Image Noise Removal using Dual Domain Approach

Developed by

C. Maher Aziz

(57-CS/MS/01)

Supervised by

Muhammad Nadeem

Department of Computer Science

Faculty of Basic & Applied Sciences

International Islamic University, Islamabad

(2008)

i Image processing – Digital techniques

ii Computer graphics

بسم الله الرحمن الرحيم

أَلَمْ نَشْرَحْ لَكَ صَدْرَكَ

وَوَضَعْنَا عَنكَ وِزْرَكَ

الَّذِي أَنقَضَ ظَهْرَكَ

وَرَفَعْنَا لَكَ ذِكْرَكَ

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا

إِنَّ مَعَ الْعُسْرِ يُسْرًا

فَإِذَا فَرَغْتَ فَانصَبْ

وَإِلَى رَبِّكَ فَارْغَبْ

صدق الله العظيم

Department of Computer Science

International Islamic University Islamabad

Dated: 03-05-2008

Final Approval

It is certified that we have read the thesis submitted by Ch. Maher Aziz, Registration number 57-CS/MS/01, and it is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the MS degree in Computer Science.

Committee:

External Examiner:

Dr. Abdul Jalil
Associate Professor / Principal Scientist
Department of Computer & IT,
PIEAS, Islamabad.

Internal Examiner:

Mr. Asim Munir
Assistant Professor
Department of Computer Science,
Faculty of Applied Sciences,
International Islamic University,
Islamabad.

Supervisor:

Mr. Muhammad Nadeem,
Assistant Professor,
Department of Computer Science,
Faculty of Applied Sciences,
International Islamic University,
Islamabad.

A dissertation submitted to the
Department of Computer Science,
Faculty of Applied Sciences,
International Islamic University, Islamabad
as a partial fulfillment of the requirements
for the award of the degree of
MS in Computer Science

This project is lovingly dedicated to

To My Supervisors

Without their help and prayers its implementation would have never been possible

Declaration

I, Ch. Maher Aziz S/O Ch. Aziz Ahmed, hereby declare and affirm that this software neither as a whole nor as a part thereof has been copied out from any source. It is further declared that I have developed this software and accompanied thesis entirely on the basis of my personal efforts, made under the sincere guidance of my teachers. If any part of this project is proven to be copied out or found to be a reproduction of some other, I shall stand by the consequences.

No portion of the work presented in this thesis has been submitted in support of an application for other degree or qualification of this or any other University or Institute of Learning.

Ch. Maher Aziz
57-CS/MS/01

Acknowledgement

This project and all my incessant efforts are of worth only by the grace of Allah Almighty, the Exalted, the Most Merciful and the Beneficial Who gave me strength beyond my capabilities to complete this task to the best of my knowledge and abilities.

I would like to take this opportunity to thank my supervisor **Mr. Muhammad Nadeem,** who imparted all his knowledge, guidance and support to boost my confidence and learning. He was always there to solve problems and helped me through this project. Being highly supportive during the research, I am thankful to him for his commendable cooperation.

Last but not the least; I would like to acknowledge **Mr. Asim Munir,** who has been the source of motivation for this project. I would like to admit that I owe all my achievements to my truly, sincere and most loving parents and friends who mean the most to me, and whose prayers have always been a source of persistence and determination towards achievement.

Ch. Maher Aziz

Project In Brief

Project Title:	Image Noise Removal using Dual Domain Approach
Undertaken By:	Ch. Maher Aziz Reg. No. 57-CS/MS/01
Supervised By:	Assistant Professor Muhammad Nadeem DCS, FBAS, IIUI
Starting Date:	Sept, 2007
Completion Date:	March, 2008
Tool Used:	MatLab 7
Operating System:	Microsoft Windows XP Professional
System Used:	Pentium IV

Abstract

Image Noise Removal using Dual Domain Approach is a research based project with the final objective of having an improved and better image result with less noise and blurring artifacts by utilizing dual domain approach i.e., harnessing the advantages from both spatial and frequency domain where local features are best preserved in spatial domain while global features are captured well in frequency domain.

The vitality and the crucial importance of image noise removal is equally appreciated in all fields. Images taken with both digital cameras and conventional film cameras will pick up noise from a variety of sources. Many further uses of these images require that the noise will be (partially) removed - for aesthetic purposes as in artistic work or marketing, or for practical purposes such as computer vision. Thus, its importance is widely recognized in the broad spectrum of applications.

The thesis contains four phases i.e, adding three kinds of noise in the original image, filtering using spatial domain, filtering through frequency domain and filtering utilizing dual domain approach.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1. Introduction

Noise corruption occurs in many scientific datasets; possibly because of the data acquisition process, or because of the other natural phenomena. The first pre-processing step in analyzing such datasets is denoising, that is, estimating the unknown signal of interest which is our target from the available noisy data. Contamination or unwanted pattern in the signal being broadcasted is a common problem in every field of signal processing and as we know that image is also a two-dimensional signal so here noise is also a critical issue and hot topic of research. [13]

Majority of the types of noise inclusive of scratches and dust particles are normally present in scanned material for instance film or photography. With the advent of digital media, revolutionary changes have come into play: digital image is not contaminated with dust and scratches. However, it has its own shortcomings and types of noises it suffers with for example: Salt and Pepper noise in poor lightening situations and Hot Pixels in comparatively long expositions of lens. [16]

As the scratches are more obvious on digital and higher definition television, so they must be removed specifically present on old films. Although a time-consuming task, but can be dealt with by the following image restoration process: known are the locations of noisy pixels to be restored and a prototype image, restoration of those noisy pixels in a natural way. This should be kept in mind that it is just a one instance. There are other types of noises which might not be dealt with using this technique emphasizing the need of having some generalized technique. [13]

1.1 Image Noise Removal

In order for material to be broadcasted or exhibited to the audience, it is desirable to remove unwanted pattern of corruption from the original material prior to broadcast. Although, there are several different approaches to denoise images and more and better techniques are being devised to remove the noise in order to have a better quality of the image but it should be kept in mind that noise can never be completely removed from the images. [13]

Existing work on image noise removal can be divided into intra-frame and inter-frame techniques reflecting the source of data needed for noise removal. In case of Inter-frame algorithms, pixels needed are copied from previous or upcoming frames resulting in compensation for motion at times of camera or even object by tracking major points of an image. Inter-frame methods fail to serve the purpose when scratches are omnipresent in consecutive series of many frames (possible reason for these scratches could be the vertical motion of the film through projectors) or when camera activity is at its peak. In both of these situations, the pixels needed are hard to find in the previous or in the upcoming frames. [13] So here now that we understand that inter-frames algorithm has its limitation which encourages our research for a newer technique to overcome this limitation. This paper describes an intra-frame algorithm under the assumption that the other frames do not contain the pixels needed at all; thus reflecting the research on both inter-frame in the sense and most definitely on intra-frame. The classification of previous intra-frame methods, based on the type of information utilized, is as follows. [13]

1. Using frequency domain information only (e.g. low pass filter).

2. Using spatial domain information only

(a) Median and similar order statistics filters.

(b) Spatial statistical texture synthesis.

(c) Cloning by copying pixels.

3. Use spatial and frequency domain information

(a) Projections onto convex sets for band limited images.

(b) Spatial and frequency based statistical texture synthesis.

1.2 Frequency Domain Only

Frequency domain algorithms of which low pass filtering is an instance is good in capturing global structure of the image but when in it comes to local control; it loses line continuity and sharpness. This result in the blurring of lines and other details. Human visual system being sensitive to details within the image especially sharp transitions, the results of noise removal are unacceptable of an image consisting of numerous contiguous noisy pixels. I intend not to show results of low pass filtering over here because of the poor performance of these kinds of filters in the case of the types of images and noise where there are textured areas or textured areas with prominent lines along with numerous contiguous noisy pixels as shown in Figure 1.1 below. [13]

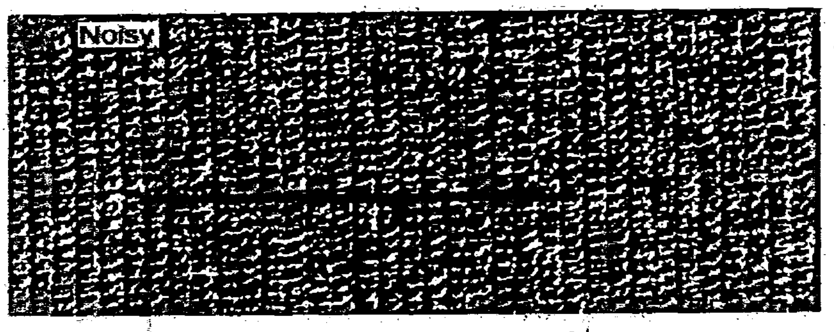


Figure 1.1: Contiguous Noisy Pixels in Textured Area

1.3 Spatial Domain Only

A problem ubiquitous in all spatial domain only methods is that they have local control and information but they ignore the global structure of the image and have no parameters to bind the global structure. Limiting the calculations to only local neighborhood has its computationally practical reasons behind it in some cases. Furthermore, methods like median filtering have an inherent inability of utilizing global information purposefully. [13]

Although in popular commercial image manipulation programs, cloning tools allow copying using paint brush strokes from another area of the image, but aligning reconstructed lines with the existing one is a very time consuming and error prone method in this way. A comparatively bigger problem that is presented during this course is of non-uniform illumination. In such instances finding uniform intensity source is quite big of a problem. [13]

These problems are demonstrated in conditions with inappropriate lightening and bad weather conditions affecting the balance of intensity across the image and the problems arise with copying from one area of the image to another area of the image. The shading as well as the alignment can be different between source and destination areas. A careful use of a manual cloning tool can mitigate the alignment problem up to some extent. Without the utilization of the frequency domain information, this kind of a tool would not do much about shading mismatch. [13]

A problem occurs with order statistic filter in the case where not enough accurate information is available for an order statistic to work purposefully, a case such as several contiguous noisy pixels. [13]

Spatial domain texture synthesis algorithms have shown outstanding results in the case of stochastic type or small regular texture, but the methods are faced with failure when the long-range structure image, such as the image of brick wall shown below in Figure 1.2, is presented to them. Due to the use of second order statistics, the computational cost shoots up excessively for long range image structure. [13]

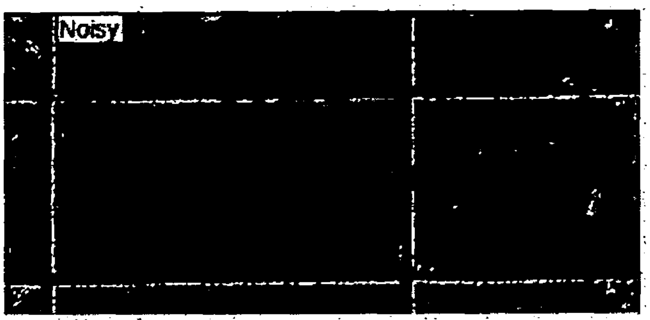


Figure 1.2: Image of brick wall

1.4 Spatial and Frequency Domains

When it comes to texture synthesis, utilization of multi-resolution directional filters and then working solely in single order statistics, such as histograms, is a possibility. This is considered a spatial and frequency domain algorithm. However, this way is functional only in the case of stochastic texture or small regular texture. Moreover, it is not a noise removal algorithm but is contrived for the purpose of generating large texture areas from sample images. It is not possible to utilize it to generate pixels maintaining the continuity of prominent lines crossing noisy pixels and at the same time retaining the noise free pixels. [13] Gerchberg-Papoulis and other related algorithms are POCS (Projection onto Convex Sets) based algorithms that utilizing both frequency and spatial domain information. [11] Nevertheless, they operate solely in the case of band-limited images with the restraint of known band limits. Furthermore, recently, extensions like "Concealment of Damaged Block Transform coded Images Using Projections onto Convex Sets," demand costly computation of lines intersecting noise pixels. In other contemporary work "Errorless Restoration Algorithms for Band-limited Images," derives a way of reduction of band-limited interpolation and extrapolation problems for finite-dimensional signals down to solution of a set of linear equations. Additionally, it has been proved that the corresponding matrix is positive-definite having a spectral radius less than 1. [13] One other method based on matrix is "On Discrete Band-Limited Signal Extrapolation." These all methods work on the restraint that the image be band limited along with the known limits. [13]

Recent work has been done in combining frequency and spatial domain for fast interactive image noise removal which is based on the framework of POCS (Projection onto convex sets) and has the property of handling images without placing the limitation of band limit and that the limits be known. This particular technique uses iterative algorithm to achieve this level of image handling while using sub image acting as a sample to restore the distorted part of the image. [13]

1.5 Literature Survey

The papers described in the literature survey are as follows:

i. **A New Algorithm for Image Noise Reduction Using Mathematical Morphology** by Richard Alan Peters II, Department of Electrical Engineering, Vanderbilt University, Nashville, TN 37235 IEEE Transactions on Image Processing Volume 4, Number 3, pp. 554-568, May 1995. [12]

ii. **Combining Frequency and Spatial Domain Information for Fast Interactive Image Noise Removal** by Anil N. Hirani, and Takashi Totsuka, Proc. Siggraph, 1996, Sony Corporation, Tokyo, Japan, Publisher: ACM New York, NY, USA. [13]

iii. **Fourth-Order Partial Differential Equations for Noise Removal** by Yu-Li You, Member, IEEE, and M. Kaveh, Fellow, IEEE, published in IEEE Transactions on Image Processing, Vol 9, and NO. 10, October 2000. [14]

iv. **Noise Removal for Degraded Image with Impulsive Noise Using Median Filters and Neural Filters** by Yamashita Noritaka cd, Lu J, Sekiya Hiroo, and Yahagi Takashi, Transactions of the Institute of Electrical Engineers of Japan. C, Volume. 123, No. 6, 2003, Japan. [15]

i. A New Algorithm for Image Noise Reduction Using Mathematical Morphology

According to the author, Richard Alan Peters, his paper is a description and analysis of a new morphological image cleaning algorithm (MIC) that preserves thin features while removing noise. The technique and the idea that has been used is the mathematical morphological algorithm which is as follows:

The basic idea employed in the morphological image-cleaning algorithm is the segmentation into features and noise, the residual image which we get by subtracting an original image from a smoothed version. Then the features that we get from the residual are added back into the smoothed image. This all procedure results in an image with edges and other one dimensional features to become sharp the same way as original has while still retaining smoother regions between them. [12]

Limitation: MIC smoothes the image in number of size-bands, subtracts these bands out of the image to create residual images, segments the residuals into features and noise, and adds the features back to the smoothed image. But this algorithm is best employed in removing noise and scanner artifacts in images where the standard deviation of the noise is less than half the standard deviation of the features. In the other cases it fails to serve the purpose. [12]

ii. Combining Frequency and Spatial Domain Information for Fast Interactive Image Noise Removal

In this paper, the writers namely, Anil N. Hirani and Takashi Totsuka have claimed their work as one of its kind and they have devised the technique for the first time which utilizes both the frequency domain and the spatial domain to obtain better results without placing band limits on the image. [13]

The algorithm technique for harnessing both domains is explained in figure 1.3 as follows:

First, consider these two parameters

r_0 = (initial repair subimage) X (noise mask) (1.1)

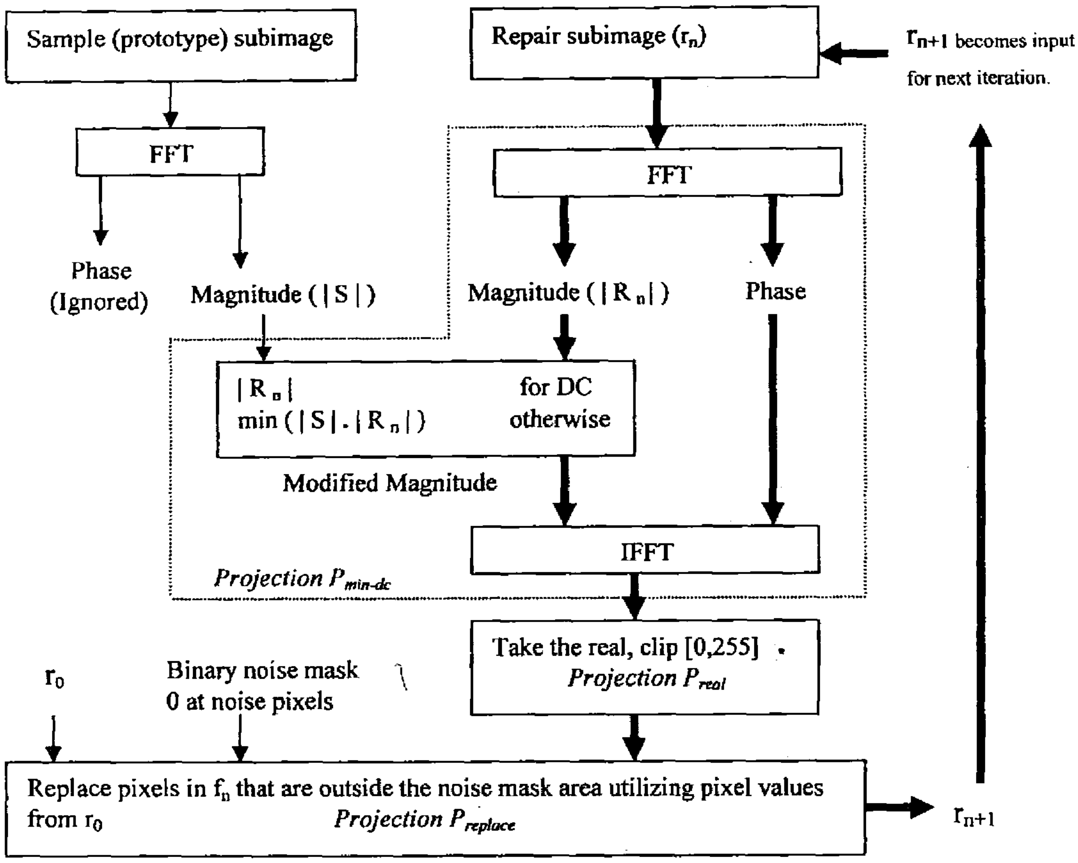
$r_{n+1} = P_{rep} P_{real} P_{min\text{-}dc}\, r_n$ (1.2)

Now,

Start at top with $n = 0$ and

Where r_0 is noisy repair subimage multiplied by binary noise mask

P_{rep}, P_{real}, $P_{min\text{-}dc}$, r_n and r_{n+1} are explained in the figure 1.3 below. [13]



Note: Thicker lines indicate main data flow.

Figure 1.3: Algorithm for utilizing dual domain approach

In current implementation of the algorithm, the user sets the number of iterations. It is easy to implement other termination area. [13]

Limitation: The shortcoming of this algorithm is that it only addresses a specific type of noise and on top of that it needs a sample sub-image prototype without which it would not work. It is mentioned in detail in problem domain as I have chosen to work on the same concept of utilizing dual domain approach for image noise removal. [13]

iii. Fourth-Order Partial Differential Equations for Noise Removal

The writer, Yu-Li You and M. Kaveh, have proposed a class of fourth-order partial differential equation (PDEs) which optimizes the trade-off between noise removal and edge preservation. The proposed PDEs tend to leave the processed images with isolated black and white speckles which may be characterized as pixels whose intensity values are either much larger or smaller than those of their neighboring pixels. [14]

The propose technique is described as follows:

First, the following function is considered which is defined in the space of continuous images over a support of Ω in equation 1.3 as follows:

$$E(u) = \int_{\Omega} f(|\nabla^2 u|)dxdy \quad (1.3)$$

Where nabla square denotes Laplacian operator. We require that the function $f(.) \geq 0$ and an increase function $f'(\cdot)>0$. So that the function is an increasing function with respect to the smoothness of the image as measured by absolute of the product of nabla square and u. Therefore, the minimization of the function is equal to the smoothing of the image. [14]

Since the Laplacian of the image intensity function is very large around speckle pixels and the function f (.) is designed such that it decreases rapidly in order to preserve edges, the function g (.) defined in will have small values around speckle pixels. The proposed fourth-order PDE, however, evolves an observed image toward a piecewise planar image which, they believe, is a better approximation to natural images. [14]

Limitation: The problem is the visibility of the speckles in the proposed PDE because piecewise planar images have less masking capability than step images and anisotropic diffusion tends to generate multiple false edges. If the proposed PDE is compared with the anisotropic diffusion (first order PDE) than it is better as far as despeckle is concerned but anisotropic diffusion creates blocky effect which gives an effect of artificiality in the images. [14]

iv. Noise Removal for Degraded Image with Impulsive Noise Using Median Filters and Neural Filters

According to the writers of this research paper, namely Noritaka Yamashita, Jianming Lu, Hiroo Sekiya, and Takashi Yahagi, nonlinear filtering, a neural filter was proposed for reducing Gaussian noise. However, a neural filter was not applied to image with impulse noise. In this paper, they have proposed a filter for removing impulse noise, Gaussian noise and mixed impulse and Gaussian noise using median filters and neural filters. [15]

The technique employed is as follows:

Median filters determine whether processed pixel is impulse noise. If processed pixel is judged impulse noise, median filters replace processed pixel with median. The method achieves removal of impulse noise and detail preservation. System configuration is shown below in the Figure 1.4. [15]

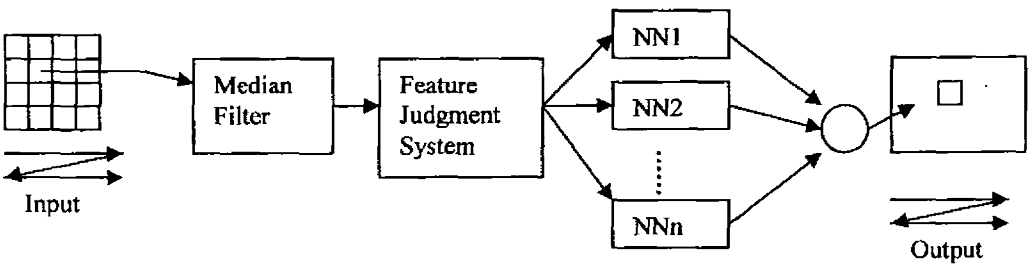


Figure 1.4: System Configuration of Median and Neural Filters

Limitation: Carrying out the simulation, they illustrated the performance of the proposed filter as being better than what other filters have achieved in the past. But the serious issue involved with neural filters is that it needed training for neural network to work and even when it was trained, the functioning of the proposed filter was a very time consuming procedure. [15]

1.6 Problem Formulation

A fast iterative algorithm for image noise removal has been built earlier. While most existing algorithms have worked solely in spatial or frequency domain, the algorithm mentioned earlier worked in both domains, making it possible to fully exploit the advantages from each domain. The shortcoming that has been the source of motivation for writing the paper is that the dual domain approach utilized here only removes specific type of noise; moreover, the algorithms works and removes only on an assumption that it would restore the image noisy part by finding a suitable sample sub-image prototype within the image and then using that prototype it would restore the image naturally. The algorithm fails to serve the purpose when a suitable sub-image prototype could not be located within the image or a different sort of noise is presented. The Figure 1.5 given below demonstrates the specific noise, noisy contiguous pixels, present which is like an object and the appropriate sub masks are selected to reconstruct the noisy image naturally. [13]

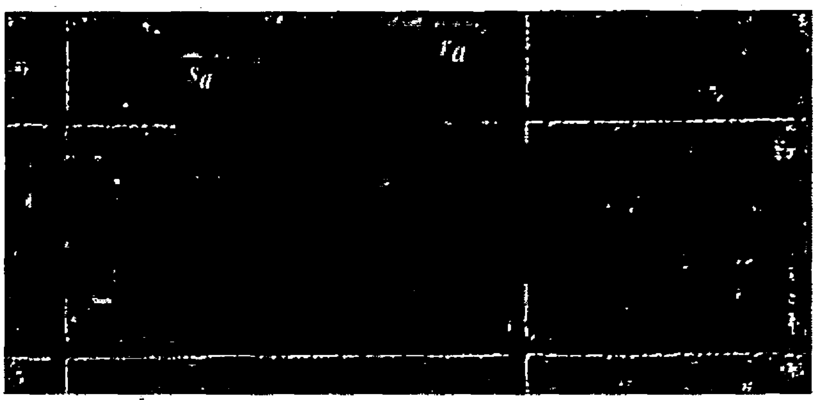


Figure 1.5: Contiguous Noisy pixels with sub masks located for restoration within the image

This image originally looked like Figure 1.6:

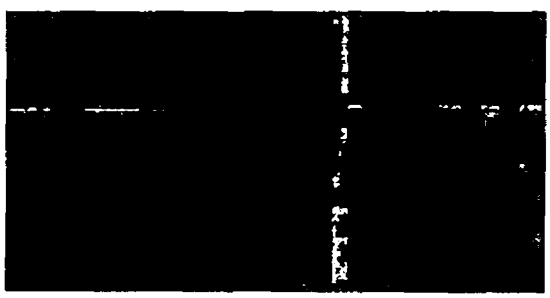


Figure 1.6: Original Image with noise

But then a Binary noise mask was passed to the algorithm to paint mask on noise, as shown in Figure 1.7, in order to later on reconstruct the whole image naturally by selecting image masks as shown above. [13]

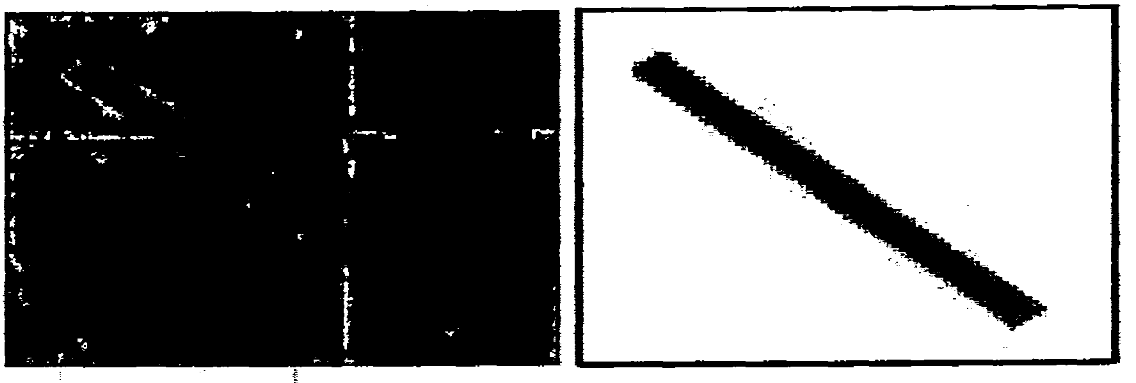


Figure 1.7: Painted Mask on Image and Binary Noise Mask

1.7 Objective of the Project

This research project aims at developing a Noise Removal System. The objective is to process and analyze Noisy Images to obtain the unprecedented results with noise removed along with the minimum amount of blurring by harnessing the Dual Domain approach. The theme of the project is to design and develop new and advanced algorithms and techniques to tackle three types of noise problems occurring during image acquisition and other electrical or electromechanical interferences. The focus of analysis will be to provide with the optimum results for further analysis whether be it for aesthetic purposes as in artistic and marketing work, or for practical purposes such as in computer vision.

1.8 Scope and Plan of the Project

Most existing algorithms for removing image noise use either frequency domain information (e.g. low pass filtering) or spatial domain information (e.g. median filtering or stochastic texture generation). The propose algorithm works in both spatial and frequency domains, making it possible to fully exploit advantages from each domain. While global features and large textures are captured in frequency domain, local continuity and sharpness are maintained in spatial domain. With a careful choice of operations and domains in which they work, the proposed dual-domain approach attempts to restore the image to serve the purpose. The propose system is harnessing characteristics of dual domain approach for creating three types of noise removal solution.

The block diagram of the propose solution is shown in the following Figure 1.8:

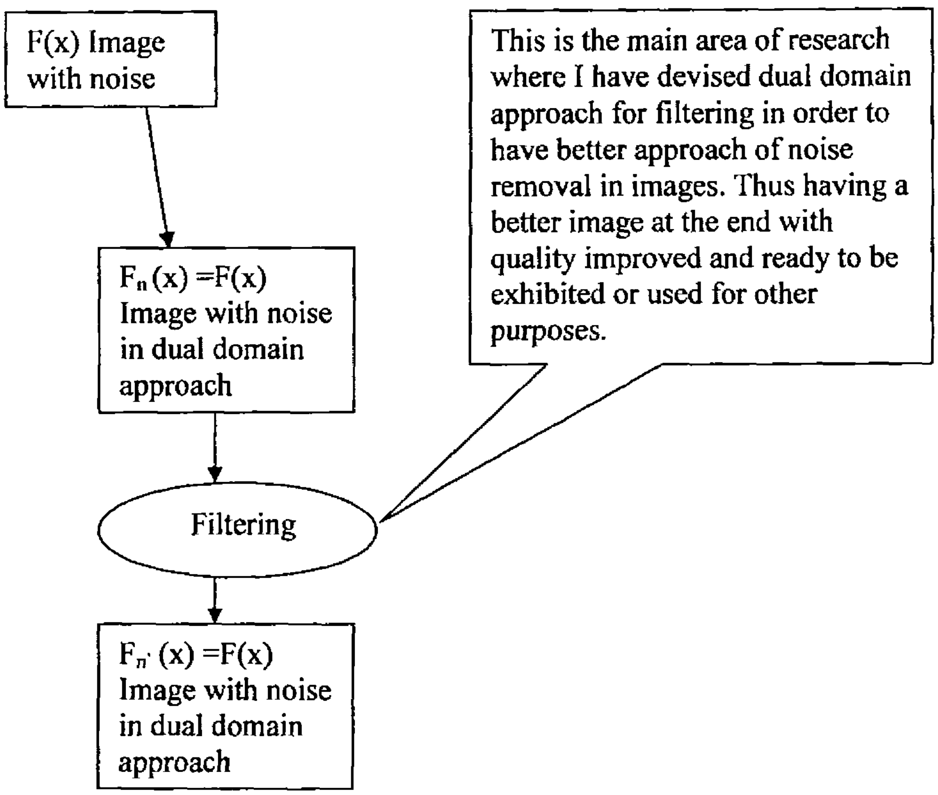


Figure 1.8: Block Diagram of the Propose Solution

CHAPTER 2

OVERVIEW OF THE TECHNIQUES

2. Overview of the Techniques

The approaches and techniques utilized have their brief descriptions given below.

2.1 Three Important Types of Noises

There are three important types of noises which have been treated in the algorithm. They are explained as follows:

2.1.1 Salt and Pepper Noise

In salt and pepper noise (also known as random noise or independent noise), pixels in the image are vastly different in color from their surrounding pixels. The discerning feature is that the color of a noisy pixel exhibits no relation to the color of surrounding pixels. Usually this kind of noise distorts only a small number of image pixels. On viewing, dark and white dots are perceived in the image, reflecting term salt and pepper noise as it is evident in the name as white like salt and black like pepper. Typically, flecks of dust on the lens or inside the camera, or with digital cameras, faulty CCD elements are all elements that contribute towards the sources of this kind of noise. [17]

2.1.2 Gaussian Noise

In Gaussian noise (dependent noise), an amount of noise is added to every part of the picture. Every pixel in the image changes in the case of Gaussian from its original value by a little intensity normally unless specified. When we take a plot of the amount of distortion of a pixel against the frequency with which it occurs produces a Gaussian distribution of noise which basically shows a normal distribution. [17]

2.1.3 Speckle Noise

Speckle noise is present innately in coherent imaging, including ultrasound imaging. Thus it is exhibited as a random, deterministic, interference pattern present in an image and is formed with radiation of a medium, a coherent radiation indeed, embracing many sub-resolution scatterers. The underlying structure does not always essentially correspond to the texture of the observed speckle pattern. The brightness which is only local, belonging to speckle pattern, does reflect the local echogenicity of the underlying scatterers. [18]

2.2 Details of Noise Removal in Spatial Domain

These two techniques are utilized for spatial domain filtering.

2.2.1 Gaussian filter

One method of removing noise is by convolving the noisy image with a mask. The Gaussian mask consists of elements projected by a Gaussian function which is given below in equation (2.1): [1]

$$G(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-x_0)^2}{2\sigma^2}} \tag{2.1}$$

It results in an image of blurred appearance as the standard deviation of the mask goes high, and has the effect of diluting the value of a single pixel over an area of the image. This eventually brings the pixel values closer to its neighbors. When compared Gaussian filtering works well, but the blurred of edges can be a problem, particularly if the output is being fed into edge detection algorithms for computer vision applications. [1]

2.2.2 Averaging filter

The method of removing noise used in this filter is by convolving the noisy image with averaging filter mask. Each pixel in the mask is set to the average value of itself and its nearby neighbors in Averaging filter. Given below in the figure 2.1 are two 3 X 3 averaging filter masks: [1]

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \qquad \frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Figure 2.1: Example of Butterworth Low Pass filtering

Averaging tends to blur an image, because pixel intensity values which are significantly higher or lower than the surrounding neighbourhood would "smudge" across the area. [1]

A special case of Gaussian filtering is averaging in a sense, where the function that defines the mask values possesses an infinite standard deviation. [1]

Averaging, similarly like Gaussian filtering, is an effective noise removal technique against Gaussian noise as the distributions of deviations are normal, and work good to restore the image near to the original value. [1]

2.3 Details of Noise Removal in Frequency Domain

Frequency domain filtering is a common technique utilized in the field of image and signal processing. According to the filters applied it can be use to achieve the effect of smoothening, sharpening, de-blurring, and restoring images. [19]

Frequency domain filtering is usually done in three basic steps as follows:

1. Firstly, the transformation of the image from the spatial domain into the frequency domain. [19]
2. Secondly, the result in step 1 is multiplied by a chosen filter (according to the need) in frequency domain. [19]
3. Thirdly and lastly, the result in step 2 is subjected to transformation back into the spatial domain for display and analysis purposes in order to visualize the effect of filter in frequency domain. [19]

2.3.1 Frequency domain methods

In frequency domain, Image enhancement is a straightforward process. We simply perform above three-step mentioned procedure to arrive at the very results. [20]

The concept of smoothing an image by diluting the effect of high frequency components or sharpening the edges by boosting the high frequency contents is intuitively easy to comprehend. However, computationally, it is often more efficient to implement these operations as convolutions by small spatial filters in the spatial domain. Understanding frequency domain concepts is important, and leads to enhancement techniques that might not have been thought of by restricting attention to the spatial domain. [20]

2.3.2 Filtering

Low pass filtering involves the elimination of the high frequency components in the image. It results in blurring of the image (and thus a reduction in sharp transitions associated with noise). An ideal low pass filter would retain all the low frequency components, and eliminate all the high frequency components. However, ideal filters suffer from two problems: *blurring* and *ringing*. These problems are caused by the shape of the associated spatial domain filter, which has a large number of undulations. Smoother transitions in the frequency domain filter, such as the Butterworth filter, achieve much better results. [20]

In the Figure 2.2 below is shown an example of Butterworth low pass filtering in frequency domain. [1]

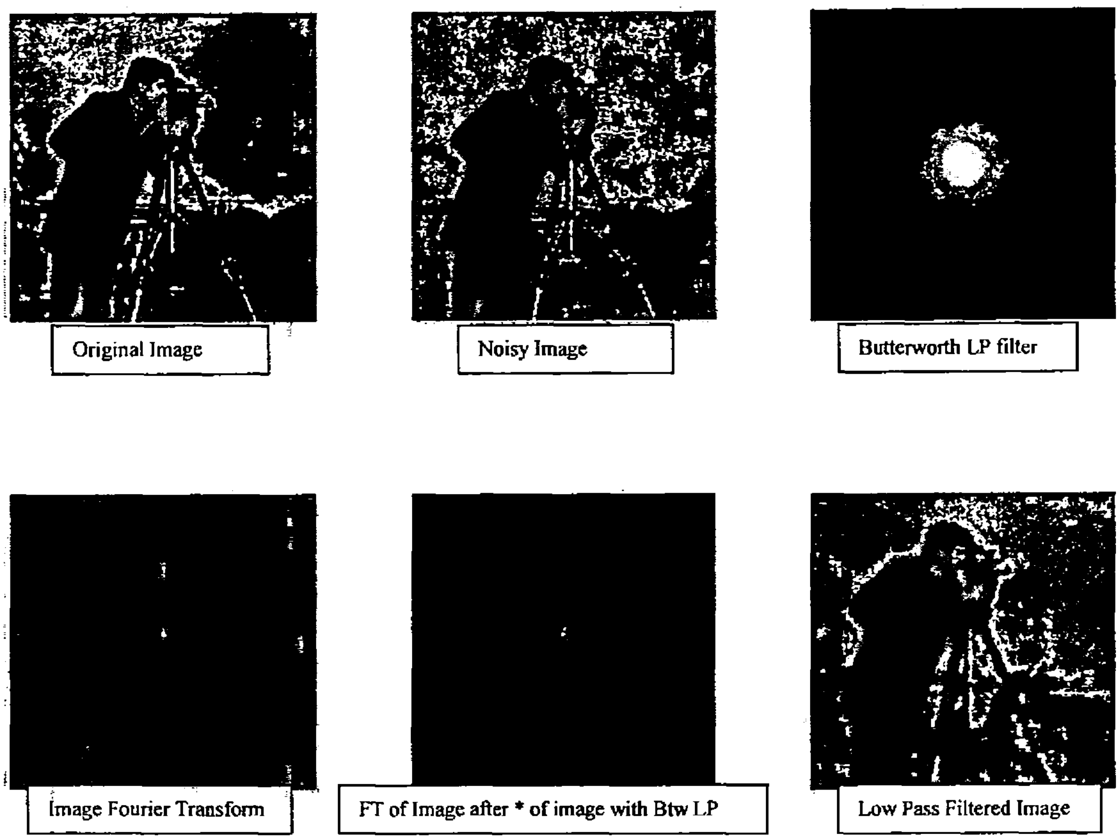


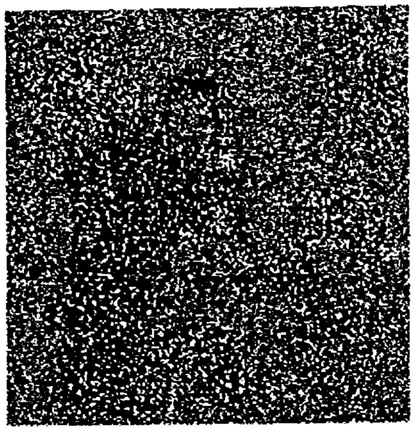
Figure 2.2: Example of Butterworth Low Pass filtering

2.4 Noise Removal using Dual Domain Approach

As this technique is a novel and its details are mentioned in implementation section, so over here only brief discussion has been done mentioning the other techniques that have been utilized in order to implement the dual domain approach. High frequency Emphasis and Butterworth low pass filter in frequency domain along with adaptive median, median and geometric mean in spatial domain have been utilized to implement the dual domain approach.

2.4.1 Adaptive Median Filter and Standard Median Filter [1]

- Along the boundaries, Impulse noise is removed and distortion is minimized.
- Variable window size to improve efficiency
- Salt and Pepper noise addressed
- Non-impulsive noise being smoothed

Original Image Image with Salt and Pepper Noise


Result by Adaptive Median Filter Result by Standard Median filter

Figure 2.3: Images filtered using Median and Adaptive Median filter.

2.4.2 Contrast Stretching

Contrast stretching, a technique where an image is stretched between any two values i.e., between an interval. One of them is the maximum value and the other is the minimum value within in which the new interval would be in. The formula for contrast stretching is as follows in the equation (2.2): [1]

$$F(x,y) = [(Rmax - Rmin) / (Gmax - Gmin)] * (G(x,y) - Gmin) + Rmin \qquad (2.2)$$

Where Rmax and Rmin are the desired interval range of gray scale and Gmax and Gmin are the existing gray scale range of the input image. G(x,y) is the pixel value at the location (x,y). [1]

2.4.3 Image Enhancement using High Frequency Emphasis

At times, it is customary to sharpen the edges of an image in order to enhance and make ir appealing to the senses. In this project, images are enhanced by using High Frequency Emphasis (HFE) Filtering. It is beneficial advantageous to emphasize the contribution of enhancement fashioned by the high frequency components of an image. In this technique, a constant multiplied by a high pass filter function is to give boost as much desired to high frequency contents and then add an offset added so that zero frequency term is not eliminated by the filter meaning the low frequency contents are also present. The filter is given as follows by the following function: [6]

$$H_{hfe}(u,v) = a + bH_{hp}(u,v) \qquad \text{where } a \geq 0 \text{ and } b > a \qquad (2.3)$$

where H_{hfe} is the High Frequency Emphasis and H_{hp} is the High Pass Frequency filter.[6]

The project work in five different phases as described below:

2.5 Adding noise

It is then sub divided into three stages:

i. *Adding Salt and Pepper Noise*

"Salt and Pepper" noise is added to the image, which affects approximately D*numel (I) pixels where I is the image and D is the noise density, 0.05. [6]

PDF of Salt and Pepper noise is mentioned below: [6]

$$p_z(z) = \begin{Bmatrix} P_a & \text{for} \quad z = a \\ P_b & \text{for} \quad z = b \\ 0 & \text{otherwise} \end{Bmatrix} \qquad (2.4)$$

$$b > a$$

ii. *Adding Gaussian Noise*

Gaussian white noise is added according to the following PDF in the equation (2.5) to the image. [6]

$$p_z(z) = \frac{1}{\sqrt{2\pi b^2}} e^{-\frac{(z-u)^2}{2b^2}} \qquad -\infty < z < \infty \tag{2.5}$$

Where mean = a = 0 and variance = $\sigma^2 = 0.01$. [6]

iii. *Adding Speckle Noise*

Multiplicative noise to the image is added using the PDF in the equation (2.6):

$$J = I + n^*I \tag{2.6}$$

Where n is uniformly distributed random noise with mean 0 and variance V. The default for V is 0.04. [6]

2.6 Filtering in Spatial Domain

It is further sub divided into three steps:

i. *Filtering noisy image corrupted with Salt and Pepper Noise*

The image is filtered using Gaussian and Average filter in the spatial domain.

ii. *Filtering noisy image corrupted with Gaussian Noise*

The same is done with this image i.e., filtered using Gaussian and Average filter.

iii. *Filtering noisy image corrupted with Speckle Noise*

The same kind of filtering is done and the results are displayed for all of these three images.

2.7 Filtering in Frequency Domain

It is further sub divided into three steps:

i. *Filtering noisy image corrupted with Salt and Pepper Noise*

The image is filtered using Gaussian and Butterworth filter in the frequency domain.

ii. *Filtering noisy image corrupted with Gaussian Noise*

The same is done with this image as well.

iii. *Filtering noisy image corrupted with Speckle Noise*

The same kind of filtering is done in this case and then the results are displayed for all of these three images.

2.8 Filtering using Dual Domain Approach

In this final phase, firstly, images are filtered using dual domain approach and then compared mathematically in order to perform comprehensive analysis, mathematical as well as visual.

It is sub divided into five steps:

i. *Filtering noisy image corrupted with Salt and Pepper Noise*

   The image is filtered and the results are stored for further analysis into dual domain approach.

ii. *Filtering noisy image corrupted with Gaussian Noise*

   The image is filtered using Dual Domain approach.

iii. *Filtering noisy image corrupted with Speckle Noise*

   The dual domain approach is used for filtering and the results are displayed. The enhanced image will be segmented to separate veins from the background.

iv. *Mathematical and Visual Analysis*

   At the end, all images are visually compared and RMSE value for every result through dual domain and other domains is calculated for comparison in order to ensure that the one that is best looking image is also mathematically the best one.

CHAPTER 3

REQUIREMENTS ANALYSIS

3. Requirements Analysis

The requirement analysis is the first step towards developing software. Analysis must be performed in a systematic and correct manner so as to have as few mistakes as possible in the software and to have an end product completely fulfilling the expectations of the client. The reliability and the robustness of the software are highly dependent on the fact that the analysis is carried out properly. The main objective of this phase is to identify all possible requirements and expectations kept of software. In requirement analysis, problems are identified and then a possible solution is proposed. [8]

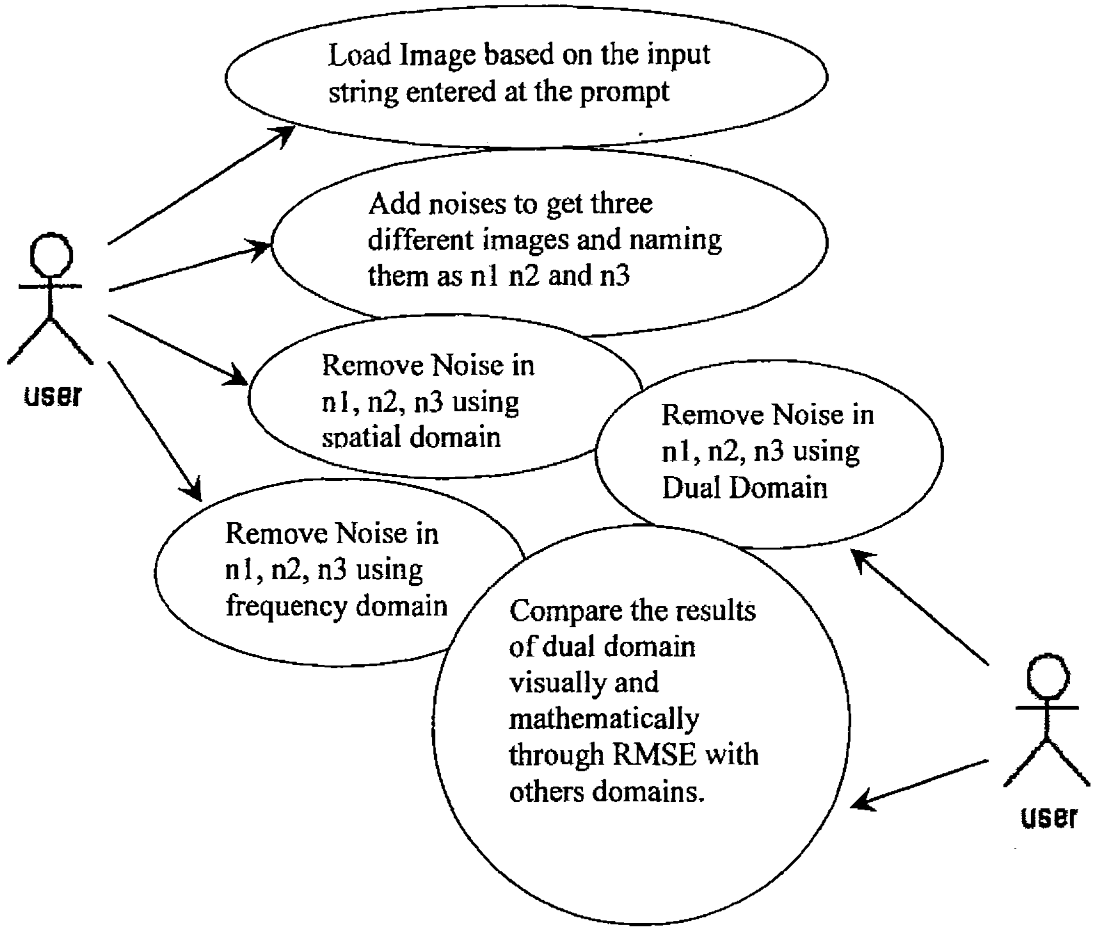
3.1 Problem Analysis

The report reveals the functional requirements of the system as given below:

- Loading the image.
- Adding noise according to the parameters chosen to have noisy images at hand.
- Removing noise using spatial domain filters, Average and Gaussian filters.
- Removing noise using frequency domain filters, Gaussian Low pass filter and Butterworth low pass filter.
- Requiring user at every step to input parameters or presenting with the choice to use default parameters.
- Finally, removing noise from noisy image using dual domain approach.
- Compare the results visually by displaying them together and mathematically by calculating the RMSE values to both visually and mathematically ensure the image quality.

3.2 Use Case Analysis

Analysis of the project is presented in terms of use case diagrams indicating the actors and use cases in expanded format. This helps to visualize the work and indicating the system boundaries while presenting the functionalities. The Use Case Model describes the proposed functionality of the new system. [8]



n1: the image corrupted by Salt & Pepper noise
n2: the image corrupted by Gaussian noise
n3: the image corrupted by Speckle noise

Figure 3.1: Use Case Diagram of Image Noise Removal using Dual Domain Approach.

Use case depicts a set of scenarios that describes an interaction between a user and a system. Use case diagram displays the relationship among actors and use cases which are the main components of a use case diagram Use case diagram of Image Noise Removal using Dual Domain Approach is shown in the Figure 3.1.[8]

3.2.1 Use Case in Expanded Format

For each module of the project several use cases are identified and the description of each use case is as follows:

3.2.1.1 Load Image Use Case

a) Name: Load Image based on the input string entered at the prompt

b) Actor: User

c) Pre-Condition: None

d) Post Condition: Image gets loaded in memory

e) Typical Course of Action:

Actor Action	System Response
1. User enters the image at the prompt.	2. System loads image in to memory. 3. System displays image on screen.

f) Alternate Course of Action:

Actor Action	System Response
1a. Image is not loaded.	2a. Display error message. 3a. Repeat step 1 to 3

3.2.1.2 Add noise Use Case

a) Name: Add noises to get three different images and naming them as n1 n2 and n3

b) Actor: User

c) Pre-Condition: Image is loaded.

d) Post Condition: Three types of noises are added to the image to produce three images.

e) Typical Course of Action:

Actor Action	System Response
1. None.	2. Based on default parameters noise is added. 3. The original Image and three new images (noisy) are displayed.

f) Alternate Course of Action:

Actor Action	System Response
1a. None.	2a. None.

3.2.1.3 Remove Noise using spatial domain Use Case

a) Name: Remove Noise in n1, n2, n3 using Spatial Domain

b) Actor: User

c) Pre-Condition: n1, n2 n3 ready to be processed.

d) Post Condition: Three new image with noise removed using spatial domain.

e) Typical Course of Action:

Actor Action	System Response
1. None.	2. Noise removed using spatial domain based on default parameters. 3. Images are displayed.

f) Alternate Course of Action:

Actor Action	System Response
1a. None.	2a. None.

3.2.1.4 Remove Noise using frequency domain Use Case

a) Name: Remove Noise in n1, n2, n3 using Frequency Domain

b) Actor: User

c) Pre-Condition: n1, n2 n3 ready to be processed.

d) Post Condition: Three new image with noise removed using frequency domain

e) Typical Course of Action:

Actor Action	System Response
1. None.	2. Noise removed using spatial domain based on default parameters. 3. Images are displayed.

f) Alternate Course of Action:

Actor Action	System Response
1a. None.	2a. None.

3.2.1.5 Remove Noise using Dual Domain Use Case

a) Name: Remove Noise in n1, n2, n3 using Dual Domain Approach

b) Actor: User

c) Pre-Condition: n1, n2 n3 ready to be processed.

d) Post Condition: Three new image with noise removed using frequency domain

e) Typical Course of Action:

Actor Action	System Response
1. User is asked to enter parameter for removing noise.	2. Noise removed using dual domain based on default parameters. 3. Images are displayed.

f) Alternate Course of Action:

Actor Action	System Response
1a. None.	2a. None.

3.2.1.6 Comparing the results of dual domain Use Case

a) Name: Comparing the results of dual domain visually, and mathematically (RMSE) with others domains.

b) Actor: User

c) Pre-Condition: Results of dual domain ready to be analyzed.

d) Post Condition: Dual domain Images analyzed both visually and mathematically.

e) Typical Course of Action:

Actor Action	System Response
1. User presses any to continue	2. Best Results of dual domain filtered images are displayed for visual analysis. 3. RMSE values for all dual domain and other domains are calculated for mathematical analysis.

f) Alternate Course of Action:

Actor Action	System Response
None	None

3.3 Conceptual Model

Conceptual objects have been described those are found in the domain of the system. This helps in further to understand the working of the system that is to be designed. Conceptual Model of "Image Noise Removal using Dual Domain Approach" is shown in Figure 3.2. [8]

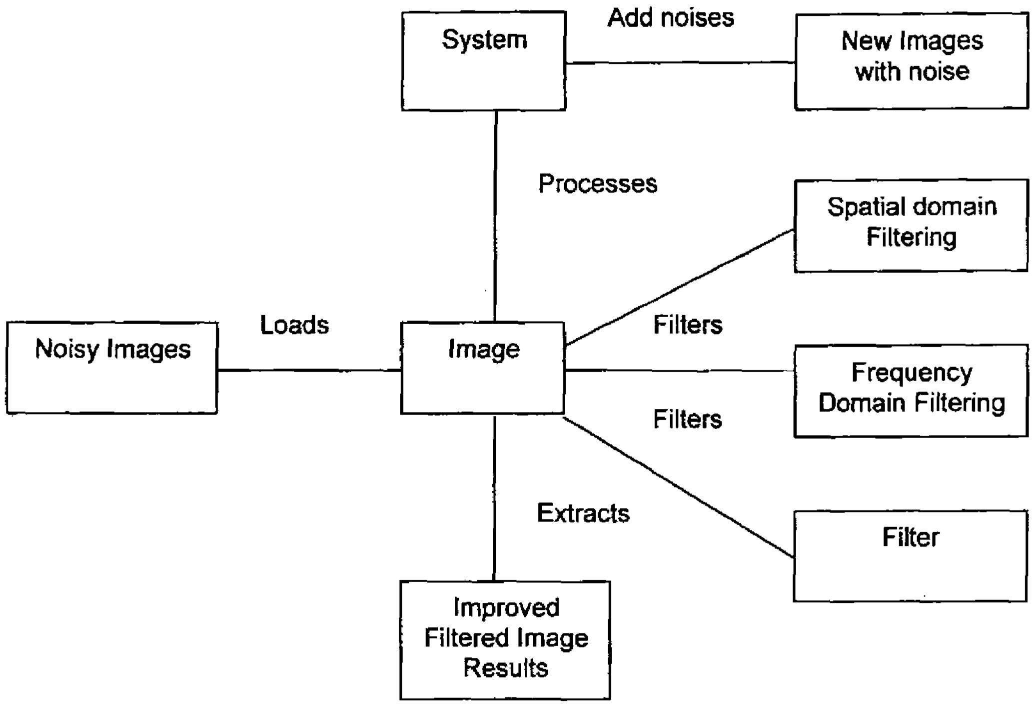


Figure 3.2: Conceptual Model of Image Noise Removal using Dual Domain Approach

CHAPTER 4

SYSTEM DESIGN

4. System Design

System design in the system analysis is the specification or construction of a technical, computer-based solution for the business requirements. The evaluation of alternative solutions and the specification of a detailed computer-based solution is the main goal. Moving from problem domain to the solution domain is reflected in the design phase. In System design, we develop the architectural detail required to implement a system or product. [8]

4.1 Object-Oriented Design Method

Object-Oriented design translates the Object Oriented Analysis (OOA) model of the real world scenarios into a practical implementation-specific model which becomes evident in software. Object-oriented design transforms the analysis model, created using object-oriented analysis method, into a design model that serves as a blueprint for software construction. For the development of the system under consideration the same technique is used. [8]

Object-oriented design (OOD) is concerned with developing an object-oriented model of a software system to implement the identified requirements. [8]

Object Oriented Design builds on the products developed during Object-Oriented Analysis (OOA) by refining candidate objects into classes, defining message protocols for all objects, defining data structures and procedures, and mapping these into an object-oriented programming language (OOPL). [8]

4.1.1 Sequence Diagrams

As some of the core objects in the system are prototyped along with the use cases being specified, the dynamic behavior of the system can be designed. Sequence diagram is a demonstration of the behavior of objects in a use case by describing the objects and the messages they pass, while it has been modified a bit to give an algorithm-specific implemental look and feel. The main purpose of the Sequence diagram is to emphasize the order in which things happen, which is self-evident in the diagrams below. [8]

The sequence diagram of Load Image is shown in Figure 4.1.

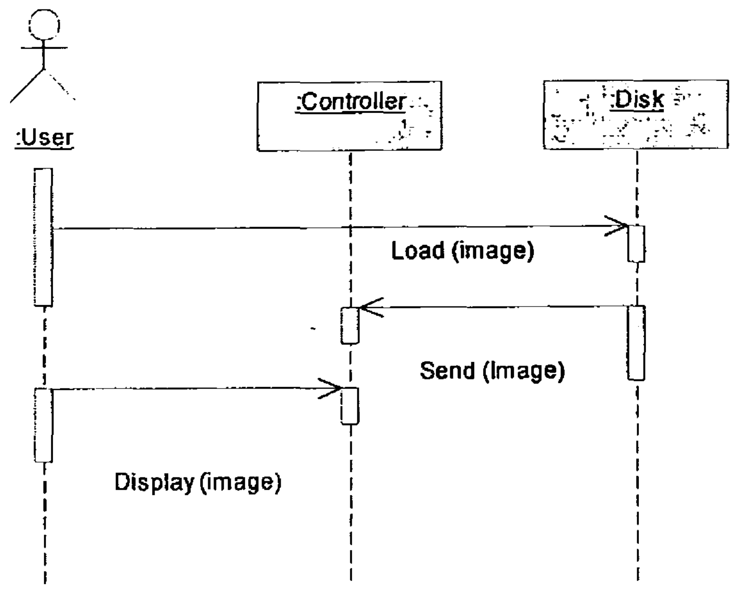


Figure 4.1: Sequence Diagram of Load Image.

The sequence diagram of Add Noise is shown in Figure 4.2.

TH. 4707

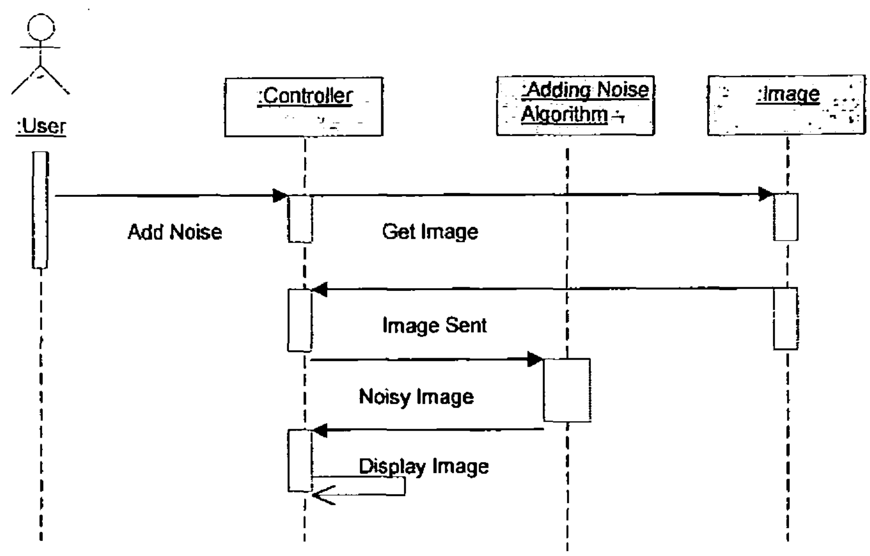


Figure 4.2: Sequence Diagram of Add Noise.

The sequence diagram of "Remove Noise using Spatial Domain" is shown in Figure 4.3.

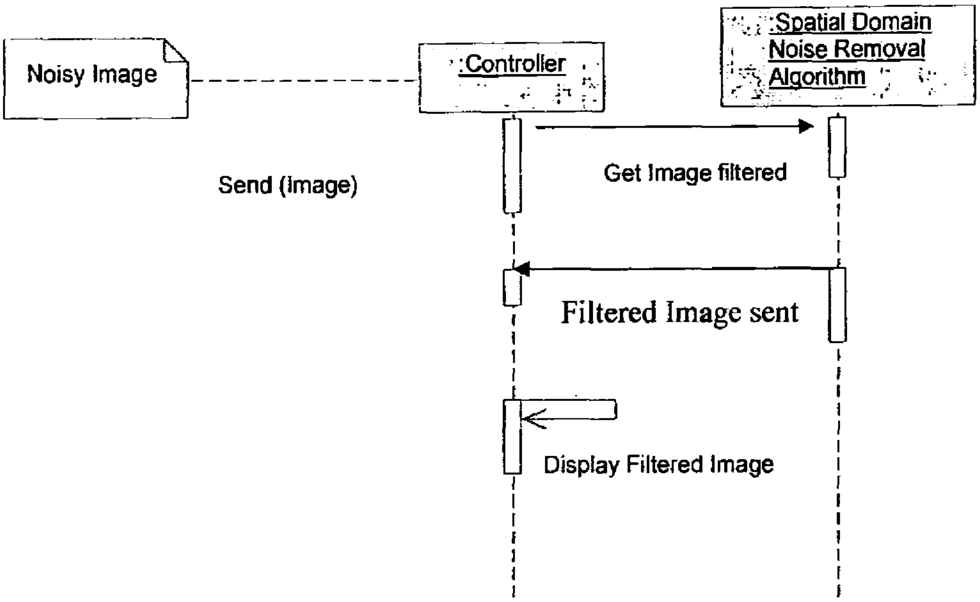


Figure 4.3: Sequence Diagram of Remove Noise using Spatial Domain.

The sequence diagram of "Remove Noise using Frequency Domain" is shown in Figure 4.4.

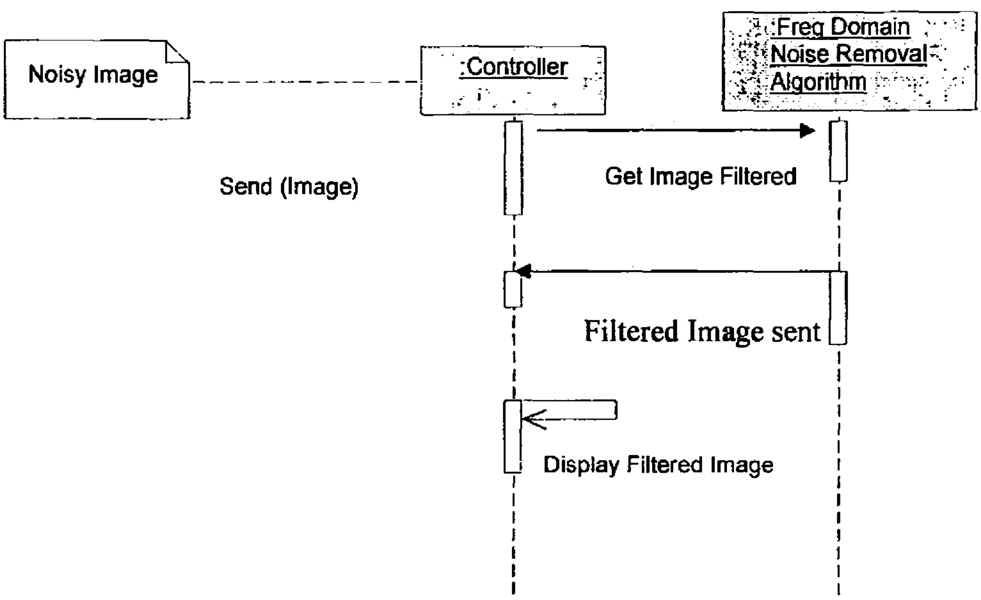


Figure 4.4: Sequence Diagram of Remove Noise using Frequency Domain.

The sequence diagram of "Remove Noise using Dual Domain" is shown in Figure 4.5.

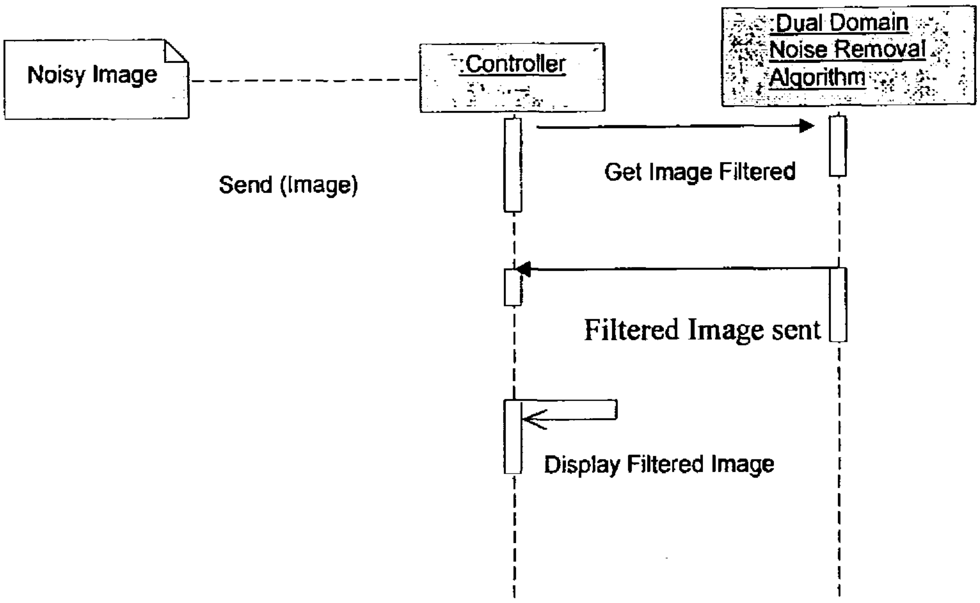


Figure 4.5: Sequence Diagram of Remove Noise using Dual Domain.

The sequence diagram of Comparison of Results is shown in Figure 4.6.

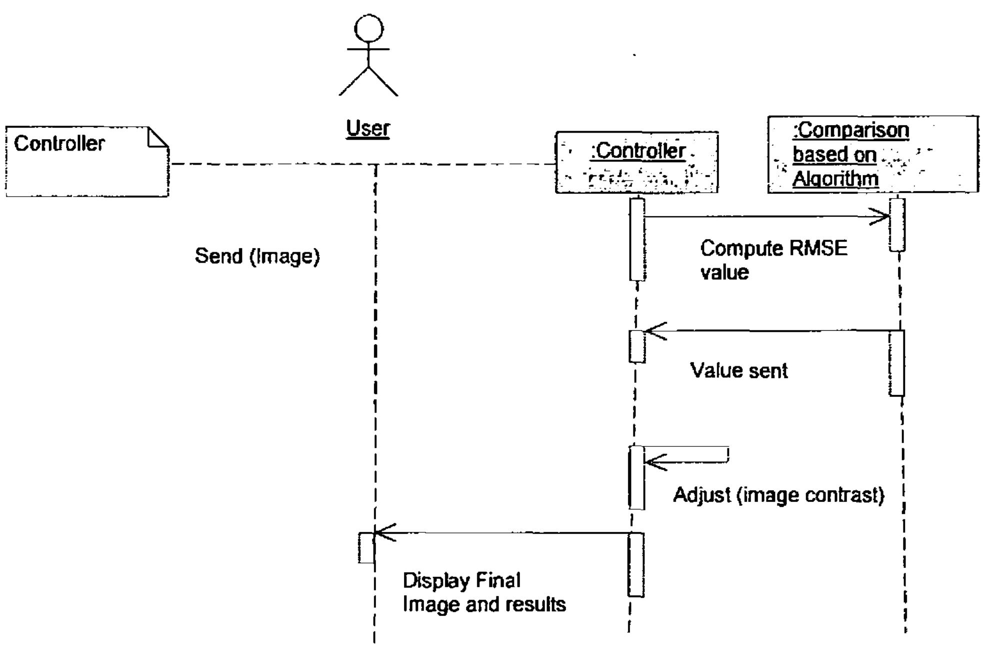


Figure 4.6: Sequence Diagram of Comparison of Results.

4.1.2 State Transition Diagram

State Transition Diagram of Image Noise Removal using Dual Domain Approach is shown in Figure 4.7.

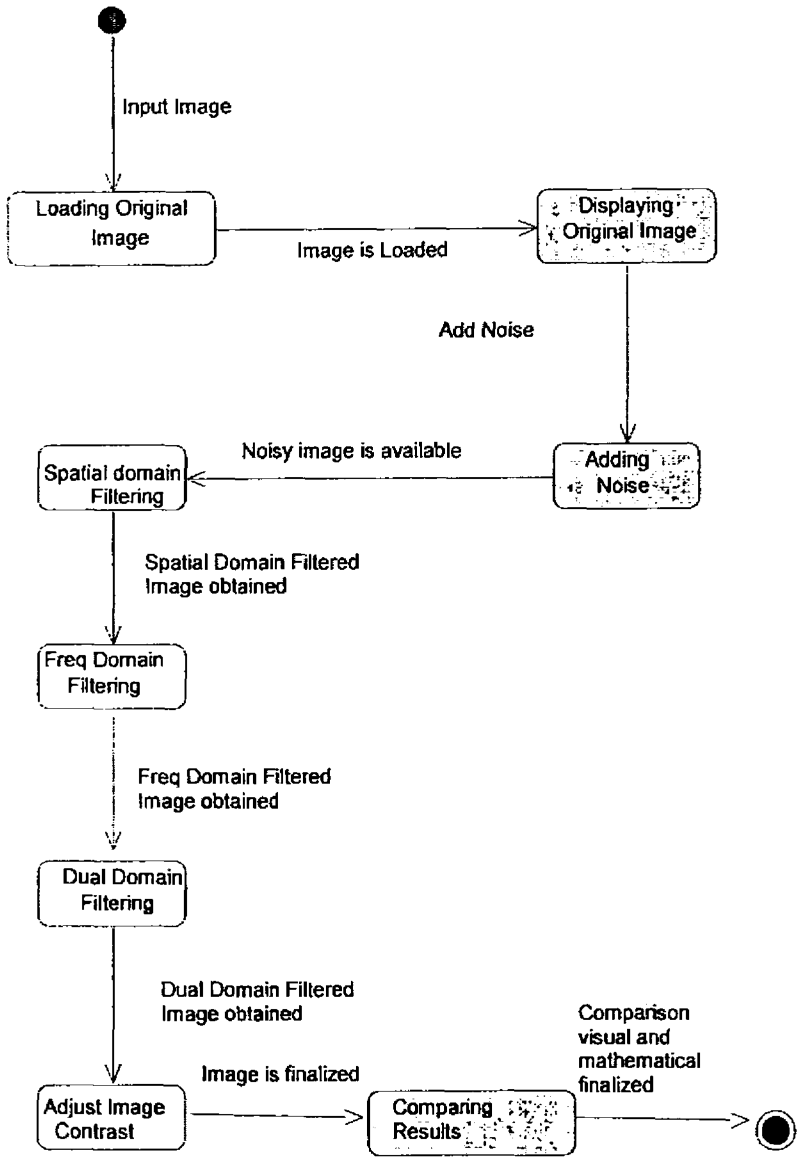


Figure 4.7: State Transition Diagram of Image Noise Removal using Dual Domain Approach

CHAPTER 5

IMPLEMENTATION

5. System Implementation

This chapter discusses the implementation details of the system. All the modules of the project are implemented in Matlab 7.0. The sample code and explanation for all the major modules is given in this chapter.

5.1 Loading Image

Image is loaded with all exceptions, in case of the file specified to be load does not exist or there is no file specified, are properly addressed the code given below:

```
%Inputting the test Image for the testing the algorithm.
in_str=input('Enter the image along with its extension that you want to
process=>','s');

while(isempty(in_str))||exist(in_str)~=2
   in_str=input('\n\nU didn''t enter the image or the image file does
not exist,\n Enter the image again along with its extension that you
want to process=>','s');
end
i=imread(in_str);
%Reading the image ends here.
```

5.2 Adding noise

In this module function has been made which is called to add three kinds of noise. All exceptions and default parameters or in case user specify them are properly addressed. The code is given below:

```
def_noise=input('Would you use the default parameters for adding all
types of noise(Y for yes and N for no)=>','s');
%enforcing the correct default value
if  (isempty(def_noise))  ||  def_noise~='n'  ||  def_noise~='N'  ||
def_noise~='Y' || def_noise~='y'
    def_noise='y';
end
%enforcing corect default value code ends here
proceed=sprintf('\n\n\n\n')  % New line added

% Putting condition in order to use default values.
if def_noise=='n' || def_noise=='N'
    disp('If you would not enter value for any parameter then default
value will be assumed');
    proceed=sprintf('\n\n')  % two line added
        M=input('Input gaussian noise mean(default 0)=>');
```

```
    if isempty(M)
     M=0;
    end

    Vg=input('Input gaussian noise variance(default 0.01)=>');
    if isempty(Vg)
     Vg=0.01;
    end

    D=input('Input salt and pepper noise density(default 0.05)=>');
    if isempty(D)
     D=0.05;
    end

    Vs=input('Input speckle noise variance(default 0.04)=>');
    if isempty(Vs)
        Vs=0.04;
    end
else
    M=0;
    Vg=0.01;
    D=0.05;
    Vs=0.04;
end;
% condition ends here


% The following is the function call for adding noise.
[n1,n2,n3]=addnoise(img,M,Vg,D,Vs);
%Adding noise code ends here.
```

5.3 Remove Noise using Spatial Domain

Now for noise removal in spatial domain a function is called on each of the noisy image i.e., n1,n2 and n3. All code is self explanatory with appropriate documentation. The function and its called code is given below:

```
[f_img1n1s,f_img2n1s]=spatial_filtering_function(n1,def);%    Code    for
%filtering n1
function [img1,img2]=spatial_filtering_function(f,def)
%This function filter the image f in spatial domain using Gaussian and
%Average filter
disp('Gaussian filter Mask')
h1=spatial_filter_function('gaussian',def)
disp('Average filter Mask')
h2=spatial_filter_function('average',def)


img1=imfilter(f,h1,'symmetric');
img2=imfilter(f,h2,'symmetric');
```

5.4 Remove Noise using Frequency domain

Now for noise removal in frequency domain a function is called on each of the noisy image i.e., n1, n2 and n3. All code is self explanatory with appropriate documentation. The function and its code are given below:

```
[f_img1n1f,f_img2n1f]=frequency_filtering_function(n1,def);%   The   code
%for filtering n1 in the frequency domain

function [filt_img1,filt_img2]=frequency_filtering_function(f,def)

pq=paddedsize(size(f));

if def=='n' || def=='N'
    percentage=input('Enter the percentage of the width of padded image
you would like D0 to be(def 7)=>');
    if isempty(percentage)
        percentage=7;
    end;
    n=input('Enter  the  order  of  the  butterworth  filter  you  would
use(def 1)=>');
    if isempty(n)
        n=1;
    end;
else
    n=1;
    percentage=7;
end;

percent=double(percentage/100);
D0=percent*pq(1);

x=pq(1);
y=pq(2);
H1=lpfilt('gaussian',pq(1),pq(2),D0);
H2=lpfilt('btw',pq(1),pq(2),D0,n);



% Code  for  displaying  filters  as  an  image  and  plotting  filters  as
function starts here
%   This  code  might  be  commented  for  being  disabled  and  can  be  used
anytime.

 figure,subplot(2,2,1),imshow(log(1+abs(fftshift(H1))),[]),
title('Gaussian LP Filt image');
 Hs1=fftshift(H1);

 subplot(2,2,2),imshow(log(1+abs(fftshift(H2))),[]),  title('Butterworth
LP Filt Image');
 Hs2=fftshift(H2);
 %figure,mesh(H2(1:10:pq(1),1:10:pq(2))),  title('Butterworth  LP  Filt
Plot');
```

```
%
% subplot(2,3,3),imshow(log(1+abs(fftshift(H3))),[]), title('Ideal LP
Filt Image');
% Hs3=fftshift(H3);
%   %figure,mesh(Hs3(1:10:pq(1),1:10:pq(2))),  title('Ideal  LP  Filt
Plot');
 subplot(2,2,3),mesh(Hs1(1:10:x,1:10:y)),   title('Gaussian   LP   Filt
Plot');
 subplot(2,2,4),mesh(Hs2(1:10:x,1:10:y)),    title('Butterworth    Filt
Plot');
%   subplot(2,3,6),mesh(Hs3(1:10:x,1:10:y)),  title('Ideal  LP  Filt
Image');

% Code for displaying filters as an image and plotting filter as
function ends here

filt_img1=dftfilt(f,H1);
filt_img2=dftfilt(f,H2);
```

5.5 Remove Noise using Dual Domain

Now for noise removal in Dual Domain a function is called on each of the noisy image i.e., n1, n2 and n3. All code is self explanatory with appropriate and sufficient documentation. The function and its code are given below:

```
res1=DDom_filtering_func(n1,def,'n1');% The code for filtering n1 in
Dual Domain

function res=DDom_filtering_func(f, def,type)

% This function to utilizes the dual domain approach for filtering
% A call to function is made as result=DDom_filtering_func(f,def,type)
% Where f is the image to be filtered
% def is the character datatype and the valid values are 'y','Y','n',
and 'N' Type is character array datatype and is used to indicate the
type of noise to be filtered, valid values are 'n1', 'n2' and 'n3'
% where n1 indicate Salt & Pepper noise, n2 indicate Gaussian noise and
n3 indicate speckle noise.
% For setting up the values to default and entering user selected
values in case of value of def is equal to 'n' or 'N' reflecting the
user's wish to enter his own selected values.

if type=='n2' | type=='n3'
    pq=paddedsize(size(f));
    if def=='n' || def=='N'
            percentage=input('Enter the percentage of the width of
padded image you would like D0 to be(def 10)=>');
            if isempty(percentage)
                percentage=10;% in case value is not input
            end;
            n=input('Enter the order of the butterworth filter you
would use(def 2)=>');
            if isempty(n)
                n=2;% in case order of the filter is not input
```

```
            end;
    else
        n=2;% in case default values will be used.
        percentage=10;
    end;

percent=double(percentage/100);
D0=percent*pq(1);

end;

% two kinds of frequency domain filters
% four kinds of spatial domain filters
% And they are utilized in the dual domain approach out of
% which two are from frequency domain and three are from spatial domain

% Switch case for handling three different types of noises
switch type
    case 'n1'%Salt and pepper, Random noise case addressed
        f1 = adpmedian(f, 7);
        [fg1,fb1]=Hfemp_filt_func(f1,.5,1.5,def);
        save ping fb1
        f2=f1+fb1;
        img1=spfilt(f1,'gmean',3,3);
        f3=f2+img1;
        res=gscale(f3,'minmax',0,1,1);
    case 'n2'%Gaussian noise case addressed
        H2=lpfilt('btw',pq(1),pq(2),D0,n);
        f1=dftfilt(f,H2);
        z=f1;
        f2=spfilt(f,'median',3,3);ding;% values from n1 is utilized
        f3=f1+f2;
        img1=f3+z;
       % img2= SL_spatial_filtering_func(img1,def,1); There is no need
       at this very moment, but may be used in the future enhancements.
        res=gscale(img1,'minmax',0,1,.9);
    case 'n3'%Speckle noise case addressed
        H2=lpfilt('btw',pq(1),pq(2),D0,n);
        f1=dftfilt(f,H2);
        z=f1;
        f2=spfilt(f,'median',3,3);ding; % values from n1 is utilized
        f3=f1+f2;
        img1=f3+z;
        res=gscale(img1,'minmax',0,1,.85);
    otherwise
        error('unknown case')

end
```

5.6 Comparison of the results

Finally, the results of the dual domain are compared with the results in the spatial and frequency domain and then images are analyzed both visually and mathematically, visually by inspecting it on aesthetic basis while mathematically by comparing the RMS values with each other.

The following code is an example of only n3 where analysis is being done by visually displaying and finally comparing the RMS values for all results of n3.

```
res3=DDom_filtering_func(n3,def,'n3');% The code for filtering n3 in Dual Domain

 figure(4), subplot(2,3,1),imshow(img),title('Original Image');
subplot(2,3,2),imshow(n3), title('n3, the Image corrupted by Speckle noise');
subplot(2,3,3),imshow(f_img2n3s),title('Filtering in Spatial Domain (Average Filter)');
subplot(2,3,4),imshow(f_img2n3f),title('Filtering in Freq Domain (Butterworth LP filt)');
subplot(2,3,5),imshow(f_img1n3f),title('Filtering in Freq Domain (Gaussian LP filt)');
subplot(2,3,6),imshow(res3),title('Result of filtering n3 using Dual Domain approach');
```

% rms is compared here

```
n3_rms1n=root_m_sq_func(img,n3);
n3_rms1f=root_m_sq_func(img,f_img2n3f);
n3_rms1s=root_m_sq_func(img,f_img2n3s);
n3_rms1dd=root_m_sq_func(img,res3);

ans3=sprintf('\n\n\nRMS n3 of noisy image = %1.5f,\n RMS n3 filtered image using butterworth LP in frequency domain = %1.5f,\n RMS n3 filtered image using average filter in spatial domain = %1.5f,\n while RMS n3 in Dual Domain = %1.5f',n3_rms1n,n3_rms1f,n3_rms1s,n3_rms1dd)
```

CHAPTER 6

RESULTS

6. Results

Noise removal techniques in spatial domain and frequency domain have been applied in order to do the comparison of the results with the proposed system. The dual domain technique employed for removing noise yielded unprecedented results. A comprehensive and thorough analysis both visual and mathematical accompanies the results presented under upcoming headings.

6.1 Overview of Image Noise Removal Results

The algorithm of filtering in dual domain processing is implemented and then applied on the test images on which then analysis is performed.

Consider the test images along with other three noisy images as shown in the Figure 6.1, 6.6, 6.11, 6.16, and 6.21. Firstly, n1, images corrupted with Salt & Pepper noise are processed and the results along with the original Image and the corrupted one are shown in Figure 6.2, 6.7, 6.12, 6.17 and 6.22. It is evident that Dual Domain in this case yielded commendable results compared to the other domains. Secondly, n2, images corrupted with Gaussian noise and their results, once again proving grounds, are shown in the Figure 6.3, 6.8, 6.13, 6.18 and 6.23. Thirdly, n3, images corrupted with Speckle noise and their self-evident results are shown in the Figure 6.4, 6.9, 6.14, 6.19 and 6.24 for visual inspection. After processing all of the three images to get the over all pictures, results are shown in the Figure 6.5, 6.10, 6.15, 6.20, and 6.25 for the integrity of the work. Then detailed visual and mathematical analysis is performed where RMSE values are displayed in tabular form for mathematical analysis; on inspection both visual and mathematical, it becomes utterly conspicuous that the dual domain algorithmic technique yields competent results. Finally, a graphical analysis is been carried out to exhibit the completeness of the analysis on the results.

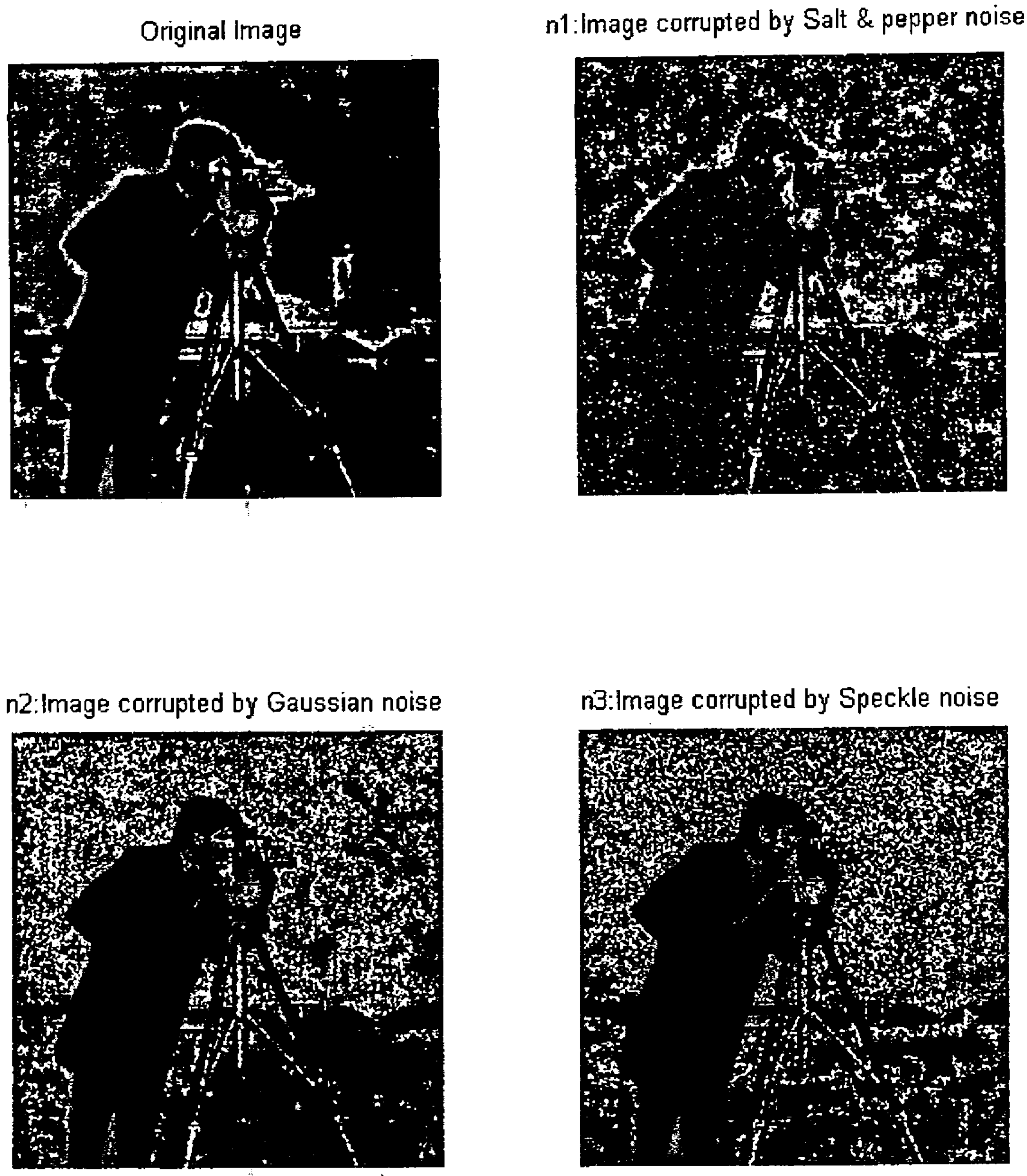
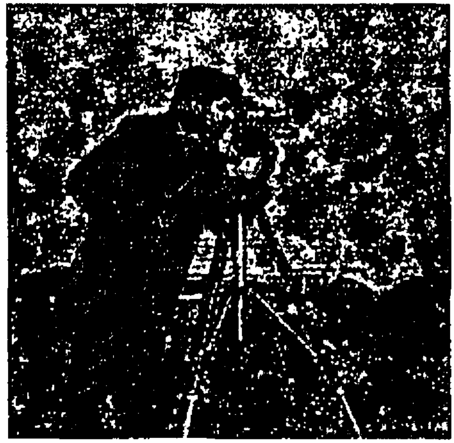


Figure 6.1: Original Image along with three Noisy Images

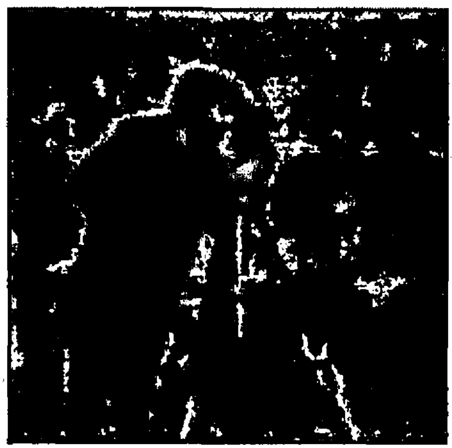
Original Image

n1:Image corrupted by Salt & Pepper noise



Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)



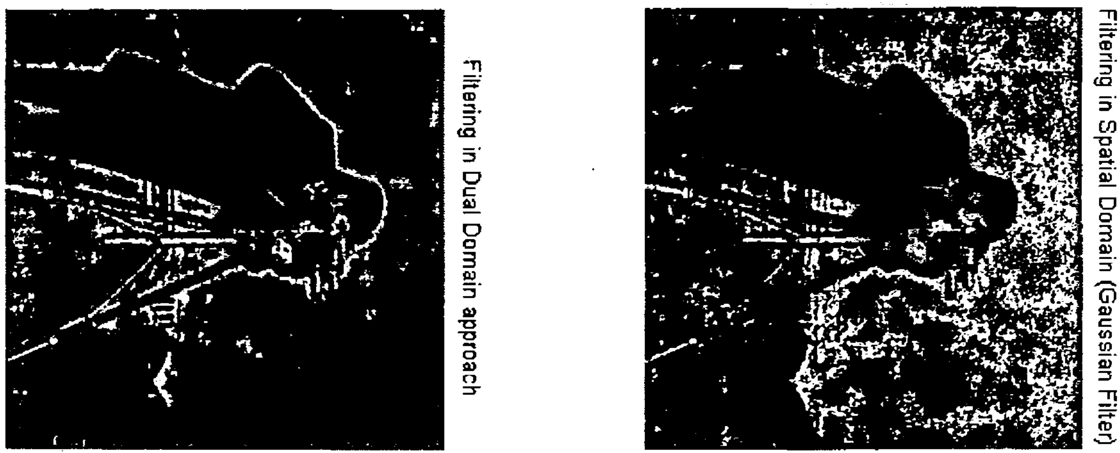
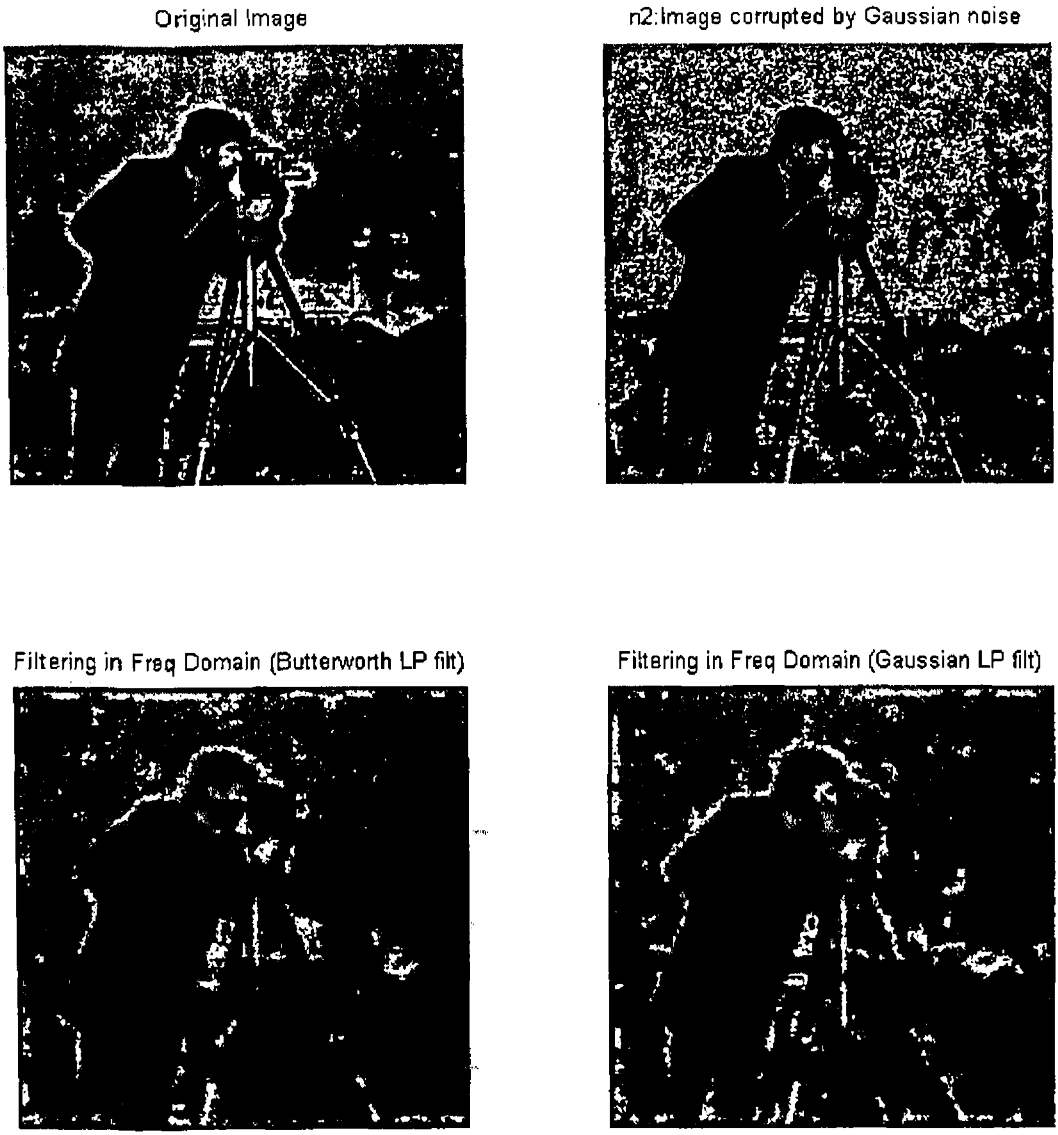


Figure 6.2: Noise Removal of n1


Original Image
n2:Image corrupted by Gaussian noise
Filtering in Freq Domain (Butterworth LP filt)
Filtering in Freq Domain (Gaussian LP filt)

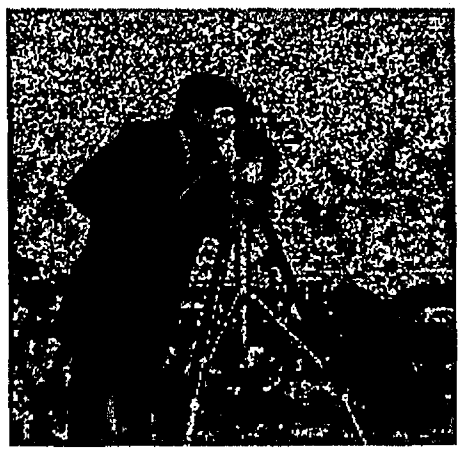
Filtering in Spatial Domain (Gaussian Filter)

Filtering in Dual Domain approach

Figure 6.3: Noise Removal of n2

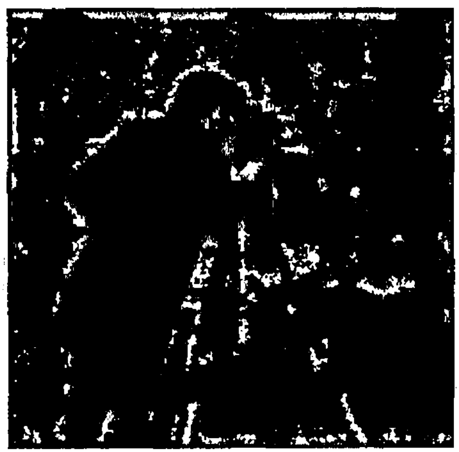
Original Image

n3, the Image corrupted by Speckle noise



Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)

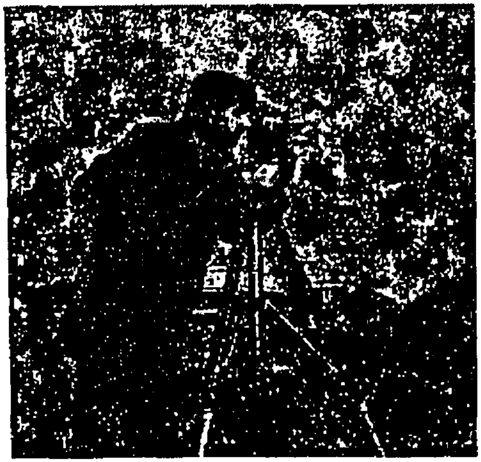


Filtering in Spatial Domain (Average Filter)

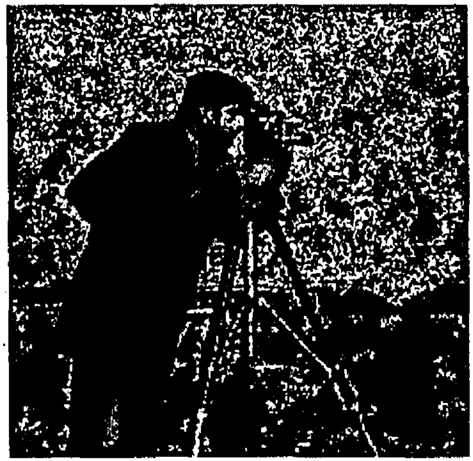
Filtering in Dual Domain approach

Figure 6.4: Noise Removal of n3

n1:Image corrupted by Salt & pepper noise



n2:Image corrupted by Gaussian noise



n3:Ima

Filtering n1 in Dual Domain

Filtering n2 in Dual Domain

F

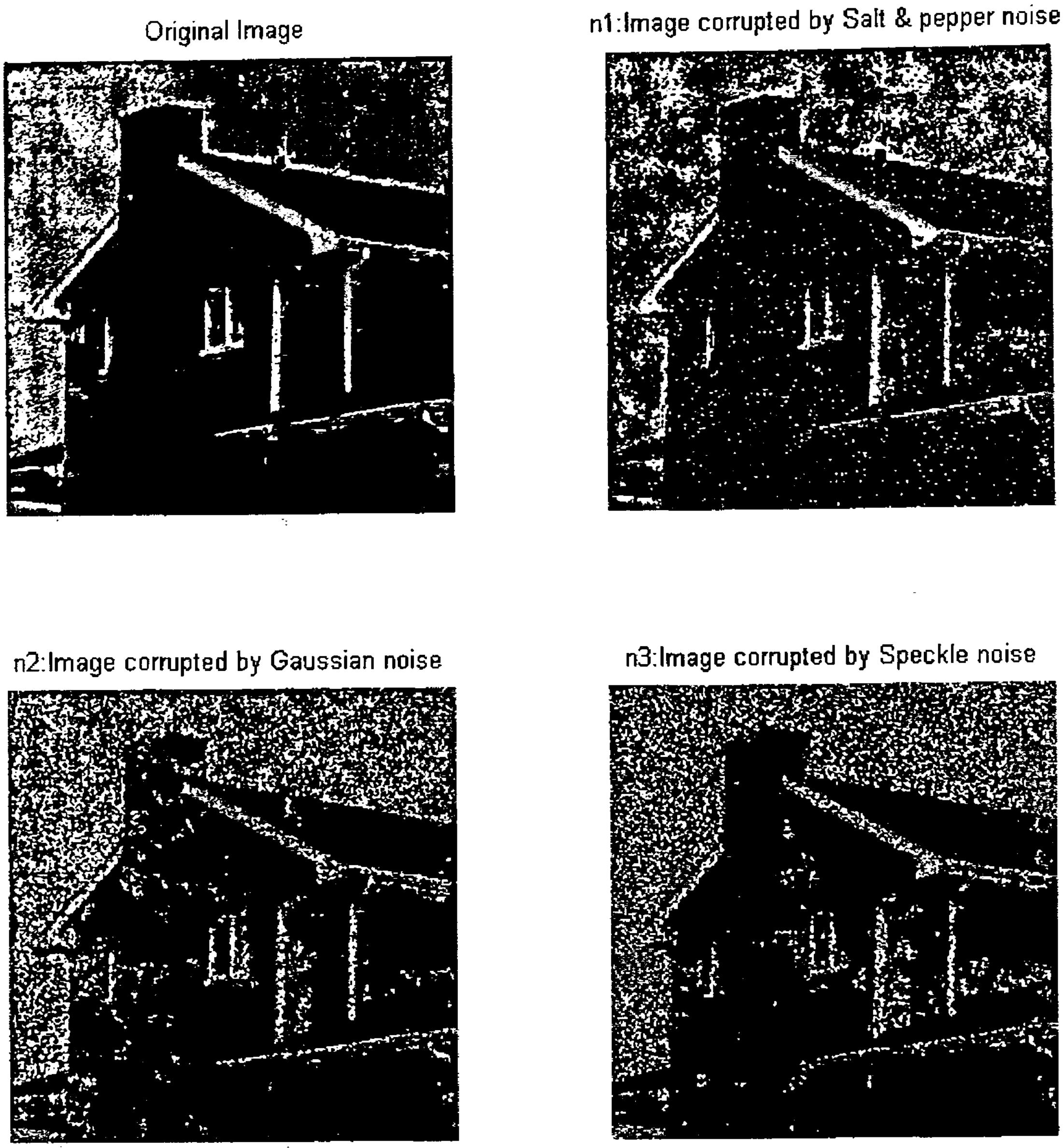
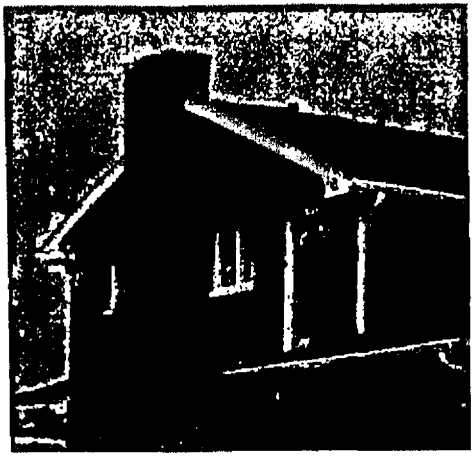
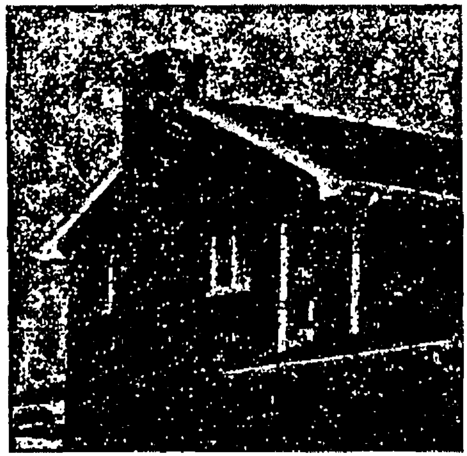


Figure 6.6: Original Image along with three Noisy Images

Original Image



n1:Image corrupted by Salt & Pepper noise



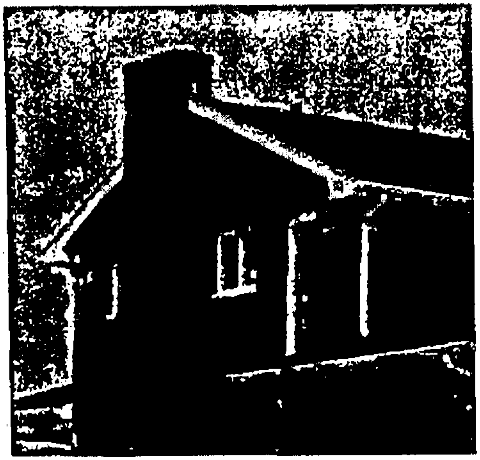
Filtering in

Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)

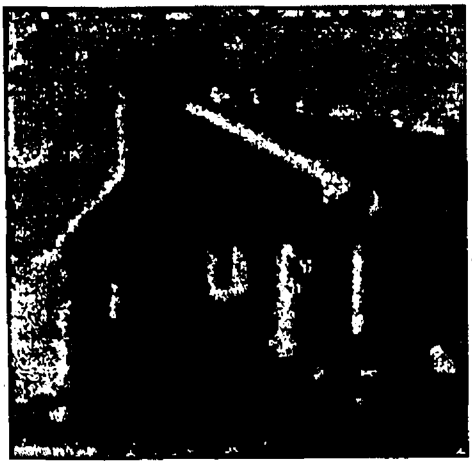
Filteri

Original Image



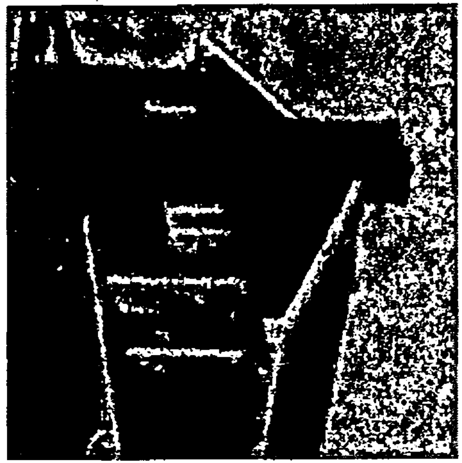
n2:Image corrupted by Gaussian noise

Filtering in Freq Domain (Butterworth LP filt)



Filtering in Freq Domain (Gaussian LP filt)

Filtering in Spatial Domain (Average Filter)



Filtering in Dual Domain approach

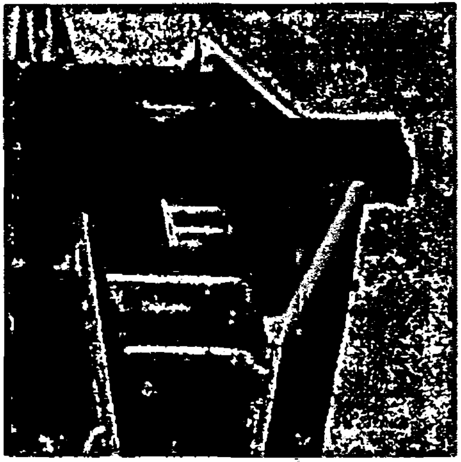
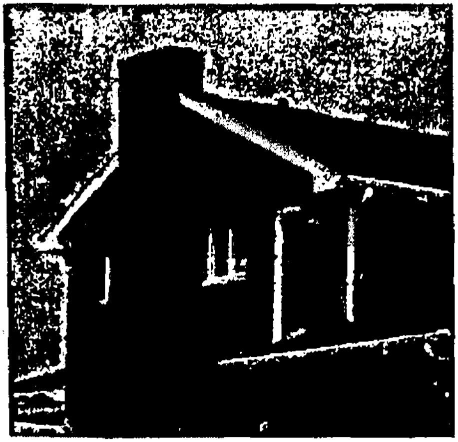


Figure 6.8: Noise Removal of n2

Original Image



n3, the Image corrupted by Speckle noise

Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)

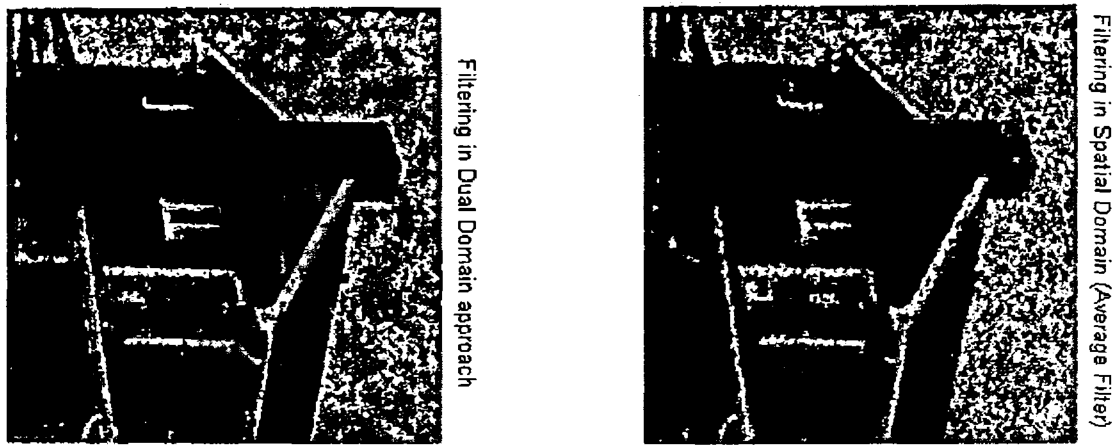
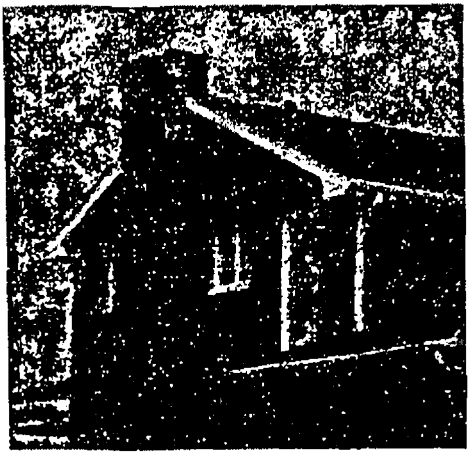
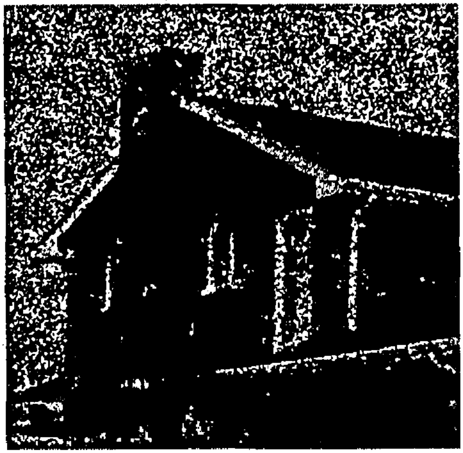


Figure 6.9: Noise Removal of n3

n1 :Image corrupted by Salt & pepper noise

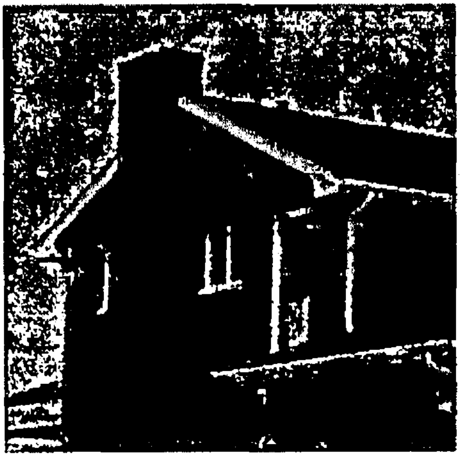


n2:Image corrupted by Gaussian noise



Filtering n1 in Dual Domain

Filtering n2 in Dual Domain



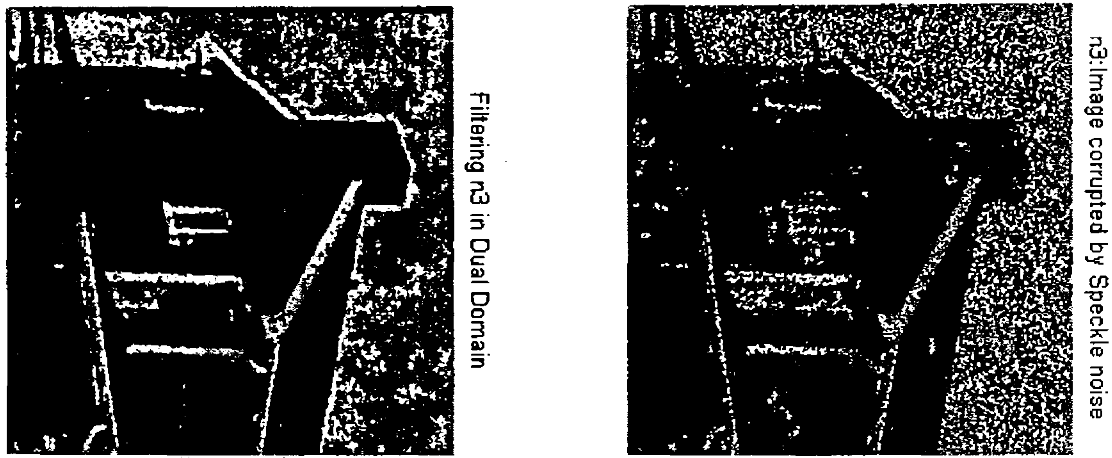


Figure 6.10: Filtered Images along with their noisy counterparts

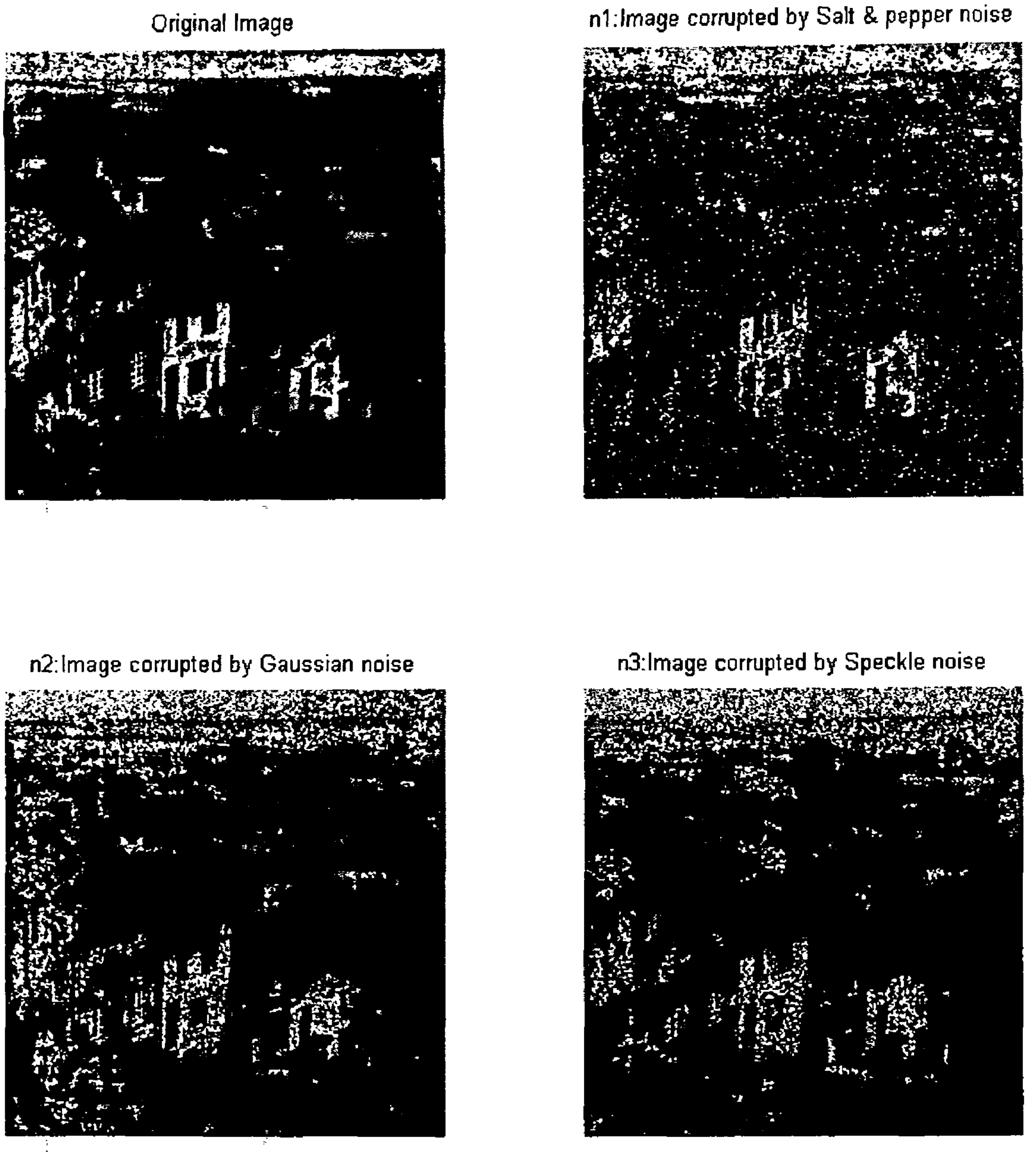
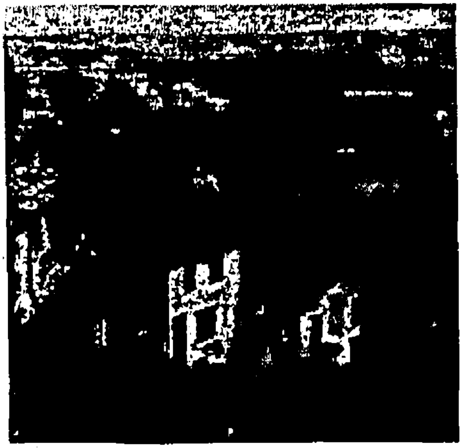


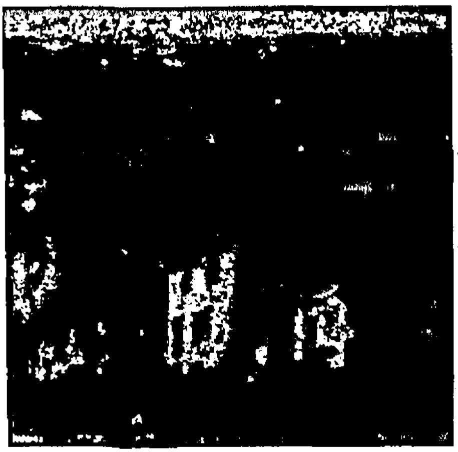
Figure 6.11: Original Image along with three Noisy Images

Original Image



n1:Image corrupted by Salt & Pepper noise

Filtering in Freq Domain (Butterworth LP filt)



Filtering in Freq Domain (Gaussian LP filt)

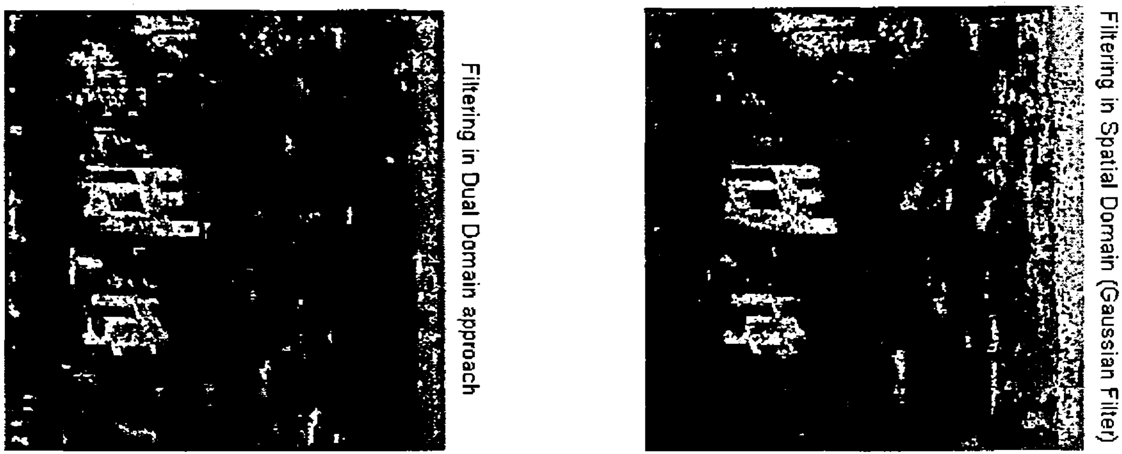
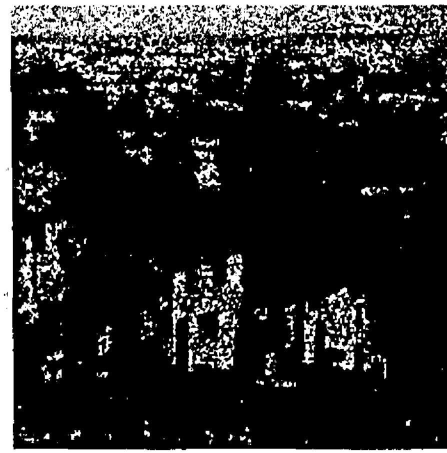


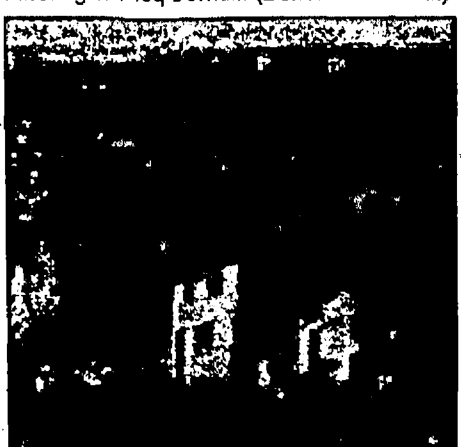
Figure 6.12: Noise Removal of n1

Original Image

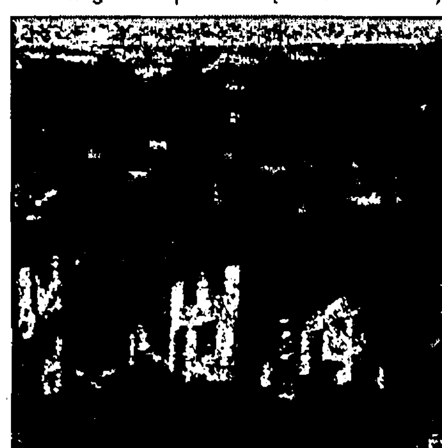
n2:Image corrupted by Gaussian noise



Filtering in Freq Domain (Butterworth LP filt)



Filtering in Freq Domain (Gaussian LP filt)



Filtering in Spatial Domain (Gaussian Filter)

Filtering in Dual Domain approach

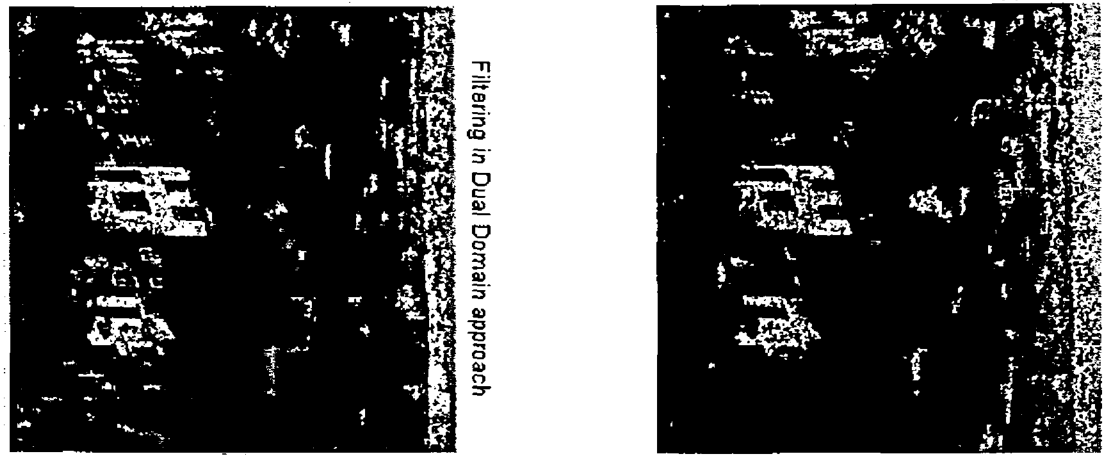
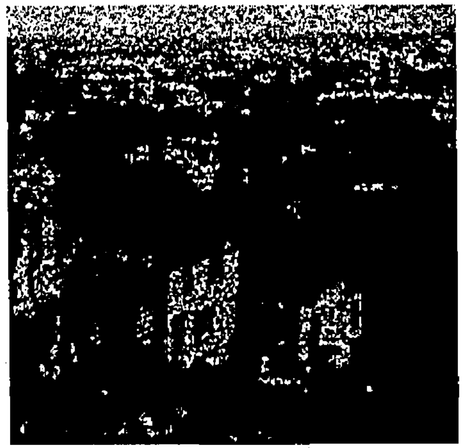


Figure 6.13: Noise Removal of n2

Original Image

n3, the Image corrupted by Speckle noise



Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)

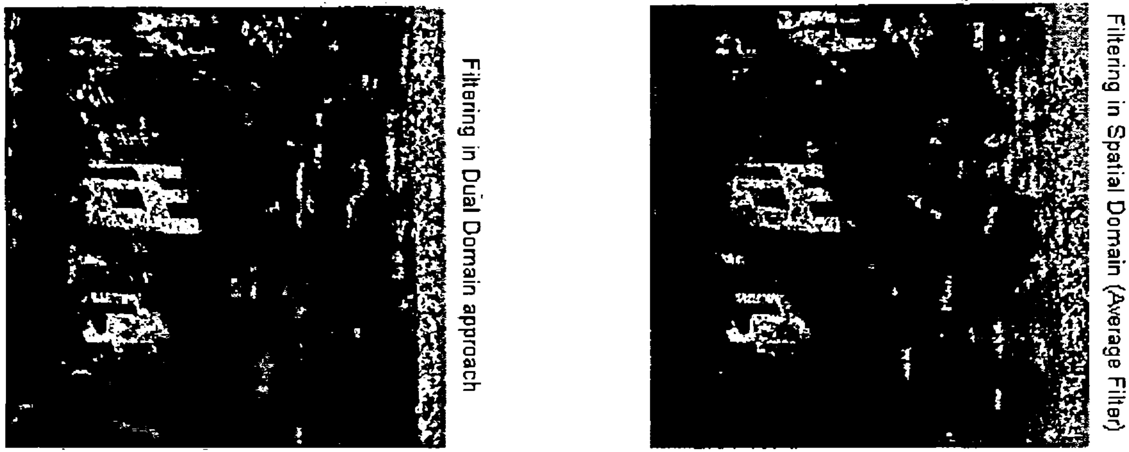
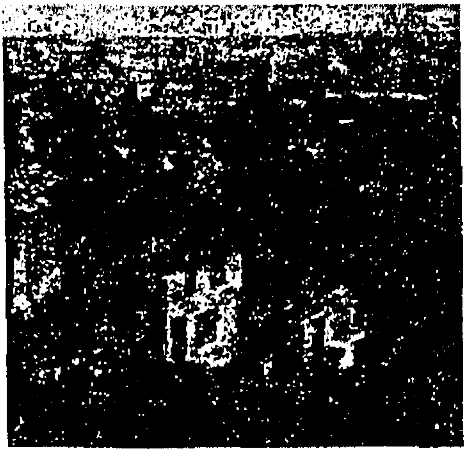
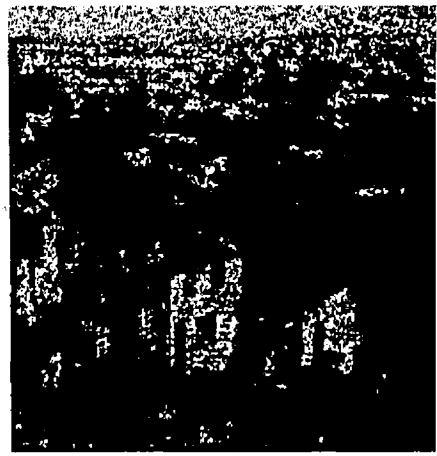


Figure 6.14: Noise Removal of n3

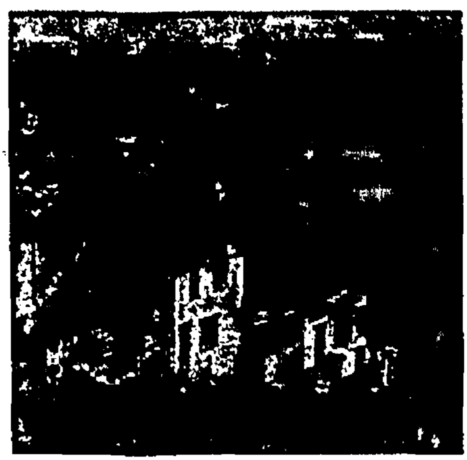
n1 :Image corrupted by Salt & pepper noise



n2:Image corrupted by Gaussian noise



Filtering n1 in Dual Domain



Filtering n2 in Dual Domain

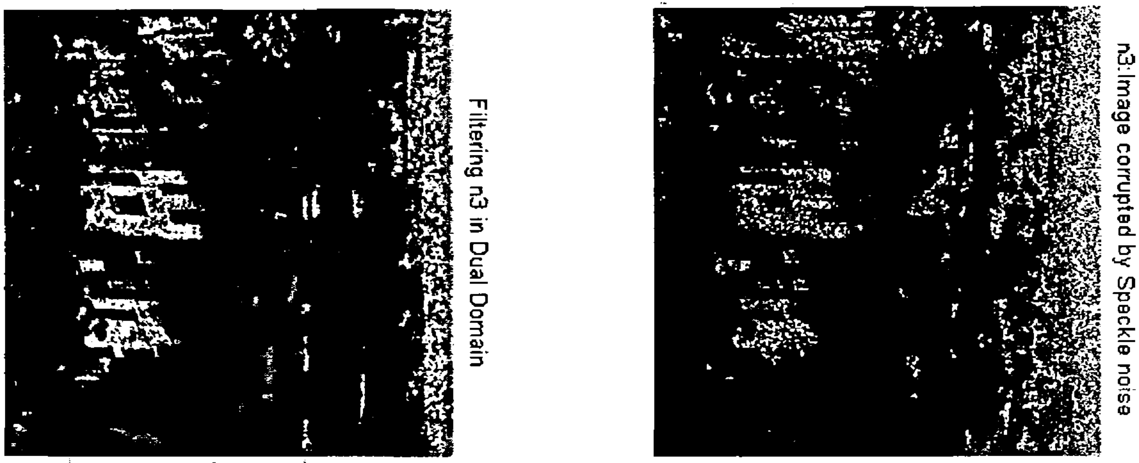


Figure 6.15: Filtered Images along with their noisy counterparts

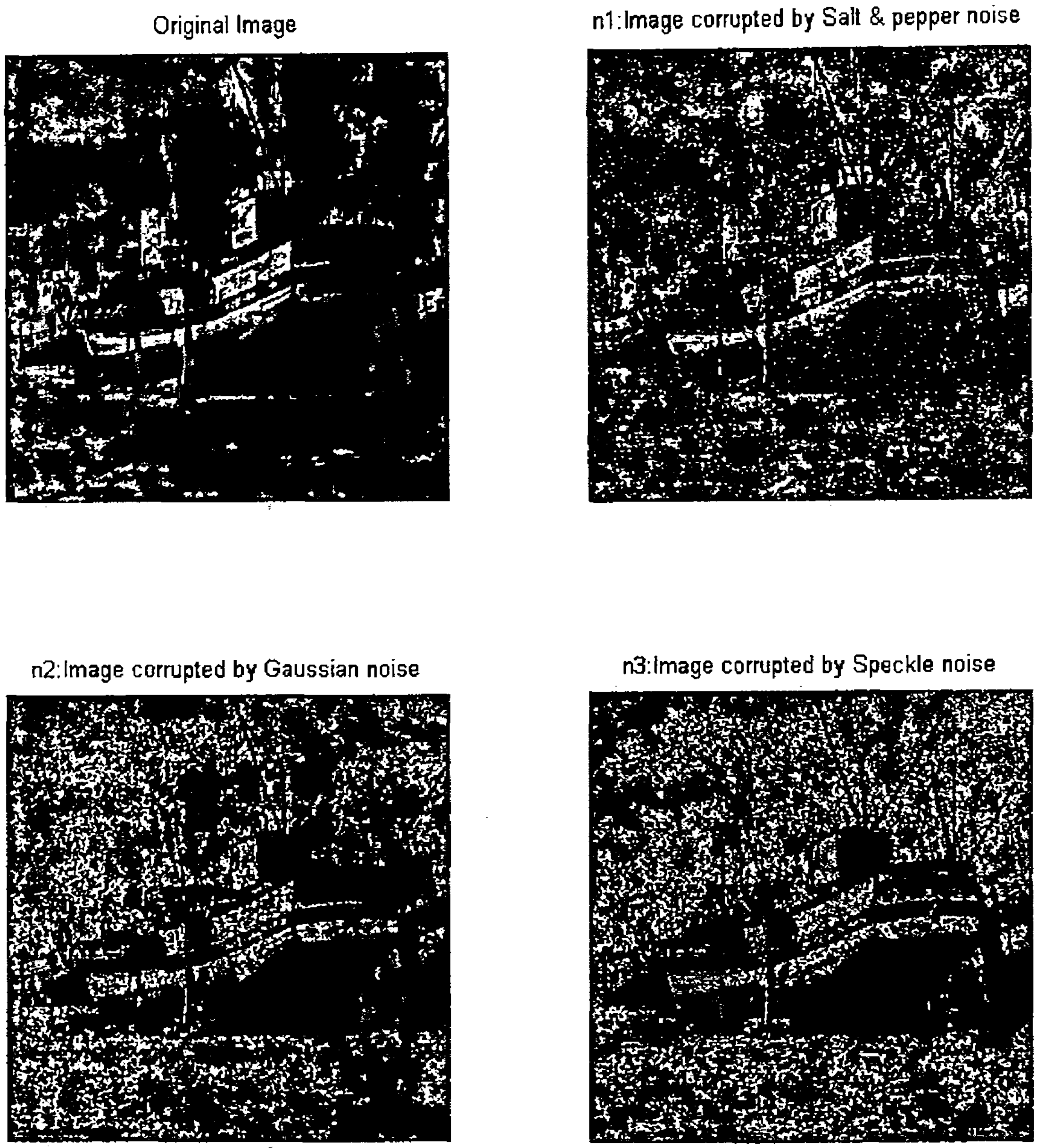
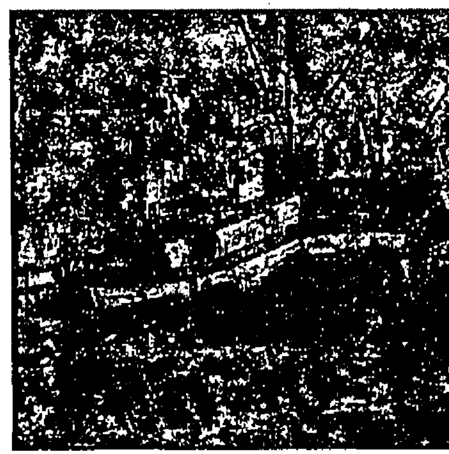


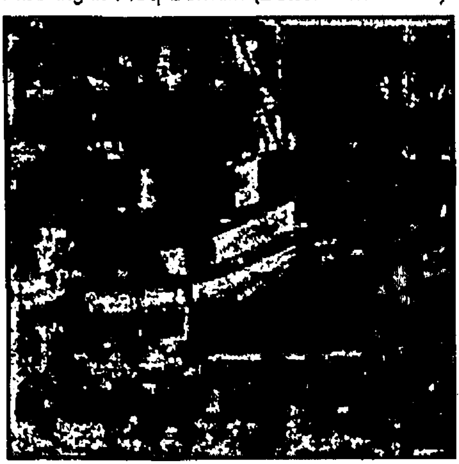
Figure 6.16: Original Image along with three Noisy Images

Original Image

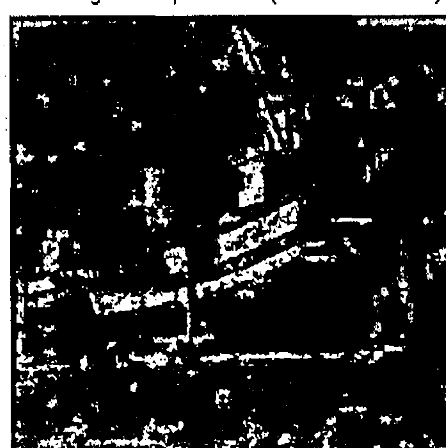
n1:Image corrupted by Salt & Pepper noise



Filtering in Freq Domain (Butterworth LP filt)



Filtering in Freq Domain (Gaussian LP filt)



Filtering in Spatial Domain (Gaussian Filter)

Filtering in Dual Domain approach

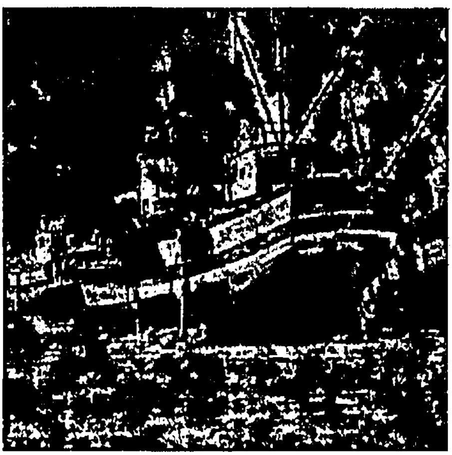
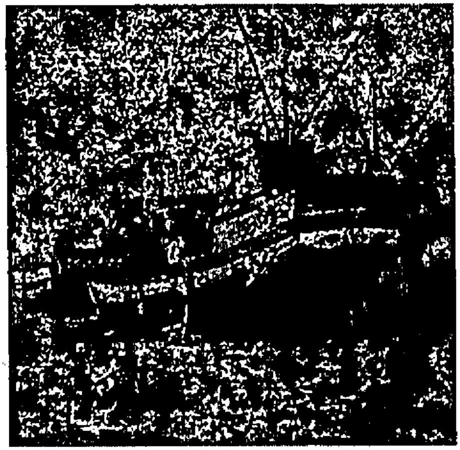


Figure 6.17: Noise Removal of n1

Original Image

n2:Image corrupted by Gaussian noise



Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)

Filtering in Spatial Domain (Gaussian Filter)

Filtering in Dual Domain approach

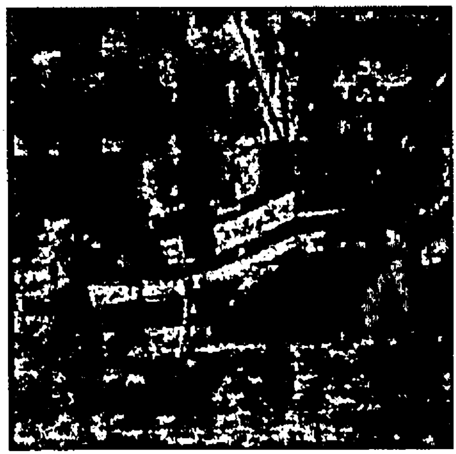
Figure 6.18: Noise Removal of n2

Original Image

n3, the image corrupted by Speckle noise

Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)



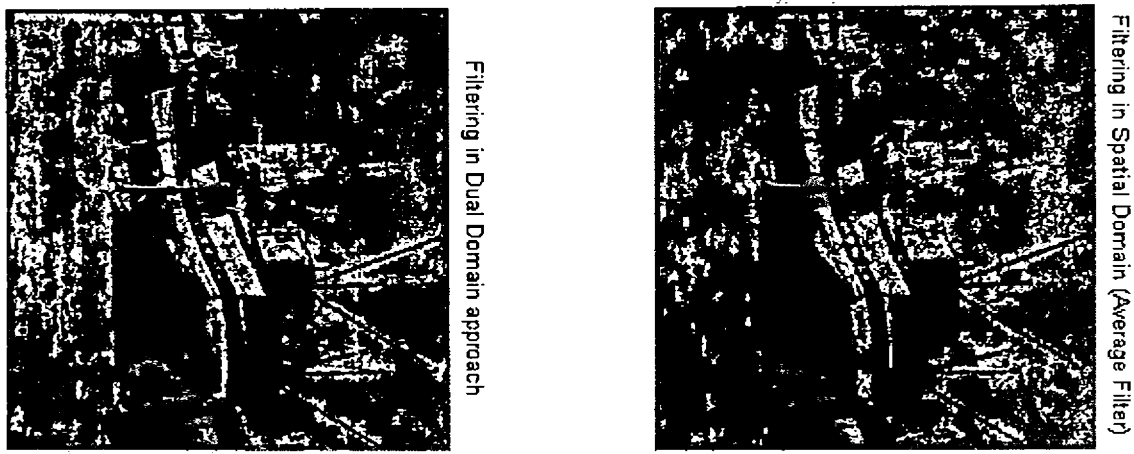


Figure 6.19: Noise Removal of n3

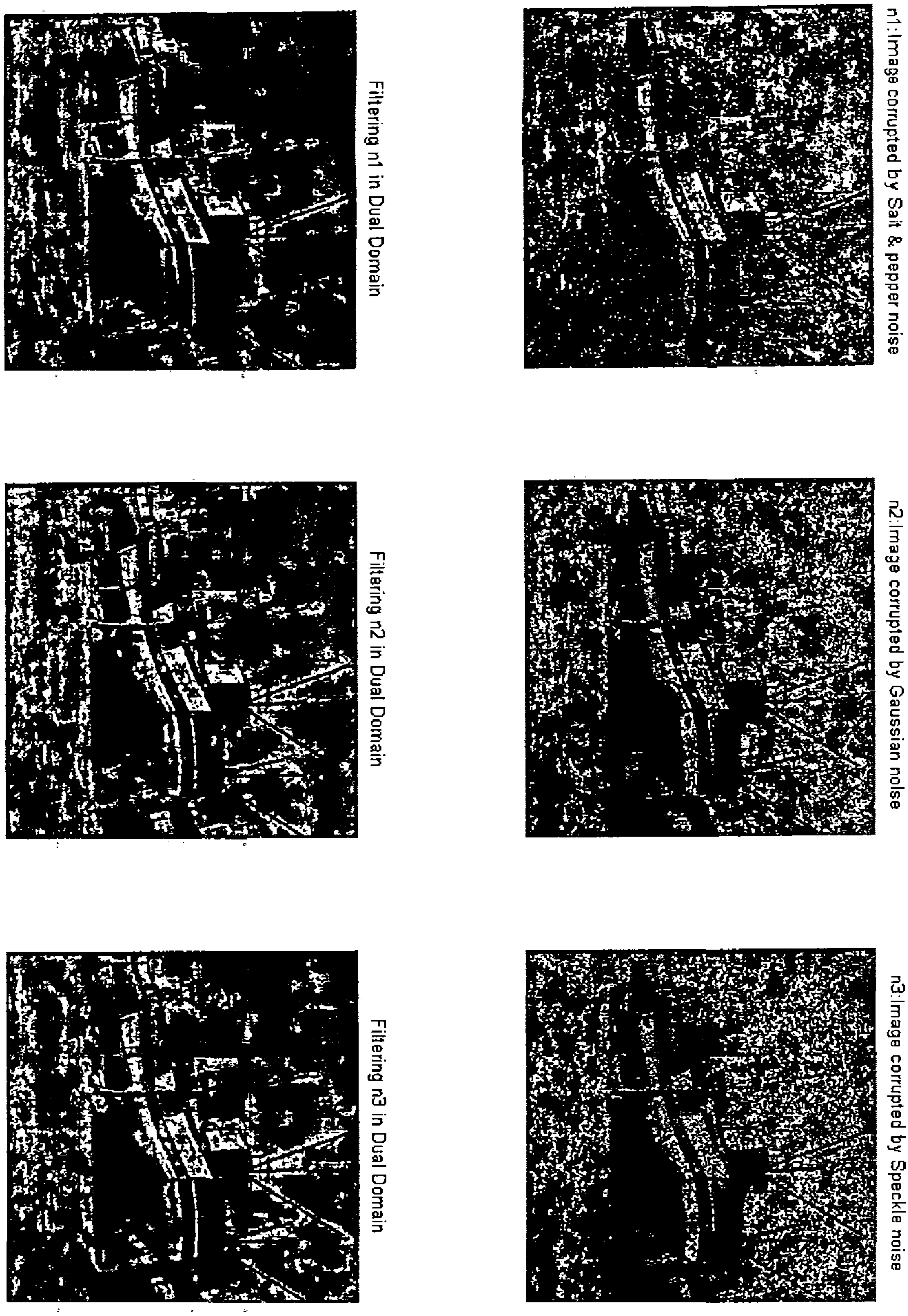


Figure 6.20: Filtered Images along with their noisy counterparts

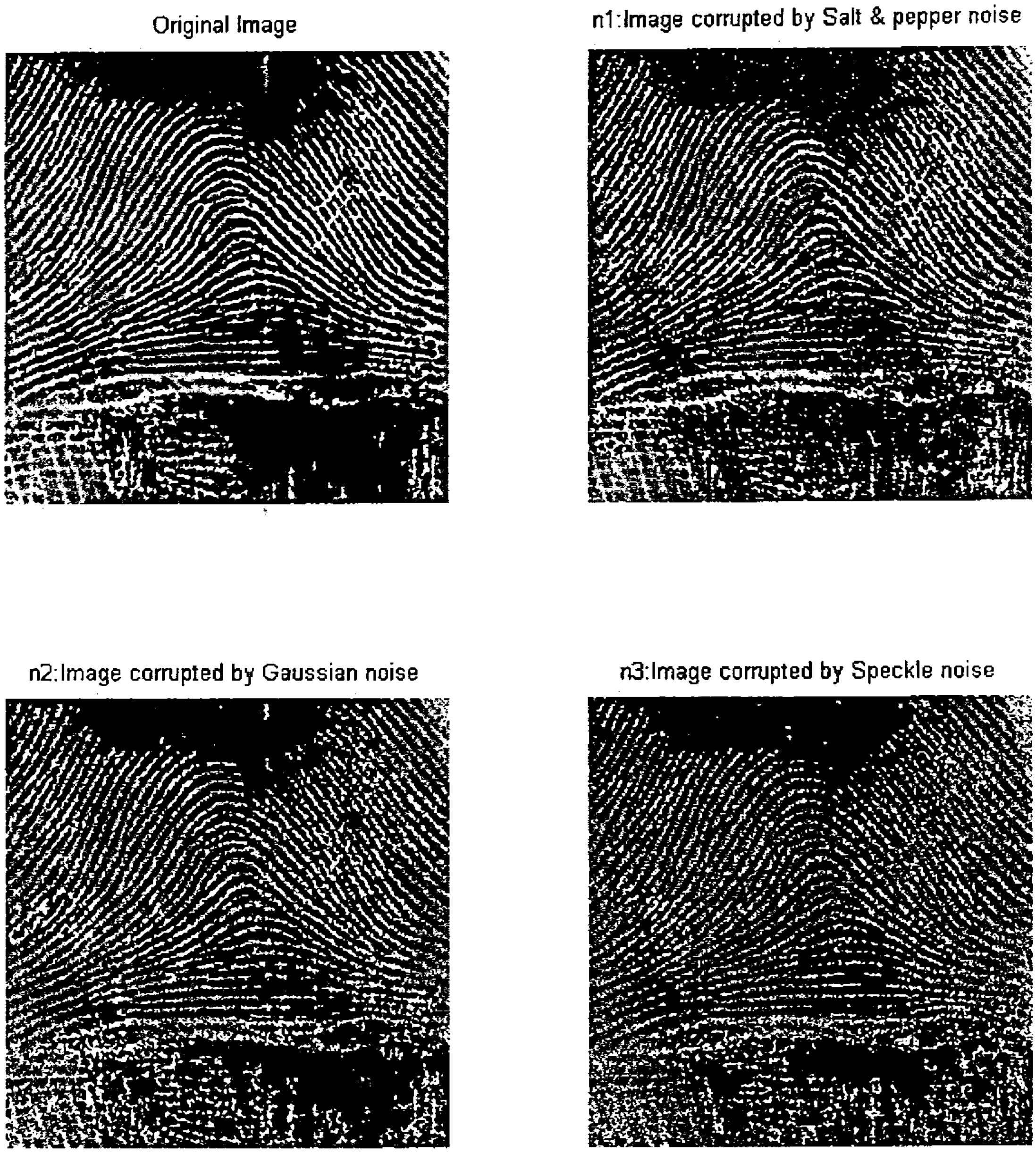
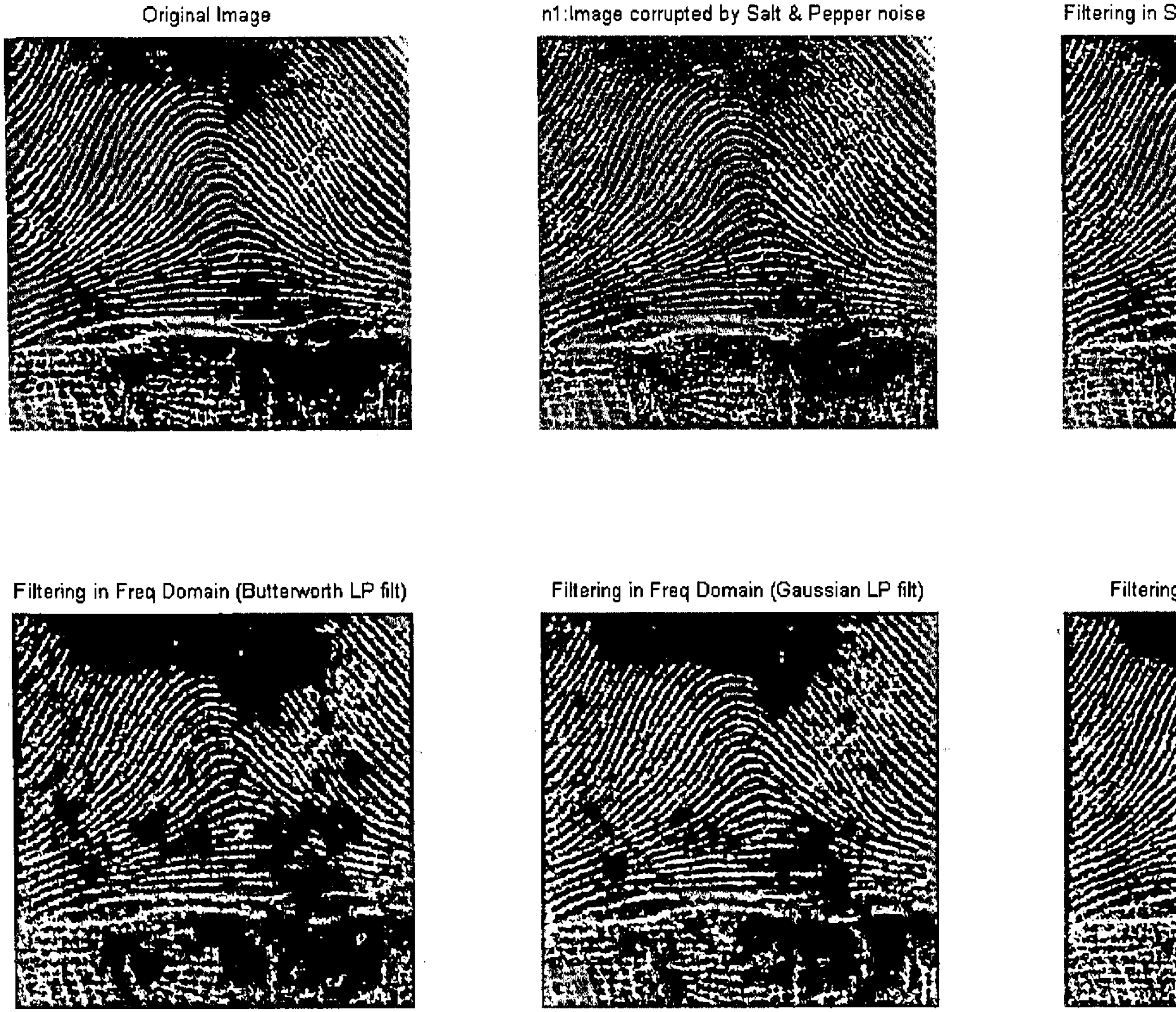
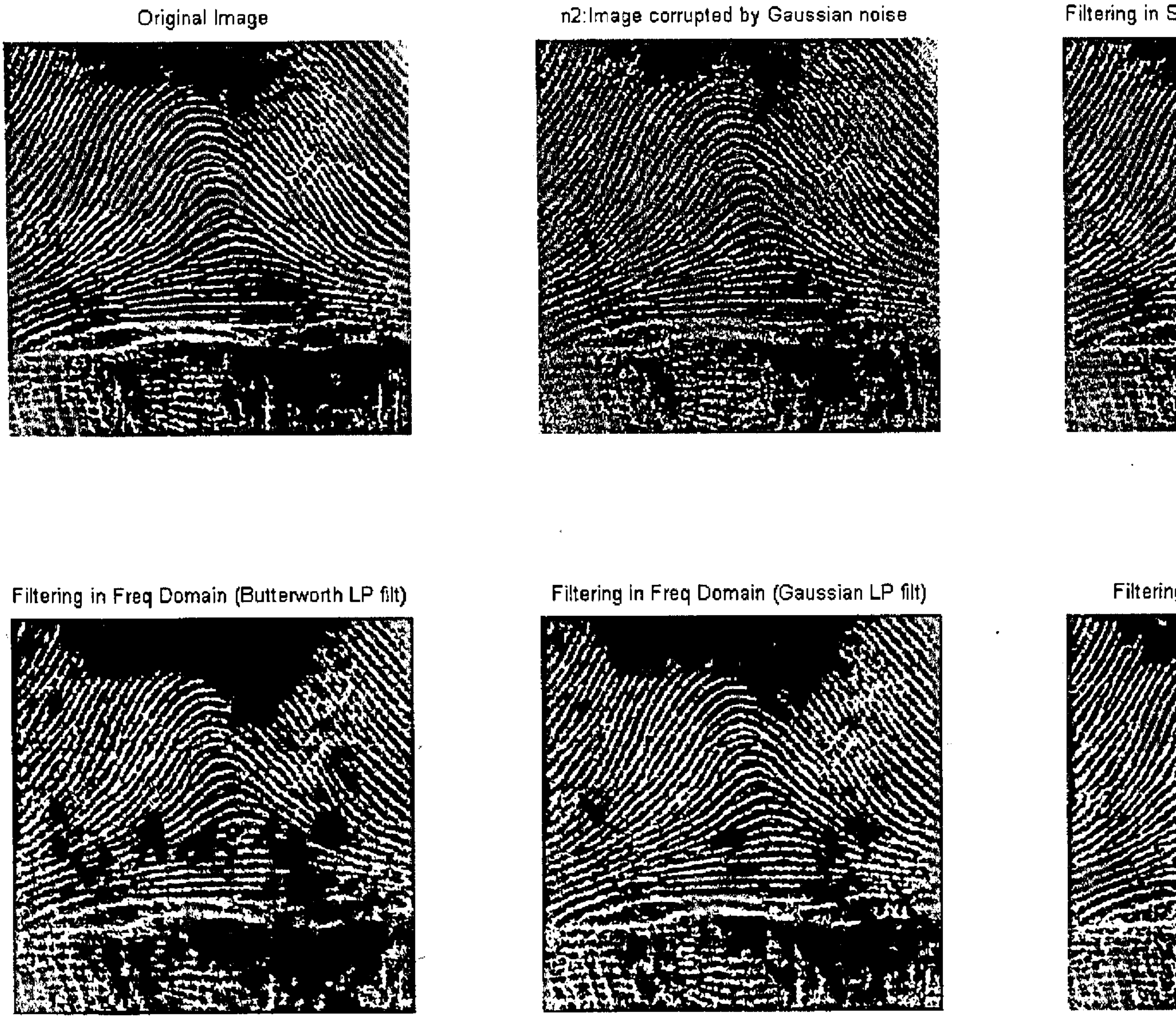
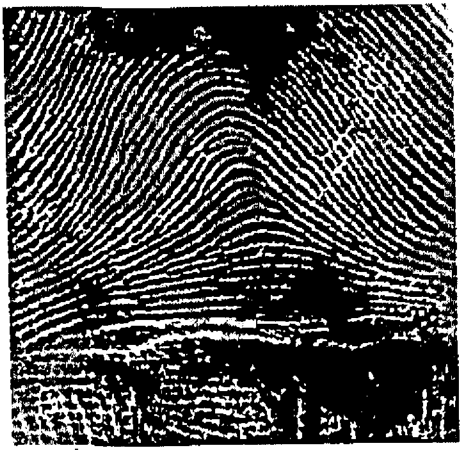


Figure 6.21: Original Image along with three Noisy Images

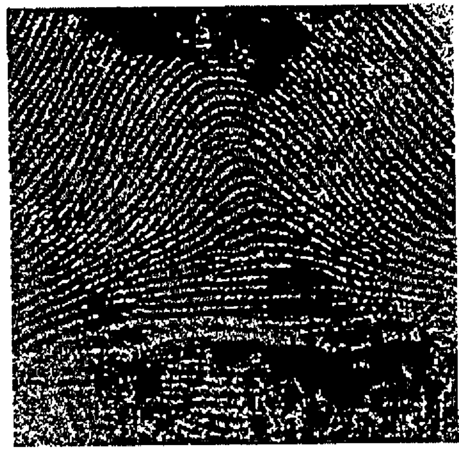

Original Image
n1:Image corrupted by Salt & Pepper noise
Filtering in S
Filtering in Freq Domain (Butterworth LP filt)
Filtering in Freq Domain (Gaussian LP filt)
Filtering


Original Image
n2:Image corrupted by Gaussian noise
Filtering in Freq Domain (Butterworth LP filt)
Filtering in Freq Domain (Gaussian LP filt)

Original Image

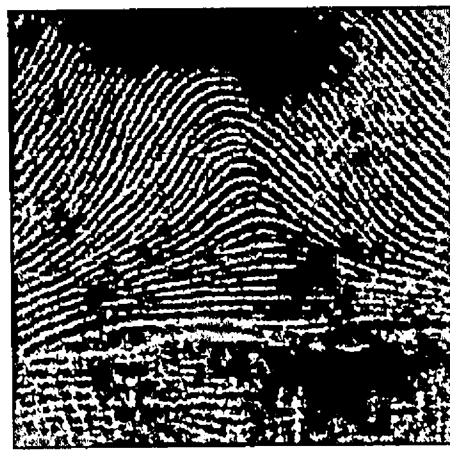


n3, the Image corrupted by Speckle noise



Filtering in Freq Domain (Butterworth LP filt)

Filtering in Freq Domain (Gaussian LP filt)



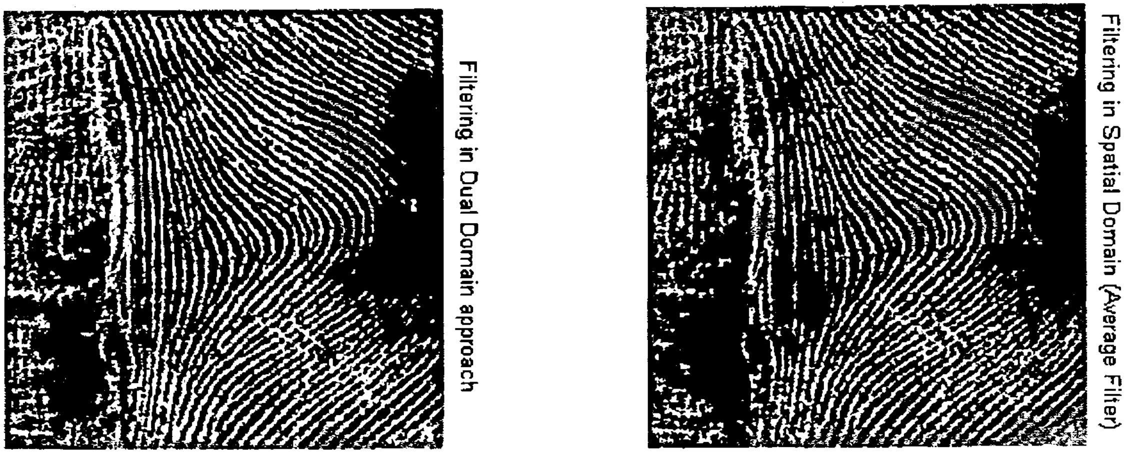


Figure 6.24: Noise Removal of n3

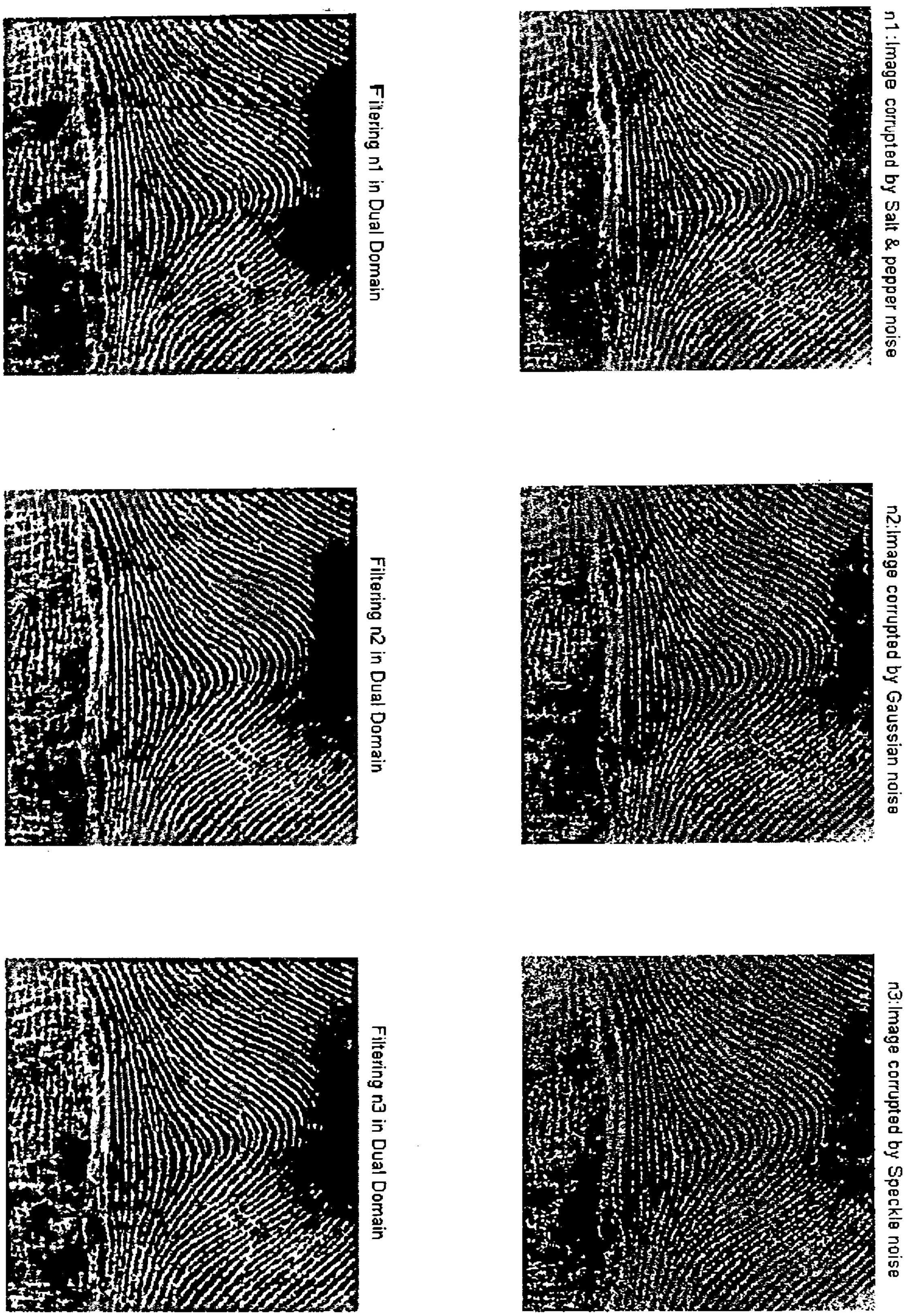


Figure 6.25: Filtered Images along with their noisy counterparts

6.2 Visual and Mathematical Analysis

Results are discussed in text and RMSE values have been displayed in the tabular form below for all images, namely, cameraman, house, hill, boat and fingerprint respectively.

Where,
n1: image with salt & pepper noise.
n2: image with Gaussian Noise.
n3: image with speckle noise.

NB: In the case of n1, n2 Gaussian filter is applied in the spatial domain while for n3 average filter is applied in the spatial domain; in all three cases n1 n2 and n3 Butterworth LP filter has been applied in the frequency domain and it is true for all Images.

RMSE values	n1	n2	n3
RMSE of the noisy image	0.12428	0.09552	0.10487
RMSE of the spatial domain	0.08300	0.06476	0.06147
RMSE of the frequency domain	0.09031	0.08860	0.08746
RMSE of the dual domain filtered image	0.06812	0.04287	0.04396

Table 6.1: Results of Image Noise Removal on Cameraman Image

As we see in table 6.1 in the case of cameraman image, it is image with medium amount of details in it and spatial domain is performing better over here than frequency domain. There are some concentration of local features lying horizontally behind the cameraman in the image which is the reason behind it but at the same time there is considerable amount of image area which has static shade or little varying shade. This reflects the fact that, subject to the parameter change, frequency domain might produce better results. While as we know that dual domain performs in both domains preserving local features along with global features, which is why it produces best results which are self-evident in all three cases of the noises.

RMSE values	n1	n2	n3
RMSE of the noisy image	0.11923	0.09942	0.11298
RMSE of the spatial domain	0.07758	0.06452	0.04527
RMSE of the frequency domain	0.05328	0.04866	0.05079
RMSE of the dual domain filtered image	0.04415	0.03382	0.03236

Table 6.2: Results of Image Noise Removal on House Image

On examining this particular image, it is clear it has less detail in terms of local features like lines or small objects; at the same time few varying shades of gray level. This all is reflected in the results of frequency domain performing better than spatial domain, as global features are heavy in this image. The only exception to this is the n3, image with speckle noise, which is due to the nature of speckle noise; the damage it does to the image is intense at the local details of the pixels so spatial domain performs better than the frequency domain. Concurrently, it is noticeable that the combination of both, dual domain is surpassing the performance in the case of speckle noise when compared with the other two, n1 and n2, in dual domain.

RMSE values	n1	n2	n3
RMSE of the noisy image	0.12172	0.09872	0.09223
RMSE of the spatial domain	0.07950	0.06430	0.04239
RMSE of the frequency domain	0.05050	0.04665	0.04753
RMSE of the dual domain filtered image	0.04201	0.02787	0.03508

Table 6.3: Results of Image Noise Removal on Hill Image

Hill image is medium detailed image with a balance in the contents of high and low frequency objects but with high frequency contents little bit prevailing the low frequency contents. As it can be seen in the results that frequency domain in the case of n1 and n2 is performing better but as we go on to n3 spatial domain exceed in performance only by a small fraction which is indicating the balance of the both local and global features in the image. At the top of the image, there is a constant shade area with little or almost no variation while in the middle of the image we can see some texture of houses adding to the local features. Final notable thing is in the case of n3 where dual domain is comparatively much nearer to the other domain results reflecting again the balance of the frequency contents in the image.

RMSE values	n1	n2	n3
RMSE of the noisy image	0.11914	0.09870	0.10695
RMSE of the spatial domain	0.07819	0.06468	0.04908
RMSE of the frequency domain	0.05744	0.05420	0.05479
RMSE of the dual domain filtered image	0.03492	0.02941	0.03574

Table 6.4: Results of Image Noise Removal on Boat Image

On visual inspection, it can be observed that both of these images Boat image and Hill image are identical in a sense that they have a balance in high and low frequency contents along with high frequency contents being little bit prevalent. Therefore, the above-mentioned analysis for Hill image also stands true in the case of frequency domain. Apart from this, in the case of dual domain for all three n1, n2 and n3, there was need of gamma being one while using intensity transformation functions i.e., adjusting image values for display purposes. This is all due to the large area of slowly varying gray shade. Any other value for gamma will cause faded small objects or could even cause complete disappearance of the small (local) features in the image resulting in the proportionate increase in the RMSE value. Lastly, a similarity between both Boat image and hill image is evident in their results of dual domain in the case of n2, which are the best when compared with n1, and n3 alluding towards the nature of Gaussian noise being a normal distribution over the whole image.

RMSE values	n1	n2	n3
RMSE of the noisy image	0.12173	0.09881	0.11325
RMSE of the spatial domain	0.08017	0.06475	0.05163
RMSE of the frequency domain	0.08910	0.08470	0.08676
RMSE of the dual domain filtered image	0.06426	0.03401	0.04404

Table 6.5: Results of Image Noise Removal on Fingerprint Image

This last image is an image of large detail with plenty of local features and contours all over the image. The local continuity and all other local features are very important in this image. The results of spatial domain are better than frequency domain naturally, as it can be seen in the image. Apart from the only one case in n1 where there is mere a difference of fraction, reflecting the very nature of salt and pepper noise in the damage it does to local continuity as it is affected by the spikes of white and black, in both other cases of n2 and n3 has attained marked results for spatial domain. Finally, dual domain achieves good results emphasizing the importance of both local features and global features even in this situation where local features are prevalent and as is the other situation where global features and high frequency are prevails.

It is evident both visually and mathematically in the results that the dual domain approach has achieved commendable performance and better RMSE values i.e., closer to the original image than either of the spatial or frequency domain.

6.3 Graphical Analysis

The results are supplemented with graphical analysis for the comprehensiveness of the work. One bird's eye view would enable us to see the performance of the dual domain and its respected PSNR in a specific type of noise. For all three cases of noises, namely, n1: image corrupted with salt and pepper noise, n2: image corrupted with Gaussian noise, and n3: image corrupted with speckle noise, separate graphs have been drawn.

One thing should be kept in mind is that the graphs are displayed according to the noises in order to get the overview of the performance of dual domain approach in the case of different noises

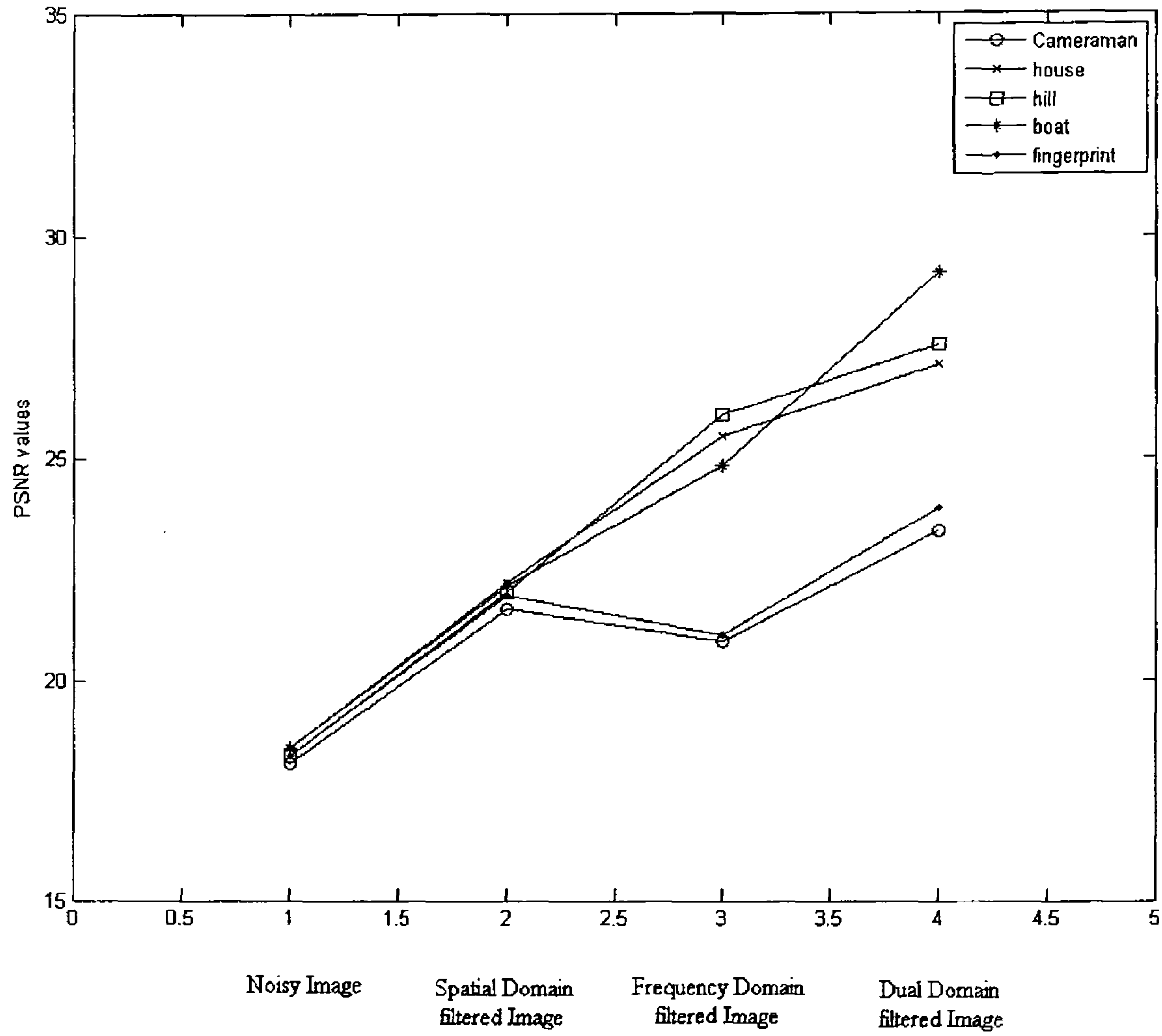


Figure 6.26: Results of Salt and pepper noise shown for all images

On observing we can see the PSNR value in the figure 6.26 for spatial domain filtering are very close to each other while in the frequency domain and in dual domain medium and high detail images of cameraman and fingerprint are close together and much lower than the other three images. Another noticeable difference is that in case of fingerprint and cameraman, frequency domain PSNR is surpassing spatial domain which is exactly the opposite of other three images.

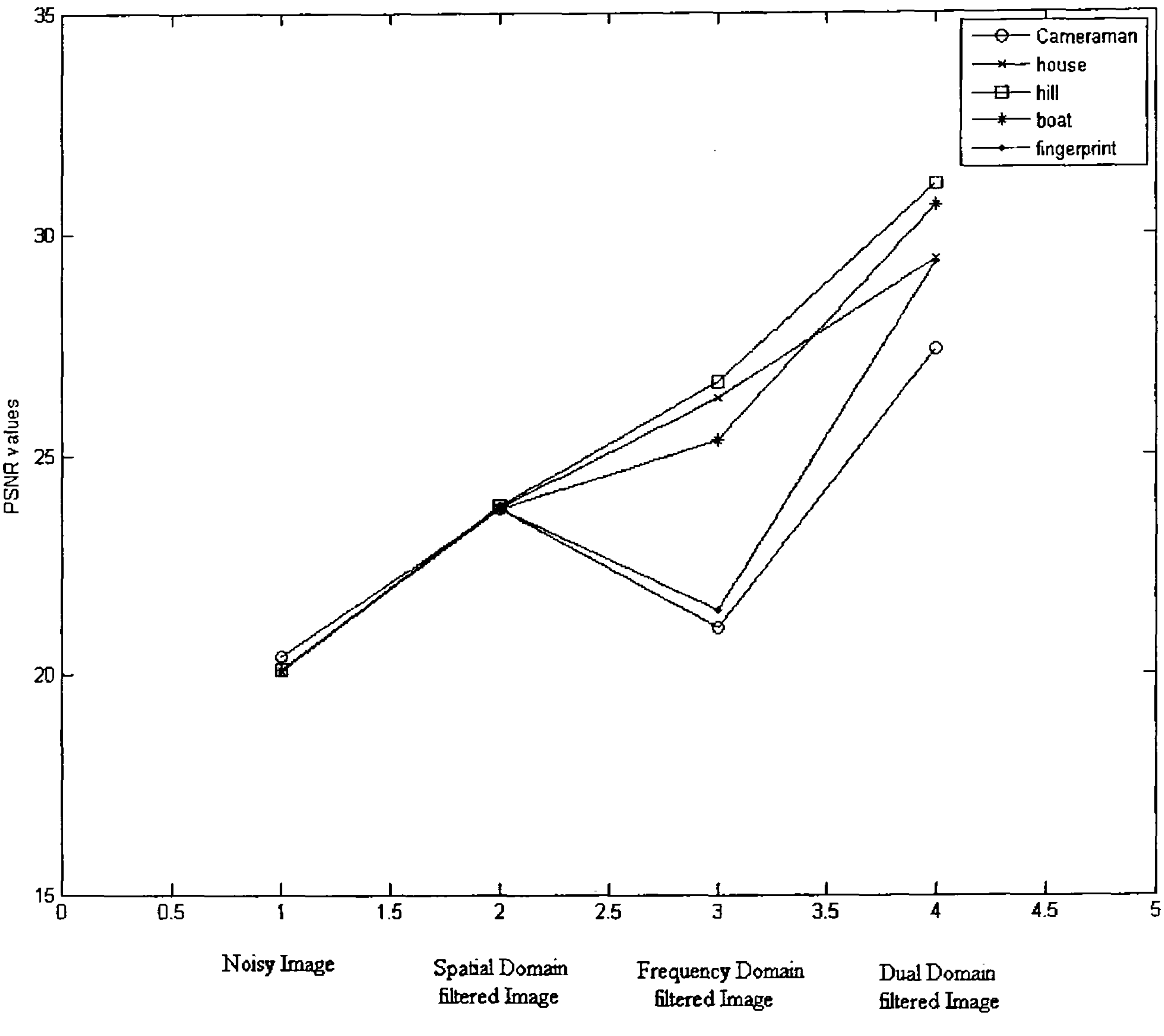


Figure 6.27: Results of Gaussian noise shown for all images

The most notable thing in the graph in figure 6.27 is convergence of all images in the case of n2 at spatial domain. So it might be concluded from this graph that in the case of Gaussian noise performance in terms of PSNR values of spatial domain filtering is same for all types of images. Furthermore, the trends in all different kind of image can be analyzed. Very interestingly, house at one extreme end of having less detail and fingerprint at the other extreme end of high detail are converging in their performance in dual domain approach reflecting the dual domain characteristic of considering both high and low frequency contents.

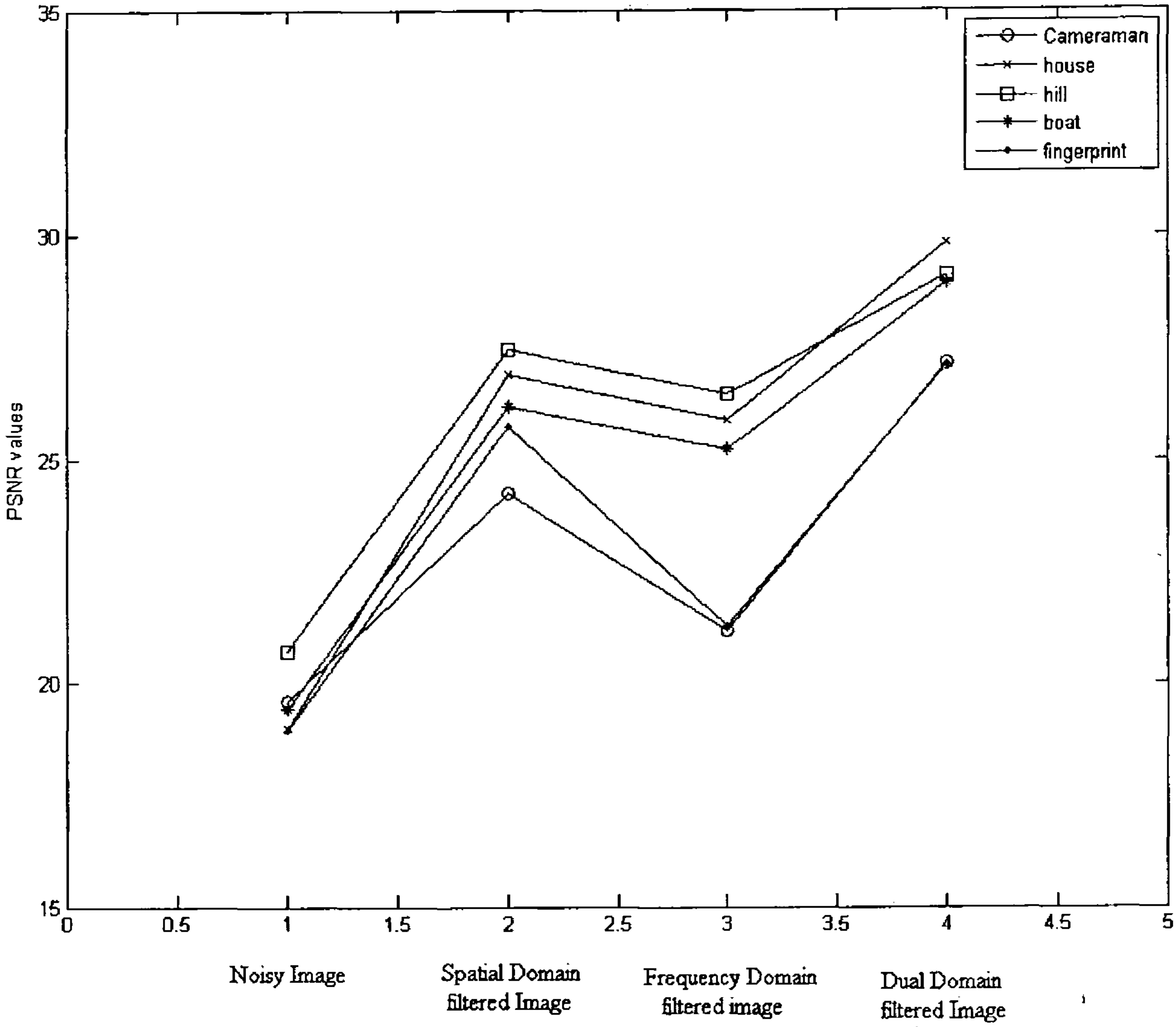


Figure 6.28: Results of Speckle noise shown for all images

The same trend, as in salt and pepper noise, can be seen in this type of noise only in the case of dual domain and frequency domain while spatial domain PSNR values are far apart exhibiting the effect of speckle noise on the images, as seen that the performance of spatial domain is better than frequency domain regardless of the types of images.

A commonality shared by all results is that in the case of dual domain the performance of always outstanding, depicting the usefulness of dual domain in preserving both global and local features alike.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENT

7. Conclusion and Future Enhancements

As the problem of noise being ubiquitous and inevitable in the process of acquisition (depending upon the circumstances), image noise removal has always been a hot topic of research.

During the acquisition process variable light effects on different parts of digital image, insufficient lighting conditions, electrical or electromechanical interference causes unwanted pattern of noise in the image. This noise becomes noticeable in large images more particularly and is undesirable for aesthetic reasons or for further processing in the broad spectrum of Computer Vision, such as segmentation where the separation of region from the background is necessary or edge detection where separation of the edges to mark off the boundaries or to do object recognition is of interest. Thus for whatever reason, it is highly desirable for image to have some noise removed before any further processing can be carried out.

So, whether be it for computer vision or aesthetic purposes, the noise removal problem is universal and understandable in its severity by a wide variety of people belonging from diversified fields.

7.1 Conclusion

The algorithm of dual domain approach has been described. While most existing algorithms have worked in isolation with spatial domain or frequency domain, this algorithm works in dual domain making it viable to harness advantages from both domains. Earlier work in dual domain has been based on POCS (projection onto convex sets) addressing a special case of contiguous noisy pixels appropriate with a need of having to specify a sub image mask in order to restore the image naturally. [13] In this algorithm, work has been carried out on three different kinds of noises in relation to each other along with the comprehensive comparison at the end. Results made it evident that utilizing dual domain approach has performed better for image noise removal.

Owing to increasing fame of Wavelet domain, this area of research has been not tapped for quite long. The purpose and source of motivation behind writing this paper was a sincere effort to revive this useful approach of dual domain in new perspective.

As evident from the results, with the critical choice of operation in both domains, the dual domain algorithms has, firstly, maintained sharpness and secondly, maintained continuity of features across the noisy region of the images better than either of the other domain.

7.2 Future Enhancements

Considering future extensions and enhancements in the dual domain approach, there are four main areas which may be worked upon. Firstly, the algorithm can be extended in its application on colour images and act as a viable tool in real world situations where noise removal is desired for aesthetic purposes.

Secondly, In today's world of multimedia where video is considered as inseparable part of our lives, an extension of dual domain approach into that arena would be met with high appreciation and applause.

On the other side, varied types of noises may be considered as well to incorporate broad variety of solutions and situations encompassing our real world needs. This extension would make dual domain approach a more comprehensive one.

Finally, three dimensional graphics, a hot topic of research and development is an another identifiable state-of-the-art technology that dual domain approach should be extended to. Its extension to 3D is considered as profitable in terms of monetary value due to the high demand of 3D in upcoming and future generations.

APPENDIX A

REFERENCES AND BIBLIOGRAPHY

References and Bibliography

Books & Thesis

1. "Digital Image Processing"; Rafael Gonzalez, Richards Woods; Second Edition; Published by Pearson Education, Singapore, 2001.

2. "Practical Computer Vision Using C", J. R. Parker, ISBN 0-471-59262-5, 2003.

3. "Pattern Recognition: Statistical, Structural and Neural Approaches"; Robert J. Schalkoff; First Edition; Published by John Wiley & Sons Inc.; 1992.

4. "Fundamentals of Computer Vision"; Mubarak Shah; University of Central Florida; 1997.

5. "Digital Image Processing"; Kenneth R. Castleman; Pearson Education; 2007.

6. "Digital Image Processing using MATLAB"; Rafael C. Gonzalez, Richard E. Woods and Steven L. Eddins; Pearson Education, 2004.

7. "Fundamentals of Digital Image Processing"; Anil K. Jain; University of California; Pearson Education, 2005.

8. "Effective Segmentation of Veins after Tackling Non-Uniform Illumination from Coronary Angiogram Images"; Muhammad Nadeem; International Islamic University; MS Thesis, 2006.

Publications

9. J.-S. Lee, "Digital Image Enhancement and Noise Filtering by Use of Local Statistics", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, pp. 165-168, 1980.

10. T-H. Yu and S. K. Mitra, "Unsharp masking with nonlinear filters," Proc. Seventh European Signal Processing Conf., EUSIPCO'94, Edinburgh, Scotland, September 1994, pp. 1485-1488.

11. FERREIRA, P.J.S.G. "Interpolation and the Discrete Papoulis-Gerchberg Algorithm", IEEE Trans. Sig. Proc., 42, No. 10, 2596-2606, Oct 1994.

12. A New Algorithm for Image Noise Reduction Using Mathematical Morphology by Richard Alan Peters II, Department of Electrical Engineering, Vanderbilt

University, Nashville, TN 37235 IEEE Transactions on Image Processing Volume 4, Number 3, pp. 554-568, May 1995..

13. Combining Frequency and Spatial Domain Information for Fast Interactive Image Noise Removal by Anil N. Hirani, and Takashi Totsuka, Proc. Siggraph, 1996, Sony Corporation, Tokyo, Japan, Publisher: ACM New York, NY, USA.

14. Fourth-Order Partial Differential Equations for Noise Removal by Yu-Li You, Member, IEEE, and M. Kaveh, Fellow, IEEE, published in IEEE Transactions on Image Processing, Vol 9, NO. 10, October 2000.

15. Noise Removal for Degraded Image with Impulsive Noise Using Median Filters and Neural Filters by Yamashita Noritaka cd, Lü J, Sekiya Hiroo, and Yahagi Takashi, Transactions of the Institute of Electrical Engineers of Japan. C, Volume. 123, No. 6, 2003, Japan.

16. http://www.mediachance.com/pbrush/help/noise.html

17. http://en.wikipedia.org/wiki/Noise_reduction

18. http://dukemil.egr.duke.edu/Ultrasound/k-space/node5.html

19. http://www.dfanning.com/ip_tips/freqfiltering.html

20. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT5/node4.html