *256 Original ')


subplot(5,4,4)
imhist(GC3)
title('512 *512 Original ')


subplot(5,4,5)
imshow(NGA1)
title ('Gaussian Noisy')

subplot(5,4,6)
imhist(NGA1)


subplot(5,4,7)
imhist(NGA2)



subplot(5,4,8)
imhist(NGA3)


subplot(5,4,9)
imshow(FMEAN1)
title('Mean Filtered')

subplot(5,4,10)

imhist(FMEAN1)

subplot(5,4,11)

imhist(FMEAN2)



subplot(5,4,12)

imhist(FMEAN3)
```

```
subplot(5,4,13)
imshow(FMED1)

title('Median Filtered')

subplot(5,4,14)
imhist(FMED1)

subplot(5,4,15)
imhist(FMED2)

subplot(5,4,16)
imhist(FMED3)

subplot(5,4,17)
imshow(FWEIN1)

title('Weiner Filtered')

subplot(5,4,18)
imhist(FWEIN1)

subplot(5,4,19)
imhist(FWEIN2)

subplot(5,4,20)
imhist(FWEIN3)
```