EDGE-ENHANCED SEGMENTATION FOR SAR IMAGES

CHEN JU

Edge-enhanced Segmentation For SAR Images

By

© Chen Ju, B. Eng.

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

August 1997

St. John's Newfoundland Canada

0-612-34190-9

Abstract

Segmentation of Synthetic Aperture Radar (SAR) images is an important step for further image analysis in many applications. However, the segmentation of this kind of image is made difficult by the presence of speckle noise, which is multiplicative rather than additive. Traditional segmentation methods originally designed for either noise-free or White Gaussian noise corrupted images can fail when applied to SAR images.

Different methods have been previously developed for segmenting SAR images corrupted by speckle. One segmentation method was proposed by Lee and Jurkevich which is quite efficient; it first smooths speckle noise to allow regions to be distinguished in the image histogram, then uses histogram thresholding to segment the filtered image. However, some problems exist with their method: in the filtered image, noise is preserved in edge areas and some fine regions are oversmoothed; while in the segmented image, region boundaries are ragged and some fine features are lost.

Based on Lee and Jurkevich's initial work, an edge-enhanced segmentation method is proposed in this thesis. The edge-enhanced segmentation method is automated and based on the iterative application of an edge-enhanced speckle smoothing filter. The edge-enhanced filters proposed in this thesis use edge information obtained by a ratio-based edge detector to improve the performance of the filters in noise smoothing as well as in edge and fine feature preservation. Due to the good performance of these edge-enhanced filters, the resulting histogram-thresholded segmented images have accurate and simple region boundaries and well separated regions of both large and small sizes. The proposed method is compared with the previous method proposed by Lee and Jurkevich, in both noise smoothing performance and in segmentation quality. The results are tested on synthetic images as well as airborne SAR images. The tests show that the proposed method produces better image segmentations. particularly in image region boundaries, homogeneous regions and for images with fine features. The proposed edge-enhanced segmentation scheme may be suitable for many SAR image analysis applications such as sea-ice segmentation. forest classification. crop identification, etc.

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 and also fellow graduate students. I would like to acknowledge Canada Centre for Remote Sensing for the SAR images they provided. 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

Chapter 1

Introduction

1.1 General

Segmentation is one of the major tasks in digital image processing and analysis. The purpose of segmentation is to divide an image into regions which are uniform and homogeneous with respect to some characteristics such as gray level or texture. Segmentation can be critical for subsequent analysis and scene description.

Synthetic Aperture Radar (SAR) utilizes a synthetic aperture to generate high resolution images of terrain from a constant emission of microwave pulses. The synthesized aperture is generated by the motion of the radar platform, which can be either an aircraft or a satellite, and by signal processing of the returned pulses. SARs have several features which make them valuable for remote sensing: they can collect images by day and at night, and in all types of weather; and the microwave region of the EM spectrum provides unique information about the terrain sensed. Because of these features, SAR has become an attactive technology for monitoring many geophysical parameters. Applications of SAR image processing include land cover classification, soil moisture measurement, forest type classification, measurement of liquid water content of vegetation, snow mapping, sea ice type classification, ice sheet dynamics, oceanography, and many others [47].

1.2 Motivation

The goal of segmenting SAR images is to produce image regions each representing ground regions such as woods, fields, roads, etc. This is essential for applications such as crop identification, terrain mapping, target detection, etc. Segmentation of SAR images is usually based on gray level or texture. Gray level based segmentation is useful for the identification of features [48] and for detecting changes between different images [9]. Texture measures are helpful, for example, in sea ice detection [46]. In this research, we focus on segmentation based on similar gray level, as will be further explained.

Segmenting a SAR image is made difficult by the presence of speckle noise which is multiplicative in the sense that the noise level increases with the magnitude of radar backscattering. There are several methods that have been developed to segment SAR images. However, some problems still exist, including the accuracy and simplicity of region boundaries, homogeneity of regions, ease of use and implementation of segmentation methods, computational complexity of segmentation algorithms, etc.

The goal of this thesis is to provide automatic, unsupervised segmentation schemes for SAR images which are efficient and accurate in dividing image regions, and to compare these new schemes with the work of past researchers to demonstrate the advantages of the proposed schemes.

1.3 Problem complexity

Segmenting SAR images is generally a difficult task. The complexity involved in the above mentioned problem arises from three major factors. First of all. the signals to be processed are two dimensional signals which are random in nature. so their statistical properties are not easy to estimate. Second. there is no general technique and complete theory for segmentation. Image segmentation techniques are basically *ad hoc* and differ precisely in the way they emphasize one or more desired properties and in the way they balance and compromise one desired property against another. Third, the presence of speckle noise makes the segmentation of SAR images more difficult. This kind of noise is multiplicative and non-additive. unlike many optical images which may have additive white Gaussian noise. Many traditional techniques for segmentation which were originally designed for optical images rely on measures based on differences between pixel intensities; hence. such methods suffer from noise artifacts when applied to SAR images [4]. For example. traditional clustering techniques for image segmentation tend to form more clusters in bright areas than in faint ones [44].

1.4 Research background

There is no general technique for segmentation. because of the differences in applications and image types. Many segmentation schemes have been developed. based on different criteria and for use with different types of images. Image segmentation techniques can be classified as: gray level thresholding. iterative pixel classification, methods based on fuzzy set theory, etc [40]. Some speckle-specific methods have been developed for segmenting SAR images. These use various techniques such as simulated annealing [25], wavelet transforms [3], hierarchical random field models [13], neural networks [22], fuzzy c-means clustering [14], etc.

A method for segmenting SAR images was proposed by Lee and Jurkevich [31], who found that the iterative application of a speckle reducing filter (based on the Lee multiplicative filter [32]) produces an image with a multimodal gray level histogram suitable for thresholding. This method is efficient, relatively easy to use and unsupervised after the setting of initial parameters. It produces good results and may offer significant advantages over techniques based on region growing with its ability to segment classes separated by a gradual change in gray level intensity [47]. However, it has several disadvantages. First of all, although the filtering scheme is unsupervised, it requires the knowledge of the ratio of standard deviation to mean in homogeneous image regions. In Lee and Jurkevich's implementation, this input parameter remains unchanged, though its actual value may change on each iteration. Second, the Lee multiplicative filter tends to preserve speckle noise in edge areas. Third, their method requires that users manually determine the valleys in the histogram of the smoothed image for thresholding purposes. To overcome these disadvantages but still retain the advantages of Lee and Jurkevich's method, in this thesis we propose and test an edge-enhanced segmentation method which is based on the initial work by Lee and Jurkevich.

1.5 Approach of the thesis

The edge-enhanced segmentation scheme proposed in this thesis is based on the repeated application of an edge-enhanced filter. My approach to the solution of the stated segmentation problem involves the following steps:

- Using a ratio-based edge detector to generate an edge map of a SAR image:
- Choosing a speckle reducing filter suitable for iterative application:
- Modifying the filter by using edge information in local statistical analysis (I call this modified filter an edge-enhanced filter);
- Applying the edge-enhanced filter iteratively:
- Segmenting the filtered image based on histogram thresholding.

The performance of the edge-enhanced segmentation method is tested on both synthetic and real multi-look airborne SAR images using several edge-enhanced filters. Comparisons of the proposed method with the previous method of Lee and Jurkevich are also provided. The test results show that the edge-enhanced method can separate the homogeneous regions very well and that the region boundaries are simple and accurate. It provides an automatic and efficient segmentation method for SAR images.

1.6 Organization of the thesis

This thesis has been organized as follows. Chapter Two provides background information on the areas related to the stated problem. Chapter Three provides descriptions of my proposed segmentation method. Test results and comparisons are presented in Chapter Four and Five, on synthetic and real SAR data, respectively. Finally, conclusions based on the results obtained and suggestions for further work are provided in Chapter Six.

Chapter 2

Literature review

2.1 Introduction

This chapter is grouped into five major sections. The second section provides a brief introduction to Synthetic Aperture Radar (SAR) imaging systems, SAR speckle. speckle noise modeling and SAR speckle reducing techniques. The third section is an overview of different existing segmentation schemes. Special attention is paid to Lee and Jurkevich's method [31] and Smith's work [47]. The next section discusses the problems related to SAR image segmentation, particularly noise smoothing, edge detection and histogram thresholding. Concluding remarks are provided in section Five.

2.2 SAR speckle noise

This section will introduce SAR imaging systems, the source of speckle noise. Also, it will provide a brief survey of previous work on SAR speckle modelling and speckle reduction techniques.

2.2.1 SAR imaging systems

A SAR is a remote sensing system used to obtain high resolution. two dimensional. microwave images of targeted terrain. Before the development of SARs. it was very difficult to acquire this type of image from microwave systems due to the extremely large antennas needed to obtain the high spatial resolution. SAR technology avoids this problem by utilizing the motion of the radar platform to synthesize a larger antenna. The platform for a SAR can be either spaceborne or airborne. As it moves over an area of interest, it constantly sends microwave pulses towards the ground at discrete intervals. These intervals are quite short, and each point on the ground is mapped numerous times by different microwave pulses as the radar platform moves past. This repeated exposure of a ground point to many different pulses gives the impression to the imaging system that the length of the antenna is no longer just its physical size, but rather, the flight length of the radar platform during which a given point is within the radar's swath. After several stages of processing such as quadrature demodulation. range compression, azimuth correlation and focus corrections. SAR images can be obtained from collected. digitized SAR data [11. 53]. Figure 2.1 shows an airborne SAR image of fields and trees.

2.2.2 Source of speckle noise

In SAR and other systems employing coherent illumination to form high resolution images, the resulting image is generally corrupted by a form of multiplicative noise known as coherent speckle. This severe form of noise, characterized by a low signal to

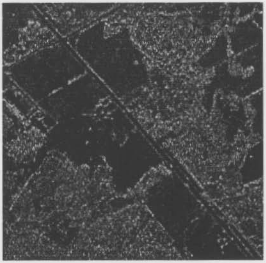


Figure 2.1: A typical multi-look airborne SAR image *Fields*

noise ratio, presents problems for image processing software of all kinds. In particular, many image analysis techniques originally designed for optical images suffer from noise artifacts and perform poorly when applied to SAR images [4, 44, 49].

Speckle is the primary source of radiometric distortion in SAR images and arises from three primary sources: first, a resolution cell is many wavelengths long at the frequency of the SAR; second, the cell typically contains not one but many elementary point scatterers at different orientations and having different areas and scattering coefficients; and third, the resolution cell is not viewed instantaneously but over a short interval and consequently the viewing angle and scattering coefficients change during imaging. To form an image pixel, the complex point spread functions contributed by the set of scatterers in the resolution cell are summed coherently with the different phases interfering constructively or destructively, yielding a speckle noise pattern superimposed on the image of interest.

2.2.3 Speckle models

To design an optimum technique for SAR image analysis, it is necessary to have an appropriate mathematical model of the speckle noise based on the method of image formation and its statistical properties. Speckle properties are discussed in several papers [1, 2, 23]. 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 equal to one for fully developed speckle. This implies that a pointwise multiplicative model would be able to describe speckle in SAR intensity images quite well. A large amount of research is found which assumes a multiplicative model of speckle [17, 30, 32]. The model can be represented in an (i, j) spatial coordinate system as,

$$z(i,j) = x(i,j) \cdot v(i,j) \tag{2.1}$$

where x is the original image, z is the recorded image and v is the random noise process.

However, it has been pointed out by Tur *et al.* [52] that a multiplicative noise model is not always complete. 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 can be seen to arise mainly from the Point Spread Function (PSF) of the imaging system. Thus a better model can be obtained by taking the PSF of the imaging system into account. Thus, the model of Equation 2.1 can be re-written as:

$$z(i,j) = (x(i,j) \cdot v(i,j)) * h(i,j) \tag{2.2}$$

where $*$ indicates spatial convolution and h is the SAR impulse response. The model described by Equation 2.2 is used by Frost *et al.* with the assumption that the PSF for SAR systems is an impulse. In addition, more complex models of SAR image can be formulated, to account for point scatterers, line features, etc [38].

Lee and Jurkevich [31] 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 in their paper.

2.2.4 SAR speckle reduction

Some image processing methods have been proposed which explicitly address the multiplicative nature of the speckle noise. These methods include the use of speckle reducing filters, the logarithm transform and techniques based on ratios of pixel intensities.

Several speckle smoothing filters have been proposed based on a statistical model of speckle noise. These are the Lee multiplicative filter [32] and its variations, *e.g.* [36], the Sigma filter [33, 34], the Weighted filter [39], Li's method [37], the Maximum A Posterior (MAP) filter [38], etc.

One way to deal with the scaling of the variance with image intensity is to work in the logarithmic domain. Under the logarithmic transformation, the noise variance becomes independent of the image intensity [4], although the logarithm changes the statistical and frequency distributions of the noise process.

Another way to work with the multiplicative aspect of the speckle noise is using similarity measures based on the ratio of average pixel intensities [18, 49]. It has been shown that the ratio of average pixel intensities within a homogeneous area follows the F distribution and is independent of the image intensity [4].

2.3 Segmentation methods

One of the most widely used steps in the process of reducing images to information is segmentation: that is, dividing the image into regions that correspond to structured units in the scene or which distinguish objects of interest. An image can be segmented into homogeneous regions based on some features such as gray level or texture. This is important for subsequent analysis and scene description [43, 45]. There are many types of images, such as optical images, magnetic resonance images (MRI), range images, infrared images, SAR images, etc. Hundreds of segmentation techniques are presented in the literature, but there is no single method which can be considered good for all images, nor are all methods equally good for a particular type of image [40]. Moreover, algorithms developed for one class of image may not perform well when applied to other classes of images. This is particularly true when the algorithm is based on a specific image formation model.

2.3.1 General

A good image segmentation requires that regions of an image segmentation be uniform and homogeneous, region interiors be simple and without many small holes, adjacent regions of a segmentation have significantly different values, and that boundaries of each segment be simple, not ragged, and spatially accurate [24].

Achieving all these desired properties is difficult because strictly uniform and homogeneous regions are typically full of small holes and have ragged boundaries. Insisting that adjacent regions have large differences in values can cause regions to merge and boundaries to be lost.

Image segmentation techniques are basically *ad hoc* and differ precisely in the way they emphasize one or more desired properties and in the way they balance and compromise one desired property against another.

Image segmentation techniques can be classified as gray level thresholding. iterative pixel classification, methods based on fuzzy set theory. etc. [40].

Thresholding is one of the oldest, simple and most popular techniques for image segmentation. Thresholding can be performed based on global information (e.g. gray level histogram of the entire image) or using local information (e.g. co-occurence matrix) of the image. If only one threshold is used for the entire image. then it is called global thresholding. On the other hand, when the image is partitioned into several subregions and a threshold is determined for each of the subregions. it is referred to as local thresholding. Typically, thresholding methods work well in situations where there are a few distinct objects having widely differing gray tone intensities and these objects appear on a near uniform background [24].

Iterative pixel classification methods include region growing, relaxation. Markov Random Field (MRF) based approaches and neural network based approaches. Region growing schemes regard some of the pixels in the image as nodes and grow regions based on some similarity critera. Relaxation is an iterative approach to segmentation in which the classification decisions about each pixel can be taken in parallel [43]. There are many image segmentation methods which use spatial interaction models such as Markov Random Fields (MRF) or Gibbs Random Fields (GRF) to model digital images [12, 20]. Several authors have attempted to segment an image using neural networks [7, 21].

Fuzzy set theory is used in fuzzy thresholding, fuzzy clustering, etc. Different histogram thresholding techniques which minimize the gray level ambiguity and geometrical ambiguity of an image are described in [41, 42]. The fuzzy c-mean (FCM) clustering algorithm [6] has been used in image segmentation [51].

2.3.2 Segmentation of SAR images

With SAR images of mixed terrain, the goal of segmentation might be to produce image regions each representing ground objects such as woods. fields. vegetation type. roads, urban areas. etc. This is essential for applications such as crop identification. terrain mapping and target detection. etc. Producing the segmentation of a SAR image is made difficult by the presence of speckle noise. Several segmentation methods specifically designed for SAR images have been proposed based on either gray level or texture.

Hegarat-Mascle *et al.* [25] proposed an algorithm for segmenting SAR images which applies simulated annealing techniques in the classification process. Benie *et al.* [5] developed a classification method which integrates two algorithms: a hierarchical image segmentation by step-wise optimization to take into account the spatial context. and an iterative conditional mode (ICM) algorithm to classify the segmented image. Barbarossa *et al.* [3] proposed a method for classifying SAR images based on a multiresolution representation of the images obtained by a wavelet transform. Derin *et al.* [13] described some algorithms based on a hierarchical random field model proposed for speckle images and a Maximum A Posterior (MAP) segmentation using simulated annealing. Fosgate *et al.* [15] presented an efficient multiscale approach to the segmentation of natural clusters. Grossber *et al.* [22] developed a neural network model of boundary segmentation and surface representation to process images containing range data which is gathered by a SAR sensor. Du *et al.* [14] used a fuzzy c-means clustering method for unsupervised segmentation of multi-look polarimetric SAR images. Texture also is important in characterizing the information in SAR images. Some texture based segmentation schemes are proposed in the literature. Ceccarelli *et al.* [10] use texture information and neural networks for SAR image segmentation. Sephton *et al.* [48] proposed a segmentation algorithm for SAR images of sea-ice based on various attibutes such as texture, shape, position, etc.

One segmentation method for SAR images was proposed by Lee and Jurkevich [31], which is simple and efficient. It is unsupervised and produces very good results. It may offer significant advantages over techniques based on region growing with its ability to segment classes separated by a gradual change in gray level intensity [47]. Smith [47] extended this method and presented a fully automatic segmentation scheme for segmenting SAR images. Based on this previous work, I propose the edge-enhanced segmentation method in this thesis. Thus, it is important to have a more detailed look at the work of Lee and Jurkevich as well as Smith's work.

Lee and Jurkevich's method

Image segmentation is often based on edge detection and region growing [9, 48]. This approach is difficult to apply to images corrupted by speckle because gaps in edges, caused by speckle, must be repaired with a bonding routine [48]. Another problem is that ill-defined edges, formed by a gradual change in intensity between two classes, will not be detected. Lowering the edge detection threshold in order to detect fewer sharp edges often results in the detection of false edges, leading to over-segmentation of the image [9].

A natural approach to unsupervised segmentation based on gray levels is to select thresholds at the valleys of a gray level histogram, if such valleys exists. For SAR images, however, the speckle appearing in them complicates the characteristics of the SAR image histogram and makes automatic segmentation of such images difficult.

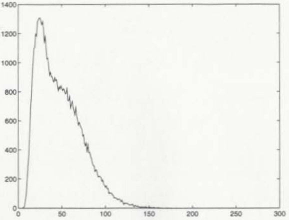


Figure 2.2: Histogram of the SAR image *Fields* of Figure 2.1.

The histogram in a SAR image is typically unimodal [31] and difficult for histogram thresholding. Furthermore, since no use is made of spatial information in simple histogram thresholding, there is no guarantee that pixels of one gray scale class will be contiguous. Figure 2.2 shows the histogram of the typical SAR image *Fields* which is shown in Figure 2.1; although this image is a relatively simple SAR image, its histogram does not reveal distinct image regions.

A method for segmenting SAR images corrupted by speckle was proposed by Lee and Jurkevich [31], who found that the repeated application of an edge-preserving speckle smoothing filter produces an image with a multimodal gray level histogram suitable for thresholding, even if the histogram of the original image is unimodal. The rationale behind this approach is that the averaging process will reduce the noise standard deviation, thus tending to produce sharp peaks in the histogram, while the edge preserving effect of the filter produces deeper valleys.

Of crucial importance to this technique is the ability of the speckle filter to smooth out the noise without destroying edges and fine features. Lee and Jurkevich iteratively used the well-known Lee multiplicative filter [32] (reviewed below in Section 2.4) in their segmentation process. This filter is quite effective in removing speckle especially in homogeneous or low variance areas. In high variance areas, however, the filter's parameters are adjusted to preserve edges; this has the effect of also preserving speckle noise near and on edges.

Overall, Lee and Jurkevich's method is quite simple and efficient. This unsupervised algorithm will segment the image into several classes without *a priori* knowledge of the number of classes and their mean values. However, there are some disadvantages to this method. The input parameter (the ratio of the standard deviation to mean) to the filtering algorithm should be calculated manually and in their implementation this parameter remains unchanged, though the physical quantity it describes (*i.e.* the speckle noise) may change on each iteration. In thresholding, the histogram valleys are user-determined. Moreover, due to the poor performance of the filtering algorithm in edge areas, the segmented image regions have ragged and inaccurate region boundaries.

Smith's work

Smith improved Lee and Jurkevich's method and proposed a fully automatic, unsupervised segmentation algorithm, based on the iterative application of the modified sigma filter [47] (also reviewed below in Section 2.4).

One contribution of Smith's work is that he proposed a method to automatially estimate the input parameter, the ratio of the standard deviation to mean, in the filtering algorithm. For estimation of this parameter, the entire image is partitioned into equal sized windows (e.g. 3×3) and the ratio of standard deviation to mean calculated in each window. Then, the histogram of these local estimates is obtained, and the histogram mode is taken as an overall estimate.

Also in Smith's work, a modified sigma filter has been proposed which is shown to be an improvement over the basic sigma filter in preserving edges while smoothing the speckle noise. Moreover, the histogram thresholding procedure is fully automatic. However, Smith's method is not able to detect small classes whose gray level falls within the two-sigma range of larger classes.

2.4 Related problems

My research is based on the segmentation method proposed by Lee and Jurkevich. As stated above, the most important component in Lee's method is the speckle reducing filter. So it is desirable to have a review of SAR noise smoothing techniques and filters suitable for iterative application. Because I also use an edge detector in my proposed segmentation scheme to improve the boundary accuracy, SAR edge detection methods will also be reviewed. In addition, the histogram thresholding method that is used in my research will be described in this section.

2.4.1 Noise smoothing

Noise can be reduced by filtering, in which a moving window is passed over each pixel in the image, and the pixel value is replaced by a value derived from the pixels within the window. Numerous speckle reducing filters have been proposed and discussed (see, for example [35, 54]). Here, we describe some spatial filtering methods that use local statistics and which may be suitable for iterative application.

Mean and median filter

Two simple filters are the mean and median filters, in which the central pixel is replaced by the mean or median, respectively, of all the pixels in the window. The main problem with the mean filter is that the edges in the image are smoothed together with the speckle noise, and may be lost completely after a few successive applications of the filter. The median filter will preserve step edges in noise-free images, however, in speckle corrupted images, it does not remove the noise very well, since speckle noise is not impulsive.

Lee multiplicative filter

The Lee multiplicative filter [32] is based on a multiplicative noise image model:

$$z(i,j) = x(i,j) \cdot v(i,j) \tag{2.3}$$

where z, x and v denote the observed image, underlying image and noise processes, respectively. Based on an assumption that the noise is white with unity mean and is uncorrelated with the image x, the Lee multiplicative filter seeks the best mean-squared estimate $\hat{x}$ of x. At each pixel (i,j),

$$\hat{x}(i,j) = \bar{x}(i,j) + k(i,j)\,(z(i,j) - \bar{x}(i,j)) \tag{2.4}$$

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

$$k(i,j) = \frac{\mathrm{Var}_x(i,j)}{\bar{x}(i,j)^2\,\sigma_v^2\, +\, \mathrm{Var}_x(i,j)} \tag{2.5}$$

where σ_v is the ratio of standard deviation to mean in homogeneous regions. The local adaptation of the filter is based on the calculation of the local statistics, $\bar{x}$ and Var_x, from the data sample estimates $\bar{z}$ and Var_z determined over a local neighbourhood window:

$$\bar{x}(i,j) = \bar{z}(i,j) \tag{2.6}$$

$$\text{Var}_x(i,j) = \frac{\text{Var}_z(i,j) + \bar{z}(i,j)^2}{\sigma_v^2 + 1} - \bar{z}(i,j)^2 \tag{2.7}$$

By adapting its parameters to both low-variance areas and high-variance areas. the filter both smooths noise and preserves edges. In order to preserve edges. the filter essentially shuts itself off in high variance areas (i.e. $k(i,j) \approx 1$) so that the estimate $\hat{x}(i,j)$ is approximately equal to the observed pixel value $z(i,j)$. This means the speckle noise is preserved in high contrast regions.

Lee and Jurkevich [31] found several iterations of the Lee multiplicative filter can greatly reduce speckle noise. However, small details may be lost due to the repetitive smoothing operation; and. as with the one-pass Lee multiplicative filter. speckle noise is preserved in edge areas.

Refined Lee multiplicative filter

To improve the performance in edge areas. Lee proposed a refinement to the original Lee multiplicative filter [36], in which the neighbourhood used in high variance areas for the calculation of the local statistics takes into account the orientation of a possible edge. For each pixel with local variance Var_z exceeding a set threshold. oriented gradients are computed and used to select a subset of the neighbourhood pixels on one side of the edge and most like the central pixel. Var_z estimated over this subset will in general be lower than the sample variance over the whole neighbourhood, allowing more accurate filtering of noise. However, the edge detection is not optimized for speckle corrupted images in which local variance is related not only to edges but also to the underlying mean intensity level.

Sigma filter

Lee [33] proposed a much simpler alternative to the Lee multiplicative filter, based on the same multiplicative speckle noise model. It is well know that 95.5 percent of normally distributed random samples fall within two standard deviations on either side of the mean value. Values outside of the two-sigma range are therefore likely to be from a different distribution. The sigma filter averages only those pixels in the filter window which lie within the two-sigma range of the central pixel value. Edges are thus preserved because pixels not belonging to the same distribution as the central pixel are excluded from the averaging process. Since the speckle is multiplicative, the two-sigma range, assuming the central pixel $z(i,j)$ to be the mean of its distribution, is bounded below and above by

$$z_{min} = (1 - 2\sigma_v)z(i,j) \tag{2.8}$$

$$z_{max} = (1 + 2\sigma_v)z(i,j) \tag{2.9}$$

Thus only pixels $z(k,l)$ whose value lies between z_{min} and z_{max} are included in the calculating the estimate of $\bar{z}(i,j)$.

A problem arises if there are no other window pixels within the two-sigma range. Such sharp spot noise is dealt with by introducing a threshold k_s such that, if the total number of pixels within the two-sigma range is smaller than or equal to k_s, then the central pixel is replaced by the average of its four nearest neighbours.

This filter is based on the assumption that speckle noise has a Gaussian distribution; therefore, it can be applied to the data processed in a large number of looks. When this condition is not satisfied, the filtered image has an asymmetric distribution [39]. Furthermore, this filter degrades point targets because it replaces the central pixel value with the mean of its four nearest neighbours whenever the central pixel has an extreme value.

Weighted filter

The sigma filter as described above is based on the assumption that the central pixel is in fact the mean of its Gaussian distribution. A more general perspective, proposed by Martin and Turner [39], is that potentially the central pixel belongs to a range of Gaussian distributions. For maximum speckle reduction it is desirable to include in the averaging process all pixels which could possibly belong to the same distribution as the central pixel, while excluding those pixels clearly from different distributions. Furthermore, it is crucial for the preservation of fine features, which by definition contain few pixels within the filter window, that pixels possibly belonging to the same distribution as the central pixel are not excluded from the averaging process. For each window pixel $z(k,l)$ it must therefore be determined whether $z(k,l)$ and $z(i,j)$ are both within the two-sigma range of the same Gaussian distribution. If so, then $z(k,l)$ should be included in the average.

Martin and Turner [39] proposed a new sigma filter, which they called the weighted filter, such that a window pixel $z(k,l)$ is included in the averaging process if a Gaussian distribution centred on $z(k,l)$ with standard deviation $\sigma_v \cdot z(k,l)$ has the central pixel value $z(i,j)$ within its two-sigma range. The weighted filter therefore averages all window pixels lying between z_{min} and z_{max} given by

$$z_{min} = \frac{z(i,j)}{1+2\sigma_v} \qquad (2.10)$$

$$z_{max} = \frac{z(i,j)}{1-2\sigma_v} \qquad (2.11)$$

The weighted filter assigns weights to each pixel based on the Gaussian probability density function centred on the pixel value. This is a computationally expensive procedure, requiring the evaluation of an exponential function for each window pixel included in the average. Because the upper limit, z_{max}, is larger for the weighted filter than for the sigma filter, more of the higher value pixels are included in the averaging process, thereby removing the low bias inherent in the sigma filter and reducing noise more effectively. The lower limit z_{min}, on the other hand, is higher for the weighted filter than for the sigma filter. This is an undesirable feature of the weighted filter because some low value pixels which are within the two-sigma range of the central pixel value will be excluded by the weighted filter. Some fine features which would have been preserved by the sigma filter will, therefore, be annihilated by the weighted filter.

The weighted filter, which is considered to be an advanced version of the Sigma filter, takes the probability of the central pixel value into account to determine the range for which the mean value can be extracted over a filtering window. It replaces the central pixel value with the weighted mean in the filtering window, in which the weighting coefficients are determined based on the probability of the pixel value. The filtered image has a larger variance in homogeneous areas compared with the other filters.

Modified sigma filter

Two simple modifications to the sigma filter has been proposed by Smith [47] in order to improve both its computational efficiency and its ability to preserve fine features. The first modification assumes the mean of a possible Gaussian distribution containing the central and a given window pixel to be a linear combination of the two. rather than equal to the central pixel $z(i,j)$ as in the standard sigma filter. or equal to the window pixel $z(k,l)$ as in the weighted filter of Martin and Turner. The two-sigma range of values to be included in the averaging process is thereby increased. enabling the speckle noise to be reduced more effectively. and ensuring that more fine feature pixels are detected so that fewer fine features are annihilated.

The second modification attempts to preserve fine features in cases in which the number of pixels within the two-sigma range is less than or equal to the sharp spot noise threshold. This is achieved by searching for the Gaussian distribution containing three or more connected pixels from which the central pixel is most likely to have been displaced.

2.4.2 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. 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. Edges can also be defined based on changes in other image features (*e.g.* texture).

In this subsection, we will first briefly review edge detectors for SAR images. Then we will focus on the Maximum Strength Edge Pruned Ratio of Averages (MSP-RoA) method [18, 19], which will be used in my proposed segmentation method.

SAR edge detectors

Many common 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 can yield very poor results because speckle is multiplicative in nature. Moreover. the problem involved with speckle is even more difficult because speckle is not only multiplicative. but can also have an inter-pixel spatial correlation to some extent. An efficient edge detector should be based on the speckle model. Hence. it is very difficult to detect edges in speckled images and almost impossible using simple gradient 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 be a better edge indicator 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 [8] which is particularly useful for detecting edges of images corrupted with a speckle noise.

Relatively few methods have been reported so far in this area. despite the importance of the problem of edge detection on speckle imagery. A simple method named the Coefficient of Variance (CoV) was proposed [49] based on the Constant False Alarm Rate (CFAR) concept which uses a coefficient of variation which can provide an edge strength measure. Frost *et al* [16] proposed an edge detecting technique for SAR images, in which the maximum likelihood ratio (LR) is the measure of edge strength. The maximum likelihood ratio is computed based on a SAR image model. Bovik's Ratio of Average (RoA) [8] 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 GSL gives fine edges but also gives rise to many false edges which is not at all desirable. A combination of these two 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 work which is simply an extension of the RoA detector is proposed by Touzi etc. [49], using the Constant False Alarm Rate (CFAR) concept.

MSP-RoA edge detector

Ratio-based edge detectors estimate edge strength at any pixel of interest in an image by calculating the ratio between neighbouring pixel values. The Maximum Strength Edge Pruned Ratio of Averages (MSP-RoA) method developed in [18, 19], is one such method which has been shown to provide accurate localized edge maps from speckled SAR images. At each pixel in the image, the method calculates the four ratio edge strengths

$$R_i = \min(P_i/Q_i, Q_i/P_i), \quad i = 1, 2, 3, 4 \tag{2.12}$$

corresponding to the four usual orientations, as illustrated in Figure 2.3, where P_i and Q_i are the averages calculated over the sub-windows denoted P and Q, respectively. The MSP-RoA then calculates a vector (R, O) characterizing a possible edge at that pixel, where the component $R = \min(R_1, R_2, R_3, R_4)$ is the edge strength and O is the orientation which yields the minimum R_i value. A candidate edge pixel is classified as an edge pixel if the magnitude $R \leq T_r$, for a preset threshold $T_r \in (0, 1)$, and if R is the minimum magnitude of all the pixels in a sub-window of $(2D + 1) \times 1$ pixels perpendicular to the orientation O.

Testing of the MSP-RoA edge detector on airborne SAR images [19, 18] has

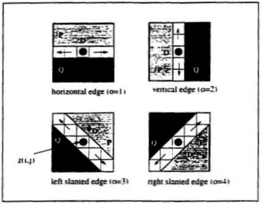


Figure 2.3: MSP-RoA scheme

demonstrated it to be an efficient ratio-based edge detector which can produce thin and accurate edge maps in the presence of speckle noise.

2.4.3 Histogram thresholding

After iterative application of a speckle reducing filter. a filtered image can be expected to have a multimodal gray level histogram suitable for thresholding. Tsai proposed a fast histogram based algorithm for multi-level thresholding [50]. and it was demonstrated to be more powerful than the widely used thresholding methods based on between-class variance or entropy. Tsai's method assumes that each desired class in the image can be represented by an approximately hill-shaped distribution in the gray level histogram. The fluctuations of an original histogram are smoothed by recursively convolving the histogram with a Gaussian kernel so that the desired peaks and valleys at varying levels of detail can be obtained. The detected valleys in the smoothed histogram indicate the location of the thresholds.

The proposed algorithm assumes that a peak shown in the gray level histogram corresponds to a homogeneous region of the image and that a valley exists between two neighboring peaks. The challenge is to locate the bottoms of the valleys that best separate the classes. Let $h_0(i)$ represent the number of pixels in the original image with gray level i for $i = 0, 1, \ldots, L$, where L is the maximum gray level. In a histogram, if

$$h_0(i) > h_0(i-1) \text{ and } h_0(i) > h_0(i+1), \qquad i = 1, 2, \cdots, L-1 \tag{2.13}$$

then a peak at gray level i is detected. Similarly, if

$$h_0(j) < h_0(j-1) \text{ and } h_0(j) < h_0(j+1), \qquad j = 1, 2, \cdots, L-1 \tag{2.14}$$

or

$$h_0(j) < h_0(j-1) \, \text{and} \, h_0(j) = 0, \qquad j = 1, 2, \cdots, L \tag{2.15}$$

and there exist two peaks at gray levels p_1 and p_2 such that $p_1 < j$ and $p_2 > j$, then a valley at gray level j is selected. The definition of peaks and valleys above assumes that the peaks and valleys will not be present at gray level 0 and L.

The fluctuation of the original gray level histogram may generate many false peaks and valleys. In order to find the proper peaks and valleys at varying levels of thresholding, we use a Gaussian kernel to smooth the histogram. The degree of smoothing is manipulated by the width of the Gaussian kernel and the number of convolutions. Function $h_0(t)$, the number of pixels with gray-level t, is convolved with a one dimensional Gaussian kernel $g(t, \sigma)$ of width σ:

$$g(t, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} exp(-t^2/2\sigma^2) \tag{2.16}$$

$H(t, \sigma)$, the convolution of $h_0(t)$ and the Gaussian kernel, is defined as

$$H(t, \sigma) = h_0(t) * g(t, \sigma) = \int_{-\infty}^{\infty} h_0(u) g(t-u, \sigma) du \tag{2.17}$$

For digital implementation, the digital Gaussian kernel with a window size of $W = 3$ is used to generate smoothing functions at various values of σ, and it is given by $g(-1) = 0.2261$, $g(0) = 0.5478$, $g(1) = 0.2261$.

The discrete convolution of $h_0(i)$ and the digital Gaussian kernel $g(u)$ is defined as

$$H(t, W) = \sum_{u=-[W/2]}^{[W/2]} h_0(t+u)g(u) \tag{2.18}$$

where $[W/2]$ is the largest integer not greater than $W/2$.

After iterative filtering with the Gaussian kernel, the valleys in the histogram are then detected. Pixels whose gray levels lie between two adjacent valleys are assigned to the same region. Therefore, we have divided the filtered image into homogenous regions and finished the segmentation stage.

2.5 Concluding remarks

This chapter attempts to provide an overall picture of the previous work most closely related to the problem of interest. It focuses on the segmentation methods on which my research is based. Because my proposed method will be involve filtering, edge detection and histogram thresholding, these related problems are also reviewed. The next chapter will provide my proposed edge-enhanced segmentation method for SAR images in detail.

Chapter 3

The proposed edge-enhanced segmentation method

3.1 Introduction

My proposed edge-enhanced segmentation method is based on the idea of Lee and Jurkevich [31] that repeated application of an edge-preserving speckle reducing filter will provide an image with a multimodal histogram suitable for thresholding even if the original image has a unimodal histogram. The problems of Lee and Jurkevich's method have been discussed in the previous chapter. First of all, due to the poor performance of the filtering algorithm in edge areas, the segmented image regions have ragged and inaccurate regions boundaries. Second, the input parameter (the ratio of the standard deviation to mean) to the filtering algorithm should be calculated manually and in their implementation this parameter remains unchanged, though the physical quantity it describes (*i.e.* the speckle noise) may change on each iteration. Moreover, in thresholding, the histogram valleys are user-determined. My approach has several advantages over Lee and Jurkevich's method. First of all, edge information is used with the filtering algorithms, which leads to a better performance of the filters in edge areas. Second, the iterative application of the filters is fully automatic. which means that no statistical parameters are required as input parameters. as all these parameters are automatically estimated from the image. Third. the histogram valleys are not manually determined in the thresholding algorithm. Finally. due to the improvement in the filtering scheme, the segmented image has simple and accurate region boundaries and there are no small holes in homogeneous regions. The following are the main components of my approach:

- Using MSP-RoA edge detector to estimate edges
- Modifying a speckle reducing filter by using edge information to improve the filtering performance in edge areas;
- Iteratively applying the modified filter (I call this filter an edge-enhanced filter) to smooth the speckle noise:
- Segmenting the filtered image based on histogram thresholding to get image regions with accurate region boundaries.

3.2 Main components

The main components of my segmentation method include edge-enhanced filters, parameter estimation, edge detection. iterative application, and histogram thresholding. These will be discussed in turn in this section.

3.2.1 Edge-enhanced filters

Of crucial importance in my segmentation method is the ability of speckle reducing filters to smooth noise while preserving the sharpness of edges. Several speckle smoothing filters have been developed by previous researchers as stated in the literature review. However, these filters usually do not use edge information and sometimes do not perform well in edge areas. In this research, I propose an edge-enhanced filtering method which utilizes edge information obtained by the MSP-RoA edge detector in local statistical analysis in the filtering window. This kind of filter smooths speckle in low variance areas as well as in high variance areas while preserving the sharpness of edges. The starting point of my research was to modify the simplest mean and median filter, with fairly good results, some of which were presented in [28]. Then the Lee multiplicative filter was modified and iteratively applied to smooth SAR images, with some results presented in [29]. These results are even better and have a significant improvement over the original iterative Lee multiplicative filter, especially in edge areas. The segmentation results based on the edge-enhanced Lee multiplicative filter, in particular, are very good when tested on synthetic images as well as real SAR images. Detailed results of all methods are presented and compared in Chapter 4 and 5 of this thesis.

Edge-enhanced mean and median filters

The main problem with the mean filter is that the edges in the image are smoothed together with the speckle noise, and may be lost completely after a few successive applications of the filter. The median filter will preserve step edges in noise-free images, however, in speckle corrupted images, it does not remove this non-impulsive noise very well.

In this research, edge-enhanced mean and median filters are presented which use the edge map generated by the MSP-RoA edge detector to determine which of the neighbouring pixels are to be included in the calculations of the mean or median. This method tends to remove the speckle noise while retaining the edges even after several iterations. The following is the full description of the modified local mean and median filters.

First, the MSP-RoA edge detector is applied to estimate the edge points. Then, for each pixel $z(i,j)$, a 3×3 window centered on the pixel is used for analysis. According to the estimated edge map, the pixels inside the window can be divided into two classes: those which are edge pixels and those which are non-edge pixels. The edge pixels are excluded in the local mean or median calculation. The next step is to decide if $z(i,j)$ is a edge point based on the edge map. If $z(i,j)$ is not a edge point, then it is replaced by the mean of the non-edge pixels. Otherwise, it is replaced by the median of the non-edge pixels. After all the pixels in an image have been examined, then one iteration of the filter is finished. The process can be repeated with subsequent iterations.

I chose small window sizes in the edge-enhanced filters because a large window size may oversmooth some regions near or on edges. The edge-enhanced mean and median filters are simple and efficient for segmentation purpose. As will be shown in Chapter 4, even after several iterations, most edges in the image are preserved while the noise is significantly reduced [28]. However, in some test images, there are still some areas that are merged. If there is a gap in the edge map, the edge tends to blur when the edge-enhanced mean filter is applied.

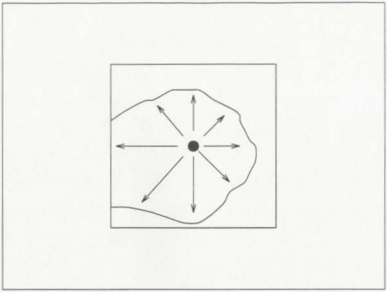


Figure 3.1: Using edge information to define the *valid* region in the filtering window

Edge-enhanced Lee multiplicative filter

Although the original Lee multiplicative filter performs very well at noise smoothing in low variance areas, it tends to preserve speckle noise near edges. Since ratio-based edge detectors are able to ignore speckle in detecting edges in SAR images, I use the edge map produced by the MSP-RoA edge detector to improve the performance of the Lee multiplicative filter.

By using the edge information obtained from the MSP-RoA edge detector, it is possible to refine the definition of the local neighbourhood over which local statistics are calculated, thus improving the homogeneity of the neighbourhood and the quality of the estimates. The improvement in the performance of the Lee multiplicative filter is most notable in high variance areas near edges. Figure 3.1 shows how knowledge of an edge contour is used in delimiting the neighbourhood for local statistical analysis.

In a filtering window of a given size, all the pixels are classified as belonging to one of two classes, thus allowing the definition of two regions within the window, the *valid* and *non-valid* regions. The *valid* region starts with the central pixel in the filtering window, and is grown in eight directions, along the arrows indicated in Figure 3.1 within a line of width one pixel. When the *valid* region reaches an edge point as determined from the MSP-RoA edge map, or when it reaches the window boundary, the region stops growing in that direction. In this way the filtering window is separated into two regions and only the pixels in the *valid* region and along the eight directions are included in statistical estimation.

In the edge-enhanced Lee multiplicative filter, edge information is thus utilized in defining the filtering window size and shape. Since $\bar{z}$ and Var_z are calculated as the mean and variance only of those pixels in the *valid* region they will yield more accurate estimates *near* an edge, so that the modified filter can smooth the speckle in edge areas while preserving the sharpness of edges.

The iterative application of the edge-enhanced Lee filter was tested on synthetic speckled images and the results compared with similar iterations of the original Lee mutiplicative filter. Detailed results are presented in Chapter 4 and 5. Some of the results have been presented at a workshop [29]. The edge-enhanced Lee filter performs better than Lee filter, especially in high variance regions, because it uses edge information to refine the filter's local statistics estimation.

3.2.2 Estimation of parameter σ_v

The Lee multiplicative filter and the edge-enhanced Lee multiplicative filter require the knowledge of the ratio of the standard deviation to mean in homogeneous areas. This parameter is seen to be σ_v for unity mean noise, and is sometimes called the

Coefficient of Variation (CoV). This value can be estimated by calculating sample values of $\mathrm{Var}_z/\bar{z}$ over several homogeneous or structure free areas of the observed image z. Alternatively, the value of the parameter σ_v can be determined from the known speckle characteristics of the number of looks and type of SAR image to be filtered (e.g. see [31, 32]).

However, the application of a nonlinear filter can significantly alter the noise characteristics, and it can be difficult to analytically update the parameter σ_v in an iterative filtering scheme. In my implementation, σ_v is automatically estimated from the image data by Smith's method, described in Section 2.3.2, thus better controlling the filtering process for each pass of the filter. For moderately busy images, the use of this estimation method tends to highlight the local estimates of σ_v in the homogeneous regions, and to exclude those erroneous estimates of σ_v in areas of high variance such as edge regions.

The automatic estimate of σ_v is quite close to its theoretical value. For example, the theoretical value for 4-look airborne SAR images is around 0.25 [56], and the estimated values for the testing images are quite close to this value (refer to Chapter 4 and 5). The window sizes in estimating σ_v have an effect on the final estimation. In experiments, it is found that for the test images presented in this thesis, a 7×7 window size can produce a good estimate of σ_v. Although estimation errors exist, the Lee multiplicative filter and the proposed edge-enhanced Lee multiplicative filter are not extremely sensitive to the value of σ_v, which means that small estimation errors are tolerable.

3.2.3 Edge detection

As noted above, the edge map obtained by the MSP-RoA edge detector is used in the edge-enhanced filters. The ability of the edge detector to find true and accurate edges while excluding false edges is very important to the performance of the filters. The MSP-RoA method makes effective use of the information available in ratio-based methods in achieving accurate, localized edge maps without additional edge thinning operations.

As with many other edge detection methods, the basic parameters of the MSP-RoA method must be determined by reference to a given image and defined carefully in order to exploit the advantages of the method. The MSP-RoA performance tends to improve if the mask size is increased, provided the resulting P and Q sub-areas are homogeneous. Therefore, the mask size should be selected as large as possible, by reference to the distances between edges and the size of image objects in the image. The value T_r can be adjusted to detect as many significant edges as possible without detecting spurious or noise edges. The parameter D can be selected by considering the likely size of objects in the image. Since the MSP-RoA method prunes candidate edge pixels within the subwindow pependicular to the edge orientation, the value of D should be small for images having fine image structure. In this research, D is set to 1 for all the test images.

Test results from some SAR images [18, 19] indicate that the MSP-RoA method has difficulty in detecting all edges such as some edges in busy image regions. Since a large window size tends to blur small regions, the MSP-RoA method has difficulty in detecting fine features when a big window size is chosen. Choosing a small window will produce more false edges due to the noise artifacts. In this research, two methods to improve the performance of the basic MSP-RoA method have been used. First, the MSP-RoA edge detector is used to generate a new edge map before each iterative application of the edge-enhanced filters. After each iteration. the MSP-RoA window size is decreased and the threshold T_r increased as the noise level decreases. so that it can detect more significant and fine edges. Second and alternatively. another way to improve the performance of the basic MSP-RoA method is to modify the basic MSP-RoA method by assigning weights to pixels in the averaging process producing P_i and Q_i in each orientation. The pixels nearest the central pixel are assigned more weight than those further from the central pixel, in order not to degrade edge strength for very fine edges when a large window size is used. Suppose a window size of $(2n+1) \times (2n+1)$ is used, the weights are assigned as $n, n-1, n-2, ...1$ from the nearest pixel to the furthest from the central pixel.

The edge-enhanced filters in this thesis are used typically with the MSP-RoA edge detector generating a new edge map on each iteration. However. since the MSP-RoA edge detector can produce very good estimations of edge maps for SAR images. even with significant speckle noise. we have also simply applied the MSP-RoA edge detector once only prior to filtering and then re-used the result on each iteration. thus saving computational time.

3.2.4 Number of iterations

The number of applications of the speckle smoothing filters has an effect on the noise smoothing performance as well as the final segmentation results. In general. the noise smoothing performance will be best at a certain iteration. Too many iterations may oversmooth the tested image. However, for segmentation purposes, more iterative applications of the filters seem to produce an image the histogram of which has deeper valleys and higher peaks, as is desirable for histogram thresholding. More detailed tests and discussion are provided in the next two chapters.

3.2.5 Histogram thresholding

After an iterative application of the edge-enhanced filters, the histogram of a filtered image typically has a mutimodal shape suitable for thresholding. Although there are many thresholding schemes that have been proposed, the Gaussian smoothing and valley seeking algorithm proposed by Tsai is a good choice, as it is faster and more accurate than other widely used between-class variance and entropy methods [50]. A detailed description of Tsai's work has been provided in the literature review in Section 2.4.3. It is worth mentioning that in Tsai's work, the number of classes in the image should be provided as an input parameter to control the convolution of the Gaussian kernel. In my research, I use the number of repeated convolutions as an input parameter for the Gaussian smoothing algorithm. The number of convolution is very important for the final segmentation stage. A large number of convolutions will produce fewer image regions in the result. More detailed results and discussion will be provided in the next two chapters.

3.3 Overall approach

Figure 3.2 shows the flow diagram of my overall approach for segmenting SAR images. The iterative application of the MSP-RoA edge detector and the modified filter is called the edge-enhanced filter. The MSP-RoA is applied on each iteration. The threshold T_r is set to a value between $0.4-0.8$ in the first iteration (detailed discussion on the setting of the threshold value can be found in [18, 19]). After each application, this value increases by 0.025 (this was obtained by testing numerous images) and

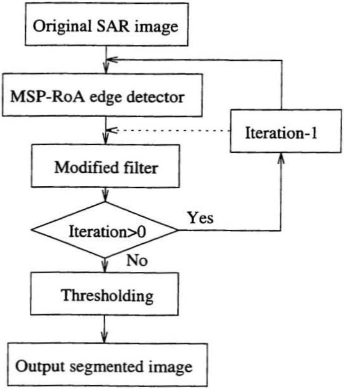


Figure 3.2: Flow diagram of overall aproach

the window size of the edge detector decreases by 2 (until window size equals 3, the smallest size) as the noise level decreases, so that the edge detector can detect more true and fine edges. These parameters are chosen by experiments. Results will be shown in the next two chapters. For computational savings, we could alternatively simply apply the MSP-RoA method once prior to filtering and re-use the result on each iteration. Also, the modified MSP-RoA edge detector can be used to produce an edge map instead of original method.

In this research, the edge-enhanced mean and median filters were initially tested. However, the edge-enhanced Lee mutiplicative filter shows the more significant improvement in filtering as well as in segmentation performance. So, more tests have been performed based on the edge-enhanced Lee multiplicative filter.

Because the edge information is used in the statistical calculation of the edge-enhanced Lee multiplicative filter, the shape and window size are redefined in the actual calculation of statistical estimates. Therefore, even if a large filtering window is chosen, it still preserves fine features and keeps computational efficiency. Moreover, σ_v is calculated automatically and used to control the filtering process on each iteration. The thresholding stage includes Gaussian smoothing and valley seeking algorithms, and produces the final segmented image. The number of convolutions in Gaussian smoothing algorithm determines the number of classes that will be detected in the segmented image. By experiments, I found that 5 or 6 times of convolutions can produce a good result in the implementation.

3.4 Concluding remarks

In this chapter. I described my proposed segmentation method based on histogram thresholding and a ratio-based edge detectior used in conjunction with an iterative application of a modified speckle-reducing filter. Test results of this edge-enhanced segmentation method on synthetic and real SAR images as well as their comparision with those from Lee and Jurkevich's method will be provided in the next two chapters.

Chapter 4

Test results on synthetic images

4.1 Introduction

The previous chapters of this thesis have discussed the development of various segmentation methods, and, to some extent, filtering and edge detection methods for speckle corrupted images. This chapter focuses on the performance of my proposed segmentation method and its comparison with Lee and Jurkevich's method for segmenting SAR images.

It is worth noting that in this chapter as well as in the next chapter, the automatic estimation of σ_v and automatic histogram thresholding are used in Lee and Jurkevich's method, making our implementation an improvement over the orignal Lee and Jurkevich's segmentation scheme. So, in fact, references to Lee and Jurkevich's method in these two chapters denote the improved Lee and Jurkevich's method. Image processing results of segmentation, filtering and edge detection are presented as well as performance measures and discussion.

The test images used in this thesis include both synthetic and real SAR images.

In this chapter, only synthetic image results are discussed. The results for real SAR images will be presented in the next chapter. The difference between real SAR images and synthetic SAR images is that there are no very high contrast regions in real SAR images as in some synthetic images. The synthetic image used in this chapter is a combination of 3 different images with different regions of contrast. so it may be more difficult for segmentation than the real SAR images.

4.2 Methodology

In this section, we will have a brief introduction to the synthetic images used for testing and the speckle simulation method used for generating these images. Methods of measuring performance will also be addressed.

All images are stored in files with pixel value 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 SunOS Release 4.1.3 operating system on a Sun SPARC 10 workstation.

4.2.1 Test data

For testing with synthetic images, the algorithms 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 gray levels and chosen to demonstrate the algorithms' ability in edge preservation. The third image is a more complicated photographic image with many shapes and edges of different contrast. These three images are combined in a single image called *Combine* which is shown in Figure 4.1.

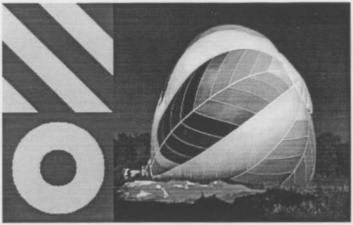


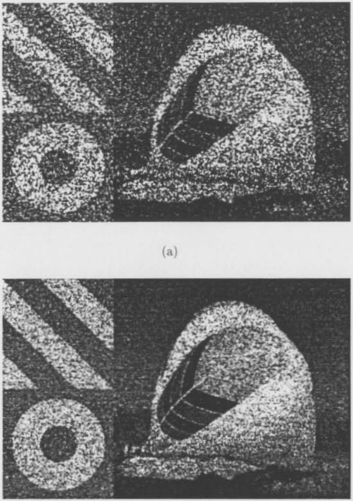
Figure 4.1: Original image *Combine*

This particular image is the clean, original image for further study.

Two different images *CombineCOR1* and *CombineCOR* are shown in Figure 4.2 (a) and (b). These are generated using a speckle model and synthetic one-look and four-look amplitude SAR speckle [55], respectively. 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 four-look is a very common choice, only the results of processing the four-look speckle degraded image of Figure 4.2 (b) will be presented.

4.2.2 Speckle simulation

The speckle in Figure 4.2 is simulated by a method proposed by Zaman and Moloney [55]. A multiplicative convolution model (Equation 2.2) has been used for speckle simulation. The first step is to generate a two dimensional noise field. Noise samples



(a)

(b)

Figure 4.2: Speckle corrupted images: (a) *CombineCOR1*, 1-look; (b) *CombineCOR*, 4-look.

from an exponential or χ-squared distribution are used for generating one-look or multi-look speckled images, respectively. The χ-squared distribution has its number of degrees of freedom twice that of the number of looks. 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 PSF 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 [27]. The resulting multiplicative image is then convolved with the given SAR PSF and a synthetic amplitude speckle SAR image is thus obtained which has the expected statistics of such an image, as verified by our automated calculation of σ_v or by manual choice of homogeneous regions [5, 6]. For the four-look speckled image, the standard deviation to mean ratio calculated over homogeneous regions of differing average intensity is approximately 0.218, close to the theoretical value of 0.25. The correlation coefficient between horizontal and vertical neighbouring pixels is approximately 0.49, and between diagonal neighbours approximately 0.25.

4.2.3 Performance measures

A traditional measure of noise smoothing is the Mean Square Error (MSE). When working with synthetic image data, the uncorrupted image is assumed to be known. Hence, the global MSE for the noisy image and filtered images can be calculated when we test the synthetic images. A commonly used speckle suppression measure for SAR images is σ_v, the ratio of the standard deviation to mean in homogeneous areas. In this work, σ_v is automatically calculated from the image using the method described in Section 2.3.2. The performance of filtering and segmentation are also evaluated by visual inspection.

4.3 Results on synthetic image

In this section, one set of test results on the image of Figure 4.2 (b) will be presented. The images resulting from the proposed edge-enhanced segmentation method as well as Lee and Jurkevich's method are compared. Fixed parameters are chosen to illustrate the performance of the algorithms. More detailed discussions on various combinations of input parameters will be addressed in the next section.

4.3.1 Segmentation of clean image

The clean image *Combine* has a multimodal histogram suitable for histogram thresholding, shown in Figure 4.4 (a). Using Gaussian smoothing and the valley seeking algorithm implemented in this research for histogram thresholding, this image can be segmented as shown in Figure 4.3 into 16 regions. The parameters for Gaussian smoothing are 5 convolutions with a Gaussian kernel of width 3. Discussions on how to choose proper Gaussian smoothing parameters will be presented in the next section.

4.3.2 Histogram

The clean, original image *Combine* has a distinctive multimodal histogram and it is a easy task to segment such an image. However, the histogram of the speckle corrupted

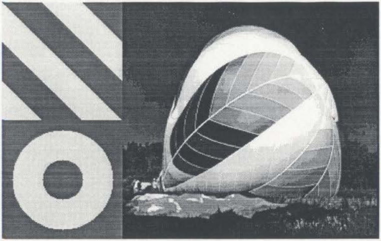
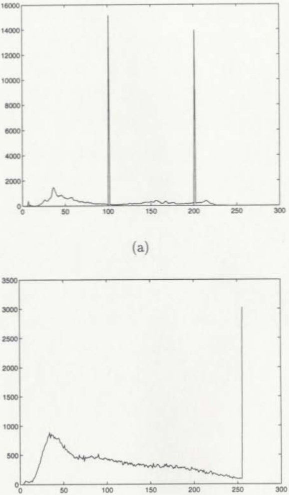


Figure 4.3: Segmentation of clean image *Combine*

image *CombineCOR* is complicated by the presence of speckle noise, which makes automatic segmentation of such an image difficult. Figure 4.4 (a) and (b) show the histograms of *Combine* and *combineCOR*, respectively.

4.3.3 Filtering results

The proposed edge-enhanced filters include the edge-enhanced mean and median filters and the edge-enhanced Lee multiplicative filter. The results of these edge-enhanced filters as well as the result of the iterative Lee multiplicative filter will be presented in this section. Discussions on the performances of these different filters will also be addressed. As noted previously, after speckle reducing filtering, the resulting image should have a multimodal histogram suitable for thresholding.



(a)

(b)

Figure 4.4: Histograms of (a) *Combine*; (b) *CombineCOR*

Iterative Lee multiplicative filter

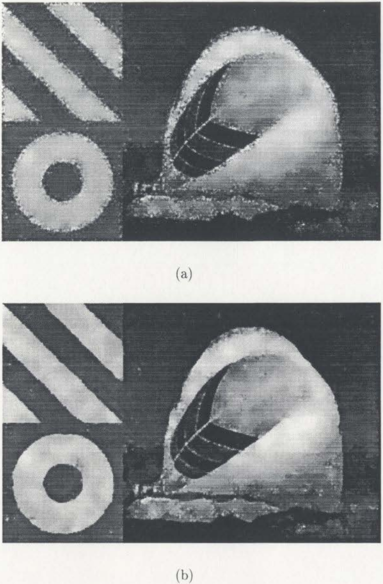
The iterative application of Lee multiplicative filter is tested on *CombineCOR*. In the original Lee and Jurkevich method [31], knowledge of the standard deviation of the speckle, σ_v is required, and in an iterative implementation, σ_v is required as an input parameter at each iteration. In the original Lee and Jurkevich implementation, this parameter was unchanged from iteration to iteration although the actual value may change. In our implementation, at each iteration σ_v is automatically estimated from the image by a method described in Section 2.3.2 and used to control the filtering of that iteration. Figure 4.5(a) shows the result of smoothing *CombineCOR* by the iterative Lee multiplicative filter, using a window size of 11×11 on each of 3 iterations.

Edge-enhanced Lee multiplicative filter

To compare the proposed edge-enhanced Lee multiplicative filter with the iterative Lee multiplicative filter, we chose the same filtering parameters: window size 11×11, 3 iterations. The MSP-RoA edge detector's input parameters are: window size 11×11, threshold $T_r = 0.72$. We chose these parameters because this combination of parameters tends to produce good filtering result. More detailed discussions on different combinations of parameters will be presented in the next section. Figure 4.5 (b), Figure 4.6 (a) and (b) show the edge-enhanced Lee multiplicative filtered result, the MSP-RoA edge map before the first iteration and the final MSP-RoA edge map before the third iteration, respectively.

Edge-enhanced mean and median filters

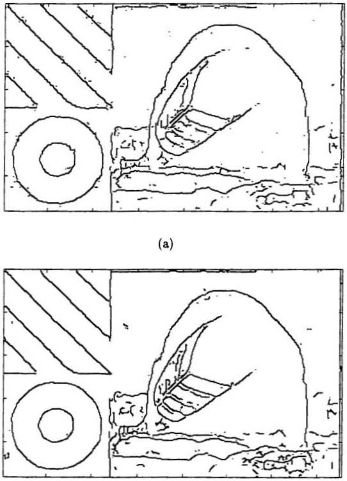
The edge-enhanced mean and median filters are the simplest edge-enhanced filters and are the starting point of this research. Although the performance is not as good



(a)

(b)

Figure 4.5: Filtering results for *CombineCOR*, mask 11 × 11, 3 iterations: (a) Using iterative Lee multiplicative filter; (b) Using edge-enhanced Lee multiplicative filter.



(a)

(b)

Figure 4.6: MSP-RoA edge map of *CombineCOR* (a) Before filtering; (b) Before the third iteration of the edge-enhanced Lee multiplicative filter.

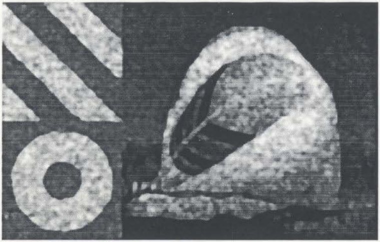


Figure 4.7: Edge-enhanced mean and median filtering result for *CombineCOR*

as that of the edge-enhanced Lee multiplicative filter, these filters are still fairly good in preserving edges and smoothing speckle noise. Figure 4.7 shows the result of the edge-enhanced mean and median filters when applied to *CombineCOR*. All the parameters are the same as the edge-enhanced Lee multiplicative filter except that a smaller filtering window size 3 × 3 is chosen in order not to oversmooth regions near fine features.

Discussion

The MSE between *CombineCOR* and *Combine* is 751, while σ_v for *CombineCOR* is 0.218. For the result of the iterative Lee multiplicative filter shown in Figure 4.5(a), σ_v is reduced to 0.014 and MSE is reduced to 230, while for the result of the proposed edge-enhanced Lee multiplicative filter shown in Figure 4.5(b), σ_v is reduced to 0.014 and MSE is reduced to 158. Although the σ_v for both results are the same indicating similar noise smoothing in homogeneous regions, the MSE indicates that the edge-enhanced Lee multiplicative filter performs better than the iterative Lee multiplicative filter in noise smoothing, especially near edges. Moreover, we can see that in Figure 4.5(a), the speckle noise is preserved in edge areas, while in Figure 4.5(b), the edges are sharp and the noise is reduced in high variance areas as well as in low variance areas. Overall, the edge-enhanced Lee multiplicative filter is an improvement over iterative Lee multiplicative filter.

For the performance of the edge-enhanced mean and median filters shown in Figure 4.7, the MSE is 275 and σ_v is 0.06. From these performance measures, it seems that this result is the worst one. However, from visual inspection, we can find that in edge areas, the noise is smoothed while edge sharpness is retained, which in this regard is better than the iterative Lee multiplicative filtering result. For segmentation purposes, after several iterations of the edge-enhanced mean and median filtering followed by histogram thresholding, the resulting segmented image will have simple and accurate edges and homogeneous regions which may be better than the iterative Lee multiplicative filter based segmentation result. Some of the results can be found in [28].

Because the edge-enhanced Lee multiplicative filter is the best of the edge-enhanced filters developed in this research, we will focus on it in the further tests and discussions.

For the MSP-RoA edge detector, we can see that the edge map is refined after several applications because the noise is smoothed. Moreover by decreasing the window size and increasing the threshold, we tend to detect more significant edges in small regions such as the narrow stripes in *Balloon*.

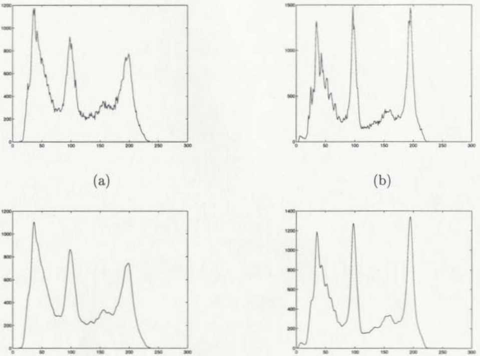
4.3.4 Histogram thresholding

The iterative application of a speckle reducing filter tends to produce an image with a multimodal histogram suitable for thresholding. Figure 4.8 (a) and (b) show the histograms of the iterative Lee multiplicative filtered image and edge-enhanced Lee multiplicative filtered image, respectively. It is clearly shown that compared with the iterative Lee multiplicative filter, the proposed method produces deeper valleys and higher peaks in the histogram of the filtered image. This is desirable for histogram thresholding.

Small fluctuations in the histograms of the filtered images may generate many false peaks and valleys. In order to find the proper peaks and valleys at varying levels of thresholding, we use a Gaussian kernel to smooth the histograms. Here, we find that the use of a one-dimensional Gaussian kernel of width 3 applied 5 times will produce good results for the test images. More discussions on this issue will be presented in the next section. Figure 4.8 (c) and (d) show the smoothed histograms of Figure 4.8 (a) and (b), respectively. It is worth noting that more small regions are preserved in Figure 4.8(d) than in Figure 4.8(c). This may be due to the ability of edge-enhanced filter to preserve fine features.

4.3.5 Segmented images

The histogram thresholding algorithm seeks valleys in the smoothed histogram of the filtered image; pixels whose gray levels lie between two adjacent valleys are assigned to the same region. Figure 4.9 (a) and (b) show the segmentation results of the iterative Lee filtered image and the edge-enhanced Lee filtered image, respectively. It can be seen in Figure 4.9 (a) that some of the stripes in *Balloon* are not very


(a)

(b)

(c)

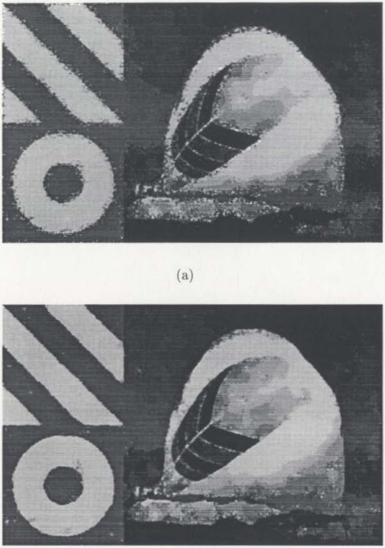
(d)

Figure 4.8: Histograms of filtered image of *CombineCOR*: (a) After iterative Lee multiplicative filter; (b) After edge-enhanced Lee multiplicative filter; (c) Gaussian smoothing of (a); (d) Gaussian smoothing of (b).

clean and homogeneous. This is because of the poor performance of the iterative Lee multiplicative filter in edge areas which did not produce deep valleys in these small regions. Another problem is that there are many small dots on the edges and the edges are not clean and sharp. This is because the Lee multiplicative filter preserves speckle noise in high variance areas. Compared to Figure 4.9 (a), Figure 4.9 (b) shows a better segmented image which separates the regions which have great gray level differences as well as small gray level differences. The edges are very sharp and clean, while most regions are homogeneous and clean. However, there are still some black and white dots in the *Bars* and *Annular*. This is because the noise is too strong in these particular areas and is not sufficiently smoothed in the filtering stage. There are 11 regions in Figure 4.9 (a) and 12 regions in Figure 4.9 (b). More false regions are produced in the edge areas in Figure 4.9 (a), while some fine regions are lost or merged. In Figure 4.9 (b), the region locations and numbers are similar to those of the segmentation of the clean image shown in Figure 4.3. Overall, the edge-enhanced segmentation method produces better result than does the Lee and Jurkevich's method.

4.4 More results

In the previous section, fixed parameters are chosen for illustrating the performance of the different stages in the filtering and segmentation algorithms, with the resulting images presented for comparison. In this section, different combinations of the input parameters are chosen for more detailed exaimination of the performance of the algorithms. Quantitative measures such as σ_v and MSE are presented as well as some resulting images to show the visual qualities.



(a)

(b)

Figure 4.9: Segmentation results on *CombineCOR*: (a) Based on iterative Lee multiplicative filter; (b) Based on edge-enhanced Lee multiplicative filter

4.4.1 Filtering results

Lee and Jurkevich's segmentation method and the proposed edge-enhanced segmentation method are all based on the iterative application of a speckle reducing filter. The Lee and Jurkevich's method uses the iterative Lee multiplicative filter while the edge-enhanced segmentation method uses the proposed edge-enhanced Lee multiplicative filter. The quality of the segmented image is highly dependant on the filter's ablity to remove noise while preserving sharp edges. In this section, different combinations of parameters of both filters are used for testing. Quantitative measures are presented for comparison.

Table 4.1 shows quantitative measures of the performance of the iterative Lee multiplicative filter when applied to *CombineCOR*, while Table 4.2 shows the same results from the edge-enhanced Lee multiplicative filter on the same image. The MSP-RoA parameters are fixed in the edge-enhanced filter, namely window size 11 × 11, threshold $T_r = 0.72$ and correlation parameter $D = 1$. The edge detector produces a new edge estimation on each iteration. I choose these parameters for the MSP-RoA method because they can produce a good edge map as shown in Figure 4.6. Moreover, choosing these parameters tends to produce good filtering results. This will be discussed in more detail later.

Discussion

For the synthetic image *CombineCOR*, Table 4.1 and 4.2 show that the iterative applications of both filters tend to suppress the speckle, and σ_v tends to reach a lowest value after several iterations. Sometimes σ_v may become a little bit higher than on a previous iteration. This is likely caused by the estimation error of σ_v which is automatically calculated from the image. However, the overall tendency is for σ_v

	σ_v after iteration (Original: 0.218)					MSE (Original: 751)				
Iteration	1	2	3	4	5	1	2	3	4	5
3 × 3 window	0.132	0.092	0.081	0.055	0.051	343	267	240	228	224
5 × 5 window	0.078	0.051	0.036	0.029	0.023	240	206	199	198	200
7 × 7 window	0.050	0.032	0.025	0.015	0.014	215	202	201	203	207
9 × 9 window	0.041	0.032	0.013	0.015	0.014	217	211	212	216	221
11 × 11 window	0.030	0.028	0.014	0.014	0.014	228	227	229	233	239

Table 4.1: Quantitative measures of iterative Lee multiplicative filter

	σ_v after iteration (Original: 0.218)					MSE (Original: 751)				
Iteration	1	2	3	4	5	1	2	3	4	5
3 × 3 window	0.132	0.083	0.070	0.059	0.050	342	264	234	219	211
5 × 5 window	0.079	0.057	0.044	0.033	0.030	249	204	188	181	179
7 × 7 window	0.068	0.035	0.036	0.027	0.023	211	178	172	169	169
9 × 9 window	0.053	0.031	0.029	0.025	0.013	192	169	165	165	166
11 × 11 window	0.045	0.048	0.014	0.013	0.016	181	165	158	160	162

Table 4.2: Quantitative measures of edge-enhanced Lee multiplicative filter

to become smaller until it reaches a lowest value. The filtering window size has a great effect on the performances of the filters. When a larger window size is chosen, σ_v goes down faster. This is because the averaging process over a large window size produces a more smoothed image more quickly.

For the iterative Lee multiplicative filter, when examining the MSE in Table 4.1, we can find that the performance becomes better when we increase the window size until it reaches a certain value, and after that, it will become worse. This is because too large a window size tends to oversmooth some small regions and to keep more noise in regions near edges. Also, the MSE has a lowest value after several iterations and then it rises again. So, it is important to choose proper parameters to get the best performance. The best performance for the iterative Lee multiplicative filter when applied to *CombineCOR* is MSE $= 198$, $\sigma_v = 0.029$. The input parameters are window size 5×5, 4 iterations.

For the MSE measure of the edge-enhanced Lee multiplicative filter when applied to *CombineCOR*, Table 4.2 shows that the change of input parameters has an influence on the performance which is similar to the Lee filter. However, the best performance is obtained when the window size is 11×11, much larger than the 5×5 for the iterative Lee multiplicative filter. This is because the edge map changes the shape and size of the neighbourhood involved in statistical calculations. So even large window sizes will not destroy image detail.

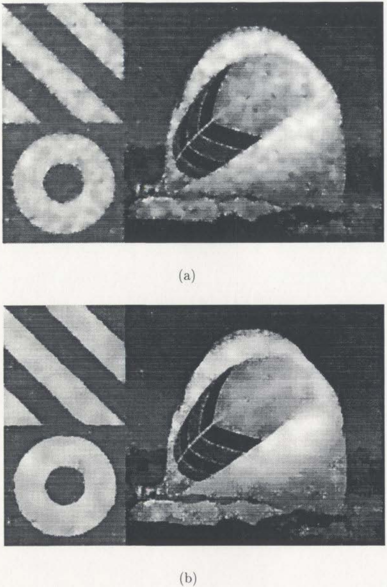
The quantitative measures shows that for the iterative Lee multiplicative filter, σ_v is reduced more quickly than with the edge-enhanced Lee multiplicative filter. This is because the edge-enhanced Lee filter only uses the samples along the eight directions in the filtering window so that the number of samples in the local window is less than that used in the iterative Lee filter. Thus, the edge-enhanced Lee filter does not smooth as quickly as the iterative Lee filter. However, when the iterations increase and the filtering window is enlarged, σ_v will reach a lowest value for both filters. Comparing Table 4.1 with 4.2, we can find that the edge-enhanced Lee filter has better performance (best is $MSE = 158$, $\sigma_v = 0.014$) than that of the iterative Lee filter (best is $MSE = 198$, $\sigma_v = 0.029$). Furthermore, visual examination of the filtered images show that my proposed filter has a great improvement over the iterative Lee filter in edge areas and in small regions, which leads to a better performance for further segmentation application. This is demonstrated, for example, in Figure 4.10, which shows these two "best" results on *CombineCOR*.

4.4.2 Use of MSP-RoA method in filtering

The edge-enhanced filter can be used with the MSP-RoA edge detector generating a new edge map in each iteration. However, since the MSP-RoA can produce very good estimations of edge maps for SAR images, even with significant speckle noise, we can choose to simply apply it once only prior to filtering and to re-use the edge map result on each iteration, thus saving computational time. Table 4.3 shows the filtering results when using the edge detector once or iteratively. The MSP-RoA parameters are the same: window size 11 × 11, threshold $T_r = 0.72$ and correlation parameter $D = 1$. The filtering window size is 11 × 11. The overall best performance for the Once set is $\sigma_v = 0.014$, MSE = 160, after 3 iterations, a result which is very competitive with best result of the filter in the Iteratively set.

Discussion

From Table 4.3, it can be seen that there is not much significant difference in the performance of the filter when using the MSP-RoA edge detector once or iteratively.



(a)

(b)

Figure 4.10: Comparison of filters applied to *CombineCOR*: (a) Best smoothing by iterative Lee filter, 5×5 window, 4 iterations ; (b) Best smoothing by edge-enhanced Lee filter, 11×11 window, 3 iterations.

	σ_v after iteration (*Original*: 0.218)					MSE (*Original*: 751)				
Iteration	1	2	3	4	5	1	2	3	4	5
MSP-RoA once	0.045	0.048	0.014	0.014	0.014	181	166	160	163	166
MSP-RoA Iteratively	0.045	0.048	0.014	0.013	0.016	181	165	158	160	162

Table 4.3: Edge-enhanced filtering performance when using MSP-RoA once or using it iteratively in each application of the filter

However, to obtain more accurate edge locations and better smoothed result in edge areas, the iterative application of MSP-RoA detector tends to provide better filtering results. This is because after each iteration, the noise will be smoothed and the re-estimation of edges will be less affected by the noise artifacts, which tends to allow more accurate edge maps to be produced.

4.4.3 MSP-RoA parameters

For the edge-enhanced Lee filter, the selection of MSP-RoA parameters is very important for the filtering results. The better the edge map is, the better is the performance of the filter. Table 4.4 shows the results of using different combinations of MSP-RoA parameters ($D = 1$ in all cases) with a filtering window size of 11×11 and 1 iteration at edge-enhanced Lee filter.

Discussion

Table 4.4 shows that the choice of MSP-RoA parameters is crucial to the performance of the proposed filter, particularly in terms of MSE. The best parameters seem to be window size 9×9 or 11×11, with a threshold T_r around 0.7. These produced the best results with the test image *CombineCOR*. Different images may have different

	σ_v (*Original*: 0.218)				MSE (*Original*: 751)			
T_r	0.5	0.6	0.7	0.8	0.5	0.6	0.7	0.8
MSP-RoA window 9 × 9	0.046	0.045	0.045	0.049	204	185	181	199
MSP-RoA window 11 × 11	0.060	0.045	0.045	0.043	206	186	181	189

Table 4.4: Performance of the edge-enhanced Lee filter when choosing different parameters for MSP-RoA edge detector

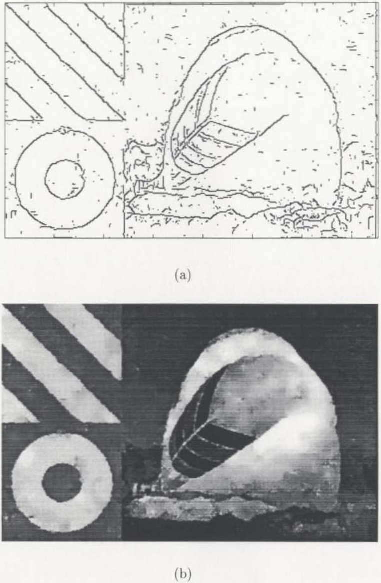
characteristics and the best MSP-RoA parameters may vary correspondingly.

4.4.4 Modified MSP-RoA method

The performance of the edge-enhanced Lee multiplicative filter depends on the performance of the ratio based edge detector. As described in Section 3.2.3. we have slightly modified the original MSP-RoA method by assigning weights to the pixels in the averaging process in order not to degrade edge strength for very fine edges when a large window size is used (refer to Section 3.2.3. Table 4.5 shows a comparison of the performance of filter using the original MSP-RoA method and the modified method. The edge detector parameters are the same: window size 11 × 11. threshold $T_r = 0.72$, $D = 1$. The edge detectors are used only once and the edge maps re-used in each iteration of the filtering process. The filter window size is 11 × 11. Figure 4.11 shows the edge map generated by modified MSP-RoA method on *CombineCOR* and the filtered results based on this edge map after 3 iterations.

Discussion

Table 4.5 illustrates that the modified MSP-RoA method improves the filtering performance compared to the original MSP-RoA method in MSE measure. This is because



(a)

(b)

Figure 4.11: Demonstration of using the modified MSP-RoA: (a) Edge map before filtering; (b) Edge-enhanced filtered result, window size 11 × 11, 3 iterations.

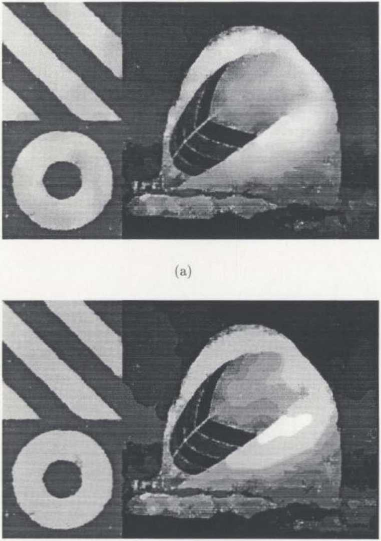
	σ_v (*Original*: 0.218) after iteration					MSE (*Original*: 751)				
Iteration	1	2	3	4	5	1	2	3	4	5
MSP-RoA	0.045	0.048	0.014	0.014	0.014	181	166	160	163	166
Modified	0.044	0.048	0.014	0.013	0.015	178	161	154	156	158

Table 4.5: Performance of filter with original and modified MSP-RoA edge detector

the weights assigned to each pixel tend to preserve fine edges. However, some false edges are also preserved in this process as shown in Figure 4.11 (a), which leads to a more spotty filtering result, as shown in Figure 4.11 (b). From visual inspection, Figure 4.11 (b) shows more small dots in apparently homogeneous image regions than are seen in Figure 4.5 (b) - an undesirable result for smoothing purposes. More work is needed in modifying the MSP-RoA method in order to improve both edge detection and the use of its edge maps in filtering application.

4.4.5 Number of iterations

For segmentation purposes, the iterative application of the edge-enhanced filter tends to produce deep valleys and high peaks in the image histogram. Too many iterations of the filter may oversmooth the regions, which is not desirable for filtering purposes. However, for segmentation purposes, more iterations tend to produce better segmented images because the valleys are deeper and peaks are higher in the histogram. Figure 4.12 show both a filtering result and the corresponding segmentation result on *CombineCOR*, using the edge-enhanced segmentation method. The filtering parameters are the same as those used in Figure 4.5 (b), except that the number of iterations is 10 (as opposed to 3 iterations for Figure 4.5 (b)). The histogram thresholding parameters are the same as was used in obtaining Figure 4.9 (b).



(a)

(b)

Figure 4.12: More iterations: (a) Filtered image; (b) Segmented image.

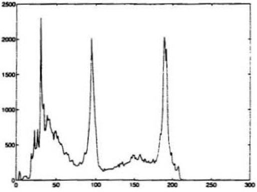


Figure 4.13: Histogram of Figure 4.12 (a).

Discussion

In terms of noise smoothing measure, the MSE for Figure 4.12 (a) is 187 which is greater than the MSE of 158 for Figure 4.5 (b). This shows that more iterations oversmooth the image which is not desirable for noise smoothing. However. the segmentation result shown in Figure 4.12 (b) is better than the segmentation result on Figure 4.5 (b), as shown in Figure 4.9 (b). For example, there are fewer black dots in *Bars* and *Annular* and the stripes in *Balloon* are more homogenous. The number of regions in Figure 4.12 (b) is 12, the same as in Figure 4.9 (b). But overall. in Figure 4.12 (b), the segmented image regions are more homogenous, have fewer holes. and with regions that are more clearly separated. These results are due to the increased filtering which resulted in much deeper valleys and higher peaks in image histogram. Figure 4.13 shows the histogram of Figure 4.12 (a), before Gaussian smoothing, which may be compared with the histogram of Figure 4.8 (b).

4.4.6 Gaussian smoothing

The number of convolutions used in the Gaussian smoothing of an image histogram has an effect on the number of regions detected. The more the histogram is smoothed. the fewer regions will be produced. For a SAR image, it can be difficult to tell how many regions are expected to be detected because we may not know the exact number of classes in the image. By experiments, we found that 5 or 6 passes of the convolution kernel may be suitable for Gaussian smoothing for the *CombineCOR*. because too many convolutions may oversmooth the histogram which leads to the loss of some regions.

4.5 Conclusion

Overall, the test results on the synthetic image *CombineCOR* show that the proposed edge-enhanced segmentation method performs better than Lee and Jurkevich's method in both filtering and segmentation. The performance of the proposed method depends on the edge detector, the number of iterations, Gaussian histogram smoothing, etc. For filtering purposes, the 'best' parameter setting may be filtering window size 11 × 11, 3 iterations, MSP-RoA window size 11 × 11, $T_r = 0.72$. $D = 1$. For segmentation purposes, all the parameters are the same except that the number of filtering processes increases to around 10. Further research is recommended on the improvement of edge detector, modification of other statistical filters and comparison with other segmentation methods.

Chapter 5

Test results on SAR images

5.1 Introduction

The previous chapter provided test results on synthetic images together with some discussion of appropriate parameter settings for various algorithms. This chapter will focus on the performance of the proposed segmentation method, and a comparison with Lee and Jurkevich's method when tested on real SAR images. The test images are four-look airborne SAR images prepared by Ward [53], with the original data as provided by the Canada Center for Remote Sensing (CCRS). Because we cannot know the noise-free images, as required for MSE calculation, only σ_v and visual inspection are used in measuring noise smoothing and segmentation performance.

5.2 SAR images

Two SAR images, called *Fields* and *Industry* are chosen for testing. These images represent typical SAR images which contain large sperated regions and/or some small busy areas. The image *Fields* is a simple image which contains only a few regions such as woods and fields. The image *Industry* is a more difficult image for segmentation because it contains many classes and some image areas are very busy. Figure 5.1(a) and (b) show the image *Fields* and *Industry*, respectively.

5.3 Results on SAR images

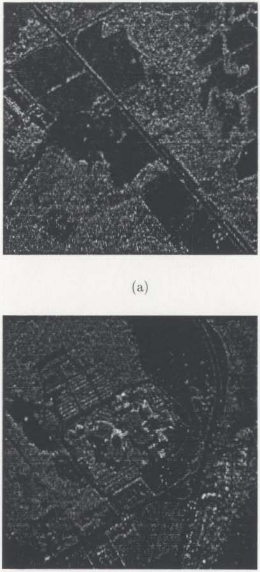
In this section, one set of test results on the images of Figure 5.1 will be presented. The images resulting from the edge-enhanced segmentation method as well as from Lee and Jurkevich's method are compared. More detailed discussions on various combinations of input parameters will be addressed in the next section.

5.3.1 Histogram

Figure 5.2 shows the histograms of *Fields* and *Industry*. These plots demonstrate that these SAR images have unimodal histograms, due largely to the presence of speckle noise. Thus it would be difficult to segment these image directly using histogram thresholding.

5.3.2 Filtering results

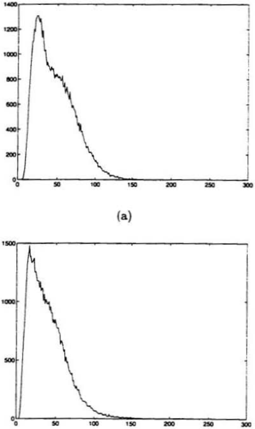
The filtering results of the iterative Lee filter and edge-enhanced Lee filter are compared. Tests on both images show that the edge-enhanced scheme performs better in noise smoothing and edge preserving.



(a)

(b)

Figure 5.1: Multi-look airborne SAR image: (a) *Fields*; (b) *Industry*.



(a)

(b)

Figure 5.2: Histograms: (a) *Fields*; (b) *Industry*.

Iterative Lee multiplicative filter

The results obtained by the iterative application of the Lee multiplicative filter to *Fields* and *Industry* are shown in Figure 5.3. The same parameters are used in both tests, namely window size of 11×11 and 10 iterations. The choice of these parameters are based on the experiments with SAR test images. These images are not too busy, so the window sizes from 7×7 to 11×11 should be fine. More iterations will produce deeper valleys and higher peaks in the image histograms which is desirable for segmentation purposes. So we choose 10 for the number of iterations.

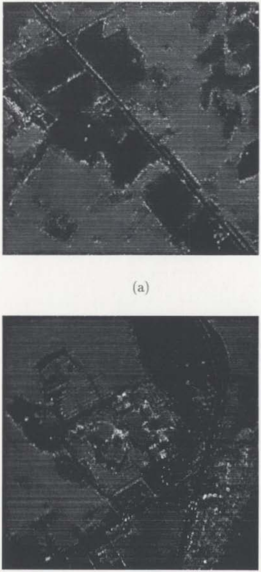
Edge-enhanced Lee multiplicative filter

The edge-enhanced Lee filter results are presented with filtering window size of 11×11, 10 iterations, MSP-RoA window size of 3×3, MSP-RoA threshold $T_r = 0.5$, correlation parameter $D = 1$. The MSP-RoA method estimates the edge map before each iteration. Figure 5.4 (a) and (b) and show the MSP-RoA edge map before the first iteration and the MSP-RoA edge map before the final iteration, respectively, when tested on *Fields*, while Figure 5.5 (a) and (b) show the same edge maps when tested on *Industry*. Figure 5.6 (a) and (b) show the filtered results of the two tested images, respectively.

Discussion

Figure 5.3 shows that the iterative Lee multiplicative filter smooths the homogenous regions. For image *Fields*, σ_v is reduced from 0.278 to 0.021, while for image *Industry*, it is reduced from 0.253 to 0.016. However, in the results of both test images, noise is preserved in edge areas and some fine features are blurred.

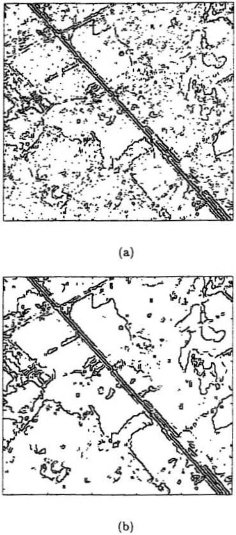
By contrast, Figure 5.6 shows that the edge-enhanced filtered images have noise



(a)

(b)

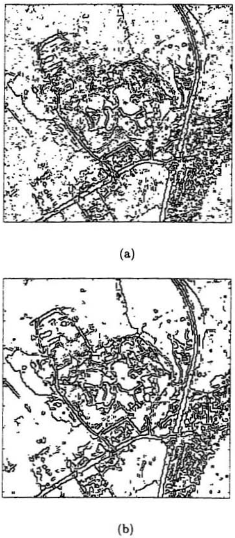
Figure 5.3: Iterative Lee multiplicative filtered result of (a) *Fields*; (b) *Industry*.



(a)

(b)

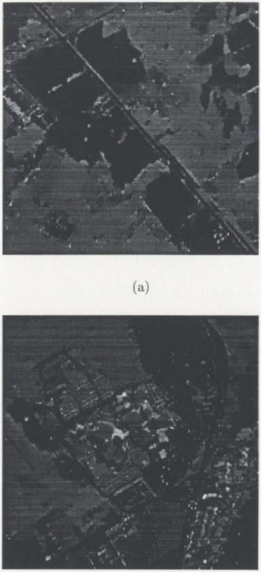
Figure 5.4: MSP-RoA edge maps of *Fields*: (a) Before filtering; (b) Before the final iteration of edge-enhanced Lee multiplicative filter.



(a)

(b)

Figure 5.5: MSP-RoA edge maps of *Industry*: (a) Before filtering; (b) Before the final iteration of edge-enhanced Lee multiplicative filter.



(a)

(b)

Figure 5.6: Edge-enhanced Lee filtered result of (a) *Fields*; (b) *Industry*.

removed in both low and high variance regions, fine features preserved, and sharp edges retained even after several iterations. This yields improvements over the iterative Lee filtered results shown in Figure 5.3. In the edge-enhanced Lee filtered images, σ_v is reduced from 0.278 to 0.021 for *Fields* and from 0.253 to 0.010 for *Industry*.

Overall, though there is no significant difference in the reduction of σ_v by both filters. Visual inspection shows that the proposed edge-enhanced Lee filter performs better than iterative Lee filter, especially in noise smoothing in edge areas and in retaining fine features.

A small window size is chosen for the MSP-RoA edge detector in the edge-enhanced Lee filter. This is because the real SAR images have many small and narrow regions and choosing a small window tends to preserve these regions after iterative filtering. Although due to the speckle noise, many false edges are included in the first iteration, as shown in Figure 5.4 (a) and Figure 5.5 (a), the final edge maps exclude false edges as the speckle is removed after several iterations. The refined edge maps shown in Figure 5.4 (b) and Figure 5.5 (b) indicate that the iterative application of the MSP-RoA method can provide good edge estimates.

5.3.3 Histogram thresholding

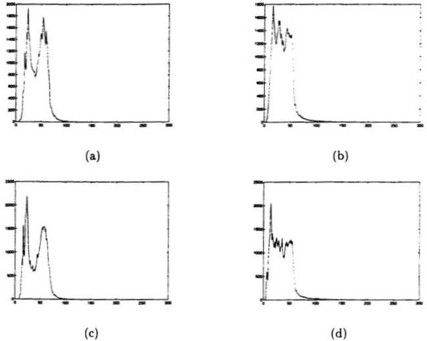
The iterative application of a speckle reducing filter will tend to produce an image with a multimodal histogram suitable for thresholding. Figure 5.7 (a) and (b) show the histograms of the Lee filtered results of *Fields* and *Industry*, respectively, while Figure 5.7 (c) and (d) show the histograms of the edge-enhanced Lee filtered results of these two images, respectively.

It is clear that compared with the iterative Lee filter, my proposed method produces deeper valleys and higher peaks in the histogram of the filtered images, which is desirable for histogram thresholding.

The fluctuation in the histograms of the filtered images may generate many false peaks and valleys. In order to find the proper peaks and valleys at varying levels of thresholding, we use a Gaussian kernel as in the previous chapter, to smooth the histograms. Figure 5.8 (a) and (b) show the Gaussian smoothed histograms of the Lee filtered results of *Fields* and *Industry*, respectively, while Figure 5.8 (c) and (d) show the Gaussian smoothed histograms of the edge-enhanced Lee filtered results of these two images, respectively. The Gaussian kernel used to smooth the histograms of the filtered image of *Fields* is one dimensional with width 3 and is convolved with the histogram 5 times. For *Industry*, the parameters for Gaussian smoothing are width 3, 6 iterations.

5.3.4 Segmented images

The histogram thresholding algorithm seeks valleys in the smoothed histogram of the iteratively filtered image; pixels whose gray levels lie between two adjacent valleys are assigned to the same region. Figure 5.9 (a) and (b) show the segmentation results based on the Lee filtered and the edge-enhanced Lee filtered images of *Fields* while Figure 5.10 (a) and (b) show the results of *Industry*. There are mainly two classes in *Fields*, which are shown in both Figure 5.9 (a) and (b). However, in (a), the edges between two adjacent regions are ragged and some small regions are lost. While in (b), almost all the regions are very clearly separated, the edges are fine and even some small regions are not merged. This is because the Lee filter tends to preserve noise in edges and it oversmooth the regions after several iterations. By contrast, the edge-enhanced Lee filter smooths noise near edges without destroying the edges and it does not merge two adjacent regions if there is an edge boundary between them.



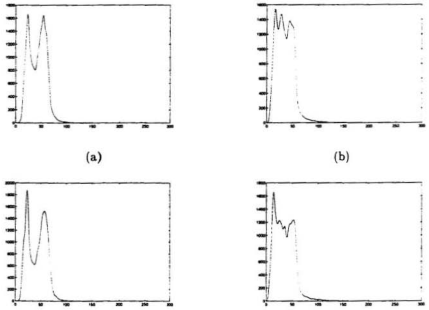
(a)

(b)

(c)

(d)

Figure 5.7: Histograms of filtered images: (a)*Fields* after iterative Lee filtering; (b)*Industry* after iterative Lee filtering; (c) *Fields* after edge-enhanced Lee filtering; (d)*Industry* after edge-enhanced Lee filtering.



(a)

(b)

(c)

(d)

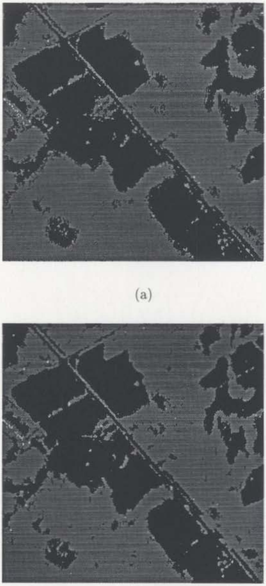
Figure 5.8: Gaussian smoothed histograms of filtered images: (a)*Fields* after Lee filtering; (b)*Industry* after Lee filtering; (c) *Fields* after edge-enhanced filtering; (d)*Industry* after edge-enhanced filtering.

Our testing on *Industry*, which contains more than three regions indicates that some regions are lost after Lee filtering, as seen in Figure 5.10 (a). This problem is more serious in busy areas. Whereas in Figure 5.10 (b), six regions are found and these regions are clearly separated even in busy areas. Also, the edges are more accurate and simple in Figure 5.10(b) than in Figure 5.10(a). Overall. the edge-enhanced segmentation method is a better solution than Lee and Jurkevich's method.

5.4 Overall discussion

The test results show that my proposed method performs better than Lee and Jurkevich's method in segmentation as well as in filtering. The test results show that when using the MSP-RoA, the choice of parameters window size 3 × 3. threshold $T_r = 0.5$, $D = 1$ can provide good estimates of the edge maps for the tested four-look airborne SAR images. So. for segmentation of this kind of real SAR images, choosing the above parameters is suitable for edge-enhanced filtering and segmentation. Also by our experiments, the filtering window size of 11 × 11 and 10 iterations are good choices. For histogram thresholding, Gaussian smoothing 5 or 6 times can provide a good thresholding result for these test images.

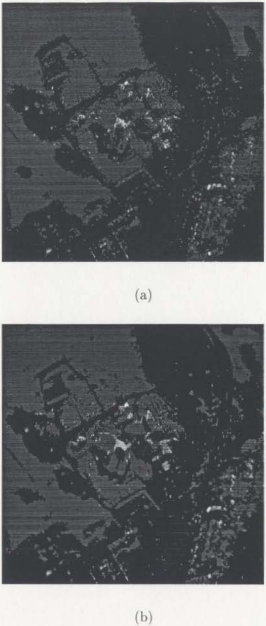
Because there are no noise-free images for comparison. MSE can not be used to measure the filtering performance. Various combinations of input parameters for filtering are used for testing the performances of the iterative Lee multiplicative filter and the edge-enhanced Lee multiplicative filter as shown in Table 5.1 and Table 5.2. These are no significant differences in the measure of σ_v for both filters. That means that in homogeneous regions, the noise smoothing performance are similar for these two filters. However, visual inspection can clearly show that the proposed



(a)

(b)

Figure 5.9: Segmentation results of *Fields*: (a) Based on iterative Lee filter; (b) Based on edge-enhanced Lee filter.



(a)

(b)

Figure 5.10: Segmentation results of *Industry*: (a) Based on iterative Lee filter; (b) Based on edge-enhanced Lee filter.

	σ_v after iteration (Original: 0.278), tested on *Fields*									
Iteration	1	2	3	4	5	6	7	8	9	10
3 × 3	0.168	0.114	0.122	0.079	0.062	0.057	0.056	0.045	0.054	0.047
5 × 5	0.111	0.099	0.048	0.031	0.027	0.023	0.023	0.023	0.021	0.026
7 × 7	0.110	0.027	0.024	0.022	0.025	0.020	0.021	0.021	0.025	0.022
9 × 9	0.051	0.026	0.020	0.021	0.021	0.023	0.020	0.023	0.023	0.023
11 × 11	0.062	0.030	0.023	0.022	0.019	0.022	0.021	0.022	0.021	0.021
	σ_v after iteration (Original: 0.253), tested on *Industry*									
Iteration	1	2	3	4	5	6	7	8	9	10
3 × 3	0.163	0.144	0.099	0.071	0.067	0.063	0.040	0.052	0.062	0.038
5 × 5	0.131	0.049	0.064	0.062	0.054	0.050	0.046	0.035	0.017	0.018
7 × 7	0.108	0.071	0.056	0.051	0.048	0.044	0.043	0.013	0.012	0.021
9 × 9	0.132	0.041	0.018	0.040	0.021	0.012	0.024	0.018	0.020	0.016
11 × 11	0.170	0.019	0.015	0.012	0.013	0.012	0.011	0.016	0.015	0.016

Table 5.1: Quantitative measures of iterative Lee multiplicative filter applied on real SAR images

	σ_v after iteration (Original: 0.278), tested on *Fields*									
Iteration	1	2	3	4	5	6	7	8	9	10
3×3	0.166	0.142	0.100	0.086	0.062	0.060	0.060	0.058	0.054	0.053
5×5	0.082	0.066	0.104	0.043	0.050	0.043	0.025	0.025	0.024	0.023
7×7	0.074	0.048	0.043	0.047	0.037	0.024	0.023	0.021	0.023	0.022
9×9	0.073	0.033	0.029	0.028	0.024	0.022	0.020	0.028	0.022	0.022
11×11	0.067	0.033	0.027	0.046	0.021	0.022	0.022	0.036	0.021	0.021
	σ_v after iteration (Original: 0.253), tested on *Industry*									
Iteration	1	2	3	4	5	6	7	8	9	10
3×3	0.229	0.100	0.074	0.101	0.052	0.047	0.086	0.054	0.038	0.037
5×5	0.202	0.085	0.045	0.067	0.045	0.048	0.054	0.052	0.051	0.019
7×7	0.123	0.060	0.026	0.022	0.019	0.042	0.042	0.041	0.037	0.038
9×9	0.253	0.078	0.059	0.058	0.014	0.014	0.013	0.012	0.012	0.041
11×11	0.100	0.047	0.015	0.014	0.013	0.012	0.046	0.011	0.036	0.010

Table 5.2: Quantitative measures of edge-enhanced Lee multiplicative filter applied on real SAR images

edge-enhanced method has better results in filtering and segmentation. especially in smoothing noise near edges, and in keeping sharp boundaries and preserving fine features. as shown in Figure 5.6. Figure 5.9 and Figure 5.10.

Chapter 6

Conclusions and recommendations

In this thesis, I have discussed the problem of segmenting SAR images. The presence of speckle noise makes the processing of this kind of image difficult. such that the techniques originally designed for either clean images or White Gaussian noise corrupted images tend to fail when applied to SAR images. Several previous techniques were developed for segmenting SAR images. One of these is Lee and Jurkevich's method [31] which is quite efficient in the segmentation of SAR images. However. there are some disadvantages in this method which have been discussed in this thesis. Based on Lee and Jurkevich's idea I have proposed the edge-enhanced segmentation method in this thesis, in order to improve the segmentation accuracy and efficiency. This method is tested and compared with Lee and Jurkevich's method. The test data include synthetic images and real multi-look airborne SAR images. The test results show a great improvement in noise smoothing (especially in edge areas), in edge and fine feature preservation, and in segmentation accuracy (especially along region boundaries and for small regions). This is because the proposed method uses edge information, obtained by a ratio-based edge detector, in its filtering algorithm in order to increase the accuracy in the estimation of local statistical parameters. Compared to Lee and Jurkevich's method, the proposed method also is an improvement in efficiency and automation. The filtering parameter σ_v is automatically estimated from the image on each iteration. Similarly, in histogram thresholding, the histogram valleys are automatically determined by the algorithm.

Further research is recommended in order to improve the performance and efficiency of the proposed edge-enhanced segmentation method. At present, the threshold T_r for the MSP-RoA edge detector is user determined. If we can develop some method to automatically estimate the value of the threshold T_r, it will no doubt improve the automation of the whole algorithm and make it much easier to use. For the filtering algorithm, the iteration number is also determined by the user experimentally, and different images may have different optimal iteration numbers. So, it is desirable to automatically control the filtering iterations. One method that may be suitable is to calculate the Root Mean Square (RMS) change in pixel values between iterations. If the RMS change is less than a certain value (*e.g.* less than 1 percent of the RMS change of the first iteration), the filtering stage may stop. For histogram thresholding, the number of convolutions with Gaussian kernel is also user determined. A fully automatic histogram thresholding algorithm is expected to develop.

More work can be done to modify other statistical speckle reducing filters such as the Sigma filter and the Weighted filter for their implementation in the proposed edge-enhanced filtering method. More performance measures specifically designed for SAR filtering and segmentation are also desirable. Comparisons of the edge-enhanced segmentation method with other segmentation methods, such as those based on region growing, fuzzy set or neural networks, are recommended. Finally, the edge-enhanced filtering and segmentation method proposed in this thesis should be tested in real application requiring SAR image analysis. such as sea-ice segmentation. forest classification. crop identification. etc.

Bibliography

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

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

[3] Barbarossa, S. and Parodi, C., " SAR image classification by wavelets", *Proceedings of IEEE International Radar Conference*, pp. 462-467, Alexandria, U.S.A., 1995.

[4] Beauchemin, M., Thomson, Keith P.B. and Edwards G., "SAR adapted techniques for image analysis", *International Geoscience and Remote Sensing Symposium*, pp. 175-177, 1995.

[5] Benie, G.B., Boucher, J.M., Plehiers, S., Massalabi, A. and Wang, S., "Pyramidal contextual classification of SAR images", *Procedings of 1995 IGRSS*, pp. 913-917, Firenze, Italy, 1995.

[6] Bezedek, J.C., *Pattern Recognition with Fuzzy Objective Function Algorithms*, New York: Plenun Press, 1981.

[7] Blanze, W.E. and Gish, S.L., "A connectionist classification architecture applied to image segmentation", *Proc. 10th ICPR*, pp. 272-277, 1990.

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

[9] Caves, R.G., Quegan, S. and White, R.G., "The use of segmentation for change detection in spaceborne SAR images", *Proceedings of 19th Annual Conference of the Remote Sensing Society: Towards Operational Application*, pp. 168-175, 1993.

[10] Ceccarelli, M., Farina, A., Petrosino, A., Vaccaro, R. and Vinelli, F., "SAR image segmentation using textural information and neural classifiers", *ONDE Electrique*, Vol. 14, pp. 24-28, 1994.

[11] Curlander, J.C. and McDonough, R.N., *Synthetic Aperture Radar: Systems and Signal Processing*, New York: John Wiley and Sons, 1991.

[12] Derin, H., Elliott, H., Cristi, R. and Geman, D., "Bayes smoothing algorithms for segmentation of binary images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, pp. 707-720, 1984.

[13] Derin, H. and Kelly, P.A., "Algorithms for the segmentation of speckled images", *Conference Record - Asilomar Conference on Circuts, Systems and Computers*, pp. 276-284, Pacific Grove, U.S.A., 1989.

[14] Du, L. and Lee, J.S., "Fuzzy classification of earth terrain covers using complex polarimetric SAR data", *International Journal of Remote Sensing*, Vol. 17, pp. 809-826, 1996.

[15] Fosgate. C.H.. Krim. H.. Irving, W.W.. Karl. W.C. and Willsky. A.S.. "Multiscale segmentation and anomaly enhancement of SAR imagery". *IEEE Transactions on Image Processing*, Vol. 6, pp. 7-20. 1997.

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

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

[18] Ganugapati, S.S., "Edge detection methods for speckled images". M.Eng. thesis. Memorial University of Newfoundland, July, 1996.

[19] Ganugapati, S.S. and Moloney. C.R., "A ratio edge detector for speckled images based on maximum strength edge pruning". *Proceedings of ICIP-95* . pp. 165-168. Washington. D.C., October 1995.

[20] Geman, S. and Geman. D.. "Stochastic relaxation. Gibbs distribution. and the Bayesian restoration of images", *IEEE Transations on Pattern Analysis and Machine Intelligence*, Vol. 6, pp. 721-741, 1984.

[21] Ghosh. A., Pal, N.R. and Pal, S.K., "Object background classification using Hopfield type neural network". *International Journal of Pattern Recognition and Artifical Intelligence*, Vol, 5, pp. 989-1008, 1992.

[22] Grossberg, S., Mingollar, E. and Williamson, J., "Multiple scale neural system for boundary and surface representation of SAR data", *Proc. the 5th IEEE Work-*

shop on Neural Networks for Signal Processing. pp. 313-322. Cambridge. U.S.A.. 1995.

[23] Goodman. J.W.. "Some fundamantal properties of speckle". *J. Opt. Soc. Am.*. Vol. 66, No. 11. pp. 1145-1150. 1976.

[24] Haralick. R.M. and Shapiro. L.G.. "Survey: image segmentation techniques". *Computer Vision, Graphics, and Image Processing*. Vol. 29. pp. 100-132. 1985.

[25] Hegarat-Mascle. S.L., Vidal-Madjav, D. and Oliver. P.. "Application of simulated annealing to SAR image clustering and classification problems". *International Journal of Remote Sensing*, Vol.17, pp. 1762-1776. 1996.

[26] Hoekman. D.H., "Speckle ensemble statistics of logarithmically scaled data". *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 29. pp. 180-182. 1991.

[27] Hudson, D.V. and Jernigan. M.E.. "Speckle suppression and texture in synthetic aperture radar images", *Techniqual Report No. 160-I-120988*. Waterloo. Canada. Aug. 1988.

[28] Ju, C. and Moloney. C.. "An edge-enhanced segmentation method for SAR images". *Proceedings of IEEE Canadian Conf on Elec and Comp Engr (CCECE-97)*, pp. 599–602, St. John's, Canada, May 1997.

[29] Ju, C. and Moloney, C., "An edge-enhanced modified Lee filter for the smoothing of SAR image speckle noise", *Proceedings of 1997 IEEE Workshop on Nonlinear Signal and Image Processing*, Mackinac Island, USA, September, 1997.

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

[31] Lee. J.S. and Jurkevich. I., "Segmentation of SAR images". *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 27, No. 6. pp. 674-680. November 1989.

[32] Lee. J.S.. "Speckle analysis and smoothing of Synthetic Aperture Radar images". *Computer Graphics and Image Processing*, Vol. 17, pp. 24-32. 1981.

[33] Lee, J.S.. "Digital image smoothing and the sigma filter". *Computer Graphics and Image Processing*, Vol. 24, pp. 255-269, 1983.

[34] Lee, J.S., "Speckle suppression and analysis for Synthetic Aperture Radar images", *Optical Engineering*, Vol. 25, No. 5, pp. 636-643. 1986.

[35] Lee, J.S.. Jurkevich. I., Dewaele. P., Wambacq, P. and Vosterlick. A.. "Speckle filtering of Synthetic Aperture Radar images: a review". *Remote Sensing Reviews*, Vol. 8, pp. 313-340. 1994.

[36] Lee, J.S.. "Refined filtering of image noise using local statistics". *Computer Graphics Image Processing*, Vol. 15. pp. 255-269. 1981.

[37] Li. C., "Two adaptive filters for speckle reduction in SAR images using the variance of ratio", *International Journal of Remote Sensing*, Vol. 9. No. 4. pp. 641-653, 1988.

[38] Lopes, A.E., Nezry, E., Touzi, R. and Laur, R., "Structure detection and statistical adaptive filtering in SAR images", *International Journal of Remote Sensing*, Vol. 14, No. 9, pp. 1735-1758. 1993.

[39] Martin, F.J. and Turner, R.W., "SAR speckle reduction by weighted filtering". *International Journal of Remote Sensing*, Vol. 14. No. 9. pp. 1759-1774. 1993.

[40] Pal, N.R. and Pal, S.K., "A review on image segmentation techniques". *Pattern Recognition*, Vol. 26, pp. 1277-1294, 1993.

[41] Pal, S.K. and King, S.K., "Image enhancement using fuzzy sets". *Electronic Letter*. Vol. 16, pp. 376-378, 1980.

[42] Pal, S.K. and Resenfeld, A., "Image enhancement and thresholding by optimization of fuzzy compactness", *Pattern Recognition Letter*, Vol. 7, pp. 77-86. 1988.

[43] Rosenfeld, A. and Kak, A.C., *Digital Picture Processing*, Academic Press, New York, 1982.

[44] Rignot, E., Chellappa, R. and Dubois, P., "Unsupervised segmentation of polarimetric SAR data using covariance matrix ". *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 30, pp. 697-705. 1992.

[45] Russ, J.C., *The Image Processing Handbook*, CRC Press, 1992.

[46] Smith, D.M., Barret, E.C. and Scott, J.C., "Sea ice type classification from ERS-1 SAR data basd on grey level and texture information". *Polar Record*. Vol. 31. pp. 135-146. 1995.

[47] Smith, D.M., "Speckle reduction and segmentation of Synthetic Aperture Radar images", *International Journal of Remote Sensing*, Vol. 17, No. 11, pp.2043-2057, 1996.

[48] Sephton. A.J.. Brown. L.M.J.. Marcklin. J.T.. Partinton. K.C. and Veck. N.J.. "Segmentation of synthetic aperture radar imagery of sea-ice". *International Journal of Remote Sensing.* Vol. 15, pp. 803-825. 1994.

[49] Touzi. R.. Lopes. A. and Bousquet, P., "A statistical and geometrical edge detector for SAR images", *IEEE Transactions on Geoscience and Remote Sensing.* Vol. 28. pp. 764-773. 1988.

[50] Tsai. D.M., "A fast thresholding selection procedure for multimodal and unimodal histograms", *Patten Recogonition Letters*, Vol. 16, pp. 653-666. June 1995.

[51] Trivedi, M. and Bezdek, J.C., "Low-level segmentation of aerial images with fuzzy clustering", *IEEE Transactions on Sys. Man Cybern.*, Vol. 16. pp. 589-598, 1986.

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

[53] Ward. S.. *Digital Signal and Image Processing of Airborne Synthetic Aperture Radar*, NSERC USRA Report. Memorial University of Newfoundland. September, 1993.

[54] Wakabayashi, H. and Arai, K.. "A new method for SAR speckle noise reduction (CST filter)", *Canadian Journal of Remote Sensing.* Vol. 22, pp. 190-197. 1996.

[55] Zaman, M.R. and Moloney, C.R., "A comparison of adaptive filters for edge-preserving smoothing of speckle noise", *Proceedings of ICASSP–93.* pp. V77-80. Minneapolis, Minnesota, April, 1993.