ADAPTIVE FILTERS FOR EDGE-PRESERVING SMOOTHING OF SPECKLE NOISE IN DIGITAL IMAGES

MARZIA RABBI ZAMAN

Adaptive Filters for Edge-Preserving Smoothing of Speckle Noise in Digital Images

By

©Marzia Rabbi Zaman

A thesis submitted to the School of Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Engineering
Faculty of Engineering and Applied Science
Memorial University of Newfoundland
December, 1992

St. John's Newfoundland Canada

ISBN 0-315-82617-7

Abstract

Speckle is a common phenomenon in all types of coherent energy imaging system such as Synthetic Aperture Radar (SAR), laser, ultrasound, acoustics, sonar etc. Speckle is a multiplicative-convolutional noise and as such is different from other commonly found types of noise such as Additive White Gaussian Noise (AWGN). Hence different methods of processing are required to restore speckled images. Moreover, in many applications, the edge structure of an image is very important, and usual filtering methods are not well suited for preserving edges particularly in speckled images. In this work, an extensive study has been made to investigate the applicability of different existing nonlinear filtering methods and also a new Quadratic Volterra Filter (QVF) based on speckle-model to solve the problem of speckled image restoration in terms of noise smoothing and edge preservation. Edge detection itself on speckled images is a major problem which has not been addressed by many researchers. This thesis attempts to provide a better approach to the solution of restoring images corrupted by speckle while preserving their edge information.

Acknowledgements

I would like to express my deep gratitude and thanks to my supervisor Dr. Cecilia Moloney for her guidance, useful discussions, criticisms and encouragements for my work throughout the program and also for the financial support she has provided me. I would like to thank the School of Graduate Studies, all faculty and staff members specially the C-CAE staff members and also all fellows graduate students.

Personally, I would like to thank Dr. J. Sharp, Associate Dean, Graduate Studies, Faculty of Engineering, for his encouragements and support specially near the end of my M.Eng. program. Thanks also go to Dr. T. Chari, the former Associate Dean, Graduate studies, Faculty of Engineering, for his kind consideration and assistance.

I would also like to thank Prof. M. A. Rahman and other CIDA officials related to MUN/CIDA/BIT project under which I initially started my Masters program.

I sincerely acknowledge Mr. Tony Parsons, a graduate of Electrical Engineering, Faculty of Engineering, MUN, for the X-image program he has developed during his final year project and also Mr. Dipankar Bhattacharya, a former graduate student, Faculty of Engineering, MUN for the useful modifications he made on the original X-image program.

I would also like to acknowledge all examiners of my thesis for their useful comments and suggestions.

Finally I thank my parents for all encouragements and help I got from them during the course of my study.

Contents

List of Figures

List of Tables

List of Abbreviations

pdf : probability density function

std : standard deviation

AWGN : Additive White Gaussian Noise

CFAR : Constant False Alarm Rate

COV : Coefficient Of Variation

GSL : Gaussian Smoothed Laplacian

HVS : Human Visual System

LMS : Least Mean Square

LOG : Laplacian Of Gaussian

LR : Likelihood Ratio

MAP : Maximum A *Posteriori*

MROA : Modified Ratio Of Averages

MSE : Mean Squared Error

Pd : Probability Of Detection

Pfa : Probability of False Alarm

PSF : Point Spread Function

QVF : Quadratic Volterra Filter

RGOA : Ratio and Gradient Of Averages

ROA : Ratio Of Averages

SAR : Synthetic Aperture Radar

SNR : Signal to Noise Ratio

List of Important Symbols

$\otimes$	: "Kronecker" product of matrices
$tr\{.\}$	: trace of a matrix
$\{.\}^T$	: transpose of a matrix
$E[.]$	: expected value operator
$dir((i,j),(k,l))$,	: direction of two pixels, (i,j) and (k,l)
$dist((i,j),(k,l))$	: distance between pixels (i,j) and (k,l)
Λ	: set of independent quadratic coefficients of the QVF
Λ_0, Λ_1, Λ_2	: independent quadratic coefficients for Type 0, Type 1 and Type 2 QVF, respectively
C	: "correct" edge factor
F	: "false" edge factor
G	: gradient magnitude
G_i	: gradient magnitude for the ith set of pair of the neighbourhoods
H_1,H_2	: linear and quadratic coefficient matrix of the QVF, respectively
L	: number of independent looks
M	: "missed" edge factor
N	: window size
N_q	: total number of pixels subjected to quadratic operation
R	: ratio magnitude
R_i	: ratio magnitude for the ith set of pair of the neighbourhoods
S	: region of support of the QVF
T	: threshold
T_g	: gradient threshold

T_r	: ratio threshold
W	: "wrong" edge factor
W_0,W_1,W_2	: weights assigned for Λ_0, Λ_1 and Λ_2, respectively
Y_1	: Input matrix for the QVF
Y_2	: $Y_1 \otimes Y_2$
σ_x,σ_n	: standard deviation of the original image and noise process, respectively
a,b	: power coefficients for two point Taylor filter
h	: PSF of the imaging system
h_0	: constant term of the QVF
$\bar{h}_1$, $\bar{h}_2$	: linear and quadratic operators, respectively
h_1, h_2	: linear and quadratic coefficients, respectively
n	: noise process
ι	: scaling factor
$x(i,j)$	: original or un-corrupted image at pixel coordinate (i,j)
$\bar{x}(i,j)$, $\hat{x}(i,j)$	: local mean and estimated value of $x(i,j)$, respectively
$y(i,j)$, $\bar{y}(i,j)$	: noisy image at pixel location (i,j) and its local mean, respectively

Chapter 1

Introduction

1.1 General

A major task in digital image processing is restoring noisy images. Images often become corrupted with different types of noises during their formation. The main aim of image restoration is to reduce the effect of noise as much as possible and thereby to produce an estimate of an image what would have been produced had the imaging system been ideal. There are many techniques for image restoration, based on different criteria and depending on different types of noises. Not only is it impossible to develop a general technique which would be able to restore an image exactly, but it is also very difficult to establish a technique which results in an image estimate reasonably close to the original. The problem is substantially more difficult when the noise is significant and/or non-additive in nature. There are several methods that have been developed to reduce the effect of multiplicative noise in images, but there is still a wide scope for further research in this area.

1.2 Motivation

The purpose of this research is to focus on the problem of restoring digital images from speckle-degraded versions. Speckle is a noise which occurs due to the physics of coherent imaging systems such as those based, for example, on laser, ultrasound or synthetic aperture radar (SAR) imaging. Speckle noise may be modelled as a convolutional-multiplicative process, and is definitely non-additive. Linear signal processing is well established for the case of additive noise but the problem arising from multiplicative noise or convolutional noise may be better suited by nonlinear methods which has not yet been fully developed. Volterra filters have been used in several nonlinear applications in digital signal processing [8, 21, 22, 50]. The quadratic Volterra filter has been used for different image processing applications such as image restoration, enhancement, edge extraction etc. [44, 49, 50]. Images corrupted with multiplicative noise have been restored using filters based on Taylor series approximation [37] which work quite effectively. In terms of its input/output relationship, a Volterra filter may be considered as a Taylor series with memory, the Volterra filter might be able to give better results for the same problem. Although, speckle noise has been studied in previous digital images processing research [16, 19, 24, 25], the problem of speckle smoothing while preserving the edge structure in digital image has not been studied. The research of this thesis is carried out with an aim to restoring speckled images using the quadratic Volterra filter which will trade off smoothing of speckle noise and preservation of edge structure. The quadratic Volterra filter is compared to other filters which have been previously proposed for the smoothing of multiplicative noise.

1.3 Problem Complexity

The complexity involved in the above mentioned problem arises from two major factors. First of all, nonlinear signal processing is difficult because there is no comprehensive and complete theory for nonlinear systems as these exist for linear systems. Second, the signal to be dealt with is a two dimensional signal and has several peculiarities. Some of the characteristics of digital images which make them difficult to treat as signals are the following :

- Digital images are two-dimensional signal.
- They are non-stationary, which may mean that locally adaptive or spatially-variant processing method is required.
- They are random in nature, so that the processing of such images may need to be based on their statistical properties which are, in general, not well defined nor easily estimated. Moreover, estimation of image statistics requires a large number of data which may not be available in a spatially-varying image or when the processing is meant to be locally adaptive.
- When images are processed for human viewers, the human visual system (HVS) becomes an important consideration in the development of an image restoration technique. There is a number of different existing models of the HVS which have nonlinear components. Hence, standard techniques such as the Mean Square Error (MSE) for performance measurement are not always adequate in judging the visual quality of the images. As a result, any general signal estimation techniques based on such standard performance measures (e.g. Least Mean Square (LMS) adaptive technique) may need to be modified to use in image

restoration algorithms.

1.4 Approach to the solution

The approach to the solution of the stated problem involves the following three steps.

- Modelling speckled images
- Designing speckle-specific filters
- Evaluating the performance of the proposed filters

It is necessary to have a realistic model of speckle to design a model-based filter. A realistic model has been developed for multi-look SAR images based on previous work [19] and the accuracy of the model has been verified. Synthetic images using the model have been created for use in testing the designed filters and other existing filters. Filters based on the quadratic Volterra series, have been designed which are made locally adaptive according to the statistical properties of speckle. To demonstrate the performance of the filters, evaluation measures have been used which focus particularly on their noise smoothing and edge preservation qualities. Also, comparisons with other well-known filters have been carried out.

1.5 Organization of the thesis

This thesis has been organized as follows. Chapter Two provides background information on four major areas related to the stated problem - i) speckle and especially SAR speckle, ii) nonlinear filtering methods for image processing, iii) the application of Volterra filters particularly in image processing, iv) edge detection algorithms particularly for speckled images. Chapter Three provides i) descriptions of the speckle

model, ii) different filtering methods including basic background on the Quadratic Volterra Filter (QVF), all of these are used later in the comparative study, iii) different speckle-specific edge detectors and finally iv) some performance criteria for evaluating filter performance. Chapter Four focusses on the proposed methodology for i) speckled SAR image modeling, ii) the design of QVFs for speckled image restoration, intended for both speckle smoothing and edge preservation and also iii) a new speckle-specific edge detector with an edge-map thinning algorithm. Results obtained from basic experiments with the QVF and from the comparative study are presented in Chapter Five. Finally, conclusions based on the results obtained and suggestions for future work are provided in Chapter Six.

Chapter 2

Literature Review

2.1 Introduction

This chapter is grouped into four major sections. The second section provides a brief survey of the pr· ·ious work on speckle-noise modeling, simulation and filtering techniques. This section also provides a brief background information on SAR image processing mainly from the point of view of the speckle smoothing. The third section is an overview of different existing nonlinear filtering methods and their importance in image restoration. The next section discusses the application of quadratic Volterra filters in various fields, particularly in image processing applications. A brief survey of different edge detection methods is presented in the fifth section. Since edge detection for speckled images is of an important consideration in this study, special attention is paid to speckle-specific edge detection methods.

2.2 Speckle Noise

Speckle is a phenomenon which occurs in images produced by all types of coherent or partially coherent imaging system such as synthetic aperture radar (SAR), laser, acoustic, ultrasound, etc. The effect of speckle noise on images is generally not desirable because it degrades images in such a way that it may be difficult to extract useful information from these. There are many image restoration and enhancement techniques used for either removing speckle from the images or emphasizing the informative aspects of images. To design an optimum image restoration technique, it is necessary to have an appropriate mathematical model of the speckle noise based on its statistical properties. Speckle properties are discussed in several literatures [5, 63], and in particular, the statistical properties of SAR speckle are well described by Arsenault *et al.* [5, 6]. It is well known that speckle is a signal-dependent noise [16, 19, 23, 28], such that its magnitude depends on the intensity of the underlying image. Lee's speckle smoothing algorithm [28, 30] for SAR images is based on employing the sigma probability of the Gaussian distribution of the image intensity. Frost *et al.* [16] developed a mathematical model for speckle and designed an adaptive filter for multiplicative noise. Although most of the literature in this area assumes that speckle is a pointwise multiplicative noise, there are cases where this assumption is not satisfactory [62]. The model derived by Kuan *et al.* [24] takes speckle-correlation effects into account where an adaptive noise-smoothing filter and a Maximum a *Posteriori* (MAP) filter have also been developed. Raney has made an extensive study of speckle and reviewed speckle modelling and simulation techniques in the literature [51]. SAR image restoration has been attempted by Jin *et al.* [20], where the segmentation and classification of SAR images are often very difficult due to the presence of speckle.

These problems have been addressed by different authors [15, 19, 25]. Hudson and Jernigan [19] proposed a model for SAR images corrupted by speckle and compared the performance of different filters in terms of their ability in texture classification on the filtered images. The model they proposed is based on the statistical property of one-look speckle as well as the impulse response of the radar which in fact causes the spatial correlation of speckled images. The model has been verified and may be considered a valid model for speckle corrupted SAR images.

2.3 Nonlinear Filtering

Linear systems have found extensive use in signal processing because they are relatively simple to analyze, design and implement. Because of their simplicity, it has also been possible to establish generalized theories of linear systems. But there are certain situations that require nonlinear processing of signals. Because, nonlinear systems are relatively difficult to design and implement, it has become usual to assume many practical systems to be linear systems and to apply linear systems theory. Such approximations are sometimes reasonable but there are cases in which performance quality is substantially reduced and in such cases, nonlinear processing may become unavoidable.

2.3.1 Nonlinear system theory

In 1968, Oppenheim *et al.* [41] made an extensive study with a view of providing a generalized nonlinear systems theory such as there exists for linear systems. They were successful in using the concept of generalized superposition to develop homomorphic filtering. Moloney [37] used the generalized superposition principle for the

problem of image restoration in the case of images degraded by pointwise multiplicative noise.

2.3.2 Application of nonlinear filters

Oppenheim *et al.* [41] described the generalized superposition principle for nonlinear filtering of convolved and multiplicative signals with two examples of its practical application, namely, i) audio dynamic range compression and expansion, and image enhancement; ii) echo removal and speech analysis.

Mathews [34] provides quite a broad overview on applications, current research trends, results, problem areas etc. regarding nonlinear filters where a lattice structure for nonlinear adaptive filters using truncated Volterra series has also been discussed.

2.3.3 Nonlinear filters in image processing applications

As mentioned above, nonlinear filtering has been used in image processing applications. One of the major reasons for using nonlinear processing of images is that the human visual system is known to have nonlinear components [3, 58]. There are other reasons why it becomes almost essential to process images using nonlinear filters. One such situation occurs in restoring noisy images. Images often get corrupted with different types of noises due to the poor imaging media, non-ideal imaging systems etc., and noise may be combined with the image either linearly or in some other fashion. But even if it is linear, nonlinear filtering is often recommended because of the non-stationary characteristics of many images. To enhance image details, specifically in preserving edges, nonlinear filtering may be required. So it is easily argued that it is important to restore images by nonlinear methods when the noise itself is not linearly combined with the signal.

At this point, it is clear that nonlinear filtering plays a very important role in image restoration. Abramatic and Silverman [1] modified the Wiener approach to the problem of restoring images degraded by white Gaussian noise. They paid special attention to preserving the edges which is another important consideration in image restoration. Quantitative and qualitative comparisons between six well-known adaptive filters used for image restoration was made by Mastin [33]. Chin and Yeh [12] made a similar kind of study with commonly used filters. Nagao and Matsuyama [39] suggested methods for the difficult task of trading off between noise smoothing and edge preservation.

Lee's multiplicative filter [27, 29], designed for a pointwise multiplicative noise model, is one example of such filters. His algorithm is based on local statistics of images and it produces very satisfactory results. Homomorphic filtering [4] is also an effective method of restoring images corrupted by pointwise multiplicative noise. Teklap and Palovik [59] also investigated several techniques for restoring images that are degraded with pointwise multiplicative noise caused by sensor nonlinearity. Moloney and Jernigan [38] developed a new approach to the solution of the same problem using the concept of multiplicative superposition. The developed algorithm is adaptive because it uses the local statistics of the image with the optimal operators calculated using Taylor series approximations.

Nonlinear filters using truncated Volterra series have been also developed and used in different applications including image processing which will be discussed next.

2.4 Volterra filters

It has almost been a century since Volterra introduced his series, which has been found to be a very useful tool for describing input/output relationships of nonlinear systems. In recent years, it has been extensively used in different areas to handle nonlinear problems. The main reason for its popularity is that it introduces a generalized theory for nonlinear systems in a relatively simple and conceptually manageable way.

2.4.1 Volterra series theory and application

Biglieri [7] discussed the nonlinear Volterra processor covering its general theory, implementation and some applications, namely, i) Mean Square Error (MSE) prediction of discrete-time random processes and ii) identification of nonlinear systems with memory.

System identification by second order nonlinear Volterra filters, has also been studied extensively by Koh and Powers [22]. The solution for the optimum filter, iterative factorization to facilitate the implementation of the filter, LMS adaptive algorithms etc. have been covered in this literature. The practical application they were interested in, was to utilize the second order Volterra filter to model and predict the dynamic behaviour of moored vessels under random sea waves. Sicuranza and Ramponi [55] also proposed an LMS adaptive algorithm for second order Volterra filters using distributive arithmetic. The main concern of their work was to provide an efficient and simple realization of the filter which could be well suited to modern technologies e.g. VLSI technology. Chiang *et al.* [11] handled the same problem using a different approach, with an implementation based on matrix decomposition. By extending the Kalman filtering approach, a fast-response second-order adaptive

Volterra filter was proposed by Davila *et al.* [13]. Nonlinear system identification has been carried out using truncated Volterra series models by several workers [22, 45], but all of them assumed Gaussian noise as the system input. Kim and Powers [21] made an attempt to solve the problem of system identification assuming generalized random input and using a truncated Volterra series model. Adaptive noise cancellation [57] for voice band data transmission, cancellation of inter-symbol interference [8] and echo cancellation [2] are three similar applications where Volterra filters have been found to be effective.

In most signal processing applications, the signals are one-dimensional, usually a function of time. Hence, work that has been discussed so far, has all been carried out for 1-D signals. However, realization of higher dimensional quadratic filters have been dealt with in [36, 56]. Mertzios *et al.* [36] proposed matrix decomposition techniques for realizing 2-D filters.

2.4.2 Volterra filter in digital image processing

Since it is very important in image processing to deal with nonlinearity, the The Volterra filter seems to be a possible choice for the nonlinear filtering of images. Nonlinear processing itself is very complex and since an image is a two-dimensional signal, nonlinear processing of images is more complex. However, considering a Volterra series only up to the quadratic term, it is feasible to realize 2-D quadratic Volterra filters. Work that has been done previously using Volterra filters in digital image processing applications has been based on Volterra series truncated up to the first nonlinear term or the quadratic term and hence the filters derived are termed quadratic Volterra filters.

2.4.3 Application areas

Since this is still a new field, there is little literature on the application of Volterra series in image processing. One of the major studies in this area has been done in 1990 by Ramponi [50]. So far, his paper could be taken as a framework providing a background of the properties and design of quadratic isotropic 2-D Volterra filters. The use of Volterra filter seems to be quite promising in both cases of additive and multiplicative noise reduction [50]. Design examples in the same literature showed how different situations could be dealt with using the Volterra filter for image restoration. The enhancement of images is often of great interest. As mentioned by Ramponi [44], most of the image enhancement techniques have used luminance scale modification after processing the image through linear or nonlinear filters. But this does not improve the quality of an image, particularly if the image contains a significant amount of noise. Because, in this situation, noise smoothing is done at the cost of losing the image details, an optimization between noise smoothing and preserving image details becomes a very important issue. A recent approach made by Ramponi [44] uses a luminance independent operator. The filter he used is a Volterra series truncated up to the first nonlinear term. Edge extraction is one of the important tasks in image processing which also needs nonlinear processing. Ramponi [49] also proposed quadratic Volterra filters for this purpose.

2.4.4 Quadratic Volterra filter design

The main problem regarding the design of quadratic Volterra filters is the complexity particularly for 2-D signals such as images. A 2-D quadratic filter has both a set of linear coefficients and a set of quadratic coefficients. The number of filter coefficients

for the quadratic part is quite large and hence difficult to design (e.g. even for a 3×3 support 2-D Volterra filter, the number of linear coefficients and quadratic coefficients are 9 and 81, respectively). But as shown by Ramponi, [50], these large numbers may be reduced by imposing certain conditions such as isotropy, symmetry, uniformity in gray level etc. But undoubtedly, it is difficult to find a straight forward design strategy such as that exists for finite support linear filters. Because of this inherent complexity of the problem, no such design technique has yet been available which is based entirely upon the theoretical properties of the Volterra series. Some literatures [46, 47, 48] were reported which give a rough idea about the design approach one can follow. The main strategy is to define the designing problem, formulate an objective function based on that, then minimize the function and thereby design the filter coefficients. An optimization approach has been suggested by Ramponi and Ukovich [47] where three existing nonlinear optimization techniques, namely i) Steepest descent, ii) Powel's conjugate and iii) Simulated annealing are used.

Ramponi in his literature [50] suggested a slightly different design strategy which he referred to as "Bi-impulse response design" based on the same concept of impulse response used in designing linear filters. For designing the quadratic part of the quadratic Volterra filter the response to a pair of impulses at appropriate locations is measured.

Lin and Unbehauen [31] introduced an adaptive approach for designing quadratic Volterra filters, that could be used for channel equalisation and image restoration. The technique used here is an LMS type where the constant, linear, and quadratic coefficients of the quadratic Volterra filters are adjusted during iterations until a minimum mean square error is achieved.

Although little work has been reported in this area, it appears to be promising

to apply the quadratic Volterra filter to restoring noisy images especially when the noise is not additive in nature. Designing the filter is the main concern in this regard. Though some previous work [46, 47, 48] proposed an optimization approach, the formulation of an objective function to obtain the filter coefficients is not yet clear. The adaptive approach discussed in [31] deals with a slightly different issue where the input/output relationship of the system is assumed to be known and channel equalization is of main interest. Since the main objective of any image restoration is to smooth out the noise while preserving the fine details as much as possible, and since an image itself is a non-stationary signal, a locally adaptive technique might be a better approach.

2.5 Edge detection

One of the major tasks in image analysis is edge detection. Edges are fundamentally important primitive features of an image because they often provide an indication of the physical extent of objects within the image. Hence, edge detection is very important for several image processing tasks such as pattern recognition, segmentation etc. An edge is defined as a variation or discontinuity in image intensity resulting from changes in some physical properties of the surface, namely, its reflectance, geometry and/or incident illumination. It is interesting to note that while the human visual system performs edge detection quite easily, it is not very easy to automate the process. However, there is no shortage of edge detection algorithms which work on digital images and provide edge maps as output. Classical edge detection operators developed during the 1960's perform quite satisfactorily on ideal (un-corrupted) images. But these edge detectors perform poorly on noisy images. Since in reality images may

be corrupted by different types of noises, recent attempts have been carried out with an aim to finding edge detectors which have been less sensitive to noise. Techniques which are developed particularly for detecting edges on noisy images are also abundant. But so far, attention has been paid mainly to images which are corrupted by a common type of noise, such as additive white Gaussian noise (AWGN). Very few methods are found which have been developed for images corrupted with noises other than AWGN.

2.5.1 General edge detection techniques

A review paper by Davis [14] discusses several edge detection approaches which have been proposed for the solution of the general edge detection problem. Most of the previous work in edge detection uses small differential operators applied to an image followed by detection operators such as the Robert, Sobel or Laplacian [18]. Such methods work quite effectively on images without noise but show poor performance on noise-degraded images. Marr and Hildreth [32] developed a theory for detecting intensity changes and proposed the Laplacian of Gaussian operator for edge detection. According to them, the Gaussian filter is the optimal filter which provides localization in both space and frequency domains, an important consideration in accurately locating edges. Shanmughan *et al.* [54] proposed a frequency domain filter for edge detection which works globally on the image. A good edge detectors should satisfy several criteria; based on these criteria, edge detection measures are usually defined to quantify the quality of an edge detector. Pratt's figure of merit [43] is one of the standard measures available for edge detectors. Canny [10] also proposed some edge detection measures in his work. A recent hierarchical edge detection approach made by McLean and Jernigan [35] is based on some pre-defined performance criteria.

2.5.2 Speckle-specific Edge Detection Techniques

Most edge detectors are of the gradient type i.e. the detectors are based on the difference between pixel values. This type of edge detector when applied to speckled images yields very poor results because speckle is multiplicative in nature. It degrades the image in such a way that the SNR becomes very low and the image loses its fine details. Hence, it is very difficult to detect edges in speckled images and almost impossible using the usual edge detectors. Since it is obvious that speckle depends on the signal, speckle is more prominent in higher intensity homogeneous areas than in darker areas. Thus a ratio between pixel values should work better than their difference. When dealing with noisy images, it is better to take the ratio of the average pixel values in two adjacent neighborhoods, opposite to the pixel of interest. A ratio magnitude image is thus formed and thresholding finally provides the edge map. This is the basic idea behind the simple Ratio of Average (ROA) edge detector [9] which is particularly useful for detecting images corrupted with a pointwise multiplicative noise. However, the problem involved with speckle is even more difficult because speckle is not only multiplicative, but also has an inter-pixel spatial correlation to some extent. An efficient edge detector should be based on the speckle model.

Very little work has been reported so far in this area, although it is necessary to pay attention to the problem of edge detection on speckle imagery. A simple method named the Coefficient of Variation (COV) has been proposed [60] based on the Constant False Alarm Rate (CFAR) concept which uses a coefficient of variation which can provide the edge strength measures [60]. Frost *et al.* [17] proposed an edge detecting technique for SAR images. The method applies the maximum Likelihood Ratio (LR) as the measure of edge strength. The maximum likelihood ratio is computed based on a SAR image model. Bovik's Ratio of Averages (ROA) [9] is

another approach which attempts to solve this problem. He suggested a combination of the ROA and the Gaussian Smoothed Laplacian (GSL) methods. According to Bovik, the ROA edge detector is quite efficient on speckle-degraded images but has a drawback of generating very thick edges. On the other hand, a general edge detector such as the GSL gives fine edges but also gives rise to many false edges which is not at all desirable. A combination of these two (a logical AND operation on the resulting images obtained from the output of these two detectors) gives a much better result than either of the individual edge detectors. However, it is worth-mentioning that the ROA edge detector is optimal if a pointwise multiplicative model having either negative exponential or Gaussian first order statistics, is considered. More recent work which is simply an extension of the ROA detector is proposed by Touzi *et al.* [60], using the Constant False Alarm Rate (CFAR) concept. This method is model based and designed particularly for SAR images. It has been shown that the performance of the ROA detector depends on the size of the neighborhood, the number of independent looks, and the ratio of mean powers. This detector uses some statistical estimates to calculate edge strength. The effect of edge orientation and neighborhood sizes are also important considerations and have been taken into account in this particular edge detector.

2.6 Concluding remarks

This chapter attempts to provide an overall picture of the previous work on areas related to the problem of interest. Since there are at least four major areas involved, it is impossible to provide more than a brief background for each of these topics. But it is felt that a more elaborate description of the filters of interest (especially

the QVF), the speckle model, speckle-specific edge detectors and filter evaluation criteria – which are all directly related to the main objective of this thesis – should be provided which are dealt with in the next chapter.

Chapter 3

Approach to the solution

3.1 Introduction

The problem dealt with in this thesis originates with coherent energy imaging systems which produce images corrupted with a special type of granular noise called 'speckle'. SAR is one of the many examples of the source of speckle. It is very important to keep in mind that these images are meant for human or computer interpreter for interpretation and analysis by extracting useful information. To serve this purpose, it is necessary to obtain reasonably good quality images. However, frequently, good quality images are not obtained directly from the practical imaging systems and hence the question of image restoration arises.

The solution of the problem requires some knowledge of mathematical modelling of image degradation and of linear and nonlinear filtering techniques. It is also important to preserve the fine details of an image while smoothing the noise which is a difficult criterion to meet. Edge detection on speckled images is essential in many applications but it seems from the previous attempts that it is very difficult to obtain a good edge map from speckled images. One of the major goals of this thesis is to investigate the

effect of filtering on the speckled images in terms of retention of fine details. A few papers that have been collected in the area of edge detection on speckled images will be discussed briefly later in this chapter. However, first it is worthwhile looking into some well known linear and nonlinear filtering techniques that might help in designing an edge preserving speckle smoothing filter.

3.2 Speckle model

A model based filter for suppressing any type of noise is always a better choice than a filter developed in an *ad hoc* manner. So it is very important to have a realistic speckle model in order to design filters for reducing speckle. It is well established that speckle noise intensity is proportional to the underlying image intensity, giving a Signal to Noise Ratio (SNR) in an observed speckled image, equals to one for fully developed speckle. This implies that a pointwise multiplicative model would be able to describe speckle quite well. A large amount of research related to this area is found which assumes a multiplicative[1] model of speckle [16, 23, 30]. Filters developed initially for multiplicative noise model have been used in speckle reduction application too. Thus the model can be represented in (i, j) spatial coordinate system as,

$$y(i,j) = x(i,j) \cdot n(i,j) \tag{3.1}$$

where x is the original image, y is the recorded image and n is the random noise process having a Rayleigh distribution function with mean one and standard deviation equal to $[(4/\pi) - 1]^{1/2}$] [19].

However, it has been pointed out by Tur *et. al* [62] that a multiplicative noise

[1]For simplicity, from now on "multiplicative" will imply "pointwise multiplicative" unless otherwise stated.

model is not always a good choice. The main disadvantage of this model is that it does not take into account the correlation of speckle which is an important consideration in some cases. The correlation results mainly from the Point Spread Function (PSF) of the imaging system. A better model can be realized by taking the point spread function into account. Thus, the model of Equation 3.1 as shown in Fig. 3.1 can be written as :

$$y(i,j) = [x(i,j) \cdot n(i,j)] * h(i,j) \tag{3.2}$$

where * denotes "convolution".

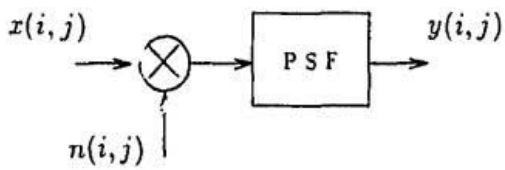


Figure 3.1: Speckle model

The model described by Equation 3.2 is used by Frost *et al.* [16] with the assumption that the PSF for SAR systems is an impulse. More recent work done by Hudson and Jernigan [19] assumes a SAR PSF which has a circularly symmetric Gaussian shape. Hudson and Jernigan used a one-look SAR image model throughout their work which is not a generalized model, due to the common case that an image obtained from SAR is the average of multiple independent looks. In such cases, the model of Equation 3.2 remains the same but with a slight change in the Probability Density Function (pdf) of $n(x,y)$. The noise pdf now takes a χ-squared distribution with N degrees of freedom. Lee [25] showed how the mean and variance of the noise process varied for one-look and multi-look SAR image models. The variation due to intensity and amplitude images is also addressed by Lee.

3.3 Image restoration filters

Most image restoration filters deal with additive white Gaussian noise (AWGN). But there are certain cases where the noise is not additive, rather multiplicative and/or convolutional or some other type and may not be Gaussian and/or white. The filters for AWGN are based on the well-established linear systems theory. These filters assume the stationarity of both image and noise which may also not be a good assumption. For example, it is very likely that images contain some high frequency components - the edges for example. Therefore, although linear filters such as low pass filters smooth out noise, they give rise to blurred images due to attenuation of high-frequency edge conten.. Nonlinear filtering appears to be an alternative when edge preservation becomes important. Also, for signal dependent noise, such as speckle, nonlinear filtering may be unavoidable. Attempts to deal with the signal dependency of noise in designing filters suffer from various limitations. One of the major limitations is that it is often very difficult to design a filter which is optimal with respect to a statistical parameter such as MSE or SNR, because estimation of such parameters might require the solution of complex mathematical equations or may not be explicitly found. Another approach that could be adopted in designing filters is to design locally adaptive filters. Although a relatively new technique, there have been many applications of this technique in digital image restoration which have been found to be quite successful. The main idea behind this approach is that a finite extent filter operates on a finite support of a corrupted image with its filter coefficients depending on local statistics of this support. The image is assumed to be stationary over the finite support used for filtering at any time. The results are significantly improved over would have been obtained by globally processing the image with one fixed filter.

$k(i,j)$ is the gain factor defined between 0 and 1 as given below :

$$k(i,j) = \frac{Q(i,j)}{Q(i,j) + {\sigma_1}^2} \tag{3.4}$$

where

$$Q(i,j) = E[(y(i,j) - \bar{x}(i,j))^2] - {\sigma_1}^2 \tag{3.5}$$

and ${\sigma_1}^2$ is an estimate of the additive noise variance and $E[.]$ denotes the operator giving the expected value or the mean value.

3.3.2 Nonlinear Filtering

Although linear filters are simple and easy to design and implement, they also suffer from some limitations as mentioned earlier. For noise which is not additive, a nonlinear filter may be the only solution. A large number of nonlinear filters have been developed so far and research is still going on in this area. This section describes some of the commonly used nonlinear filters which can be grouped into the two categories,

1. General and
2. Model based
    - Pointwise multiplicative
    - Speckle Reducing

a) Median Filter : This is one of the most classical and commonly used general-purpose nonlinear filters. This algorithm was first proposed by Tukey [43] and used extensively in time series analysis. Later on, it found application in digital image processing. The principle of operation of this filter is very simple. The filter is

Although, the improvement achieved is at the price of complexity and time requirement in design and implementation, in most cases, it is worth while to design an adaptive filter. The following section will focus on some of the commonly used linear and nonlinear filters. As mentioned before, since the adaptive quadratic Volterra filter is of main interest here and has both linear and nonlinear parts, it seems important to understand how these two parts function in smoothing noise and preserving edge structure. For this reason, a linear filter is also included in the following study. It is intended to make a comparative study of the performance of several nonlinear adaptive filters including the quadratic Volterra filter in edge-preserving noise smoothing of speckled images. Hence, it seems worth while to briefly describe some of the existing nonlinear methods which will be used later. The general theory and properties of the quadratic Volterra filter will also be discussed.

3.3.1 Linear Filters

Linear filtering has been used extensively for removing Gaussian additive noise from images. The popularity of linear filtering stems from the fact that it is very simple in structure and hence easy to design and implement. The filter developed by Lee [27] for additive Gaussian noise of zero mean is probably a good example of adaptive linear filters. Its input/output relationship can be represented by

$$\hat{x}(i,j) = \bar{x}(i,j) + k(i,j)[y(i,j) - \bar{x}(i,j)] \tag{3.3}$$

where $\hat{x}(i,j)$ is the estimated pixel intensity at (i,j), $y(i,j)$ is noise corrupted image pixel, based on additive noise model $y(i,j) = x(i,j) + n(i,j)$, and $n(i,j)$ is the noise component. $\bar{x}(i,j)$ is the mean of the un-corrupted image, given by $\bar{x}(i,j) = \bar{y}(i,j)$,

composed of a sliding window encompassing an odd number of pixels. The pixel of interest is replaced by the median of the pixels within the window. Although simple, the filter has been proven to be a good image enhancer specially in the case of impulse or shot noise.

b) Homomorphic Filter : Its technique could be used quite effectively for image restoration when the image is subject to multiplicative interference or degradation. A "logarithmic" operation performed on the noisy image converts multiplicatively combined signals to a sum of signals to which linear filtering technique can be applied before a final "exponential" operation gives back the estimate of the original image [4]. The algorithm is best illustrated in Fig. 3.2. At this point, it is worth mentioning that it is quite difficult to design the linear filter in the middle of the cascade because once the "logarithmic" operation is performed, the noise statistics are changed and simplifying assumptions could lead to a poor overall estimate.

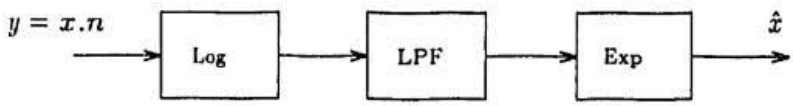


Figure 3.2: Homomorphic filtering

c) Lee's Multiplicative Filter : This is a very well known adaptive nonlinear filter proposed by Lee [27] which is based on a multiplicative noise model. As with Lee's additive filter, the mean and variance of the original image can be estimated from local mean and variance. The multiplicative model uses the following relation,

$$y(i,j) = x(i,j) \cdot n(i,j) \tag{3.6}$$

where y, x and n denote the noisy image, original image and noise processes, respectively. The estimated image is calculated as :

$$\hat{x}(i,j) = \bar{x}(i,j) + k(i,j)[y(i,j) - \bar{x}(i,j)] \tag{3.7}$$

where the gain factor $k(i,j)$ can be obtained as

$$k(i,j) = \frac{Q(i,j)\bar{n}}{x^2(i,j)\,\sigma_n^2 + Q(i,j)\,\bar{n}^2} \tag{3.8}$$

in which $\bar{n}$ and σ_n are the mean and standard deviation of the noise process, respectively. Also since the signal and noise are statistically independent, $\bar{y}(i,j) = \bar{x}(i,j) \cdot \bar{n}(i,j)$, with the assumption of $\bar{n}(i,j) = 1$, which leads to

$$\bar{x}(i,j) = \bar{y}(i,j) \tag{3.9}$$

and

$$Q(i,j) = \frac{\sigma^2{}_y + \bar{y}^2(i,j)}{\sigma^2{}_n + \bar{n}^2} - \bar{x}^2(i,j) \tag{3.10}$$

d) Two-point Taylor Filter : This algorithm [38] is developed for estimating images from their noisy observation that are corrupted by multiplicative noise. It is based on the calculation of local statistics estimates, Taylor series approximations to optimal formulas and normalization of data by local multiplicative sample means. The degradation model can be expressed as,

$$y(i,j) = x(i,j) \cdot n(i,j) \tag{3.11}$$

where x is the original image, y is the noisy image, and n is the noise process.

A pixel estimate $\hat{x}(i,j)$ is defined with two point estimators as,

$$\hat{x}(i,j) = y^a(i,j) \cdot z^b(k,l) \tag{3.12}$$

where $z(k,l)$ is a neighbouring pixel of $y(i,j)$, and $a = a(i,j)$ and $b = b(i,j)$ are two estimating power coefficients. The error function to be minimized thus takes the following form,

$$J(a,b) = E[(x - y^a z^b)] \tag{3.13}$$

The term $y^a z^b$ can be expanded using Taylor series with the bivariate expansion of

$$y^a z^b = 1 + a\ln y + b\ln z + ab\ln y \ln z + \cdots, \quad \forall a\ln y, b\ln z \tag{3.14}$$

A simple version of the two point estimator only takes the first three terms of the series i.e. $y^a z^b = 1 + a\ln y + b\ln z$. Minimizing the above function with respect to a and b i.e setting $\frac{dJ}{da} = 0$ and $\frac{dJ}{db} = 0$ the following system of equations is obtained. The truncation of the series ensures that the system of equations is linear.

$$\begin{aligned} aE(\ln^2 y) + bE(\ln y \ln z) &= E(x\ln y) - E(\ln y) \\ aE(\ln y \ln z) + bE(\ln^2 z) &= E(x\ln z) - E(\ln z) \end{aligned} \tag{3.15}$$

which when solved [2] yields,

$$\begin{aligned} a &= \frac{E(\ln^2 z)[E(x\ln y) - E(\ln y)] - E(\ln y \ln z)[E(xlnz - E(\ln z)]}{E(\ln^2 y)(\ln^2 z) - E[(\ln y \ln z)]^2} \\ b &= \frac{E(\ln^2 y)[E(x\ln z) - E(\ln z)] - E(\ln y \ln z)[E(xlny - E(\ln y)]}{E(\ln^2 y)(\ln^2 z) - E[(\ln y \ln z)]^2} \end{aligned} \tag{3.16}$$

Speckle Reducing Filters : There is a large number of nonlinear filters [16, 19, 23, 24, 28] which are developed particularly for reducing speckle. Any model-based filter should be derived from an accurate model of the degraded image. Most of the speckle suppression methods using nonlinear filters assume speckle to be a pointwise multiplicative process which in some cases does not hold true. It has been quite well

[2] For complete details, see [38].

established that speckle in an image is spatially correlated. However, a few of the multiplicative filters as described in previous sections have been used and found to be effective. The following sections provide an overview on three filters based on speckled image model.

e) Sigma filter : Lee [28] suggested this filter for smoothing speckle in SAR images who observed that SAR speckle suppression techniques fall into major two categories. First, SAR images may be improved by averaging several frames from non-overlapping spectra. But when such means are not available, speckle smoothing could be done after the image has been formed by methods based on the statistics of image and noise. Lee's Sigma filter is based on the sigma probability of the Gaussian distribution. Unlike usual linear filtering, in this case, the pixel to be processed is replaced by the average intensity of only those pixels which have an intensity within two noise standard deviations from the intensity of the centre pixel. Thus, the estimated pixel intensity $x(i,j)$ can be expressed

$$\hat{x}(i,j) = \frac{1}{N_e} \sum_{k=i-n}^{n+i} \sum_{l=j-m}^{m+j} \delta(k,l) \cdot y(k,l) \tag{3.17}$$

where y is the noise degraded image, N_e is the total number of non-zero points in the following summation. and $\delta(k,l)$ is defined as :

$$\delta(k,l) = \begin{cases} 1 & \text{if } (1-2\sigma(i,j)) \cdot y(k,l) < y(i,j) < (1+2\sigma(i,j)) \cdot y(k,l) \\ 0 & \text{otherwise} \end{cases} \tag{3.18}$$

f) Frost filter : Although they began with a convolutional-multiplicative model of speckle noise, Frost *et al.* [16] concluded that the SAR transfer function can be assumed as constant over some finite bandwidth and hence the SAR impulse response can be assumed as an impulse. If $f(i,j)$ denotes the impulse response of the filter,

the signal estimate is :

$$\hat{x}(i,j) = y(i,j) * f(i,j) \tag{3.19}$$

Based on the simplified model of speckle (Equation 3.1), Frost *et al.* proposed a locally adaptive speckle suppressing filter. The basic idea behind this technique is similar to any other filtering technique based on a multiplicative noise model. The following equation represents the impulse response of the filter :

$$f(i,j) = K \cdot \alpha_f(i,j) \cdot \exp\left(-\alpha(i,j)\sqrt{i^2 + j^2}\right) \tag{3.20}$$

where K is a constant chosen such that the filter provides zero flat field response (i.e. all filter coefficients must sum to zero). The parameter α_f is the decay constant which depends on the local mean and variance and is given by the following equation,

$$\alpha_f(i,j) = \frac{4\,\sigma_x^2(i,j)}{L\,\sigma_n^2\,\bar{x}^2(i,j)} \tag{3.21}$$

where L is the number of looks used to form the image, σ_x and σ_n are respectively the local standard deviation of the image and the noise, and $\bar{x}$ is the local mean of the image. It is quite evident from equations (3.20) and (3.21) that the filter weights decrease exponentially with distance from the centre and that the relative weight is controlled by the variance to mean squared ratio. In uniform regions (smaller variance) and brighter region (greater mean), the speckle is more prominent and hence the weights are more evenly spaced than they are in darker and non-uniform regions.

g) Kuan filter : Kuan *et al.* [23] proposed a multiplicative filter which can be used for suppressing speckle, if it is considered as multiplicative noise. Hudson and Jernigan [19] considered this model as one of the best in terms of texture classification

for speckled images. The algorithm is similar to Lee's multiplicative filter [19]. The recorded signal $y(i,j)$ is considered as ,

$$y(i,j) = x(i,j) + u(i,j) \tag{3.22}$$

where the noise process $u(i,j)$ depends on the signal $x(i,j)$ by the following equation,

$$u(i,j) = x(i,j)[n(i,j) - 1] \tag{3.23}$$

Assuming local stationarity of the image, the resulting estimate of the image is,

$$\hat{x}(i,j) = \bar{x}(i,j) + k[y(i,j) - \bar{x}(i,j)] \tag{3.24}$$

with

$$k(i,j) = \frac{\sigma^2_x(i,j) - \bar{x})^2(i,j)\sigma^2_n}{\sigma^2_x(i,j)[1 + \sigma^2_n]} \tag{3.25}$$

3.3.3 Quadratic Volterra Filter (QVF)

A relatively new technique for nonlinear filtering is based on truncated Volterra series. Volterra series is inherently nonlinear in nature which has stimulated researchers to apply them for nonlinear filtering purposes. To avoid the complexity of using higher order terms, the series can be truncated at second order, leading to the Quadratic Volterra Filter (QVF). Ramponi *et al.* [41]-[50] made an extensive study of this filter and proved its robustness in restoring images corrupted by multiplicative noise as discussed in details in Section 2.4. Based on Ramponi's work [50], a quadratic Volterra filter is proposed in present work to restore speckled images. Before approaching the design strategy for a speckle specific filter based on the quadratic Volterra series, it seems necessary to have an idea about the general theory and properties of the quadratic Volterra series as will be discussed next.

3.3.3.1 Theory and property

The I/O relationship of a quadratic Volterra series can be described by the following expression :

$$\hat{x}(n_1, n_2) = h_0 + \bar{h_1}[y(n_1, n_2)] + \bar{h_2}[y(n_1, n_2)] \tag{3.26}$$

where y and $\hat{x}$ are respectively the input and output of the system, (n_1, n_2) is the pixel of interest and h_0 is the constant term of the Volterra filter.

The linear operator $\bar{h_1}$ and quadratic operator $\bar{h_2}$ are given by,

$$\bar{h_1}[y(n_1, n_2)] = \sum_{(i_1, i_2 \in S)} h_1(i_1, i_2) y(n_1 - i_1, n_2 - i_2)$$

$$\bar{h_2}[y(n_1, n_2)] = \sum_{(i_1, i_2 \in S)} \sum_{(j_1, j_2 \in S)} h_2(i_1, i_2, j_1, j_2) y(n_1 - i_1, n_2 - i_2) \cdot y(n_1 - j_1, n_2 - j_2) \tag{3.27}$$

where S is the finite support of the filter. For example, the finite support of size $N \times N$ in 2D space is defined as,

$$\begin{aligned} i_1 &\geq -(N-1)/2, \quad j_1 \leq (N-1)/2, \\ i_2 &\geq -(N-1)/2, \quad j_2 \leq (N-1)/2, \end{aligned} \tag{3.28}$$

for N taken to be odd.

Equation (3.26) can be described more conveniently in matrix[3] form :

$$Y = H_0 + tr\{H_1 Y_1^T\} + tr\{H_2 Y_2^T\} \tag{3.29}$$

where H_1 and Y_1 are $N \times N$ matrices, Y_2 and H_2 are $N^2 \times N^2$ matrices; $tr\{.\}$ indicates the trace of a matrix and $\{.\}^T$ denotes matrix transposition. The element (i_1, i_2) of

[3]To avoid negative indexing in the matrices H_1 & H_2, all indices are offset by $(\frac{N-1}{2})$, (N odd).

the input matrix Y_1 corresponds to the input sample $(n_1 - i_1, n_2 - i_2)$. The matrix Y_2 is obtained from Y_1 using the following formula,

$$Y_2 = Y_1 \otimes Y_1 \tag{3.30}$$

where $\otimes$ denotes the 'Kronecker' product. The matrix Y_2 contains the products of all possible pairs of elements in Y_1.

As seen in the previous analysis, the number of linear and nonlinear coefficients for a finite support of $N \times N$ are N^2 and N^4 respectively. Thus, even for $N = 3$, the number of coefficients for the quadratic part of the filter is 81, a large number. Fortunately, some valid conditions can be exploited without losing the generality of the filter, so that the number of independent coefficients can be reduced to a large extent under the following conditions :

i) uniformity in grey level which implies for a homogeneous region, the linear coefficients and nonlinear coefficients should sum up to one and zero respectively as expressed mathematically by :

$$\begin{aligned} \sum_{i_1,i_2 \in S} h_1(i_1,i_2) &= 0 \\ \sum_{i_1,i_2 \in S} \sum_{j_1,j_2 \in S} h_2(i_1,i_2,j_1,j_2) &= 0 \end{aligned} \tag{3.31}$$

ii) symmetry condition which is obvious from the following equations :

$$\begin{aligned} h_1(i_1,i_2) &= h_1(i_2,i_1) \\ h_2(i_1,i_2,j_1,j_2) &= h_2(j_1,j_2,i_1,i_2) \end{aligned} \tag{3.32}$$

which reduces the number of coefficients by half; and

iii) isotropy condition which means that the orientation of the filter mask must not make any difference to the output; this particular condition gives a large reduction in the number of coefficients.

Exploiting these conditions for $N = 3$ the total number of independent linear and nonlinear terms drop to 6 and 11, respectively.

It should be clear that the quadratic Volterra filter has mainly two parts – a linear part and a quadratic part and the filter response is the combination of these two parts except for a possible constant offset. The linear part's role is smoothing the noise, whereas, the quadratic part compensates for the damage caused by the low-pass filter on fine details of the image. This seems to give an improved performance provided that the filters - both linear and quadratic can be well designed. However, this is not an easy task. The following analysis is done based on Ramponi's work [50] which arranges the quadratic coefficients into three major classes, referred to as Type 0, Type 1 and Type 2 coefficients. This concept will be used later in designing the quadratic part of the filter used in this research.

The quadratic coefficients can be arranged in a $3^2 \times 3^2$ lexicographically ordered matrix H_2. Thus, H_2 can be considered as a matrix which is composed of 3×3 elements, with each element a sub-matrix of H_2 denoted by $H_2[i,j]$, where i,j represents the row and column of matrix H_2. Each sub-matrix $H_2[i,j]$ of matrix H_2 is in fact a matrix composed of 3×3 elements denoted by $H_2[i,j](k,l)$, where k,l refers to the row and column of the sub-matrix $H_2[i,j]$. Thus, the element of H_2, $H_2[i,j](k,l)$ acts on the product of the image pixels $y(i - \frac{N-1}{2}, j - \frac{N-1}{2})$ and $y(k - \frac{N-1}{2}, l - \frac{N-1}{2})$.

A possible set λ of the 11 independent coefficients are :

$$\begin{aligned}\lambda \;=\; \{&H_2[1,1](1,1), H_2[1,1](1,2), H_2[1,1](1,3),\\ &H_2[1,1](2,2), H_2[1,1](2,3), H_2[1,1](3,3),\\ &H_2[1,2](1,2), H_2[1,2](2,1), H_2[1,2](2,2),\end{aligned}$$

$$H_2[1,2](3,2), H_2[2,2](2,2)\} \tag{3.33}$$

Depending on the relative position of the pair of pixels on which the coefficients act, these can be divided into 6 groups and the independent coefficients then further reduced to 6. The set of all quadratic coefficients Λ can now be defined as ,

$$\Lambda = \{\alpha, \beta, \delta, \epsilon, \theta, \mu\} \tag{3.34}$$

where

$$\begin{aligned}
\alpha &= 4H_2[1,1](1,1) + 4H_2[1,2](1,2) + H_2[2,2](2,2) \\
\beta &= 4H_2[1,1](1,2) + 4H_2[1,2](2,2) \\
\delta &= 2H_2[1,1](1,3) + H_2[1,2](3,2) \\
\epsilon &= 2H_2[1,1](2,2) + 2H_2[1,2](2,1) \\
\theta &= 2H_2[1,1](2,3) \\
\mu &= H_2[1,1](3,3)
\end{aligned} \tag{3.35}$$

To render the subsequent design simpler, the responses are again classified into three major categories based on the distance measured using 8 connectivity as a metric between the couple of pixels on which the corresponding quadratic coefficient acts. The distance between two pixels (i,j) and (k,l) denoted by $dist((i,j),(k,l))$ is defined as :

$$dist((i,j),(k,l)) = max(abs(i-k), abs(j-l)) \tag{3.36}$$

And hence the three types are,

$$\begin{aligned}
&\text{Type 0:} \quad \Lambda_0 = \{\alpha\} \quad \text{having } dist((i,j),(k,l)) = 0 \\
&\text{Type 1:} \quad \Lambda_1 = \{\beta, \epsilon\} \quad \text{having } dist((i,j),(k,l)) = 1 \\
&\text{Type 2:} \quad \Lambda_2 = \{\delta, \theta, \mu\} \quad \text{having } dist((i,j),(k,l)) = 2
\end{aligned} \tag{3.37}$$

3.4 Speckle-specific edge detection techniques

As mentioned in Section 2.5, edge detectors are mainly of the gradient type i.e. the detectors are based on the difference between pixel values. This type of edge detector, when applied to speckled images, yields very poor results because of multiplicative-convolutional nature of speckle noise. Very few speckle-specific edge detection methods have been developed as discussed in detail in Section 2.5.2. Brief analytical descriptions of some of these methods are included here.

3.4.1 Coefficient of Variation

It is well known that for radar imagery, the coefficient of variation ($\sigma/\mu = \sqrt{var(x)}/E(x)$) is constant i.e. independent of mean power over a homogeneous area [60, 61]. The Probability of False Alarm (Pfa) is defined as the probability that a pixel of a homogeneous area is detected as an edge pixel [61]. If Pfa is dependent on the mean power as it is in the case of speckled image, it is better to use an edge detector based on an operator which is independent of the mean power and such an operator has a property of Constant False Alarm Rate (CFRA). An edge detector of CFAR could be developed using the coefficient σ/μ as an edge strength measure. In practice, an estimate of coefficient of variation (i.e. the standard deviation to mean ratio denoted by $\bar{\sigma}/\bar{\mu}$) is computed over a window of N neighboring pixels and the centre pixel of the window is replaced by this value. σ and μ are computed as,

$$\bar{\mu} = \frac{1}{N}\sum_{i=1}^{N} y_i \tag{3.38}$$

and

$$\bar{\sigma} = \sqrt{\left(\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{\mu}^2 - 1\right)} \tag{3.39}$$

In practice since the distribution of σ/μ is around $\sqrt{\frac{1}{L}}$, the threshold is set to a value $T = \sqrt{1/L} + \epsilon$, where ϵ is very small number and L is the number of independent looks. So the larger σ/μ is, the greater is the probability that the pixel of interest is a part of an edge. The pixel is assigned to an edge if the edge strength $\frac{\sigma}{\mu} \geq T$.

3.4.2 Frost' CFAR edge detector

Frost *et al.* [17] proposed an edge detecting technique for SAR images. The method applies maximum Likelihood Ratio (LR) as the measure of edge strength. Maximum likelihood ratio is computed based on a SAR image model and works quite satisfactorily. The likelihood ratio is given by,

$$\Lambda(y_1 y_2 y_3 \cdots y_N) = \frac{2^N \prod_{i=1}^{N} \hat{\beta}_0^{-L} \exp(-y_i/\hat{\beta}_0)}{\prod_{i=1}^{N} (\hat{\beta}_1^{-L} \exp(-y_i/\hat{\beta}_1) + \hat{\beta}_2^{-L} \exp(-y_i/\hat{\beta}_2))} \tag{3.40}$$

where y_i the ith pixel value and

$$\hat{\beta}_0 = M_y/L, \tag{3.41}$$

with a local mean of M_y using all pixels $y_1 y_2 \cdots y_N$, and L independent number of looks.

$$\hat{\beta}_1 = \hat{\beta}_0 - R_f$$

$$\hat{\beta}_2 = \hat{\beta}_0 + R_f$$

and R_f is defined as,

$$R_f = (\frac{\Psi^2}{L(L+1)} - \hat{\beta}_0^2)^{1/2} \tag{3.42}$$

where $\Psi^2 = S_y^2 + M_y^2$ with S_y^2 being the variance of neighborhood.

Frost *et al.* claimed that the method can be generalized and applied to any noisy images with some knowledge of the first order statistics of the noise. The given test

image model assumes only the multiplicative model of speckle with correlation fully ignored and the likelihood ratio is calculated based on this assumption. A thinning operator [17] should be used after the LR operator in order to produce fine edges.

3.4.3 Bovik's ratio of Averages (ROA)

Another approach to the solution of this problem was made by Bovik [9]. He suggested a logical combination of a ratio of averages (ROA) and the Gaussian Smoothed Laplacian (GSL) edge detectors. According to Bovik, the ratio of averages edge detector is quite efficient on speckle-degraded images but has a drawback of generating very thick edges. On the other hand a general edge detector such as the Gaussian Smoothed Laplacian gives fine edges but gives rise to many false edges which is not at all desirable. A combination of these two edge detectors (logical AND operation on the resulting images obtained from the output of these two detectors) gives better results than either of the individual edge detectors. The ratio of averages edge detector is computed as described below :

Horizontal component :

$$H(i,j) = max[R(i,j)/L(i,j), L(i,j)/R(i,j)] \tag{3.43}$$

where $R(i,j)$ and $L(i,j)$ are the average values of the neighboring pixels immediately to the right and left to the image coordinate (i,j) respectively.

The vertical component $V(i,j)$ is similarly computed and finally the overall edge magnitude could be given by,

$$R(i,j) = \sqrt{[H^2(i,j) + V^2(i,j)]} \tag{3.44}$$

Now, a predefined threshold is used to obtain the final the edge map. Thus, $R(i,j) > T$ implies that an edge is likely present or otherwise there is likely no edge.

3.4.4 Extended Ratio of Averages CFAR

Another edge detector which is simply an extension of ROA detector is proposed by Touzi *et al.* [60, 61] which uses the Constant False Alarm Rate (CFAR) concept. This method is model-based and designed particularly for SAR images. It has been shown that the performance of an ROA detector depends on the size of neighborhood, number of independent looks and ratio of mean powers. The detector uses some statistical estimates to calculate edge strength. A contrast ratio of two homogeneous areas of the same intensity, is defined as $C_r = max(P_1/P_2, P_2/P_1)$, with $C_r = 1$ for two homogeneous areas. Given a threshold T, it is then possible to compute the conditional probability of detection (Pd) within the boundary between two homogeneous areas of contrast ratio C_r :

$$Pd(T, C_r) = Prob(r < T/C_r) = \int_0^T p(r/C_r)dr \tag{3.45}$$

Hence,

$$Pfa(T) = Pd(T, 1) \tag{3.46}$$

The effect of edge orientation and neighborhood size are also important considerations and have been taken into account. Edge orientation in 4 different directions is suggested and this is different from Bovik's method which uses only horizontal and vertical directions. Although the concept of using different thresholds with increasing window size is also proposed, but it is not clear how the different thresholds are computed [60, 61]. In present work, only the idea of four different orientations is used. A thinning edge process based on grey-tone morphology [52] is also suggested following the edge detection because the resulting edge map is quite thick.

3.5 Performance measures

Evaluating a filter's performance both qualitatively and quantitatively is important. A comparison between several filters developed for the same purpose can be made based upon some pre-established measures. There are a number of features on which a filter can be evaluated. As it is usually not possible to find one filter which is the best in all respects, a filter is considered as a good one if it performs reasonably well according to most of the desired features. Depending on the purpose of filtering, different performance criteria can be chosen. However, image restoration filters are usually evaluated in terms of their noise smoothing ability. Images of interest in this thesis are meant for interpretation either by human or machine and resolution is also very important consideration. So equal attention is paid to investigating both the noise smoothing and edge preserving ability of the filters.

3.5.1 Noise smoothing measures

A traditional measure of noise smoothing is the Mean Squared Error (MSE). When working with synthetic image data, the un-corrupted image is assumed to be known. Hence, the global MSE for the noisy image and filtered images can be calculated. However, it is well known that reduction of MSE does not necessarily indicate an improvement in interpretability (human or machine) [38]. Another commonly used noise smoothing measure is Signal to Noise Ratio (SNR). The SNR for a whole image as well as for two sample homogeneous regions of low and high intensities can be calculated too.

3.5.2 Edge preserving measures

A filter can be evaluated in terms of its edge preserving capability. It is often the case that filtering reduces the resolution and causes blurred edges which might lead to mis-interpretation in the analysis of the image. Several edge detection measures are available which can be used to compare edge maps obtained before and after filtering. Thus an evaluation of different filters can be made in terms of their edge preservation ability and possibly even their edge enhancement ability. One such well known measure proposed by Pratt is his 'figure of merit' [43] which yields a single value with respect to edge displacement, ambiguity and incorrect classification, etc. is defined by :

$$R = \frac{1}{max(N_A, N_I)} \frac{1}{1 + \beta d^2} \tag{3.47}$$

where N_A and N_I are respectively the total number of edge pixels in the actual and ideal edge maps, d is the perpendicular distance from an actual edge pixel to the ideal edge pixel and β is a scaling constant. Pratt's figure of merit is very compact and is quite biased [35]. Some more accurate measures for evaluating edge detectors are proposed by McLean and Jernigan [35]. Some similar edge detector's measures are suggested here to evaluate the edge maps resulting from speckled image and filtered images. The factors named "correct", "ambiguous", "missed" and "wrong" are defined as,

$$\begin{aligned} C &= \frac{2 \times \#correct}{\#true + \#found} \\ A &= \frac{\#ambiguous}{\#true} \\ M &= \frac{\#missed}{\#true} \end{aligned}$$

$$W = \frac{\#false}{\#found} \tag{3.48}$$

where

- #true= number of total edge pixels in the ideal or true edge map,
- #found= number of edge pixels found in the edge map of interest,
- #correct= number of true edge pixels which were found with either one or two edge pixels within a 5×5 window centered at corresponding true edge pixel location.
- #ambiguous= number of true edge pixels which were found with more than one edge pixels within a 5×5 window centered at corresponding true edge pixel location.
- #missed= number of true edge pixels which were not found with at least a single pixel within a 5×5 window centered at corresponding true edge pixel location.
- #wrong= number of pixels found which were not in the ideal edge map within a 5×5 window centered at corresponding pixel location.

A good edge detector should provide a large value of C and small values of A, M and W.

3.5.3 Other measures

Visual interpretability is no doubt another important consideration in evaluating a filter's performance. Images of interest in this thesis are meant to be interpreted by humans or machines. If available, real data can be gathered and processed and

expert viewers can be asked to evaluate the quality of the image before and after filtering. This will likely give a better understanding of the improvement in terms of the information content. Time consumption and complexity of filters in design and implementation are also worth investigating as other measures of filter performance.

3.6 Concluding remarks

Based on the above discussion of SAR speckle modeling and of different nonlinear filtering theory including the QVF, it is intended to design a speckle model-based QVF. As is clear from the previous discussion, the QVF has the most generalized structure among all the filters that have been discussed and hence it is flexible from the design point of view. The next chapter focusses mainly on the proposed methodology for modeling SAR speckle and suggests some speckle-specific QVF designs with detailed descriptions for implementation. Also a new edge detector based on Bovik's and Touzi's ROA edge detector and a thinning algorithm [53] are described in detail.

Chapter 4

Methodology and Implementation

4.1 Methodology

Although speckle has been discussed and dealt with in the literature [16, 19, 23, 24, 28], there still remains the question of removing speckle from an image while preserving the underlying informative structure. The present thesis is an attempt to the solution of this problem. Some of the previously discussed image restoration filters will be applied to compare their performance on speckled images in terms of edge preserving noise smoothing. Some of these filters are meant for speckle suppression and some are not. However, those filters which seem to be appropriate in terms of reducing speckle while preserving the fine details are investigated. A relatively new technique which has not been applied previously to solve the speckle problem is demonstrated as well.

4.1.1 Choice of speckle model

To choose an image restoration filter specifically for speckle degraded images, one must know the speckle statistics and have a speckle model. A realistic model of

speckled SAR images is one which is described by Hudson and Jernigan [19]. A slight modification of this model (Section 3.2) will be used throughout the remainder of this thesis. The validity of the model will be tested. It is well established that speckled images have a property of maintaining constant SNR as the mean intensity of the image varies. However, the value of the constant depends on the number of independent looks, the SAR PSF and also on the type of image (amplitude, intensity etc.). A test will be performed to confirm that the model used would provide constant SNR which is independent of the mean intensity.

4.1.2 Quadratic Volterra filter

It has been proven that speckle is a multiplicative and partially correlated noise and argued that a linear filter may not work optimally in suppressing speckle. Most of previous reported work in this area assumes a multiplicative model of speckle and hence uses multiplicative filters for removing speckle. The fact that speckle is partially correlated, has largely been ignored although it is an important consideration. The inter-pixel correlation of speckle leads to a conclusion that the edge preserving speckle smoother must have a spatial memory in its structure. The Quadratic Volterra filter originating from the truncated Volterra series has been used widely and has been proven to be a very effective way of dealing with nonlinear problems. As mentioned in Section 2.4.1, some of the successful applications in which the quadratic Volterra filter proved its robustness are echo cancellation [8], modelling low frequency drift oscillations under sea wave excitation, electromagnetic scattering from quadratically nonlinear targets and in general modeling quadratically nonlinear systems [21, 22]. The application of the Volterra filter in image processing is also found in many previous papers (as illustrated in section 2.4.2). Specially, applications of the Quadratic

Volterra filter in image enhancement [44], edge extraction [49] and edge preserving smoothing [50] seem to be very satisfactory. Moloney's filter uses a truncated Taylor series in accordance with a pointwise multiplicative noise model [38]. Since the Volterra series can be considered as a Taylor series with memory, the Volterra filter might be able to take into account the correlation property of speckle and might yield better results for speckled images. The Volterra filter has a generalized nonlinear structure which provides flexibility in different types of filter design. However, most Volterra filters, used in various applications, are designed only up to the quadratic term because of the huge complexity of further higher orders, and also because most of the practical nonlinear problems are not so complex that they require higher than a second order filter to obtain an adequate results. A quadratic Volterra filter is therefore proposed here to solve the problem of speckle.

Though named a *quadratic filter*, it has both a linear part and a quadratic part. The linear part which is intended to play a role in noise smoothing, can be any low-pass filter. A simple box-type local averaging filter is proposed because the quadratic part is of main interest here,rather than the linear part. However, it is difficult to design the quadratic part which compensates for the blurring effect caused by the linear part, because of the large number of independent coefficients. The present design is restricted to a filter support of $N = 3$. For a fairly homogeneous area in a speckled image, it is perhaps best not to use the quadratic part of the filter at all. Hence, a decision-directed implementation of the filter is suggested here as was also proposed by Ramponi [45]. The filter coefficients can be either set to some fixed values beforehand or updated in a regular fashion throughout the image as will be discussed later in detail in Section 4.2.2. The effect of the three types of responses on the speckled images are investigated.

4.1.3 Choice of filters

The other filters which are chosen as candidates for the present study of edge preserving speckle smoothing are:

1. Linear average filter
2. Median filter
3. Homomorphic filter
4. Lee's multiplicative filter
5. Two-point Taylor filter
6. Lee's sigma filter and
7. Frost filter.

The reason the linear average filter has been chosen is that it is important to see how the other filters work with respect to a simple linear average filter. For a homogeneous region, a linear filter should perform quite well regardless of the type of noise. However, it is expected that edge retention will be much better in other filters. Although median filter is an *ad hoc* choice, it plays a good role in image enhancement which explains why it has been chosen. The multiplicative property is very prominent in speckle and multiplicative filters have been previously used in many previous speckle suppression techniques, homomorphic filter, Lee's multiplicative filter and the two-point Taylor algorithm. The remaining two filters were developed for SAR speckle smoothing purpose and hence are worth investigating for their smoothing and edge retention ability. Finally the proposed filter will be applied to speckled images and a comparative study of it and all of the above filters is reported.

4.1.4 Types of data

To test the filters' performance, it is necessary to have a set of images. It is worth mentioning that in image processing, it is important to test a developed algorithm on more than one image, since it might happen that an algorithm works well for a particular image, but may perform very poorly for some other images. It was decided to use created images of three or four different types and to simulate speckle corrupted versions of each image using the speckle model (Equation 3.2). The choice of synthetically created images need not necessarily be very complex. In practice, SAR images are not very complex in terms of their number of grey levels or fluctuation of image intensities. Most of the work previously done on synthetic SAR images [9, 19, 23] use images of relatively simple structure. Attention must be paid while choosing images so that the edge detecting capability can be measured effectively even in the worst possible situations. An annular ring and stripe pattern are appropriate for this purpose. It is also important to show a filter's performance in homogeneous areas of both bright and dark intensities. Images with two contrast grey levels would be appropriate from this point of view. Also a more complex image is used to illustrate the overall performance of the filters. In addition, to test the validity of the speckle model some tests are performed. These tests require an image of gradually varying intensities in the horizontal direction. This particular image is used only for the purpose of validity of the speckle model as will be discussed later in Section 4.2.1.

4.1.5 Edge detection

Investigating various edge detection techniques for speckle corrupted images is an important issue in this thesis, because a lack of good edge detector for speckled images

makes it very difficult to detect edges in such images. Edge detection is necessary if one must determine either qualitative or quantitative measures for the edge preserving ability of a filter. For original un-corrupted images (known for synthetically generated images), any well known edge detector e.g. Sobel, Robert's gradient, Gaussian smoothed Laplacian etc. [18], will give substantially good performance. However, to detect edges in speckled images, a slight modification of Bovik's Ratio of Averages detector and a combination of Ratio and Gradient of Averages (RGOA) edge detectors are proposed, since most of the common edge detection techniques perform poorly on speckled images.

4.1.6 Performance measures

Filter performance needs to be evaluated quantitatively and qualitatively. Two very common noise smoothing measures are the global Mean Square Error (MSE) and the Signal to Noise Ratio (SNR). These are computed for a corrupted image and its filtered versions to examine the filter performance in terms of noise smoothing. A lower value of MSE and higher value of SNR are expected after filtering. Edge detection ability is determined before and after filtering by the correctness of the edge map, based on some measure; namely number of *correct*, *missed*, *ambiguous*, *wrong* etc. as defined earlier in Section 3.5.2.

4.2 Implementation

4.2.1 Speckle simulation

A multiplicative convolutional model (Equation 3.2) has been used for speckle model simulation. The first step is to generate a two-dimensional noise field. There is

a provision for generating noise having different probability distributions [42] (e.g. χ, χ-squared, Gaussian, exponential etc.). However, noise having exponential and χ-squared distribution are used for generating one-look and multi-look speckled images. The next step is to multiply each ideal image pixel pointwise with samples drawn from the noise file of interest depending on the desired number of looks. Thus the multiplicative part of the model is generated. To take the correlation property of speckle into account, it is necessary to know or assume the point spread function or impulse response of an imaging system. Since the present work involves SAR speckle simulation, a hypothetical but realistic impulse response for a SAR system is chosen. The assumed impulse response has a circularly symmetric Gaussian shape, the edge of which is taken to be 10 decibels below the peak value. This response was initially used by Hudson and Jernigan [19]. The resulting multiplicative image is then convolved with the given PSF of SAR and an amplitude speckle model of SAR image is thus obtained with the expected statistics of such an image. At this point, it seems almost essential to test the validity of the model generated. This is done using a 200×128 image with gradually varying intensities from 0 to 200 corrupted with speckle as shown in Fig. 4.1(a). Both one-look and four-look speckle have been considered (Figs. 4.1(b) and (c)). The ratio of the standard deviation (std) to mean for all columns in this image are expected to be equal with theoretical values of 0.522 and 0.25 for one-look and four-look images, respectively [19, 25]. The plot of the ratio against the mean intensity in Fig. 4.2. shows a close match between the expected value and the values found. Moreover, the horizontal distribution of the ratios confirm the fact that the standard deviation of the image is directly proportional to the mean intensity. Another image with uniform intensity of 40 throughout the image has been synthetically created and corrupted with one-look and four-look speckle (not shown).

(a) (b) (c)

Figure 4.1: (a) Original image "bands"; (b) One-look speckle corrupted version; (c) Four-look speckle corrupted version

The histograms of these images are shown in Fig. 4.3. These figures illustrate the pdf of these images which seem to follow a Rayleigh and χ-distribution as expected for one-look and multi-look speckled images, respectively, [19, 25].

4.2.2 Quadratic Volterra filter implementation

A variety of different implementations of the quadratic Volterra filter have been carried out as a part of this work and although not all of them perform satisfactorily, it is worthwhile to describe some of the previous experiments to illustrate the development of the successful QVFs has been done. Finally, the implementation of the filter which gives the most satisfactory results is discussed in detail.

As mentioned earlier, it is not difficult to implement the linear part of a QVF. A simple box type low pass filter performs well for this purpose, and has been used in all implementations. It is important to remember that the quadratic part may not

necessarily perform any better over homogeneous areas, and in fact, it may cause some artifacts in these areas. For this reason, it was decided to turn off the quadratic part in these areas. A threshold is thus provided in all implementations which determines whether or not the pixel to be estimated is required to be processed through the quadratic part. Thus,

$$H_2[i,j](k,l) = \begin{cases} 1 & \text{if } L(\sigma_y/\mu_y) \geq T \\ 0 & \text{otherwise} \end{cases}$$

where, σ_y and μ_y are the local standard deviation and mean calculated from the noisy image, respectively; L is the independent number of looks and T is a multiplying factor which must be chosen interactively.

The difficulty arises in designing the quadratic part because it has a large number of independent quadratic coefficients ; 11 even for a small mask size of 3×3. The following discussion presents some methods used in this work used in setting these coefficients. Some of these methods are based on trial and error.

- **Fixed coefficients QVF:**

  One possible implementation is to choose the quadratic coefficients such that the filter fulfills three conditions namely uniformity, symmetry and isotropy (Equation 3.29 and 3.33). An algorithm is thus developed which takes the set of these coefficients as one of the inputs and maps them to the 9×9 quadratic coefficient matrix creating a fixed mask. The same filter coefficients are used throughout the image to be filtered. However, it is difficult to choose all these values without having much information about the shape of the quadratic mask. As described by Ramponi [50], some of these coefficients can be considered as zero when an edge preserving smoother is of great importance.

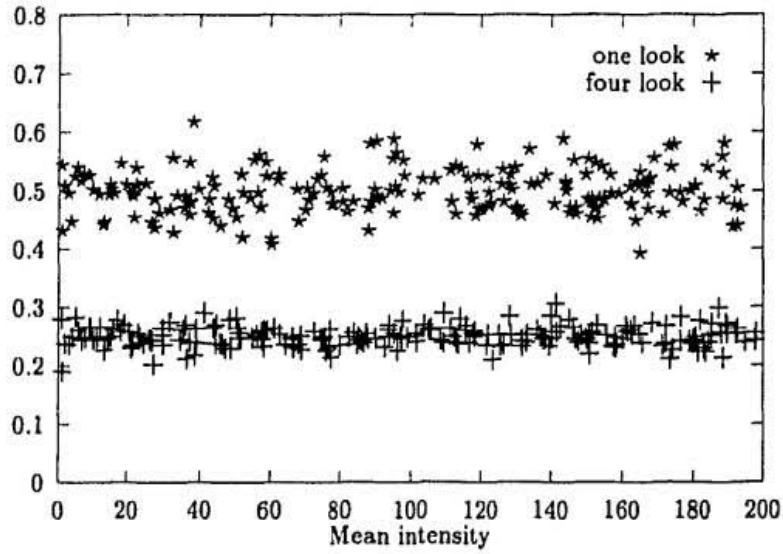
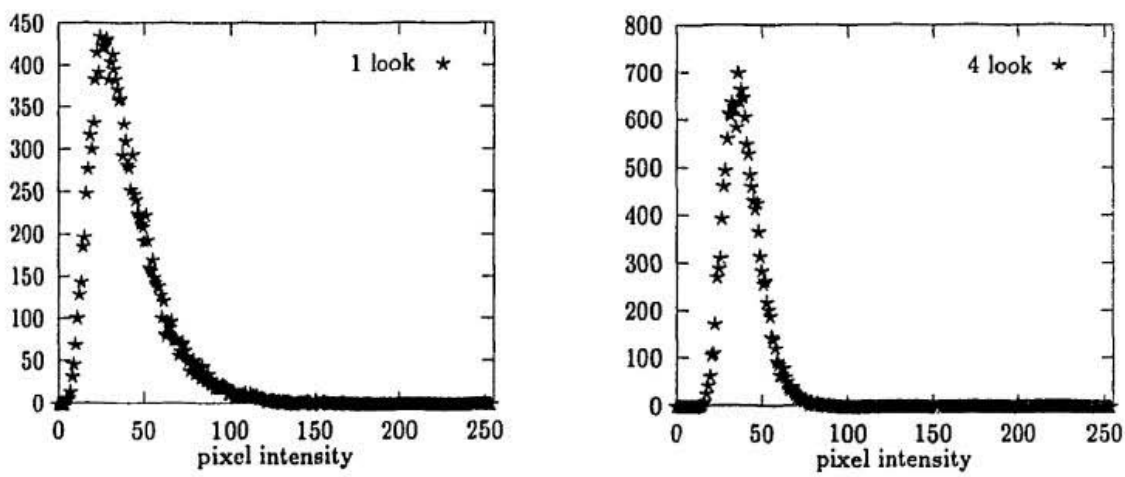


Figure 4.2: Scatter plot of std/mean ratio vs. mean intensity (Image: "bands")



(a) (b)

Figure 4.3: Histogram of a uniform image corrupted by speckle noise (mean = 40)

- **Kronecker product coefficients QVF:**

  The next implementation avoids choosing all 11 coefficients. In this implementation, the 9×9 quadratic matrix is formed by taking the "Kronecker" product of two equal symmetric 3×3 matrices. Since the matrices are equal and symmetric, one needs to know only 3 out of 9 of their elements. Thus if H_{23} is a symmetric 3×3 matrix then the 9×9 quadratic matrix is given by the following equation,

$$H_{29} = H_{23} \otimes H_{23} \tag{4.1}$$

  where $\otimes$ stands for the "Kronecker" product. The following constraint is important to meet the symmetry, grey-level uniformity and isotropy conditions on the quadratic coefficients. Thus,

$$\sum_{(i,j)} H_{23}(i,j) = 0 \tag{4.2}$$

  This condition however reduces the number of unknowns to 2. So a trial and error process is used to choose the best values of these two unknowns which gives a satisfactory output.

- **Proportional weight coefficients QVF:**

  Based on Ramponi's idea of three types of filters [50], coefficient values W_0, W_1, W_2 for the three types (discussed in details in Section 3.3.3) are arbitrarily chosen such that $\sum_{i=0}^{2} W_i = 0$. The weight calculated for a given type Λ_i of the response (Equation 3.37) is then equally distributed to all elements of that type. Hence, the elements of Λ_i (Equation 3.37) are assigned to have the following

values,

$$\begin{aligned} \alpha &= W_0 \\ \beta = \epsilon &= \frac{W_1}{2} \\ \delta = \theta = \mu &= \frac{W_2}{3} \end{aligned} \tag{4.3}$$

Again, each weight of a given type can be proportionally distributed to all elements of the corresponding type. The independent coefficients (Equation 3.35) are evaluated as shown below,

$$\begin{aligned} H_2[1,1](1,1) = H_2[1,2](1,2) &= \frac{4}{9}\alpha \\ H_2[2,2](2,2) &= \frac{1}{9}\alpha \\ H_2[1,1](1,2) = H_2[1,2](2,2) &= \frac{1}{2}\beta \\ H_2[1,1](2,2) = H_2[1,2](3,2) &= \frac{1}{2}\epsilon \\ H_2[1,1](1,3) &= \frac{2}{3}\delta \\ H_2[1,2](3,2) &= \frac{1}{3}\delta \\ H_2[1,1](2,3) &= \theta \\ H_2[1,1](3,3) &= \mu \end{aligned} \tag{4.4}$$

Knowing the values for all 11 coefficients, the 9×9 quadratic coefficient matrix H_2 is formed.

- **Mean square estimated coefficients QVF:**

Let $x(i,j)$ denote the SAR image which is corrupted by speckle noise process $n(i,j)$. Therefore, the corrupted image $y(i,j)$ at pixel (i,j) can be written as,

$$y(i,j) = [x(i,j) \cdot n(i,j)] * h(i,j) \tag{4.5}$$

where $h(i,j)$ is the point spread function of the SAR.

Assuming that the estimated value of any pixel can be expressed as a QVF, operating on the speckled image, a minimum Mean Square Error (MSE) estimation is presented to obtain the coefficients of the QVF. A simpler structure of the estimation for the problem is attempted here as given in the following equation,

$$\hat{x} = \bar{x}(i,j) + \sum_{r=0}^{N-1} \alpha_r \sum_{[(i,j),(k,l)] \in S_r} y(i,j)y(k,l) \tag{4.6}$$

where S_r is a set which contains all possible pairs of pixels (i,j) and (k,l) such that the distance (Equation 3.36) between these pixels are r i.e.

$$dist((i,j),(k,l)) = r \tag{4.7}$$

For simplicity, the index for the second summation will be omitted throughout the following analysis. For $N = 3$, three coefficients α_0, α_1 and α_2 have to be determined. One simple way of estimating these coefficients is illustrated below. To further simplify the process, one coefficient is estimated at a time while setting the others to zero.

Determination of α_r

An objective function $J(\alpha_r)$ can be defined as,

$$\begin{aligned} J(\alpha_r) &= E[(x(m,n) - \hat{x}(m,n)]^2] \\ &= E[(x(m,n) - \bar{y}(m,n) - \alpha_r \sum y(i,j)y(k,l)]^2 \\ &= E[(x(m,n) - \bar{y}(m,n))^2] - 2\alpha_r E[(x(m,n) \\ &\quad -\bar{y}(m,n)) \sum y(i,j)y(k,l) \\ &\quad +{\alpha_r}^2 E[(\sum y(i,j)y(k,l))^2] \end{aligned} \tag{4.8}$$

By minimizing the objective function, an optimum solution for α_r will be obtained,

$$\frac{d}{d\alpha_r} J(\alpha_r) = 0 \tag{4.9}$$

This yields,

$$\alpha_r = \frac{E[(x(m,n) - \bar{y}(m,n)) \sum y(i,j)y(k,l)]}{E[\sum y(i,j)y(k,l)]} \tag{4.10}$$

However, it is impossible to determine $E[x(m,n) \sum y(i,j)y(k,l)]$ unless the noise statistics are well known. In this case, the overall noise statistics of the model (Equation 3.2) are not well known. Nevertheless, if a purely multiplicative model is assumed, the above mentioned expression can be evaluated easily from the noisy image process. Denoting $x(m,n)$, $y(i,j)$ and $y(k,l)$ as x_1, y_1 and y_2, respectively, the following relation can be derived,

$$\begin{aligned} y_1 y_2 y_3 &= n_1 x_1 y_2 y_3 \\ E[y_1 y_2 y_3] &= E[n_1] E[x_1 y_2 y_3] \\ E[x_1 y_2 y_3] &= E[y_1 y_2 y_3] \end{aligned}$$

It is also assumed here that the noise process is independent of the original image.

Therefore,

$$\alpha_r = \frac{E[(y(m,n) - \bar{y}(m,n)) \sum y(i,j)y(k,l)]}{\sum (y(i,j) \cdot y(k,l))^2} \tag{4.11}$$

- **4-direction-oriented QVF:**

The next implementation is an attempt to make the quadratic part more sensitive to the noise particularly in the edgy regions. The quadratic part in both

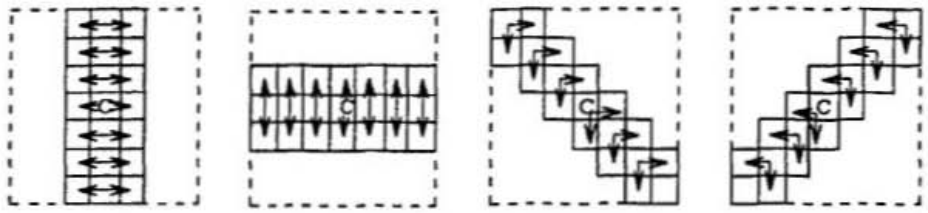


Figure 4.4: 4 one-dimensional filters in different orientations

of the above implementations has worked on a 3×3 region of support which does not seem to be large enough. But a large region of support means a larger mask size of the quadratic filter. However, to avoid the huge complexity in dealing with a two-dimensional quadratic filter having an $N \times N$ filter mask, it seems much simpler as well as effective to use four one-dimensional filters in four different orientations, - horizontal, vertical, diagonal and counter-diagonal as shown in Fig. 4.4.

In this way, it is possible to cover a larger portion surrounding the centre pixel to be estimated in 4 particular directions without increasing the complexity. This has been done where the estimated value of a pixel located at (m, n) is described by the following equation,

$$\hat{x}(m,n) = \bar{y}(m,n) + \sum_{d_s=0}^{2} \sum_{d_r=0}^{3} w(d_s, d_r) Y(d_s, d_r) \tag{4.12}$$

where, $\hat{x}(m,n)$ and $\bar{y}(m,n)$ are the estimated value of the original image and the expected value of the corrupted image, respectively; $Y(d_s, d_r)$ is the expected value of the product of pair of the pixels (i,j) and (k,l) having inter-pixels distance of d_s and direction of d_r, and $w(d_s, d_r)$ is a weight assigned to $Y(d_s, d_r)$. A definition slightly different than the previous one (Equation 3.36) of *dist* is

used here and is given by :

$$dist((i,j),(k,l)) = abs(i-k) + abs(j-l) \tag{4.13}$$

The direction of a pair of pixels, (i,j) and (k,l) is denoted here by $dir((i,j),(k,l))$ and the arbitrary values assigned to four standard directions are,

0 - horizontal

1 - vertical

2 - diagonal

3 - counter-diagonal

Thus $Y(d_s,d_r)$ can be expressed as shown below,

$$Y(d_s,d_r) = \frac{1}{N(d_r)} \sum_{(i,j),(k,l)} y(i,j) \cdot y(k,l) \tag{4.14}$$

with $dist((i,j),(k,l)) = d_s$ and $dir((i,j),(k,l)) = d_r$; where, $N(d_r)$ denotes the number of such pixels in the direction of d_r. The summation is taken over a $2l+1$ long strip, 3 pixels wide, symmetric around the pixel (m,n) and the strip is along the direction of d_r', where the d_r' indicates the direction orthogonal to d_r. For $d_s = 0$, only those pixels which lie on the line of d_r' passing through (m,n) are used for calculating $Y(0,d_r)$. Thus $N(d_r)$ for $d_r = 0, 1$ and 2 are,

$$N(0) = 2l+1 \tag{4.15}$$

$$N(1) = 2(2l+1) \tag{4.16}$$

$$N(2) = 2l+1 \tag{4.17}$$

The product $Y(d_s,d_r)$ is normalized with the local mean $\bar{y}(m,n)$.

To make the algorithm simpler, the weights can be made independent of the direction and in this case, the weights denoted by $w(d_r)$ become a function only

of the distance d_r. The weights $w(0)$, $w(1)$ and $w(2)$ provided before processing the speckled image. Different weights were chosen to test speckled images and the empirical relations found among the three weights are as follows :

$$\begin{aligned} w(1) &= Kw(0) \\ w(2) &= -(w(0) + w(1)) \end{aligned} \tag{4.18}$$

where K is a positive integer between the range 2 to 5.

The algorithm can be further refined to make it more sensitive to edges. This is done by determining the probable occurrence of an edge at a pixel and the probable direction and strength of the edge. A ratio $r(d_s)$ which measures approximately the probability that the pixel estimated belongs to an edge, is defined as :

$$r(d_r) = max\left(\frac{Y(0.d_r)}{Y(2,d_r)}, \frac{Y(2,d_r)}{Y(0,d_r)}\right) \tag{4.19}$$

The larger the value of $r(d_r)$, the higher is the probability that the estimated pixel is a part of an edge in the direction of d_r. So, calculating d_r's for all possible 4 directions, the direction d_{rm} which gives the maximum ratio is determined and only the products ($Y(d_s, d_{rm})$s) are used for subsequent processing. This method has a big disadvantage, since in the vicinity of an edge in a particular direction, the ratio corresponding to the orthogonal direction is most likely to be maximum which may affect the resulting image in a severe manner. To avoid this, it is suggested that it be checked whether or not the least ratio is below a certain range (typical values found are (0.01 - 0.1)). A very low value of $r(d_r)$ implies that the pixel is near the edge and it is better to avoid nonlinear filtering on it. So, a second check is provided to determine whether or not a pixel should be subjected to nonlinear filtering.

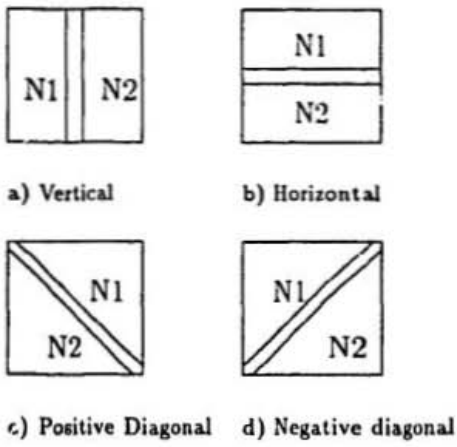


Figure 4.5: Neighborhood pair oriented in four different directions

In all the above methods, an overall scaling factor c is applied to the quadratic response of the filter. It can be adjusted properly after some trials; with the images tested, the scale is found to be in the range of $0.05 - 0.5$. In the following chapter, only the "Proportional weight coefficient" and "4-direction-oriented" QVFs are used. To be consistent, the weights for the three types will be denoted as W_0, W_1 and W_2 for both methods.

4.2.3 Ratio of Averages edge detector

The typical gradient type edge detector is not suitable for detecting edges on speckled images. Since speckle is multiplicative in nature, a ratio of averages edge detector seems to work better. One such edge detector is proposed by Touzi *et al.* [60, 61]. Based on these studies, a simplified form of a ratio of averages is used in the following edge detector.

- **Modified Ratio of Averages (MROA) edge detector:**

  To determine whether an edge is present at a given location of an image, an

edge strength measure is needed. A window of a given size $N \times N$ centred at the pixel of interest, is split into two contiguous neighbourhoods for each of the four usual directions as shown in Fig. 4.5. The averages of the pixels in the contiguous neighbourhoods for the four pairs are calculated. Thus, the edge strength measured at a particular pixel located at (x, y) for ith pair of neighbourhood can be defined as :

$$R_i = min(P_i/Q_i, Q_i/P_i); \quad i = 1 \cdots 4 \tag{4.20}$$

where, P_i and Q_i are the averages of the pixels in the two neighbourhoods. The edge strength is measured for four directions and since a lower value of R_i implies that an edge is more likely to be present at that location, the minimum value among the four ratios is determined and finally the ratio R is obtained,

$$R = min(R_1, R_2, R_3, R_4) \tag{4.21}$$

The estimated edge strength has to be thresholded using a predefined threshold T and the desired edge map is obtained. Too low and/or too high values of T and N, respectively, might result in an edge map having a large number of missing edges. On the other hand, too high and/or too low values of T and N, respectively, might yield large numbers of spurious edges and very thick lines in the edge map. So it is very important to choose optimal values of T and N which compromise between missing edges and ambiguous and/or wrong edges in the edge map. For good performance, the range of T and N has been found to be $0.2 - 0.8$ and $5 - 11$, respectively.

- **Ratio and Gradient of Averages (RGOA) edge detector**

It has been noticed that above method suffers from a limitation. It tends to

locate more edges in darker regions than edges in brighter regions even if the contrast is held constant. This makes the method very susceptible to noise (false edges) in darker areas if edges belong to a brighter area in the same image, have to be detected. The effect cannot be noticed and hence can be ignored in high contrast images. But this effect is very prominent in the images with low contrast regions. This is understandable because the ratio of averages rather than the gradient is being used as an edge measure. But for correlated speckled images the multiplicative nature of noise is compensated partially by the correlation property of the speckle. Undoubtedly, a gradient type edge detector would not work properly. However, the combination of these two types of detectors might work better than any one of these. This is attempted in this work and much better results have been obtained. An edge is detected if *either* the ratio-magnitude $R < T_r$ or the gradient-magnitude $G > T_g$, where R is calculated using Equation 4.20 and 4.21 and G is calculated in a similar fashion as shown below. The gradient magnitude for the four directions ($i = 1, 2, 3$ and 4) are calculated as :

$$G_i = \ abs(P_i - Q_i); \ \ i = 1, \ldots, 4 \tag{4.22}$$

and hence the gradient magnitude is obtained by :

$$G = max(G_1, G_2, G_3, G_4) \tag{4.23}$$

Therefore, two thresholds must be set to use this edge detector. But as N increases, the edge map becomes thicker making it necessary to apply a method to thin the edge map. A simple edge thinning process developed by Shanmugham *et al.* [54] is used following the edge detection. A brief illustration of the algorithm is given in the following.

4.2.4 Edge thinning algorithm

As mentioned in the previous section, it is very important to have a thinning process following the edge detection particularly when it is required to have an edge map for noisy images. A very fast two-pass algorithm is proposed by Shanmughan *et al.* [53]. The algorithm uses a binary image which is in this case the output of the edge detector and scans through this image twice – horizontally and vertically. The algorithm as proposed requires two inputs for each pixel - i) the edge magnitude $R(i,j)$ and ii) the edge direction $D(i,j)$. The edge magnitude is obtained from the edge map which is the output of the edge detector. For a binary edge map, it is standard to assume $R(i,j) = 1$ for edge pixels and $R(i,j) = 0$ for pixels which do not belong to an edge. Most edge detectors usually do not provide edge direction information. A simple strategy is proposed here to set the direction of the edge at a given pixel point (i,j). The algorithm is summarized below.

Horizontal thinning:

- For a given row index i, a connected string of 1s is searched for. A string of 1s in any horizontal scanning implies :

$$R(i,j) = 1 \tag{4.24}$$

  for all $j = j_1, j_2, \cdots j_n$ corresponding to a string of 1s.

- The direction of the edge is checked. $D(i,j)$ is set to 1, if it is in the horizontal direction, otherwise to 0. The value of $D(i,j)$ is determined using the following expression :

$$D(i,j) = \begin{cases} 1 & \text{if } n < W_m \\ 0 & \text{otherwise} \end{cases} \tag{4.25}$$

where n indicates the number of 1s connected in a string and W_m is the maximum possible width of an edge. It depends on the type of edge detection used and directly on the window size used in detecting edges. In present application, it is suggested that W_m be about twice of N, where $N \times N$ is the window size used by the edge detector.

- If $D(i,j) = 1$, for a string of 1s, all values of $R(i,j)$ are set to 0 except,

$$R(i,k) = 1 \tag{4.26}$$

where k is the midpoint of the string given by :

$$k = \frac{j_1 + j_n}{2} \tag{4.27}$$

Vertical thinning:

- For a given column index j, a connected string of 1s is searched for. A string of 1s in vertical scanning implies :

$$R(i,j) = 1 \tag{4.28}$$

for all $i = i_1, i_2, \cdots i_n$ corresponding to the string of 1s.

- A nonzero value of $D(i,j)$ indicates a vertical edge. The value of $D(i,j)$ is determined using the following expression,

$$D(i,j) = \begin{cases} 1 & \text{if } n < W_m \\ 0 & \text{otherwise} \end{cases} \tag{4.29}$$

- If $D(i,j) = 1$ for a string of 1s all values of $R(i,j)$ are set to 0 except,

$$R(k,j) = 1 \tag{4.30}$$

where k is the midpoint of the string given by :

$$k = \frac{i_1 + i_n}{2} \tag{4.31}$$

Thus the output R is replaced by the thinned edge map.

4.3 Concluding remarks

So far, different methods for implementing speckle-specific QVFs have been discussed. To avoid the high complexity in design and computation, simple experimental methods have been proposed to set the quadratic coefficients. The justification for the experimental design method comes from the results of filtering as will be shown in the next chapter. The next chapter presents the results obtained with the QVF and also with other filters of interest. A comparative study is made and based on that a critical discussion is included at the end of the next chapter.

Chapter 5

Results

5.1 Introduction

The previous chapters of this thesis have discussed the development of various filtering methods and, to some extent, edge detection methods for speckle corrupted images. This chapter focuses on comparison between the performance of the filters in terms of their noise smoothing and edge preserving abilities. Image processing results of the various filters are presented as well as quantitative measures of noise smoothing and edge preserving. A critical review of the performance of the filters is also provided in this context. Edge maps obtained from the Ratio and Gradient of Averages (RGOA) edge detection method for speckled images, before and after filtering, are presented too.

Outputs are obtained for the following filters,

- Liuear Average
- Median
- Homomorphic

- Multiplicative Lee
- Two-point Taylor
- Sigma
- Frost
- Quadratic Volterra

All filters are applied to three different images; named, "bars", "annular" and "balloon". The first two images are fairly simple, artificially created with high contrast, containing two grey levels and chosen to demonstrate the filters' ability in edge preservation. The third image is more complicated with many shapes and edges of different contrast. These three images are combined in a single image called "combine" which is shown in Fig. 5.1. This particular image is presented throughout this study.

Two different images shown in Fig. 5.2(a) and Fig. 5.2(b) are generated for one-look and four-look speckle, respectively. It is assumed in all algorithms that the noise characteristics and the independent number of looks are known.

It is not feasible to provide all results for both one-look and four-look speckled images. However, since in most cases SAR images are multi-look and since $L = 4$ is a very common choice, only the results of processing the four-look speckle degraded image will be presented.

Different edge detection methods are attempted on the various filtered images to show their ability to preserve edges. Most of the usual edge detectors perform poorly as mentioned earlier due to the special type of noise involved. Therefore,

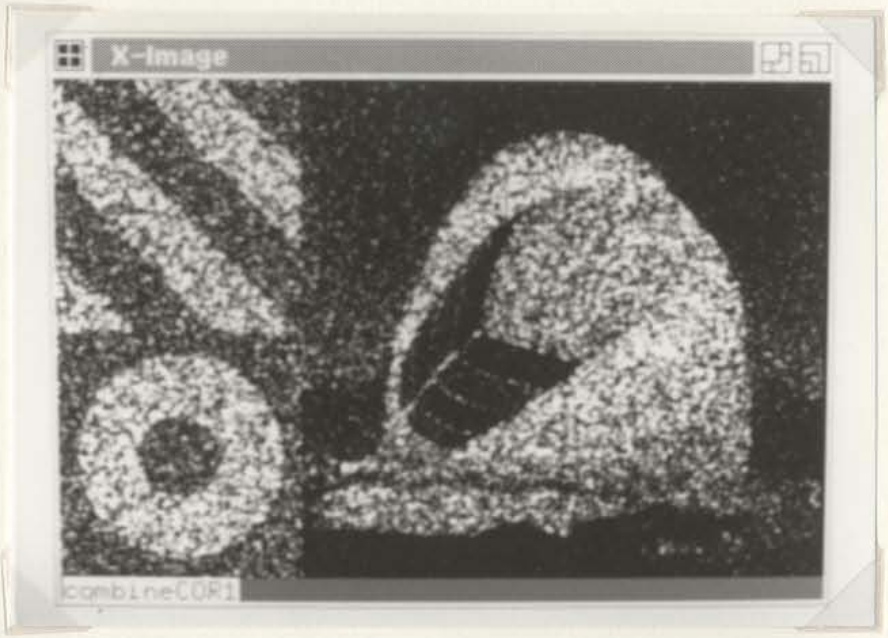
Figure 5.1: Original "combine.pic" image

only the results obtained with the Ratio and Gradient of Averages (RGOA) detectors (discussed in detail in Section 4.2.3) are presented in this chapter.

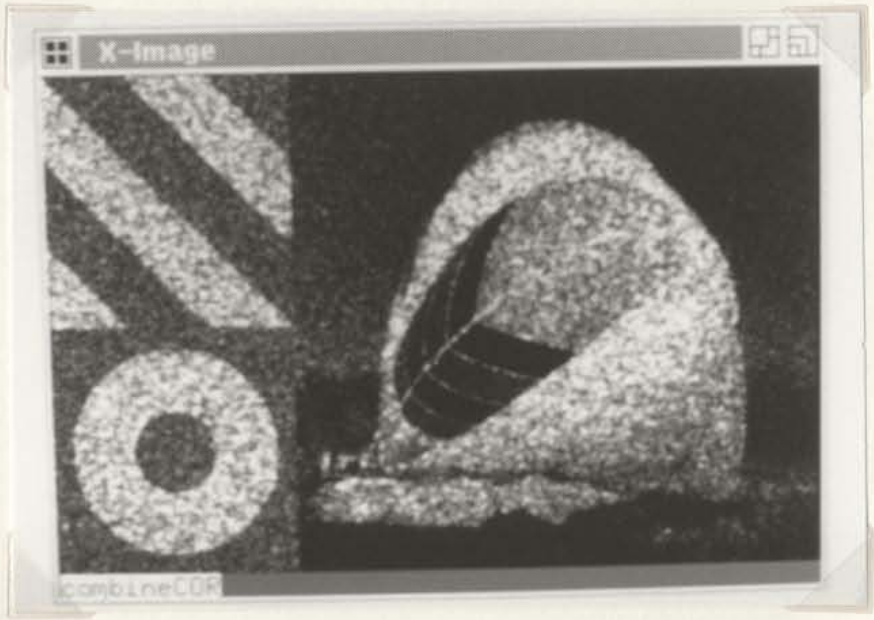
All images are stored in files with pixel values ranging from $[0-255]$. Although the values are stored in "byte" ("char" in "C") format to minimize space, all processing was done with double precision. All algorithms are written in the "C" programming language to run under the Ultrix operating system on DEC-2000/5000 stations.

5.2 Filtering results

The results obtained upon applying the above mentioned filters on the test image are presented in this section. All filters other than Quadratic Volterra Filter (QVF) are applied for two different window sizes. Since the QVF is the main interest of this thesis, some basic experimental results are also shown for this particular filter. A



(a)



(b)

Figure 5.2: Speckle corrupted image (a) one look , (b) 4 look

different image "border" is used for this purpose. Finally the QVF has been applied to the test image "combine" and the results obtained for different combinations of the filter parameters are presented.

5.2.1 Linear filter

This is the only linear filter in the above list, and has been used to provide a benchmark. It demonstrates the importance of nonlinear filtering for this specific problem of speckle. For $N > 3$, the output image becomes very hazy. The images obtained with two different window sizes of 3×3 and 5×5 are shown in Figs. 5.3(a) and (b)

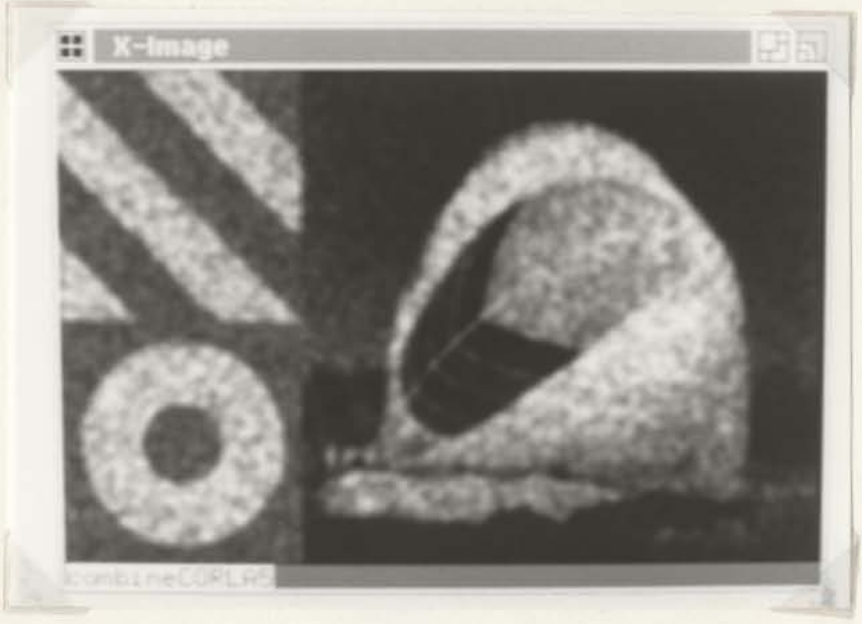
5.2.2 Median filter

Although very simple, this filter provides good image enhancement results in some cases, and has been selected in different comparison studies [12, 19, 33, 38, 50]. In present implementation, various window sizes have been used in an attempt to find a best trade-off between noise smoothing and edge enhancement; a value of $N = 5$ seems to be the best choice. However, output images obtained with $N = 5$ and $N = 7$, are presented in Figs. 5.4(a) and (b).

5.2.3 Homomorphic filter

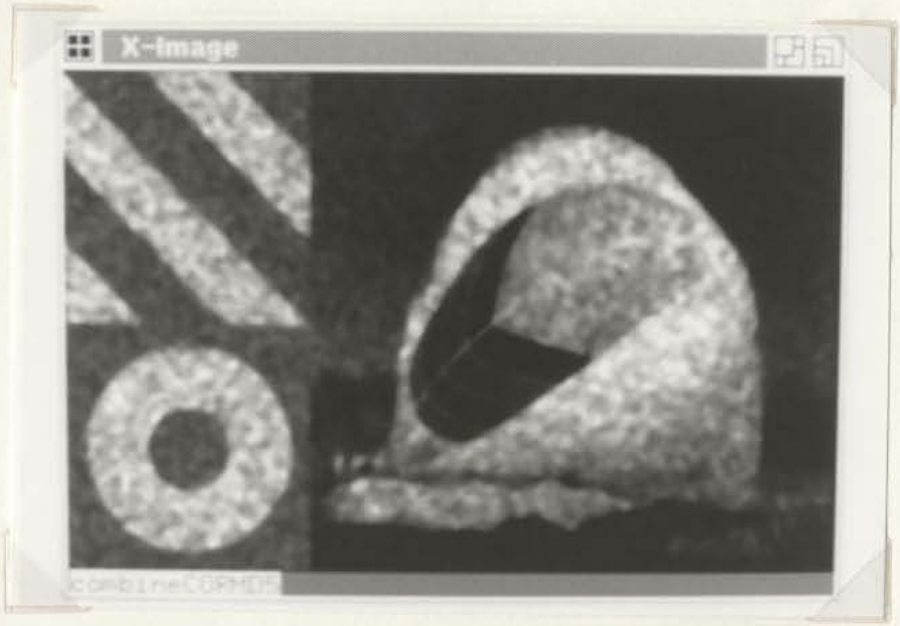
This is theoretically good filtering technique for the multiplicative noise model. A simple box-type, 5×5 linear average filter is used as the low pass filter in the interior of the homomorphic cascade. The resulting images are shown in Figs. 5.5(a) and (b) for $N = 5$ and $N = 7$, respectively.

(a)



(b)

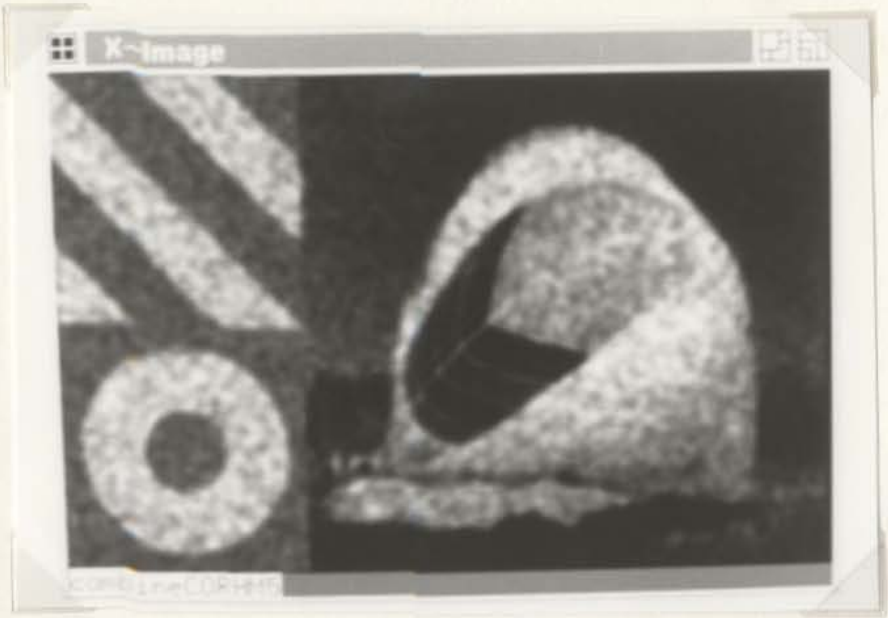
Figure 5.3: Linear average filter, window size (a) 3×3 , (b) 5×5



(a)

(b)

Figure 5.4: Median filter, window size (a) 5×5 , (b) 7×7



(a)

(b)

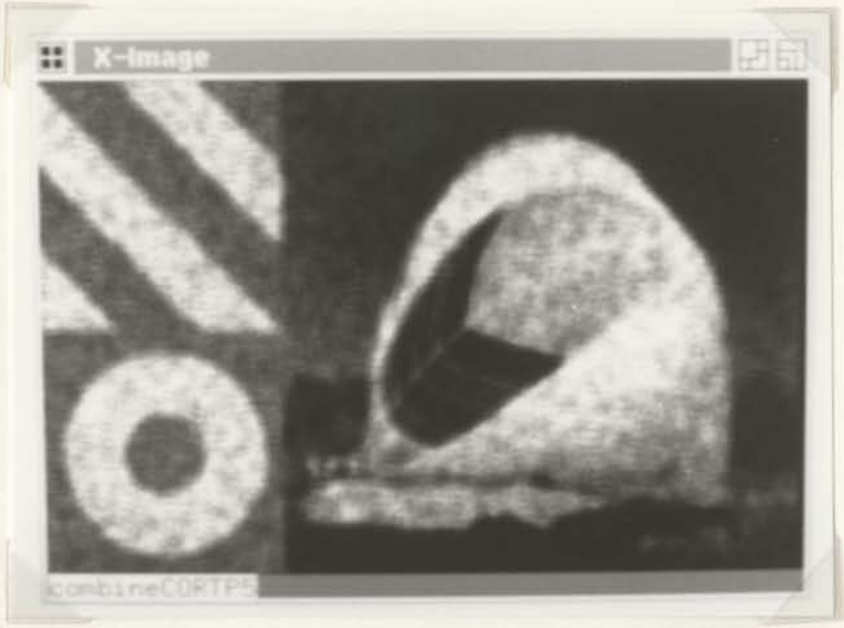
Figure 5.5: Homomorphic filter, window size (a) 5×5, (b) 7×7

5.2.4 Two-point Taylor filter

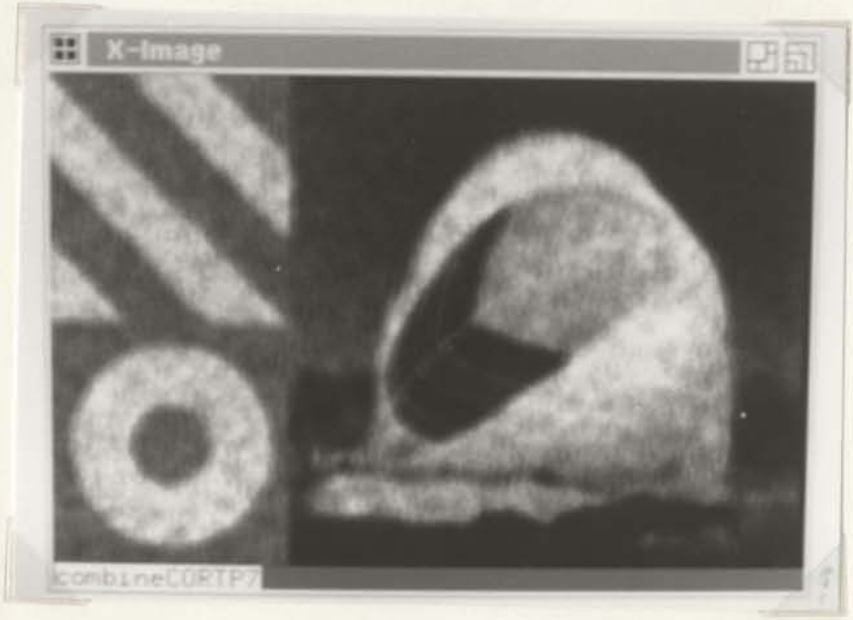
Although it is very sensitive to the noise model as mentioned in the literature [38], it is considered as one of the possible choices as an edge preserving speckle smoother. Various "process descriptors" are required for this particular algorithm. These are calculated from the noise file which has been generated for modelling speckle and has been stored in a separate file. These values are used later while processing the image. Also, it seems important to provide lower and upper limits for the power coefficients a and b to ensure stability. After several trials, $(-0.2,0.2)$ is chosen as the boundary values for both a and b. The output in Figs. 5.6(a) and (b) are the results of using window sizes of 5 and 7, respectively.

5.2.5 Multiplicative Lee filter

As the name implies, this filter is based on a multiplicative noise model. However, there are many previous studies [16, 23, 30] that use multiplicative filters on speckled image since the multiplicative property of speckle is prominent. It is however difficult to supply the proper value of the noise variance. The overall image yields a standard deviation to mean ratio of $1/\sqrt{L}$ (L is the number of independent looks), so a variance of $\sigma_n = 1/L$ seems to be appropriate. But from the model of speckle noise, the multiplicative noise process has a χ-square distribution with a standard deviation of $1/M$. The standard deviation of the latter distribution, $\sigma_n = 1/L^2$, is also worth attempting. Both of these values have been attempted and it appears that the filter works better at $\sigma_n = 1/L$. The images shown in Figs. 5.7(a) and (b) represent the output at $\sigma_n = 0.25$ and for $N = 5$ and $N = 7$, respectively.

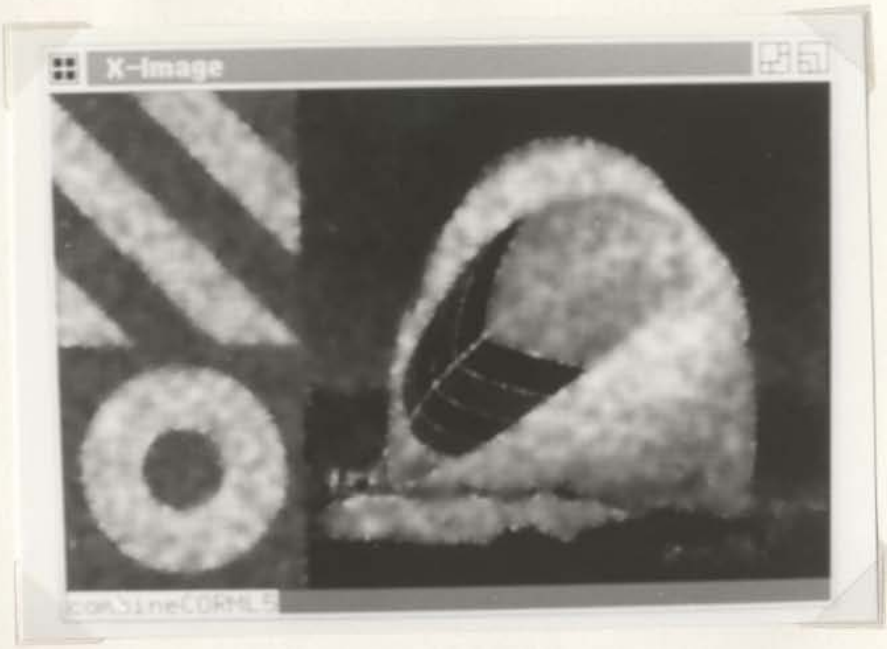


(a)

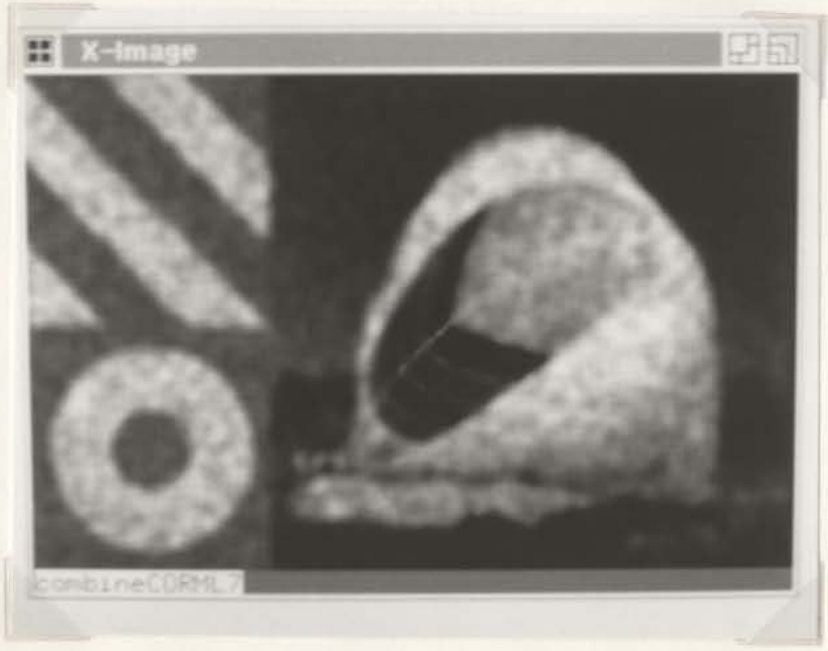


(b)

Figure 5.6: Two point Taylor filter, window size (a) 5×5, (b) 7×7



(a)



(b)

Figure 5.7: Multiplicative Lee filter, window size (a) 5×5, (b) 7×7

5.2.6 Sigma filter

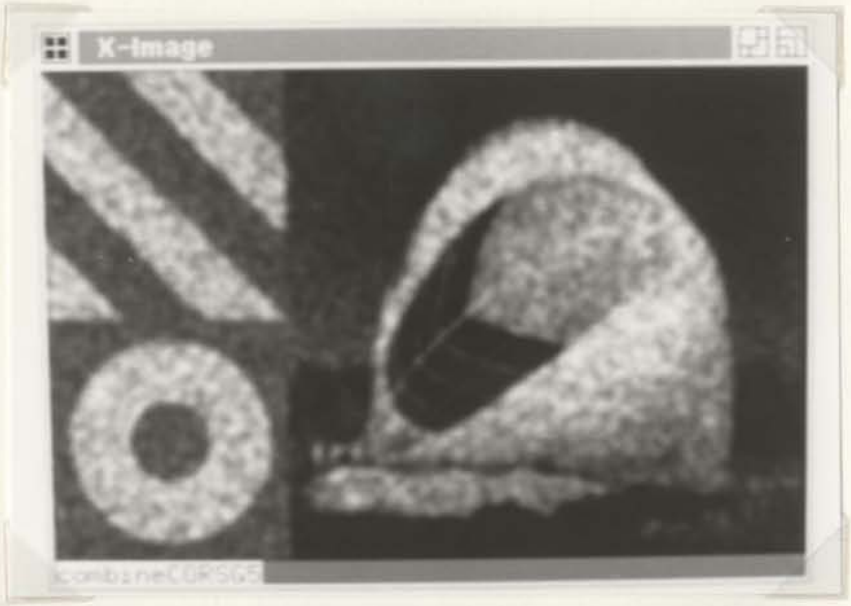
Among the few speckle specific filters, this filter has been chosen. Both 5×5 and 7×7 regions of support are used to estimate the local mean and variance. The resulting images are presented in Figs. 5.8(a) and (b).

5.2.7 Frost filter

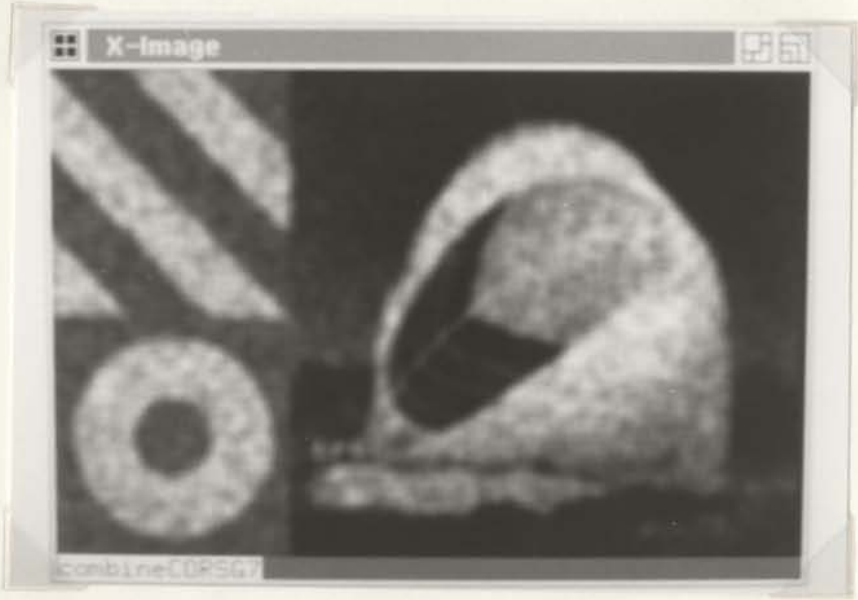
This is another speckle specific filter which has been designed and used for speckle smoothing purpose. Although this filter is based on a multiplicative model of speckle, it seems worth investigating its performance on images satisfying the multiplicative-convolutional speckle model. The filter requires the knowledge of the noise variance and the number of independent looks. Both values of 0.25 and 0.0625 have been used for the reason described above in Section 5.2.5. However the output images of Figs. 5.9(a) and (b) show the result at $\sigma_n = 0.25$ and for $N = 5$ and $N = 7$, respectively.

5.2.8 Quadratic Volterra filter

The quadratic Volterra filter (QVF) has been applied to the speckled images. It was intended to investigate the impact of the different coefficient values, namely W_0, W_1 and W_2, the scaling factor c, the threshold T and the window size $N \times N$ as described in details in Section 4.2.2. Only the 4-direction-oriented method is chosen for this experiment because this implementation can give an overall idea of the basic principle of the Volterra filter. A simple 128×128 image called "border" is used to carry out some basic experiments. The original "border" image and its corrupted version are depicted in Figs. 5.10(a) and (b). One-dimensional horizontal and vertical filters are applied separately to demonstrate the effect of these filters on horizontal and

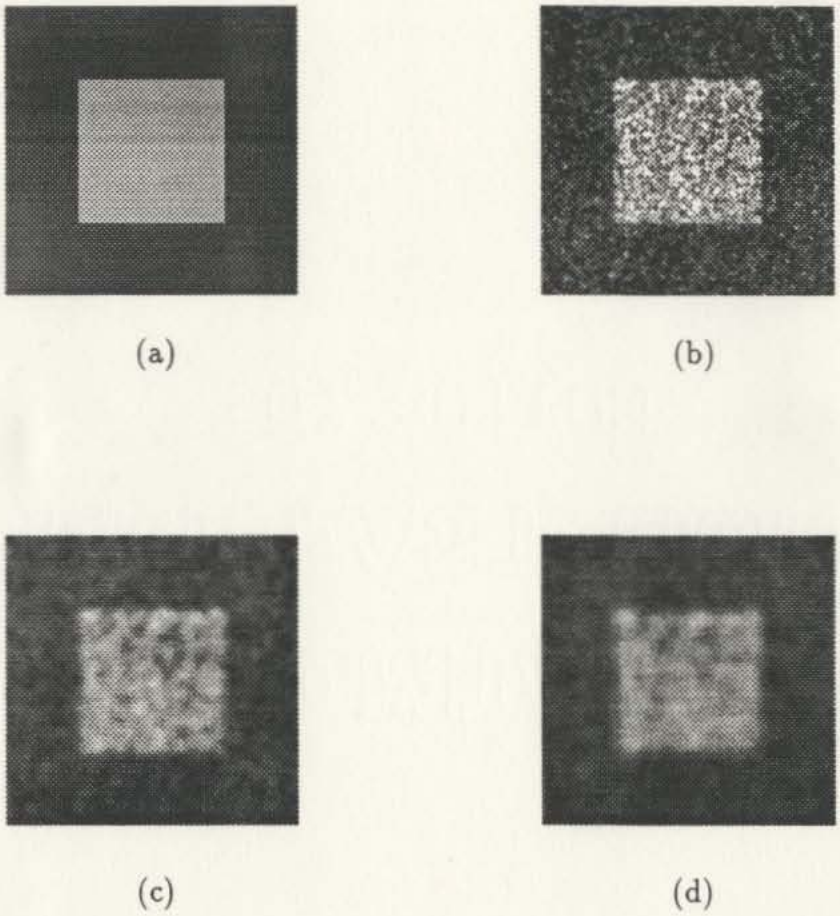


(a)



(b)

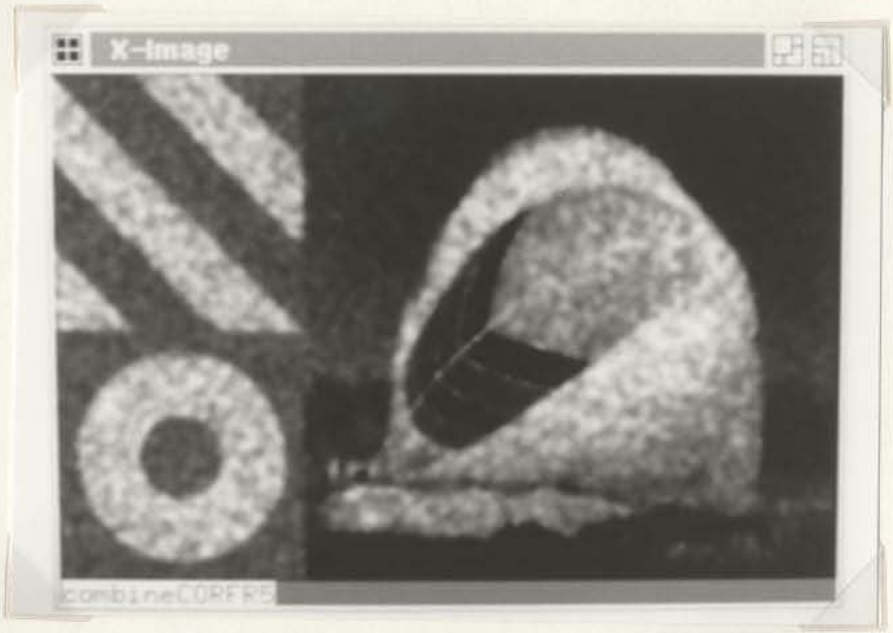
Figure 5.8: Sigma filter, window size (a) 5×5, (b) 7×7



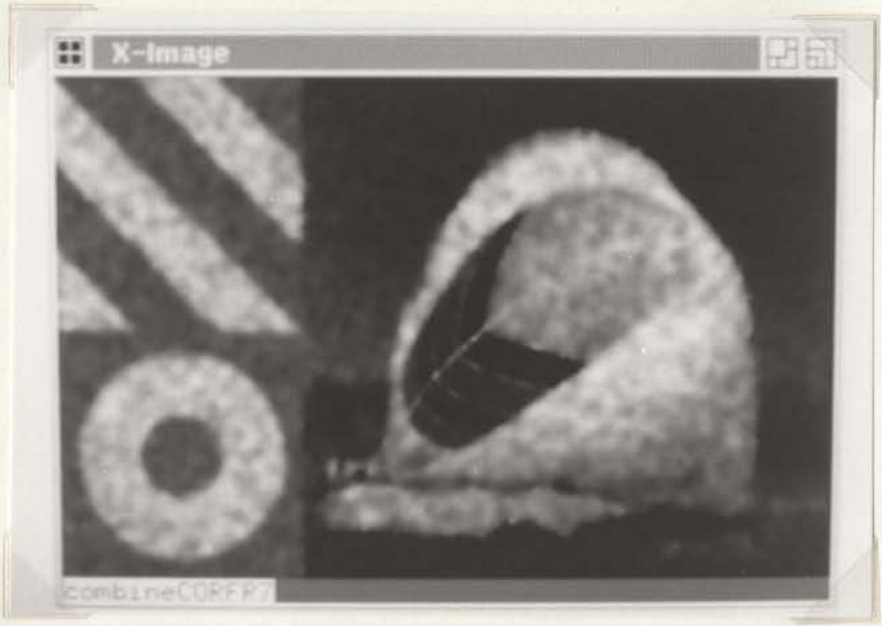
(a) (b)

(c) (d)

Figure 5.10: "Border" image (a) Original, (b) Speckle corrupted, Linear average filter (c) 5×5, (d) 7×7.



(a)



(b)

Figure 5.9: Frost filter, window size (a) 5×5, (b) 7×7

vertical edges. Different combinations of these parameters are chosen as shown in Table 5.2.8. Due to thresholding, only some parts of the total pixels are subjected to nonlinear operations. The thresholding is based on the local statistics of the image. For homogeneous regions, the quadratic operator is better to avoid the quadratic operation, although it is very important for the edgy regions. The value N_q shows the number of such pixels out of the total pixels of $128 \times 128 = 16384$.

The effect of the linear lowpass filter alone is also demonstrated in Figs. 5.10(c) and (d). It is interesting to note that the QVF; horizontal, and the QVF; vertical, play an important role in enhancing the edges in the horizontal and vertical directions respectively and that they present a great improvement over the linear lowpass filter which does the smoothing quite well but causes severe blurring effects. Figs. 5.11(a),(b) and (c) show the effect of vertical filtering on the speckled image. As expected, the vertical edges are quite clear. Different thresholds have been attempted to demonstrate the effect of thresholding on the speckled images. A low threshold of $T = 0.9$ causes unnecessary filtering on some pixels which belong to homogeneous regions ($N_q = 6453$), resulting in some bright spots within those areas. Whereas a higher threshold of $T = 1.1$ reduces this effect, but a very high threshold of $T = 1.3$ results in a blurry image because the quadratic part is inactive for most of the pixels ($N_q = 827$ only).

Figs. 5.12(a), (b) and (c) show the horizontal filtering on the same speckled image with different window sizes. As expected the bigger the region of support, the smoother the homogeneous areas. But the edges do not appear as sharp as they are for smaller window sizes. Figs. 5.13(a), (b) and (c) show the results of the simultaneous application of horizontal and vertical filters on the speckled image. Figs. 5.13(a)-(c) illustrate the effect of scaling factor c on the performance of the filter. As seen from

QVF	Figs.	T	N	c	W_0	W_1	W_2	N_q	MSE
Original	5.10(a)	-	-	-	-	-	-	16384 (100%)	0.00
Corrupted	5.10(c)	-	-	-	-	-	-	-	803.80
Vertical-I	5.11(a)	0.9	7	0.25	1	3	-4	6453 (39.40%)	225.91
Vertical-II	5.11(b)	1.1	7	0.25	1	3	-4	1882 (11.48%)	171.54
Vertical-III	5.11(c)	1.3	7	0.25	1	3	-4	827 (5.04%)	164.13
Horizontal-I	5.12(a)	1.1	7	0.25	1	3	-4	1882 (11.48%)	157.80
Horizontal II	5.12(b)	1.1	5	0.25	1	3	-4	1622 (9.90%)	191.65
Horizontal-III	5.12(c)	1.1	9	0.25	1	3	-4	2190 (13.36%)	157.16
Combined-I	5.13(a)	1.1	7	0.25	1	3	-4	1882 (11.48%)	175.58
Combined-II	5.13(b)	1.1	7	0.35	1	3	-4	1882 (11.48%)	152.57
Combined-III	5.13(c)	1.1	7	0.15	1	3	-4	1882 (11.48%)	246.10
Edge Adap.-I	5.14(a)	1.1	7	0.25	1	3	-4	1745 (10.65%)	166.54
Edge Adap.-II	5.14(b)	1.1	7	0.25	3	1	-4	1745 (10.65%)	197.60
Edge Adap.-III	5.14(c)	1.1	7	0.25	0.2	4.8	-5	1745 (10.65%)	159.78

Table 5.1: QVFs with different parameters (Image: "border")

T - Decision-directed threshold

N - Length of the filter

c - Overall scaling factor of the quadratic output,

W_i - Weight assigned to Type-i quadratic coefficients,

N_q - Total number of pixels subjected to quadratic operations,

MSE- Mean Squared Error.

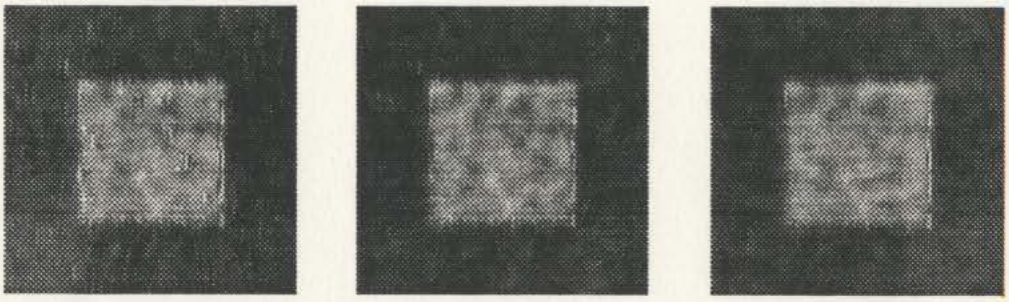
Fig. 5.13(b), very large scaling factor ($c = 0.4$) might cause some pixels to have very high intensities. On the other hand, too small a scaling factor ($c = 0.1$) might not be enough to compensate for the excess smoothing done by the linear part of the filter in the edgy areas.

It is important to mention here that there is an adverse effect of using both filters simultaneously as described in Section 4.2.2 and seen in Figs. 5.13(a)-(c). In the vicinity of vertical edges , there is a series of horizontal stripes and similarly, in there is a series of vertical stripes near horizontal edges. The horizontal response of the QVF is much smaller than the vertical response around horizontal edges (one or two pixels away from true edge pixel) and the reverse is also true. Recalling Section 4.2.2, if instead of combining two responses, the maximum response is taken in order to have more prominent edges, this effect will become even more severe. However, the effect is substantially minimized by checking whether a pixel belongs to an edge or is near an edge during filtering.

Finally, Figs. 5.14(a)-(c) show the results obtained from this particular implementation. The results are for different combinations of W_0, W_1 and W_2. It is obvious that for $W_0 = 3W_1$ (Fig. 5.14(a)), the resulting output is more prone to error, whereas for $W_1 = 3W_0$, the output (Fig. 5.14(b)) is quite satisfactory.

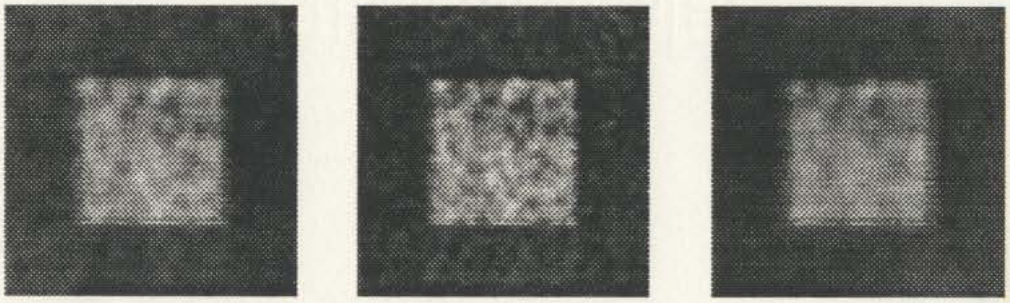
Seemingly, it is rather difficult to choose a proper set of values of threshold T, scaling factor c, the window size N etc., from the above experiments. The following factors are found to be very useful in choosing these values.

- Too large a threshold T might result in some pixels being missed which should be subjected to nonlinear filtering; on the other hand, too low a threshold might cause some pixels to undergo nonlinear filtering which should only be filtered with the linear part. Thus, an optimization may be required to achieve the



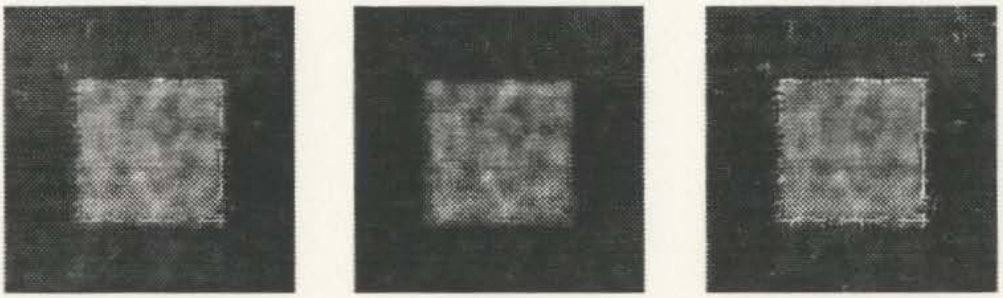
(a) (b) (c)

Figure 5.11: QVF (vertical), (a) $T = 0.9$, (b) $T = 1.1$, (c) $T = 1.3$



(a) (b) (c)

Figure 5.12: QVF (horizontal), (a) $N = 7$, (b) $N = 5$, (c) $N = 9$



(a) (b) (c)

Figure 5.13: QVF (horizontal and vertical), (a) $c = 0.25$, (b) $c = 0.35$, (c) $c = 0.15$

(a) (b) (c)

Figure 5.14: QVF (horizontal and vertical edge adaptive)

(a) $W_0 = 1.00, W_1 = 3.00, W_2 = -4.00$ and $c = 0.25$,

(b) $W_0 = 3.00, W_1 = 1.00, W_2 = -4.00$ and $c = 0.25$,

(c) $W_0 = 0.20, W_1 = 4.80, W_2 = -5.00$ and $c = 0.25$.

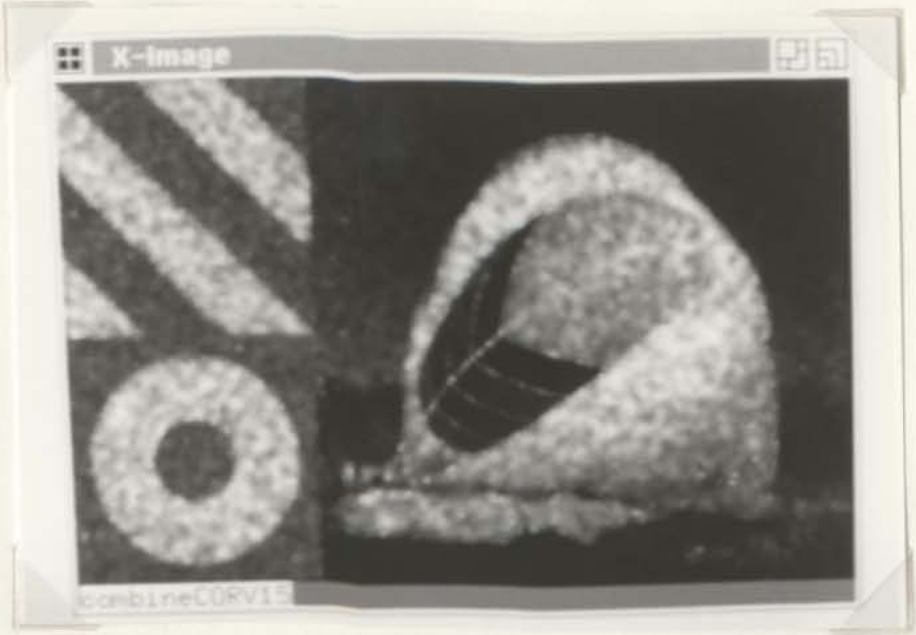
Filters	Figs.	T	t	W_0	W_1	W_2	N_0
QVF-I, 5×5	5.15(a)	1.0	0.5	0.5	1.5	-2.0	15968 (17.69%)
QVF-I, 7×7	5.15(b)	1.2	0.5	0.5	1.5	-2.0	7903 (8.75%)
QVF-II, 5×5	5.16(a)	1.2	0.25	1.0	2.0	-3.0	12698 (14.07%)
QVF-II, 7×7	5.16(b)	1.2	0.3	1.0	1.0	-2.0	16897 (18.72%)

Table 5.2: QVFs with different parameters (Image: "combine")

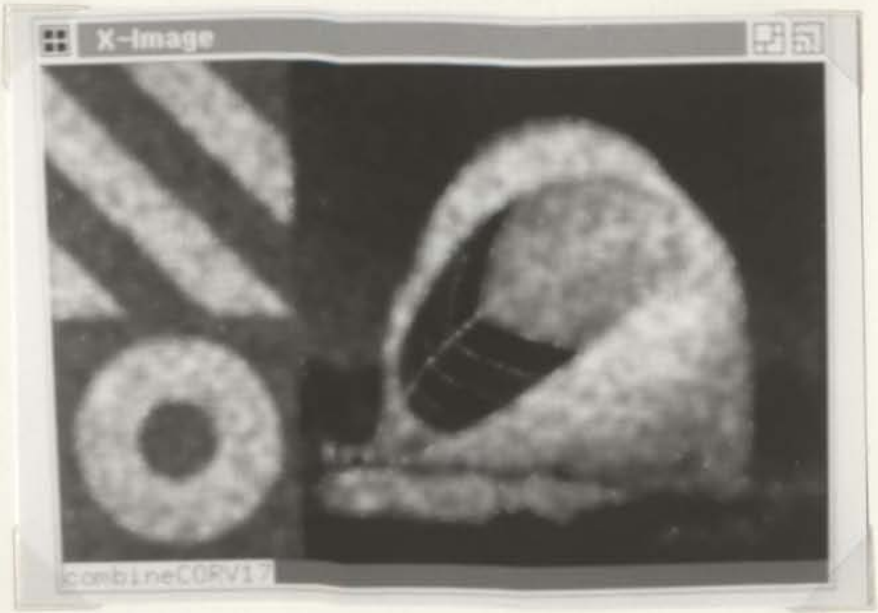
required performance.

- For a larger window size, a smaller threshold is required.
- Too high a value W_0 might result in presence of impulse noise in the processed the image. To avoid this, it is necessary to reduce W_0.

After acquiring some knowledge about choosing the various parameters characterizing a QVF, different QVFs have been selected and used in smoothing the speckle while preserving the underlying structure from the test image named "combine". All methods described in Section 4.2.2, have been attempted. The best results are obtained with the "Proportional weight coefficients" and the "4-direction-oriented" methods, denoted by "QVF-I" and "QVF-II", respectively. Table 5.2 shows the different combinations of the filter parameters used to obtain the outputs shown in Figs. 5.15 and 5.16. The MSEs are included in Table 5.3.

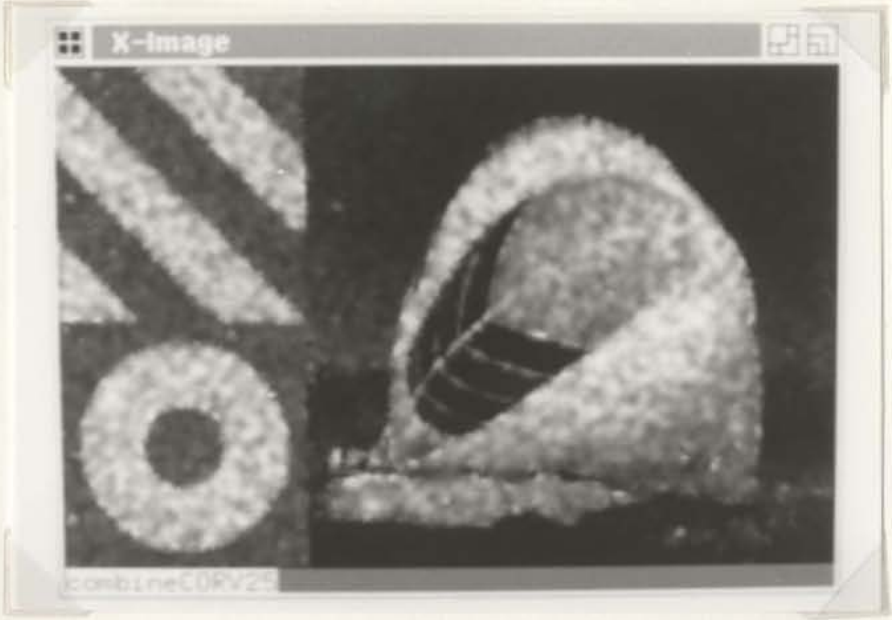


(a)

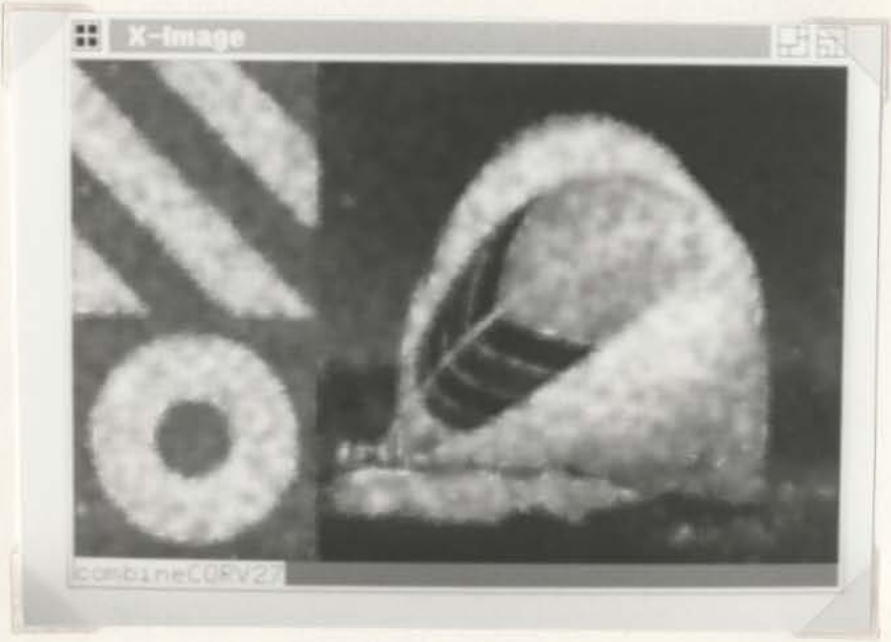


(b)

Figure 5.15: QVF-I, window size (a) 5×5, (b) 7×7



(a)



(b)

Figure 5.16: QVF-II, window size (a) 5×5, (b) 7×7

Filters/Images	Figs.	Ranges		Mean, Std			SNR		MSE
		min	max	Global	Dark	Bright	Dark	Bright	
Original	5.1	5–238		111,64.15	100,0.00	200,0.00	∞	∞	0.00
Speckled	5.2(b)	3–255		110,67.55	100,23.33	197,37.37	4.29	5.30	751.13
Lin. Av., 3 × 3	5.3(a)	5–255		109,63.23	100,13.43	197,20.92	7.51	9.44	335.72
Lin. Av., 5 × 5	5.3(b)	9–247		109,61.38	102,11.06	197,13.95	9.24	14.13	266.98
Median, 5 × 5	5.4(a)	7–255		108,62.20	99,9.38	197,15.75	10.63	11.15	278.68
Median, 7 × 7	5.4(b)	9–253		107,61.04	100,8.32	196,12.78	12.08	15.37	258.46
Homo., 5 × 5	5.5(a)	7–246		106,60.56	99,9.65	193,14.31	10.30	13.54	295.88
Homo., 7 × 7	5.5(b)	10–237		105,59.33	100,9.27	192,11.39	10.82	16.91	318.39
Mul. Lee, 5 × 5	5.7(a)	4–238		109,61.41	101,9.47	196,10.53	10.70	18.67	203.35
Mul. Lee, 7 × 7	5.7(b)	9–238		109,60.39	103,11.14	196,10.85	9.28	18.10	259.77
Two pt, 5 × 5	5.6(a)	6–255		121,68.77	114,14.20	220,18	8.05	11.85	505.74
Two pt, 7 × 7	5.6(b)	2–255		120,68.23	114,13.16	219,17.97	8.68	12.21	503.86
Sigma, 5 × 5	5.8(a)	9–247		109,61.38	102,11.05	197,13.94	9.25	14.13	266.95
Sigma, 7 × 7	5.8(b)	13–238		109,60.18	103,11.14	196,10.85	9.28	18.10	276.86
Frost, 5 × 5	5.9(a)	3–247		109,61.78	101,10.46	197,14.21	9.71	13.87	244.75
Frost, 7 × 7	5.9(b)	3–238		109,60.96	102,9.61	196,10.81	10.64	18.17	222.92
QVF-I, 5 × 5	5.16(a)	7–252		110,61.73	102,13.62	197,14.24	7.54	13.84	279.43
QVF-I, 7 × 7	5.16(b)	13–250		110,60.27	103,12.20	196,10.85	8.48	18.10	279.40
QVF-II, 5 × 5	5.15(a)	1–255		111,62.13	103,16.04	197,13.94	6.46	14.13	333.91
QVF-II, 7 × 7	5.15(b)	1–255		111,60.75	104,14.20	196,10.72	7.36	18.34	327.21

Table 5.3: Quantitative measures for noise smoothing

5.2.9 Quantitative measures

As mentioned in previous chapters, it is important to have some quantitative measures to evaluate the filters' performance and to make a comparison between them. To evaluate the noise smoothing of the filters, the global MSE and SNR are calculated for the speckled image and all filtered images. Also MSE and SNR are measured for two sampled 25×25 regions which in the original "combine" image has homogeneous intensities of 100 and 200. Various noise smoothing measures are presented in Table 5.3.

5.3 Edge detection results

For the edge preservation measures, it is necessary to first produce the edge maps of the images before and after filtering. Edge maps are first generated using the modified Bovik's ratio of averages edge detector (MROA) (Section 4.2.3) for the speckled image and all the filtered images. It has been observed that the edge detector based on only the ratio of averages does not perform a good job on speckled images. Therefore, the combination method named RGOA in Section 4.2.3 is used instead and much better results are obtained. Most of the filters do not perform very well. However, the best four found are those which are derived from the following filters,

- Median
- Multiplicative Lee
- Frost
- QVF-I

Hence, only the edge maps obtained from the outputs of the above filters are presented here. Since it is important for a quantitative study to have a true or ideal edge map, an edge map of the original image is generated using the Robert's method [18] as shown in Fig. 5.17. The edge map for the corrupted image is shown in Fig. 5.17. Other edge maps for the filtered images are presented through Figs. 5.18 - 5.22.

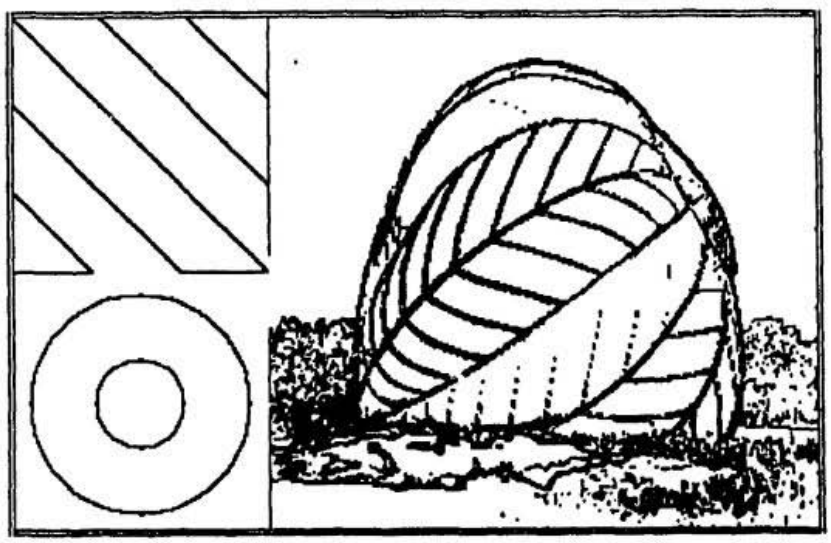


Figure 5.17: True edge map of the original image "combine"

It should be mentioned here that the quality of an edge map is quite sensitive to the edge detector mask N and the thresholds chosen for the ratio and gradient magnitudes T_r and T_g respectively. Edge maps have been generated for several different combinations of N, T_g and T_r for each filtered output. However for the same filtered image, there is no significant difference among the various edge maps obtained with different parameters. Only one from each filter and the corrupted image have

been shown in Figs. 5.18-5.22, however, quantitative measures are presented for two different combinations of N, T_r, and T_g in Table 5.4.

5.3.1 Quantitative measure

Recalling Section 3.5, different factors denoted by C, M, W and A, which determine the quality of edge maps, are calculated as shown in Table 5.4. The number of edge pixels found is also included in this table. A good edge preserving filter should yield a high value of C, and low values of M, W and A, as discussed in Section 3.5.2.

5.4 Critical Review

In this section, a detailed discussion is provided based on the obtained results. As emphasized before, the intention was to compare different filtering algorithms for their solution of the problem of speckle in terms not only of noise smoothing but also edge preservation. However, there is always a question of tradeoff between noise smoothing and edge preservation and hence it is difficult to achieve both at the same time. Most of the commonly used image restoration filtering methods work very well in terms of smoothing but often the resulting image loses its edge resolution to some extent. If the noise is not significant (e.g. additive Gaussian noise or even multiplicative with unity mean and a small variance between 0.01 and 0.04), the price paid by losing resolution for smoothing is not noticeable, but for a significant noise like speckle, smoothing itself cannot provide a good quality image unless the edge preservation is also substantially good. The comparison presented in this thesis is based on both quantitative measures and subjective visual quality of the image. Although standard, the quantitative measures used are not very effective in judging

Images /Filters	Figs.	N	T_g	T_r	$\#found$	C in %	M in %	W in %	A in %
Original	5.17	5	35	—	12827	100.00	0.00	0.00	0.00
Speckled	—	7	60	0.7	5724	68.28	22.16	18.71	28.45
,,	5.18	9	55	0.7	4509	67.87	27.65	7.52	26.40
Median	—	7	35	0.72	4569	66.24	32.35	15.86	22.72
,,	5.19	9	32	0.75	4893	66.64	28.04	16.84	25.92
Mul. Lee	5.20(b)	7	22	0.75	4668	67.65	27.34	11.00	26.51
,,	—	9	22	0.75	4600	69.91	27.51	12.78	25.00
Frost	5.21	7	32	0.70	4793	67.31	28.72	14.98	25.04
,,	—	9	30	0.68	4606	66.66	29.81	11.27	24.89
QVF	5.22	7	32	0.7	4715	67.85	28.35	10.28	25.25
,,	—	9	30	0.68	4553	67.28	30.07	11.07	24.34

Table 5.4: Quantitative measures for edge preserving

$$C = \frac{2 \times \#correct}{\#true + \#found}$$

$$M = \frac{\#miss}{\#true}$$

$$W = \frac{\#wrong}{\#true}$$

$$A = \frac{\#ambig}{\#true}$$

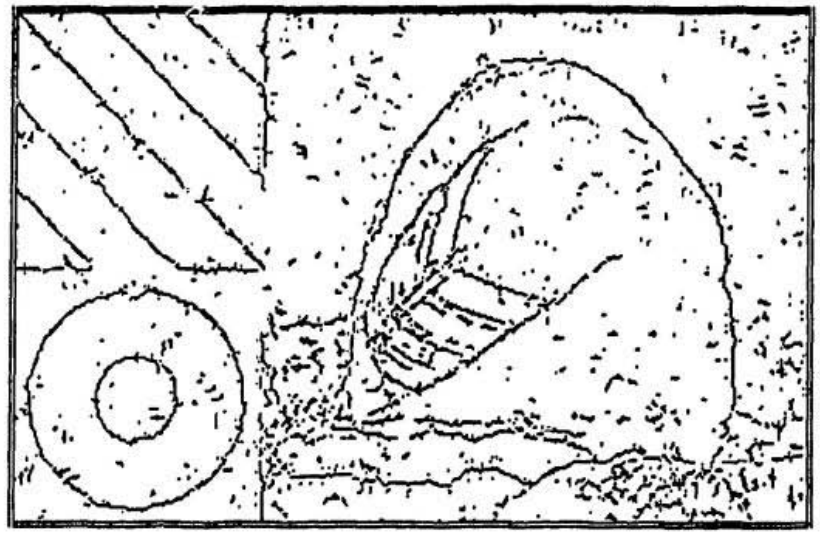


Figure 5.20: Edge map of the speckle corrupted image

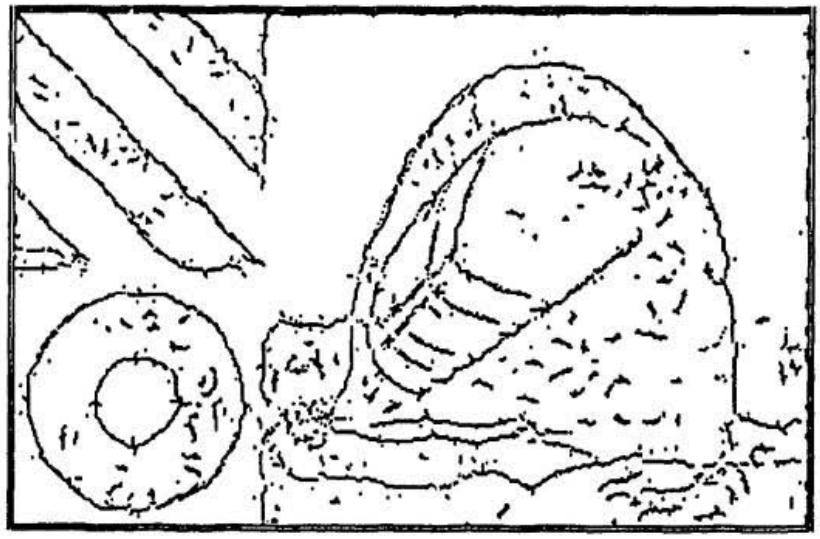
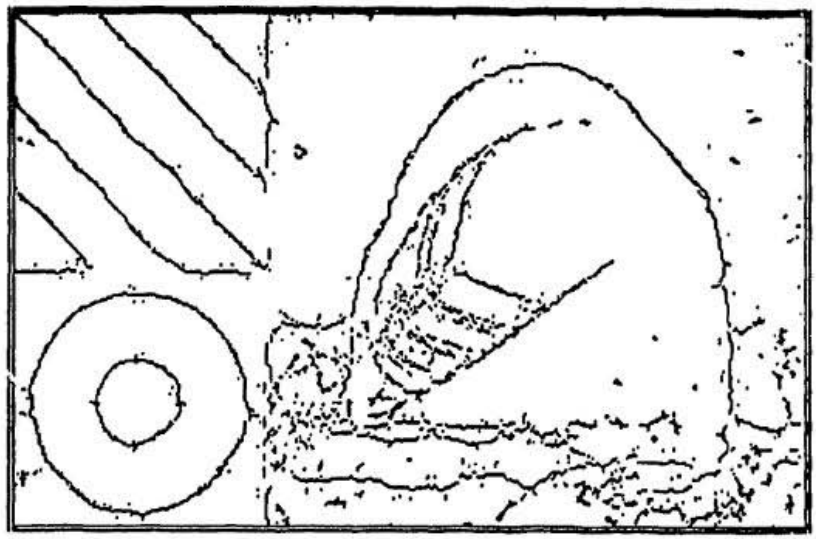
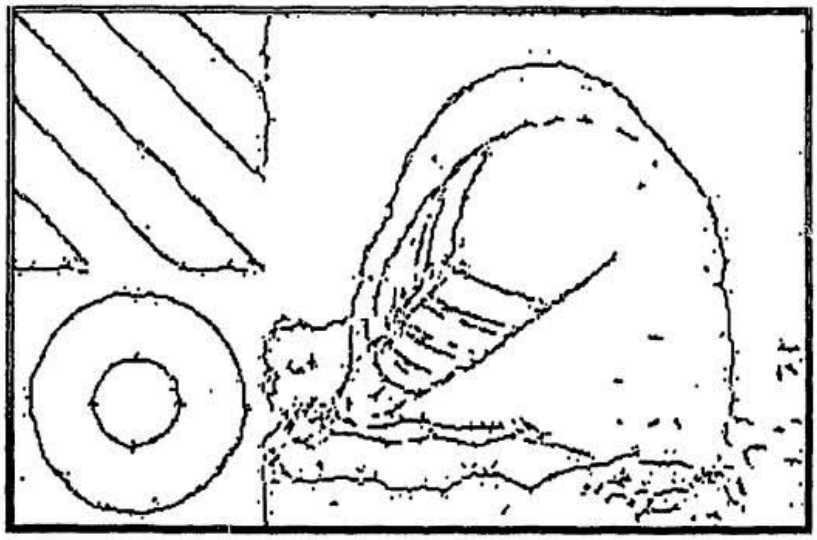


Figure 5.21: Edge map of the filtered image (Median)



(a)



(b)

Figure 5.22: Edge map of the filtered image (Multiplicative Lee) using (a) MROA, (b) RGOA edge detectors

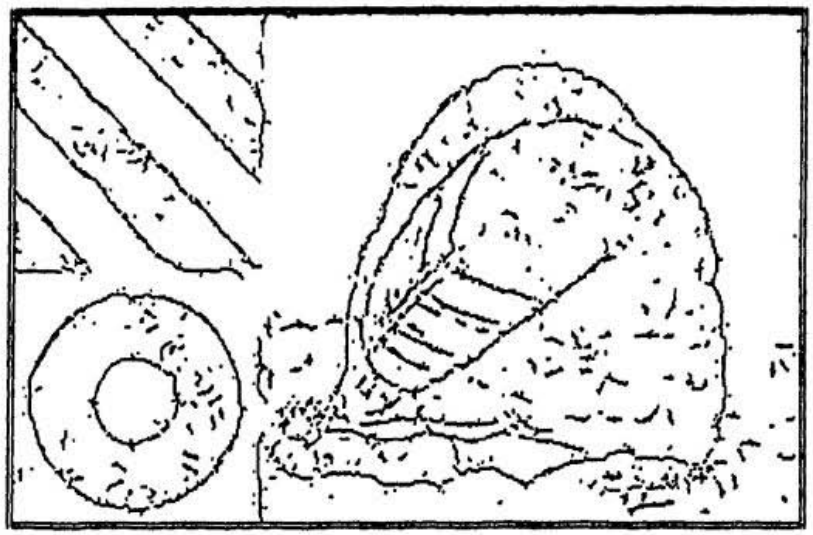


Figure 5.23: Edge map of the filtered image (Frost)

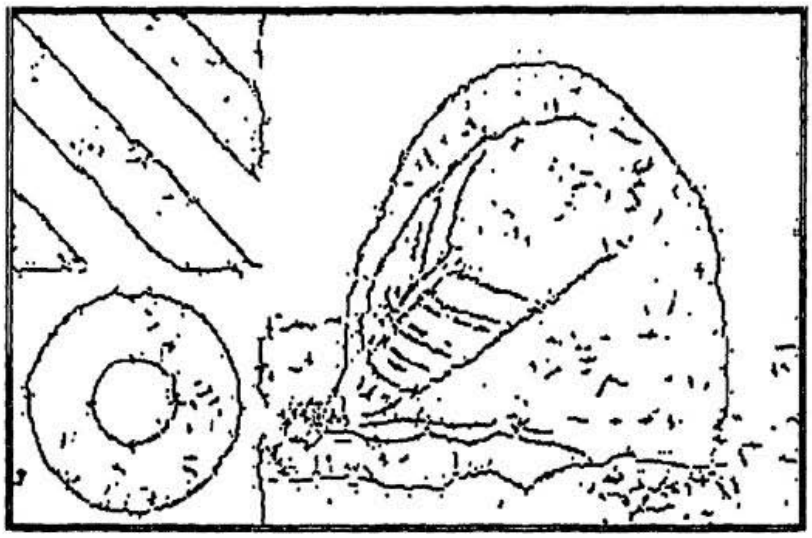


Figure 5.24: Edge map of the filtered image (QVF)

the quality of the images [3, 38]. So, visual interpretation is equally or even sometimes more important.

Although it was anticipated that the linear average filter would yield a smoothed image, it does not perform the smoothing very well. For a window size of $N = 3$, the homogeneous areas (e.g. on the slant bar or the annular section), are still quite noisy. However, a larger window size of $N = 5$ gives a better result in terms of the smoothing but shows a loss of resolution. For a window size higher than $N = 5$, the picture becomes very blurry and the fine edges are almost lost as shown in Fig. 5.3. This filter was not expected to work well because it is a linear filter and the problem of interest here is undoubtedly a nonlinear one.

The median filter performs quite a good job on the speckled image. It is interesting to notice that even for a window size $N = 7$, the filter attempts to preserve the edges and the overall picture does not seem to blur as it does for linear average filtering. However, the fine edges are mostly lost even for a window size of $N = 5$ as shown in Fig. 5.4. Although it performs better than the linear average filter to some extent, the output is not very satisfactory in terms of both smoothing and edge retention. The poor performance of the filter is no surprise because it is an *ad hoc* nonlinear filter and not matched to any specific model. Nevertheless, the nonlinear nature of the filter makes it possible to work better than a linear average filter which again provides evidence of the need of a nonlinear filter for the problem.

The next filter used in this study is the homomorphic filter which is based on a multiplicative noise model. But since speckle is not purely multiplicative, it does not perform very well for speckle noise, as it has a very poor edge preserving capability as shown in Fig. 5.5. Another problem with the homomorphic filter is the difficulty of choosing an appropriate filter for the intermediate step between the "logarithmic"

and "exponentiation" operation. In this study, a box type average filter is used and it does not seem to do a good job even in smoothing out the noise.

The multiplicative Lee filter performs much better job as shown in Fig. 5.7. Although it is meant for multiplicative Gaussian or uniform noise [27], it works quite satisfactorily for speckle too. However, the thin stripes on the balloon seem to have almost disappeared, especially for the higher window size of $N = 7$. A window size of 5×5 gives the best overall result. For this particular filter, the smoothing is excellent even for $N = 5$, whereas there is no substantial improvement on edge preserving ability over the other filters used in this study. The vicinities of the edges seems to be more prone to errors. However, the filter's optimization between noise smoothing and edge preservation is praiseworthy.

The two-point Taylor filter which is based on the Taylor series does not work properly in the case of speckle noise as shown in Fig. 5.6. However, this is not very surprising as it was mentioned in the literature [38] that the algorithm is very sensitive to the noise model. The algorithm is developed for pointwise multiplication and has been tested previously with images corrupted with Gaussian multiplicative noise and having a small variance in the range of $(0.01 - 0.04)$. But speckle has a much higher variance depending on the mean of the image and also has a statistical distribution which is different from Gaussian. The two coefficients used in this algorithm are restricted to be in a given range. After a few trial and error experiments, a boundary of $(-0.2,0.2)$ has been fixed for both the power coefficients a and b which give reasonable output. But the mean square estimated values of both the coefficients a and b are beyond this range for most of the pixels in the test image. The image appears to be very bright for a wider range of coefficients. So it is suspected that this filter has a stability problem in filtering speckled images.

Although the Frost filter was proposed for speckle noise removal, the filter does not take the correlation property of speckle into account. The output for $N = 5$ and $N = 7$ shown in Figs. 5.9(a) and (b) seem to be better than the first three filters and competes quite well with that of the multiplicative Lee filter. But as seen in all previous cases, it also cannot overcome the limitation of low resolution output. The fine edges have almost disappeared from the image after filtering with $N = 7$ and are also not very clear for $N = 5$.

The results obtained from the Sigma filter as they appear in Figs. 5.8(a) and (b) are not at all good in terms of edge preservation. The main reason is probably that the filter is very simple and based on a multiplicative model of speckle. For a moderate window size, the pixels belonging to a window are more likely to be within two standard deviations for speckle because speckle has inter-pixel correlation and hence the filter works almost much like a linear average filter for speckled images.

The Quadratic Volterra filter is however the one which was expected to perform very well. Although all methods discussed in Section 4.2.2, have been attempted on the test image, only the results obtained from the "Proportional weight" and "4-direction-oriented" methods as denoted by QVF-I and QVF-II in Table 5.2, have been presented. The results agree with the theory quite well. As mentioned earlier, it was expected to perform better especially from the edge preservation point of view. This is quite obvious from the output. Although the quantitative measures do not show a major difference in the measuring factors from the other filters of interest, the visual quality of the image shows the difference. Since the quadratic Volterra has a spatial memory, it takes into account the correlation property of the speckle and works according to this correlation. Although, it was intended to make this filter edge-adaptive, it has been found that it is very difficult to get the edge information out of

speckled images, since the standard deviation of the resulting image is very high for any realistic speckled image. Moreover, the inter-pixel correlation property of speckle makes it more difficult to distinguish between true and false edges. The inter-pixel correlation of speckle gives rise to many dark and bright patches scattered randomly all over the image with different shapes and sizes. So, any edge-adaptive approach responds to the edges of these patches before it can respond to low-contrast but true edge pixels. So if a lower threshold is used to take all true edge pixels into account, the resulting image becomes contaminated with randomly scattered bright spots in some of the areas which were originally homogeneous. With the QVF-I, another difficulty arises since the filter uses only a 3×3 region for estimating a particular pixel which does not seem to be large enough. However, the QVF-II compensates for this problem. To avoid the huge complexity of using the whole 2-D QVF structure for $N > 3$, four one-dimensional QVFs are used in four different directions. Since they are each basically 1-D filters, they combine with an adverse effect which is quite visible near the edges of the image in Fig. 5.16(a), where the reason is discussed in Section 5.2.8. A better result was also anticipated in terms of smoothing. A box-type linear filter works on the homogeneous areas; this may not be the best choice.

So far the discussion involves the visual interpretation of the output images from different filters. At this point, it is important to ask whether the different quantitative measures are consistent with the visual interpretation. The noise smoothing measures give some important information about the noise smoothing ability of the filters. The corrupted image has a global MSE of 751.13. The SNR is also quite poor for (dark,bright) regions having a value of (4.29, 5.30). However, it should be pointed out that the standard deviation of 37.37 in the bright region is much higher than that of 23.33 in the dark region. This was expected because of the multiplicative nature of

speckle. The filters, in general, show a substantial lowering of the MSE. As evident from the Fig. 5.7(a), the multiplicative Lee filter shows the lowest MSE of 203.35 for $N = 5$. This is interesting because normally the higher the window size, the lower is the MSE. The (dark,bright) *std*, *mean* and SNR are also satisfactorily improved for this filter. However, some other filters (*e.g.* the Frost and the Sigma) have these values which are quite close although visually the smoothing does not appear as good as it is for the multiplicative Lee filter. The noise smoothing quantities for the QVF-I and QVF-II are also consistent with their visual qualities. Although, the MSEs are slightly higher than some of the filters, they are reasonably good.

The edge maps are generated using the modified ratio of averages edge detector which combines two previous edge detectors [9, 60] as mentioned earlier in Section 4.2.3. It has been observed that this edge detector does not perform well. The detector used, makes only slight differences among the edge maps obtained for different filters. Fine details are almost all lost in the edge map obtained, although they are quite clear in some of the filtered images with the human eye. A threshold and the window size are the two variables in the edge detector and they have a great influence on the quality of the edge maps . Although, several different combinations have been attempted, none of which could provide truly satisfactory results. Also, since all edge detection measures are calculated using the edge maps, (i.e. fully dependent on the quality of the edge map), the quantities obtained are contradictory. Previous studies [9, 60, 61] which employed the ratio of averages edge detector for speckled images, take into account the multiplicative property of the speckled images. But, the problem does not seem to be resolved satisfactorily for correlated speckle noise. It has been observed that the detector does not work properly in brighter regions. At a low value of the threshold, for the ratio magnitude, some edges which are visually quite distinct,

do not appear in the edge map. On the other hand, for a higher threshold (for higher threshold, more edges are found), some spurious edges start appearing in the darker areas. This is quite evident from the edge map shown in Figs. 5.20(a). Some spurious edges begin to appear at the top of the "balloon" image (the sky over the balloon) as shown in Fig. 5.1, which looks very uniform and dark with no distinct edges. The middle segment of the balloon is quite distinguishable but the upper portion of this segment which has a brighter mean intensity does not appear very clearly. Also, all thin edges on the bottom part of the middle segment are broken in the edge map for the same threshold. However, the ratio magnitude used in the ratio of averages edge detector tends to detect more edges in the overall darker region than it does for the same contrast in the brighter regions. This may happen due to because speckle being not purely multiplicative rather having inter-pixel correlation. Hence, the combined method which detects the edges both from the gradient and the ratio magnitude as described in Section 4.2.3 would be more appropriate. Therefore, this edge detector is applied and much better results are obtained. Figs. 5.20(a) and (b) confirm the effectiveness of the proposed RGOA edge detector. The edge measures calculated using these edge maps agree quite well with the visual interpretation.

Using RGOA edge detector, maximum number of correct edges are detected after filtering with the multiplicative Lee filter ($C = 69.91$) as shown in Table 5.4. However, the QVF works satisfactorily giving a correction factor $C = 67.85$. The number of wrong edge pixel detected is also quite low ($W = 10.28$) which is slightly higher for the best performance of the multiplicative Lee filter ($W = 12.78$) as shown in Table 5.4.

The other two filters under examination are not as good as the two filters, mentioned above. Edge maps generating from the Median filter tend to produce many

false edge pixels with relatively small number of correct pixels (*e.g.* an edge map from the Median filter yields $W = 15.86$ and $C = 66.24$ as shown in Table 5.4). The Frost filter works better than the Median filter. However, if the number of correct pixels increases, it produces more false edge pixels. For example, as shown in Table 5.4, an increase of $C = 67.31$ from $C = 66.66$ is obtained with an increase in false edge pixels ($W = 11.27$ to $W = 14.98$).

In terms of overall performance, the multiplicative Lee is the best among all these filters under study. Although the quadratic Volterra filter does not outperform as it was expected to do, it does perform well and attempts to trade off between smoothing and edge preservation. The limitations of the QVFs are described earlier in this chapter. Some suggestions are provided in the next chapter for future extension and improvement of present work.

Chapter 6

Conclusions

This thesis focuses on the statement and complexity of the speckle problem in SAR images. It also investigates different possible filtering techniques to solve the problem of restoring speckled images without losing much of the image edge details. It was not expected that an absolute solution of the problem would be developed; rather, the intention was to investigate the magnitude of the problem, probable solution methods using existing filtering techniques and the applicability of a relatively new filtering approach based on quadratic Volterra filter to the solution of the stated problem. Present work deals with a model of SAR images. It has been emphasized throughout this thesis that the filters used for smoothing speckle should also provide good edge preservation. These criteria are important particularly for certain applications such as image segmentation, coast line detection, accurate volume measurement, etc. [15, 25, 26]. Poor resolution is a common problem with any filtering method as it may lead to misinterpretation of detail in these images. As mentioned earlier, the existing speckle-specific filters are mainly based on a multiplicative model and they have not been developed with special attention to their ability for edge preservation. Since a realistic SAR model involves a multiplicative-convolutional model rather than a

purely multiplicative model, the commonly used nonlinear filtering methods are not optimal for this purpose. Consequently, further investigation into this problem is necessary. The new technique using quadratic Volterra filters has already been proven capable of providing satisfactory answer to some problems [2, 8, 57] similar to speckle. The applicability of this filter to present problem also seems very promising. The results presented in the last chapter shows its effectiveness as an edge-preserving speckle smoother. However, even better results may be possible if the QVF design is optimized. Throughout the course of study, it has been strongly felt that there is a wide scope for research in this area. Theoretically, the Volterra filter is very powerful but its use and methods for its design need to be further explored. Present thesis is a good start towards that goal. In terms of the speckle problem, the following are some of the ideas which could be further investigated,

- A 2-D quadratic Volterra filter with a local window size greater than 3×3.
- A truncated Volterra filter of higher than second order (e.g. cubic Volterra filter).
- A MSE estimation using the quadratic Volterra filter taking the inter-pixel correlation of speckle into consideration.
- A good edge detection method and an edge-adaptive filtering scheme based on this method.
- A different lowpass filter other than simple box type as the linear part of the quadratic Volterra filter. For example, a Gaussian shaped filter could be used.

Apart from its use in the speckle problem, the basic design approach may be used with little or no modification to solve any other nonlinear problems similar to speckle.

The present work undoubtedly provides a guidelines to pursue further study of this topic.

Modeling speckle is also one of the major contributions of this thesis. At some point it is felt that a thorough knowledge of SAR image processing is important to handle speckle. It was also intended to process real SAR data but unfortunately due to some unavoidable circumstances and time constraint it has not been possible. Hence, it is suggested that future researchers work with real SAR data to investigate the applicability of different filters in the real situation. However, the model used here agrees with certain practical criteria, so it can be assumed that the filters will show much the same performance with real data as they have with synthetically generated speckled images. It is also important to test any filtering algorithm on synthetic data in order to study its performance quantitatively. This has been done effectively in this thesis.

As mentioned earlier, there is a great demand for good literature and work in edge detection methods for speckled images. Present work also focusses on the solution to some extent, but the problem seems to be acute enough to justify more time and study.

The problem investigated in this thesis involves many complexities and so cannot be expected to be solved in a final complete form. In particular, the nonlinearity of this problem is very unusual and there has not yet been any established generalized nonlinear system theory which matches the problem well. However, it is important to understand the magnitude of the problem and to realize its effect and also to investigate possible ways of overcoming the problem. This has been illustrated in this thesis. The quadratic Volterra filtering techniques proposed here, are mainly based on experiments, however, this is not very uncommon in image processing research

especially in image restoration which often involves a number of trials and errors before reaching a specific goal. It is strongly believed that the thesis will contribute to at least two different directions of research – firstly, the solution of similar nonlinear problem using quadratic or higher order Volterra filters, and secondly, the solution of speckle related problems and particularly those which involve modelling, smoothing and edge detection for SAR images.

Bibliography

[1] Abramatic, J.F. and Silverman, L.M., "Nonlinear restoration of noisy images", *IEEE Trans. on Pattern Recognition and Machine Intelligence*, Vol. 4, No. 2, 1982, pp. 141-149.

[2] Agazzi, O., Messerschmitt, D.G., and Hodges, D.A., "Nonlinear echo cancellation of data signals," *IEEE Trans. on Communications*, Vol. 30, No. 11, 1982, pp. 2421-2433.

[3] Anderson, G.L. and Netravali, A.N., "Image restoration based on subjective criterion", *IEEE Trans. Systems, Man and Cybernetics*, Vol. 6, 1976, pp. 845-853.

[4] Andrews, H.C. and Hunt, B.R., Digital Image Restoration", Prentice-Hall signal processing series, Englewood Cliffs, NJ, Prentice-Hall, 1977.

[5] Arsenault, H.H. and April, G.V., "Information content of images degraded by speckle noise", *Optical Engineering*, Vol. 25, No. 5, 1986, pp. 662-666.

[6] Arsenault, H.H. and Denis, M. "Image processing in signal dependent noise", *Can. Journal of Physics*, Vol. 61, 1983, pp. 309-317.

[7] Biglieri, E., "Theory of Volterra processors and some applications",*Proc. ICASSP-82*, Paris, France, May 1982, pp. 294-297.

[8] Biglieri, E., Gersho, A., Gitlin, R.D., and Lim, T.L., "Adaptive cancellation of nonlinear inter-symbol interference for voiceband data Trans.ission", *IEEE Journal on Selected Areas in Communications*, Vol. 2, No. 5, 1984, pp. 765-777.

[9] Bovik, A.C., "On detecting edges in speckle imagery", *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 36, No. 10, 1988, pp. 1618-1627.

[10] Canny, J. "A computational approach to edge detection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 8, No. 6, 1986, pp. 679-697.

[11] Chiang, H.H., Nikias, C.L., and Venetsanopoulos, A.N., "Efficient implementation of quadratic digital filters", *IEEE Trans. on Acoustic, speech and signal processing*, Vol. 34, No. 6, 1986, pp. 1511-1528.

[12] Chin, R.T and Yeh, C.L., "Quantitative evaluation of some edge-preserving smoothing techniques", *Computer Vision, Graphics, and Image Processing*, Vol. 23, 1983, pp. 67-91.

[13] Davila, C.E., Welch, A.J., and Rylander, H.G., "A second order adaptive Volterra filter with rapid convergence," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol.35, No. 9, 1987, pp. 1259-1263.

[14] Davis, L. "A survey of edge detection techniques", *Computer Graphics and Image Processing*, Vol. 4, 1975, pp. 248-270.

[15] Durand, J.M., Gimonet, B.J., and Perbos, J.R., "SAR data filtering for classification", *IEEE Trans. on Geoscience and Remote Sensing*, Vol. 25, No. 5, 1987, pp. 629-637.

[16] Frost, V.S., Stiles, J.A., Shanmugan, K.S., and Holtzman, J.C., "A model for radar image and its application to adaptive digital filtering of multiplicative noise", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 4, No. 2, 1982, pp. 157-166.

[17] Frost, V.S., Shanmugan, K.S., and Holtzman, J.C., "Edge detection for SAR and other noisy images", *Proc. IGARSS'82*, Munich, 1-4, June 1987, pp. 4.1-4.9.

[18] Gonzalez, R.C. and Wintz, P., *Digital Image Processing*, 2nd edition, Addison-Wesley Publishing Company, 1987.

[19] Hudson, D. V. and Jernigan, M.E., "Speckle suppression and texture in synthetic aperture radar images", *Technical Report #160-I-120988*, Aug. 1988, Waterloo, Canada.

[20] Jin, S., Wear, S. and Raghuveer, M.R., "Reconstruction of speckled image using bispectra", *Journal of the Optical Society of America*, Vol. 9, No. 3, 1992, pp. 371-376.

[21] Kim, K.I. and Powers, E.J., "A digital method of modeling quadratically nonlinear systems with a general random input," *IEEE Trans. on Acoustic , Speech and Signal Processing*, Vol. Vol. 36, No. 11, 1988, pp. 1758-1769.

[22] Koh, T. and Powers, E.J., "Second-order Volterra filtering and its application to nonlinear system identification," *IEEE Trans. on Acoustic, Speech and Signal Processing*, Vol. 33, No. 6, 1985, pp. 1445-1455.

[23] Kuan, D.T., Sawchuk, A.A., Strand, T.C., and Chavel, P., "MAP speckle reduction filter for complex amplitude speckle images", *Proc. IEEE Pattern Recognition Image Processing conf.*, 1982, pp. 58-63.

[24] Kuan, D.T., Sawchuk, A.A., Strand, T.C., and Chavel, P., "Adaptive noise smoothing filter for images with signal dependent noise", *IEEE Trans. on Pattern Analysis Machine Intelligence*, Vol. 7, No. 2, 1985, pp. 165-177.

[25] Lee, J.S. and Jurkevich, I., "Segmentation of SAR images", *IEEE Trans. on Geoscience and Remote Sensing*, Vol. 27, No. 6, 1989, pp. 674-679.

[26] Lee, J.S. and Jurkevich, "Coastline detection and tracking SAR images", *IEEE Trans. on Geoscience and Remote Sensing*, Vol. 28, No. 4, 1990, pp. 662-668.

[27] Lee, J.S., "Digital image enhancement and noise filtering by use of local statistics", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 2, 1980, pp. 165-168.

[28] Lee, J.S., "Digital image smoothing and sigma filter", *Computer Vision, Graphics, and Image Processing*, Vol. 24, 1983, pp. 155-169.

[29] Lee, J.S., "Refined filtering of image noise using local statistics", *Computer Graphics Image Processing*, Vol. 15, 1983, pp. 380-389.

[30] Lee, J.S., "Speckle suppression and analysis for synthetic aperture radar images", *Computer Graphics and Image Processing*, Vol. 17, 1981, pp. 24-32.

[31] Lin, J.N. and Unbehauen, R., "2-D Adaptive Volterra filter for 2-D nonlinear channel equalisation and image restoration", *Electronic Letters*, Vol. 28, No. 2, 1992, pp. 180-182.

[32] Marr, D. and Hildreth, E.C., "Theory of edge detection", *Proc. R. Soc. London Ser. B. 207*, 1980, pp. 187-217.

[33] Mastin, G., "Adaptive filters for digital image noise smoothing: an evaluation", *Computer, Vision, Graphics, and Image Processing*, Vol. 31, 1985, pp. 103-121.

[34] Mathews, V.J., "Adaptive polynomial filters", *IEEE Signal Processing Magazine*, 1991, pp. 10-26.

[35] McLean, G.F. and Jernigan, M.E., "Hierarchical edge detection", *Computer Vision, Graphics, and Image Processing*, Vol. 44, 1988, pp. 350-366.

[36] Mertzios, B.G., Sicuranza, G.L., and Venetsanopoulos, A.N., "Efficient realization of two-dimensional quadratic digital filters", *IEEE Trans. on Acoustic, Speech and Signal Processing*, Vol. 37, No. 5, 1989, pp. 765-768.

[37] Moloney, C.R. ,"Nonlinear systems for the restoration of images with multiplicative noise", *Ph.D. thesis*, University of Waterloo, Canada, 1988.

[38] Moloney, C.R. and Jernigan, M.E., "Adaptive image estimation based on multiplicative superposition", *Optical Engineering*, Vol. 29, No. 5, 1990, pp. 478-487.

[39] Nagao, M. and Matsuyama, T., "Edge-preserving smoothing", *Computer Graphics and Image Processing*, Vol. 9, 1979, pp. 394-407.

[40] Nahi, N.E., "Role of recursive estimation in statistical image enhancement", *Proc. IEEE*, Vol. 60, 1972, pp. 872-877.

[41] Oppenheim, A.V., Schaffer, R.W., and Stockham, T.G. Jr., "Nonlinear filtering of multiplied and convolved signals", *Proc. IEEE*, Vol. 56, 1968, pp. 1264-1291.

[42] Papoulis, A. *Probability, random variables, and stochastic processes*, second edition, McGraw-Hill, new York, 1984.

[43] Pratt, W.K., *Digital Image Processing*, New York, NY:J. Wiley & Sons, 1978.

[44] Ramponi, G., "Quadratic filters for image enhancement", *Proc. EUSIPCO-88*, Grenoble, France, 1988, pp. 239-242.

[45] Ramponi, G. and Sicuranza, G.L., "Decision directed nonlinear filter for image processing", *Electronic Letters*, Vol. 23, No. 23, 1987, pp. 1218-1219.

[46] Ramponi, G. and Sicuranza, G.L., and Ukovich, W., "A computational methods for the design of 2-D nonlinear Volterra filters", *IEEE Trans. on Circuits and Systems*, Vol. 35, No. 5, 1988, pp. 1095-1102.

[47] Ramponi, G. and Ukovich, W., "Quadratic 2-D filter design by optimisation techniques", *Proc. 1987 int. conf. on Digital Signal Processing*, Florence Italy, 1987, pp. 59-63.

[48] Ramponi, G., Sicuranza, G.L., and Ukovich, W., "An optimization approach to the design of nonlinear Volterra filters", *Proc. EUSIPCO-86*, The Hague, The Netherlands, Sept. 1986, pp. 151-153.

[49] Ramponi, G., "Edge extraction by a class of second-order nonlinear filters", *Electronic Letters*, Vol. 22, No. 9, 1986, pp. 482-484.

[50] Ramponi, G.,"Bi-Impulse response design of isotropic quadratic filters", *Proc. of IEEE*, Vol. 78, No. 4, 1990, pp. 665-677.

[51] Raney, R.K. and Wessels, G.J., "Spatial considerations in SAR speckle simulation", *IEEE Trans. on Geoscience and Remote Sensing*, Vol. 26, No. 5, 1988, pp. 666-671.

[52] Serra, J., *Image Analysis and Mathematical Morphology*, Academic Press Inc., New York, 1982.

[53] Shanmugan, K.S. and Paul, C., "A fast edge thinning operator", *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 12, No. 4, 1982, pp. 567-569.

[54] Shanmugan, K. S., Dickey, F.M., and Green, J.A., "An optimal frequency domain filter for edge detection in digital pictures", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 1, No. 1, 1979, pp. 37-59.

[55] Sicuranza, G.L. and Ramponi, G., "Adaptive nonlinear digital filters using distributed arithmetic," *IEEE Trans. on Acoustics, Speech and Signal Processing*, Vol. 34, 1986, pp. 518-526.

[56] Sicuranza, G.L. and Ramponi, G., "Theory and realization M-D nonlinear digital filters", in *Proc. ICASSP-86*, Tokyo, Japan, Apr. 1986, pp. 20.20.1-20.20.4.

[57] Stapleton, J.C. and Bass, S.C., "Adaptive noise cancellation for a class of nonlinear, dynamic reference channels", *IEEE Trans. on Circuits and Systems*, Vol. 35, No. 2, 1985, pp. 143-150.

[58] Stockham, Jr., T.G., "Image processing in the context of a visual model", *Proc. IEEE*, Vol. 60, 1972, pp. 828-842.

[59] Teklap, A.M. and Pavlović, G., "Image restoration with multiplicative noise incorporating sensor nonlinearity", *IEEE Trans. on Signal Processing*, Vol. 39, No. 9, 1991, pp. 2132-2136.

[60] Touzi, R., Lopes, A., and Bousquet, P., "A statistical and geometrical edge detector for SAR images", *IEEE Trans.on Geoscience and Remote Sensing*, Vol. 26, No. 6, 1988, pp. 764-773.

[61] Touzi, R., Lopes, A., and Bousquet, P., "A statistical and geometrical edge detectors for SAR image segmentation", *Proc. of IGARSS'87 symp.*, Ann Arbor 1987, pp. 1469-1474.

[62] Tur, M., Chin, K.C., and Goodman, J.W., "When is speckle noise multiplicative ?", *Applied Optics*, Vol. 21, No. 7, 1982, pp. 1157-1159.

[63] Ulaby, F.T., Moore, R.K., and Fung, A.K., *Microwave Remote Sensing – Active and Passive*, Vol. III, Addison and Wesley Publishing Company, 1981.