A Thesis

entitled

Extraction of Linear Features Based on Beamlet Transform

by

Yuan Zhu

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the
Master of Science Degree in Electrical Engineering

Dr. Ezzatollah Salari, Committee Chair

Dr. Junghwan Kim, Committee Member

Dr. Jackson Carvalho, Committee Member

Dr. Patricia R. Komuniecki, Dean
College of Graduate Studies

The University of Toledo
May 2011

Copyright 2011, Yuan Zhu

This document is copyrighted material. Under copyright law, no parts of this document may be reproduced without the expressed permission of the author.

An Abstract of
Extraction of Linear Features Based on Beamlet Transform

by

Yuan Zhu

Submitted to the Graduate Faculty as partial fulfillment of the requirements for the
Master of Science Degree in Electrical Engineering

The University of Toledo
May 2011

Over the past few decades, Synthetic Aperture Radar (SAR) images have been widely used to estimate the various features on the ground. As SAR is a radar system, its images are degraded by noise which limits the application of SAR images. Therefore, extracting features from SAR images with noise is an important issue. The goal of this thesis is to develop and implement a beamlet based method to extract the linear features from Synthetic Aperture Radar images with noise and a low signal-to-noise ratio (SNR). The basic concept of beamlet analysis is to approximate the linear objects by line segments called beamlets. Beamlet analysis performs well in a noisy environment; therefore, a beamlet based algorithm for linear feature extraction is proposed in this thesis. The proposed method uses digital image pre-processing techniques to offset the noise and low-contrast problems and to recalculate pixel values. Linear features such as a road network are then extracted by applying the beamlet transform based algorithm. The algorithm recursively partitions the image into sub-squares to build a beamlet dictionary to perform the transform. The complete linear features are then obtained with the post-processing algorithm to link the

discontinuities. Experimental results have demonstrated the effectiveness of this method.

I would like to dedicate this thesis to my grandparents, who passed away during my graduate level study at University of Toledo. They gave me a happy childhood and so much tender loving care.

Acknowledgments

First of all, I would like to express the deep appreciation to my academic advisor Dr. Ezzatollah Salari for his guidance and help. He always patiently answered my various questions about research and work, carefully read the thesis, corrected errors, pointed out the problems and provided valuable suggestions. I would never have completed this work without his advice and encouragement. In addition, I would like to thank Dr. Kim and Dr. Carvalho for being my defense committee.

I would like to give my thanks to my dear friends at UT, Guanqun Bao, Xinren Yu, Xueying Chen, Xianan Feng, and so on. Thanks for their help and friendship, the time spent with them will be my life-long treasure and I will always cherish it.

I also want to dedicate my deepest gratitude to my parents. I do not have words to adequately describe my appreciation for all they have done for me in the past twenty-five years. They are the greatest parents and I will always love them.

Finally, I would like to thank my husband, Ao Tang, who is always there for me through all the ups and downs.

Table of Contents

Abstract	iii
Acknowledgments	vi
Table of Contents	vii
List of Tables	x
List of Figures	xi
1 Introduction	1
1.1 Background.....	1
1.2 Motivation of the Research	2
1.3 Outline of Thesis.....	3
2 Literature Review	4
2.1 Edge Detection Operators	4
2.2 Hough Transform.....	8
2.3 Detection of Linear Features Based on Wavelet Analysis	10
2.4 Active Contour Models	13
2.5 Detection of Linear Features Based on Mathematical Morphology.....	14
3 Pre-Processing Techniques	17
3.1 Various Types of Distortions in SAR Images	17

3.1.1 Radiometric Distortion	17
3.1.2 Geometrical Distortion	19
3.1.3 Atmospheric Distortion.....	20
3.2 Proposed Pre-Processing Algorithm	21
3.2.1 Conversion from Original Image to Grayscale Image	22
3.2.2 Histogram Equalization	25
3.2.2.1 Implementation of the Histogram Equalization	25
3.2.2.2 Example of Histogram Equalization.....	26
3.2.3 Median Filtering.....	28
3.2.3.1 Overview of Median Filtering.....	28
3.2.3.2 Example of Median Filtering in SAR Images.....	29
3.3 Summary of the Pre-Processing.....	30
4 Detection of Linear Features Based on Beamlet Transform	31
4.1 Overview of Beamlet Analysis	31
4.2 Introduction of Beamlet Transform	33
4.3 Linear Feature Extraction with Beamlet Transform in Noisy Images	35
4.4 Implementation of Linear Feature Extraction Based on Beamlet Transform	38
4.4.1 Image Partitioning	39
4.4.2 Build Beamlet Dictionary	39
4.4.3 Beamlet Transform.....	40
4.4.4. Threshold	40
4.4.5 Gradient Test.....	40

4.4.6 Summary of the Algorithm Based on Beamlet Transform for Linear Feature Extraction.....	42
4.5 Example of Applying the Algorithm Based on Beamlet Transform for Linear Feature Extraction.....	43
5 Post-Processing Techniques	44
5.1 Description of Post-Processing Algorithm	44
5.1.1 Initialization of Data Structure.....	44
5.1.2 Calculation of Beamlet Orientation.....	45
5.1.3 Calculation of Orientation Difference Between Beamlets	45
5.1.4 Connectivity Analysis using Angle and Distance Measures	46
5.2 Summary of the Post-Processing Algorithm	47
5.3 Example of Applying the Post-Processing Algorithm	49
6 Experimental Results and Analysis	50
6.1 Experimental Results	51
6.1.1 Experiment 1	51
6.1.2 Experimental 2	54
6.1.3 Experimental 3	57
6.2 Analysis of Experimental Results.....	59
7 Conclusion and Future Work	61
References	63

List of Tables

2.1 Some advantages and disadvantages of edge detection operators.....	7
4.1 Kernels of two gradient operators	41
5.1 Data structure for post-processing	45
5.2 Results of vector dot product.....	46

List of Figures

1-1 Synthetic Aperture Radar (SAR) images	1
2-1 Hough transform for straight lines	9
2-2 Image decomposition based on wavelet transform	13
3-1 Flow chart of pre-processing steps	22
3-2 Conversion from RGB image to grayscale image	23
3-3 An example of conversion from color image to grayscale image	24
3-4 Histogram equalization of Figure 3-3(a).	27
3-5 The histogram equalization result of Figure3-3(a).....	27
3-6 Example of median filtering with an odd number of entries	29
3-7 Results of applying the median filtering.....	29
4-1 Approximating a line segment by beamlets	31
4-2 Beamlets at different scales.....	34
4-3 Discrete beamlet transform can be viewed as a weighted sum of pixel values	35
4-4 Nearly identical sensitivity of the maximum beam statistic and the maximum beamlet statistics.....	38
4-5 Flow chart of linear feature extraction using beamlet transform	42
4-6 Result of applying beamlet based method to Figure 3-7(a).....	43
5-1 Flow chart of post- processing techniques.....	48

5-2 Result of applying the post-processing algorithm to Figure 4-6.....	49
6-1: Results of Experimental 1	53
6-2 Results of Experimental 2	56
6-3 Results of Experimental 3	59
6-4 An example of Figure 6-1(f) with linked area marked	60

Chapter 1

Introduction

1.1 Background

Synthetic Aperture Radar (SAR) images provide useful information for many applications, they also present problems where the objects of interest consist of many linear features, for example, a road network or a river network seen from above[1], as shown in Figure 1-1.

Figure 1-1: Synthetic Aperture Radar (SAR) images

Many applications require the up-to-date information on road network data, for example, updating for geographic information systems (GIS), traffic monitoring, or

navigation, and so on. As an active system, Synthetic Aperture Radar (SAR) which can operate independently of daylight and good weather conditions [2] holds some advantages over optical image acquisition. Therefore, the road network information extracted from SAR images offers a suitable alternative to road extraction from optical images. In the past few years, attention has been brought to the extraction of road networks from SAR images and much research have been done about the extraction and update of road networks.

1.2 Motivation of the Research

As a fundamental tool, the edge detectors identify points in the digital image where the brightness changes or discontinuities exist. This involves convolution of the image with an operator which is constructed to be sensitive to large gradients in the image while returning values of zero in uniform regions. These classical ‘edge detectors’ seem to work well when noise is nonexistent or weak [3], [4]. However, many imaging systems produce images with noise, which reduce the detectability of targets and impede further investigation of SAR images [5]. Thus, extracting linear features from a noisy environment becomes an important issue in SAR image processing.

The beamlet transform, in which the role of line segments is similar to the role of points in wavelet transform, is based on dyadically-organized line segments with a wide range of scales, locations and orientations. The beamlets are a data structure

forming a multi-scale framework for analyzing the linear and curvilinear features [6]-[8]. Thus, a method based on beamlet transform of linear feature extraction is proposed in this thesis.

1.3 Outline of Thesis

The primary goal of this thesis is to implement a method based on beamlet transform to extract the linear features in SAR images. The rest of the thesis is organized as follows:

Chapter 2 is a literature review of some major approaches of linear feature extraction.

Chapter 3 describes the pre-processing steps aiming to offset problems with noise or brightness and recalculate pixel values that minimize these problems.

In Chapter 4 and Chapter 5, a method, based on beamlet transform, is proposed for linear feature extraction followed by a post-processing algorithm to link the discontinuous line segments.

In Chapter 6, experimental results are presented to illustrate the effectiveness of the approach.

The conclusion and future work are presented in Chapter 7.

Chapter 2

Literature Review

During the past few years, many approaches have been developed to perform linear feature extraction. In image analysis, linear feature extraction (edge detection) is a problem of fundamental importance. Linear features (edges) are useful for segmentation, registration and identification of objects in a scene as these features characterize object boundaries. Edge detecting an image can significantly reduce the amount of data and filter out useless information, while preserving the important structural properties and information in an image [9].

There are many ways to perform edge detection. Some of them are different edge detection operators, Hough transform, and the related applications of edge detection can also be found in wavelet analysis, morphology, and so on. The following sections will review some of the major approaches of linear feature extraction.

2.1 Edge Detection Operators

The majority of the edge detection operators can be grouped into two categories, gradient based and zero-crossing based. The gradient based method detects the edges

by looking for the maximum and minimum in the first derivative of the image, while the zero-crossing based method finds edges by searching for zero crossings in the second derivative of the image [10].

Robert, Prewitt, and Sobel are commonly used gradient based operators in image processing techniques [9]. The location of an edge can be highlighted by calculating the derivative of the image. In order to recover the edges after applying the edge operators, the global or local threshold operator must be applied to the gradient images for segmentation. The resulting segmentation and perceived quality of the edge detector is determined by the choice of the threshold value, therefore, a cumulative histogram of the gradient image is usually considered in selecting an appropriate threshold value.

The zero-crossing edge operator was originally proposed by Marr and Hildreth [11]. They suggested that the operator needs to have two essential characteristics to effectively detect intensity changes (edges). First, it must be a differential operator, taking either a first or second spatial derivative of the image. Second, it must be capable of adapting to any desired scale, so that blurry shadow edges can be detected by large filters and sharply focused details can be detected by small filters.

One of the zero-crossing operators is the compound Laplacian-of-Gaussian edge operator that combines a smoothing operation and a differentiation operation. This operator uses a Gaussian-shaped, linear-phase FIR filter and discrete Laplacian filter separately [9], [10] to identify the edges by the locations of zero crossings.

Canny proposed the Canny edge detection operator [12] in his paper “A

Computational Approach to Edge Detection”. In this paper, he followed a list of criteria to improve methods of edge detection.

(1) Low error rate.

This is important because edges occurring in images should not be missed and there are no responses to non-edges either. Therefore, the algorithm should mark as many real edges in the image as possible.

(2) Good localization of edge points.

The second criterion is that the edge points be well localized. In other words, the marked edges should be as close as possible to the edges in the real image.

(3) Minimal response.

The third criterion is that there is only one response to a single edge, and no false edges are created by the noise in the image. The third criterion was implemented because the first two were not substantial enough to completely eliminate the possibility of multiple responses to an edge.

Based on these criteria, the Canny edge detector uses a filter based on the first derivatives of Gaussian to smooth the image and eliminate the noise. It then finds the image gradient to highlight regions with high spatial derivatives, and the direction of gradient is rounded to one of four angles representing vertical, horizontal and the two diagonals in this stage. After the edge directions are known, non-maximum suppression is applied to trace along the edge in the edge direction and suppress any pixel value that is not considered to be an edge, a set of points that are often referred as thin edges is obtained in the stage of non-maximum suppression. The gradient

array is further reduced by hysteresis which tracks along the remaining pixels that have not been suppressed. Hysteresis uses two thresholds—high and low. The high threshold is applied and any pixel in the image that has a value greater than this high threshold is presumed to be an edge pixel, the low threshold is then applied to trace any pixels that are connected to this edge pixel until it reaches the starting point.

The Canny algorithm contains a number of adjustable parameters, which affects the computation time and effectiveness of the algorithm. The parameters are chosen depending on the particular requirements of a given environment. Two key parameters are explained in the following.

(1) The size of the Gaussian filter: the smoothing filter used in the first stage directly affects the results of the Canny algorithm. Smaller filters cause less blurring, and allow detection of small, sharp lines, while larger filters cause more blurring, smearing out the value of a given pixel over a larger area of the images.

(2) Thresholds: two thresholds with hysteresis are more flexibility than a single-threshold approach, but general problems of threshold approaches still exist. For example, a threshold that is set too high will result in missing important information. On the other hand, a threshold set too low will falsely identify irrelevant information such as noise. It is difficult to give a generic threshold that works well on all images. No tried and tested approach to this problem yet exists.

Table 2.1 presents some advantages and disadvantages of edge detection operators that are discussed in this section.

Table 2.1: Some advantages and disadvantages of edge detection operators

Edge Operator	Advantages	Disadvantages
Classical (Sobel, Prewitt...)	Simplicity. Detection of edges and their orientations.	Sensitivity to noise. Inaccurate.
Zero Crossing (Laplacian second directional derivative)	Detection of edges and their orientations. Having fixed characteristic in all directions.	Responding to some of the existing edges. Sensitivity to noise.
Laplacian of Gaussian(LoG)	Finding the correct places of edges, Testing wider area around the pixel.	Malfunctioning at the corners, curves and where the gray level intensity function varies. Not finding the orientation of edge because of using the Laplacian filter.
Gaussian (Canny)	Using probability for finding error rate. Localization and response. Improving signal to noise ratio. Better detection especially in noise conditions.	Complex computations. False zero crossing. Time consuming.

2.2 Hough Transform

Duda and Hart invented the widely-used ‘Generalized Hough Transform (GHT)’ which is related to the patent of Paul Hough [13]. Hough transform estimates the parameters of the linear features from boundary points; in other words, it finds the imperfect instances of objects within a certain class of shapes by a voting procedure. This voting procedure is carried out in a parameter space, from which object candidates are obtained as local maxima in an accumulator space that is explicitly constructed by the algorithm for computing the Hough transform [14]. An example is given in Figure 2-1 to illustrate the Hough Transform.

Figure 2-1: Hough transform for straight lines--the data space (a) is mapped into model space (b).

For all points $[x_n, y_n]^T$ on a straight line, the following equation is used to represent the straight line:

$$y_n = a_0 + a_1 x_n \quad (2.1)$$

where a_0 and a_1 represent the offset and the slope of the line, respectively.

Equation (2.1) can also be read as a condition for the parameters a_0 and a_1 :

$$a_1 = \frac{y_n}{x_n} - \frac{1}{x_n} a_0 \quad (2.2)$$

The equation (2.2) can represent a line in a new space spanned by the parameters a_0 and a_1 , this new space is named the model space. In the model space, the line has the offset (y_n / x_n) and a slope of $(-1 / x_n)$. Each point reduces the model space to a line, thus, a line can be drawn in the model space for each point in the data space. All lines in the model space meet at one point which gives the parameters a_0 and a_1 of the lines. Since a line segment contains many points, a reliable estimate of the

two parameters of the line can be obtained. In this way, a line in the data space is mapped onto a point in the model space. The transformation from the data space to the model space via model equation is the Hough transform.

However, the process of Hough transform is very time-consuming for a large-sized image; because for each point in the image, a line must be computed in the parameter space, and the increment of each point in the model space through which the line passes must be computed as well. Moreover, the false peaks mixing up with the short filaments, which are generated by random alignments, also become the inevitable disadvantage of the GHT [15], [16]. Much of the efficiency of the Hough transform is dependent on the quality of the input data, that is to say, the linear features must be detected well for the Hough Transform to be effective. Otherwise, the Hough transform may not be able to provide accurate results.

The Radon transform is sometimes preferred to detect linear features when the image is corrupted by noise, because it attenuates the noise through summation [17]. The Radon transform is named after the Austrian mathematician Johann Karl August Radon [18]. The main application of the Radon transform is Computed Axial Tomography (CAT) scans, where the inverse Radon transform is applied.

2.3 Detection of Linear Features Based on Wavelet Analysis

As edges in digital images represent the abrupt changes of the brightness, they

can be mathematically defined as local singularities. The Fourier transform was a useful mathematical tool for analyzing singularities, however, the Fourier transform is global and not well adapted to local singularities. Thus, it is difficult to find the location and spatial distribution of singularities with Fourier transforms. In other words, the Fourier analysis transforms a signal based in time to one based in frequency, but the disappearance of time information when a signal is transformed to the frequency domain becomes a serious disadvantage, meaning that when analyzing the frequency component, it is impossible to know when an event occurred in time [19].

Time-frequency analysis is a critical issue in the area of image analysis, and the frequency properties of a function in a local time interval are needed. The wavelet analysis provides a solution to overcome the shortcomings of the Fourier transform. A wavelet is a short wave that has an average value of zero [12]. Different from sinusoids that theoretically extend from minus to plus infinity, wavelets, which are irregular, of limited duration, and often non-symmetrical, have a beginning and an end. With the growth of wavelet theory, the wavelet transform has been found to be a remarkable tool to analyze singularities including the edges, and further, to detect them effectively.

This idea of edge detecting with wavelet analysis is similar to that of Canny's algorithm [12]. The Canny approach selects a Gaussian function as a smoothing function θ , while the wavelet-based approach chooses a wavelet function to be θ' . Mallat, Hwang, and Zhong [20], [21] proved that the maxima of the wavelet

transform modulus can detect the location of the irregular structures. One and two-dimensional signals can be reconstructed from the local maxima of their wavelet transform modulus with a good approximation.

In wavelet analysis, the local regularity of signals is characterized by decomposing signals into elementary building blocks that are well localized both in space and frequency. This explains the basic mechanism of classical edge detectors, and indicates a way of constructing optimal edge detectors under specific conditions. Based on the image decomposition model of wavelet transform, the original image can be divided into low-frequency information and high-frequency information. After two-dimensional frequency decomposition of wavelet transform, low-frequency information can be further decomposed to low-frequency area LL and high-frequency area LH. Similarly, high-frequency information can be decomposed to low-frequency area HL and high-frequency area HH.

LL shows the smooth sub-band of the original image which contains most of the information in original image, LH and HL represent the vertical and horizontal edge information of the original image, and HH preserves the diagonal details which are greatly influenced by noise. The process of the image decomposition based on wavelet transform is shown in Figure 2-2.

Figure 2-2: Image decomposition based on wavelet transform

Generally, the idea of edge detection using wavelet transform is described as below [20], [21].

- (1) Choose a suitable wavelet function.
- (2) Use the function to transform images into decomposition levels.
- (3) Filter out the wavelet detailed coefficients containing significant energy at noise scales are.
- (4) Detect the edges from the filtered detailed coefficients.

Note that this traditional edge detection method based on wavelet transform extracts the low-frequency sub-image to further detect the edges, which will discard some important details. In addition, the performance of extracting edge from a sub-image will be affected by the noise in the high-frequency area of this sub-image [22].

2.4 Active Contour Models

An active contour model, also known as the snake, which transforms the problem of extracting the image contours into minimizing the energy [23], was presented by

Kass. Snakes are curves defined in the image domain that can move actively toward the image contours under internal, external, and constrained forces which are pre-defined until its energy is minimal. Actually, no prior knowledge is needed because this model uses the global information of the image contours to converge to the curve [24]. On the other hand, some weaknesses still exist in the minimization procedure of the active contour model [25]. First, the initial contour should be as close as possible to the real contour; otherwise, the result may be inaccurate or misleading. Second, moving a part of the snake into the boundary concavity is of great difficulties. There are several improvement methods aiming to solve those defects, such as pressure forces, distance potentials, and gradient vector flow (GVF). Each of these methods builds a new external force [26]. However, these improved methods also show a number of limitations as well. For example, calculating the GVF force field always results in obvious high computational complexity.

2.5 Detection of Linear Features Based on Mathematical Morphology

Developed from set theory, mathematical morphology was introduced by Matheron [27] as a technique for analyzing the geometric structure of metallic and geologic samples. It was extended to the area of image analysis by Serra [27]. Unlike the traditional mathematical modeling and analysis, mathematical morphology

provides an alternative approach to image processing based on the shape concept. In the theory of mathematical morphology, the images are treated as sets, whose operations are defined by set arithmetic, meaning that the image which will be processed by mathematical morphology theory must be changed into a set, and the operations on the image transform one set to another. Mathematical morphology carries out the image processing by using structuring elements to measure the shape of an image.

Dilation and erosion are two basic mathematical morphological operators; other morphological operators are various combinations of these two. Several mathematical operators are introduced in the following.

Let $F(x,y)$ denote a gray-scale image, B denotes the structuring element which can be any shape. The following four morphological operations are commonly used in image analysis:

(1) Dilation

The dilation operation of a gray-scale image $F(x,y)$ by the structuring element $B(s,t)$ is given by

$$(F \oplus B)(x, y) = \max\{F(x - s, y - t) + B(s, t)\} \quad (2.3)$$

(2) Erosion

The erosion operation of a gray-scale image $F(x,y)$ by the structuring element $B(s,t)$ is given by

$$(F \ominus B)(x, y) = \min\{F(x - s, y - t) - B(s, t)\} \quad (2.4)$$

(3) Opening

The opening operation of a gray scale image $F(x,y)$ by the structuring element $B(s,t)$ is given by

$$F \circ B = (F \ominus B) \oplus B \quad (2.5)$$

(4) Closing

The closing operation of a gray scale image $F(x,y)$ by the structuring element $B(s,t)$ is given by

$$F \bullet B = (F \oplus B) \ominus B \quad (2.6)$$

Dilation increases the gray-scale values of an image as it is a transform of expanding, while erosion decreases the gray-scale values of an image as it is a transform of shrinking. These two basic morphological operators are sensitive to the changes in intensity, therefore, they can be used for edge detection. Generally, the opening operation smoothes the contour of an image and breaks narrow gaps; in contrast, the closing operation tends to fuse narrow gaps and eliminate small holes.

Appropriate structuring elements are essential to design an effective morphological edge detector. Therefore, selecting the structuring element and designing the morphological filter are the two most important issues that determine the results of edge detection.

Chapter 3

Pre-Processing Techniques

The data can be recovered by a series of pre-processing techniques to overcome the flaws and deficiencies in the raw data. The pre-processing techniques consist of operations to compensate for systematic errors, including the removal of the noise and other distracting effects to restore the image to its original condition.

3.1 Various Types of Distortions in SAR Images

This section will discuss the distortions in the original SAR images and the solutions to these issues. The problems that are considered in the pre-processing step are briefly categorized as below [28]:

- (1) Radiometric distortion.
- (2) Geometric distortion.
- (3) Atmospheric distortion.

3.1.1 Radiometric Distortion

The radiometric distortion is mainly due to the instruments used for recording the

data and the effect of the atmosphere. Radiometric processing influences the brightness values of an image to adjust the values to compensate for atmospheric degradation or to correct for sensor malfunctions. There are two types of radiometric distortion: the first one is that the relative distribution of brightness over an image in a given band can be different from that in the ground scene; the second one is the relative brightness of a single pixel from band to band can be distorted compared with the spectral reflectance character of the corresponding region on the ground.

Generally, line-dropouts and de-striping are two methods targeting the removal of such defects. Line-dropouts usually correct the problem either by replacing the defective line with a duplicate of the preceding or subsequent line, or taking the average of the two. If a spurious pixel located at (x,y) has a value $N(x,y)$, then the algorithms are simply:

$$N(x,y) = N(x,y-1), \quad (3.1)$$

or

$$N(x,y) = [N(x,y-1) + N(x,y+1)]/2 \quad (3.2)$$

If one or more detectors go out of adjustment in a given band, the banding or striping are likely to occur. Electro-mechanical scanners such as Landsat's MSS and TM produce the systematic horizontal banding pattern seen on images and resulting in repeated patterns of lines with consistently high or low digital numbers. Therefore, two reasons are put forward in favor of applying a 'de-striping' correction. First, it can improve the visual appearance and interpretability of the image. Second, equal pixel values in the image are more likely to represent areas of equal ground leaving

radiance while other things being equal. Two methods of de-striping are explained as follows:

(1) A histogram is created for each detector of the problem band. For instance, there are six histograms generated from six detectors, and then the mean and standard deviation are calculated for each of the six histograms. Assuming that the proportion of the pixels representing different objects in the image are the same for different detectors, the mean and standard deviation should be the same for all of the six histograms. However, the stripes are characterized by distinct histograms. Equalization of the means and standard deviations of the six detectors is performed to reach the equal selected values, which are usually the mean and standard deviation of the entire image.

(2) The second method is a non-linear method in the sense that the relationship between radiance r_{in} (received at the detector) and r_{out} (output by the sensor) is not describable in terms of a single linear segment.

Noisy pixels can be replaced by the average value of its neighborhood. Usually, moving windows of 3×3 or 5×5 pixels are applied to suppress random noise.

3.1.2 Geometrical Distortion

Geometrical distortion is likely to occur due to earth curvature, non-linearity in scanning and platform motion. The geometrical distortion can be categorized as non-systematic and systematic. The non-systematic distortions are raised by variations in spacecraft variables, while the systematic distortions are constant and predictable.

Therefore, the systematic distortions are well understood and easily corrected by applying formulas derived by modeling the sources of distortions mathematically.

Rectification and registration are two methods both dealing with the geometrical distortion. The process of projecting data onto a plane and correcting the distortion by transforming the data into a standard coordinate system is called rectification, while registration makes an image to conform to another image.

3.1.3 Atmospheric Distortion

The quality of the output image from the instrument on a satellite is determined by the intensity and spectral distribution of the energy received at the satellite. Since the intensity and spectral distribution of energy and radiation travel some distance through the atmosphere and suffer both attenuation and augmentation, when the data generated by the remote sensing fails to regenerate the correct radiation properties of the object, the problem of atmospheric distortion occurs.

Necessary corrections in brightness and geometry are needed for accuracy when utilizing an image. Several reasons are stated for the necessity of correction.

(1) First, since the effect of atmospheric scattering depends on the wavelength, the computed ratios derived in two bands of multi spectral image will not reflect the true ratio accurately due to the unequally effect from the two channels.

(2) Second, there are situations in which the land surface reflectance and the sea surface temperatures are needed from the image.

(3) Third, two images taken at different times are needed to make a comparison.

There are a number of ways of performing the atmospheric correction to correct the image data if the effect of the atmosphere cannot be ignored. One important step in the correction is collecting the ground information such as target temperature and reflectance and calibrating these values or quantities on the ground. In addition, modeling the absorption or scattering effects for the measurement of the composition and temperature profile of the atmosphere, and utilizing the information about the atmosphere inherent to remotely sensed data are also effective solutions for atmospheric correction.

3.2 Proposed Pre-Processing Algorithm

Pre-processing includes a wide range of operations from the very simple to the extremely abstract and complex, the appropriate technique to apply to the raw data is determined based on the nature of the image. A flow chart of the pre-processing steps used in this thesis is shown in Figure 3-1.

Figure 3-1: Flow chart of pre-processing steps

3.2.1 Conversion from Original Image to Grayscale Image

A grayscale digital image is an image in which the value of each pixel is a single sample, meaning that each pixel carries the intensity information only. Also known as black-and-white images, grayscale digital images are composed of shades of gray, varying from black at the weakest intensity to white at the strongest. Distinct from one-bit black-and-white images (also known as binary images), the grayscale images have many shades of gray in between.

Color images are often built of several stacked color channels; each of these color

channels represents values of its own. For example, RGB images are composed of three independent channels that are red, green and blue primary color components.

A common strategy for the conversion is matching the luminance of the grayscale image to the luminance of the color image. For example, in order to perform the conversion from RGB color to grayscale representation of its luminance, the values of the red, green and blue (RGB) primaries in linear intensity encoding must be obtained by Gamma expansion. Figure 3-2 illustrates that 30% of the red value, 59% of the green value and 11% of the blue value in the RGB image are added together for the conversion, as shown below:

$$Gray=(R*30+G*59+B*11)/100 \quad (3.3)$$

Figure 3-2: Conversion from RGB image to grayscale image

The reverse conversion is also possible; the process can be done by simply setting all the primary color components to gray value, and correcting to the different gamma

if necessary.

Figure 3-3 (a) is an original SAR image, Figure 3-3 (b) shows the result after conversion from a color image to a grayscale image.

(a)

(b)

Figure 3-3: An example of conversion from color image to grayscale image
(a) The original SAR image (b) Grayscale image after conversion

3.2.2 Histogram Equalization

A histogram equalization algorithm determines the ideal number of times each frequency should appear in the image and, theoretically, re-plots the histogram appropriately. Histogram equalization is usually applied to increase the global contrast of many images, especially when the usable data of the image is represented by close intensity values. Through this adjustment, the intensities can be better distributed on the histogram. The process allows areas of lower local contrast to gain a higher contrast, and it is accomplished by effectively spreading out the most frequent intensity values.

Histogram equalization is useful in images with backgrounds and foregrounds that are both bright or both dark. Specifically, this method can bring better views of bone structures in X-ray images and better details in under-exposed photographs. One of the key advantages of this method is that it is a straightforward technique and an invertible operator, meaning that the original histogram can be recovered if the histogram function is known. The disadvantage of this method is that it may increase the contrast of background noise, while decreasing the usable signal. In addition, unrealistic effects are usually produced after histogram equalization. However, this method is still very useful for scientific images like thermal, satellite or X-ray images.

3.2.2.1 Implementation of the Histogram Equalization

Given a discrete grayscale image x , let n_i denote the number of occurrences of gray level i . The probability of occurrences of a pixel of level i in the image is defined

as:

$$p_x(i) = p(x=i) = \frac{n_i}{n}, \quad 0 \leq i < L \quad (3.4)$$

In equation (3.4), L is the total number of gray levels in the image, n is the total number of pixels in the image, and $p_x(i)$ represents the histogram for pixel value i .

Then the cumulative distribution function (CDF) is defined as follows:

$$cdf_x(x) = \sum_{j=0}^i p_x(j) \quad (3.5)$$

A transform in the form of $y=T(x)$ is created to produce a new image, and the CDF of the new image will be linearized across the value range,

$$cdf_y(i) = iK \quad (3.6)$$

$$y = T(x) = cdf_x(x) \quad (3.7)$$

Notice that function T maps the levels into the range $[0,1]$. The following transform (3.8) is needed for mapping the values back into their original range.

$$y' = y \cdot (\max\{x\} - \min\{x\}) + \min\{x\} \quad (3.8)$$

3.2.2.2 Example of Histogram Equalization

For the SAR image given in Figure 3-3(a), the original histogram is shown in Figure 3-4(a), and the histogram after the equalization is shown in Figure 3-4(b). It can be seen that the intensity values are better distributed after the equalization. The corresponding result of histogram-equalization is shown in Figure 3-5.

(a)

(b)

Figure 3-4: Histogram equalization of Figure 3-3(a)

(a)The original histogram (b) The histogram after equalization

Figure 3-5: The histogram equalization result of Figure3-3(a)

3.2.3 Median Filtering

Noise reduction is often desirable in image processing. Such noise reduction is a typical pre-processing step to improve the results of the later processing.

3.2.3.1 Overview of Median Filtering

As a very widely used filtering method in digital image processing, the median filtering can preserve the edges while removing the noise under certain conditions. The main idea of the median filter is to run through the image entry by entry, replacing each entry with the median of the neighboring entries. The pattern of neighbors is called the ‘window’. Usually, a window that has an odd number of entries is used, thus, the median is the middle value after all the entries in the window are sorted numerically. Median filtering is very effective, although it sometimes tends to be a slow operation.

Figure 3-6 illustrates an example of median filtering of a 3×3 window. When the median filter is applied to the values in Figure 3-6(a), these values are sorted in an ascending order, which is 2, 2, 3, 4, 5, 6, 9, 13 and 89. Obviously, the median value is 5. Therefore, the previous center value 89 is replaced by 5, as shown in Figure 3-6(b).

9	3	2	--	--	--
4	89	5	--	5	--
13	2	6	--	--	--

(a)
(b)

Figure 3-6: Example of median filtering with an odd number of entries

3.2.3.2 Example of Median Filtering in SAR Images

Given the SAR image Figure 3-3(a), a median filter of 3×3 is applied, and the result is shown in Figure 3-7(a). Figure 3-7(b) is the result of applying a median filter of 5×5 .

Figure 3-7: Results of applying the median filtering

- (a) The result of 3×3 window size median filter
- (b) The result of 5×5 window size median filter

3.3 Summary of the Pre-Processing

As only the image with radiometric distortion is considered in this thesis, the proposed pre-processing algorithm consists of three steps.

In the first stage, the original image is converted to a gray-scale image. The second stage is to obtain the intensity histogram of the gray-scale image. Because the histogram represents the distribution of the values of each pixel, histogram equalization is applied in this stage to adjust the global contrast of the image. Thus, a higher local contrast is obtained. In the third stage, median filtering is used to remove the noise while preserving the edges [9].

However, the proposed pre-processing algorithm is intended for dealing with SAR images, some other types of images may require other pre-processing techniques.

Chapter 4

Detection of Linear Features Based on Beamlet Transform

4.1 Overview of Beamlet Analysis

The Beamlet analysis was first introduced by Donoho and Huo as a tool for multiscale image analysis [6], [7]. The concept of beamlet analysis deals with the approximation of the linear objects by line segments, as shown in Figure 4-1.

Figure 4-1: Approximating a line segment by beamlets

Beamlet is a framework for multiscale image analysis in which line segments

play a role analogous to the role played by points in wavelet analysis. The framework has five key components:

(1) Beamlet dictionary is a dyadically organized collection of line segments, occupying a range of dyadic locations and scales, and occurring at a range of orientations.

(2) Beamlet transform of an image $f(x,y)$ is the collection of integrals of f over each segment in a beamlet dictionary.

(3) Beamlet pyramid stores the resulting information.

(4) The beamlet graph is the graph structure with pixel corners as vertices and beamlets as edges; a path through this graph corresponds to a polygon in the original image.

(5) Beamlet-based algorithms are formulated by exploiting those components to identify and extract beamlets and chains of beamlets with special properties.

The beamlet dictionary, beamlet transform, beamlet pyramid and beamlet graph build up a four-level hierarchy of beamlet algorithms. The first level consists of simple procedures which ignore the structure of the beamlet pyramid and beamlet graph; the second level exploits only the parent-child dependence between scales; the third level incorporates co-linearity and co-curvity relationships; and the fourth level allows global optimization over the full space of polygons in an image.

4.2 Introduction of Beamlet Transform

Given an image of $n \times n$ pixels, the image domain can be viewed as the continuum square $[0,1]^2$, and the pixels as an array of $1/n$ -by $1/n$ squares arranged in a grid in $[0,1]^2$. The following definitions are helpful for better understanding the beamlet transform.

Definition 1: Consider two pixels v_1, v_2 within an image of $n \times n$ pixels. The line segment $b = \overline{v_1 v_2}$ is called a beam. There are $O(n^4)$ such beams in $[0,1]^2$.

Definition 2: Partition the image into windows at different scales $2^m \times 2^m$, where $0 < m < \log_2 n$. All the beams that connect the vertices on boundaries of these windows are called beamlets.

Although the family of beamlets is built from $O(n^2)$ vertices, it contains many fewer than $O(n^4)$ beams; the collection of beamlets has a cardinality only logarithmically larger than $O(n^2)$. It follows that the speed of exhaustive searches through the collection of beamlets is much faster than the speed of exhaustive searches through beams. Despite reduced cardinality, the dictionary of beamlets is expressive. It is proved in [29] that a relatively small number of beamlets can be used as a substitute for any single beam, as shown in Figure 4-1. Figure 4-2 presents the beamlets at different scales.

Figure 4-2: Beamlets at different scales

Let $f(x_1, x_2)$ be a continuous function on $[0, 1]^2$, the beamlet transform of f is the collection of all line integrals,

$$T_f(b) = \int_b f(x(l)) dl, b \in B_E, \quad (4.1)$$

where B_E is the collection of beamlets, and $x(l)$ traces the beamlet b along a unit length path.

The digital beamlet transform of an $n \times n$ array f_{i_1, i_2} is understood to be the beamlet transform of function f defined on the continuum by interpolation of the values of f_{i_1, i_2} .

$$f(x_1, x_2) = \sum_{i_1, i_2} f_{i_1, i_2} \phi_{i_1, i_2}(x_1, x_2) \quad (4.2)$$

Where ϕ_{i_1, i_2} is a specified family of continuous interpolation functions. There are several ways to select the interpolation function. However, Average-Interpolating

Function is used in this thesis. Let $P(i_1, i_2)$ represent the sub-square $[i_1/n, (i_1+1)/n) \times [i_2/n, (i_2+1)/n)$. The interpolation function ϕ_{i_1, i_2} should be chosen to satisfy the following condition,

$$n^2 \int_{P(i_1, i_2)} \phi_{i_1, i_2}(x_1, x_2) dx_1 dx_2 = \delta_{i_1, i_1'} \cdot \delta_{i_2, i_2'}, \quad (4.3)$$

where $\delta_{i, i'}$ is the Kronecker symbol, and f is defined to be,

$$f_{i_1, i_2} = Ave\{f \mid P(i_1, i_2)\}. \quad (4.4)$$

Where $P(i_1, i_2)$ is the sub-square as explained above. In other words, the value of discrete beamlet transform f_{i_1, i_2} can be viewed as the sum of the pixel-level average of f , as shown in Figure 4-3.

Figure 4-3: Discrete beamlet transform can be viewed as a weighted sum of pixel values

4.3 Linear Feature Extraction with Beamlet Transform in Noisy Images

It is stated in Chapter 1 that the further investigation of the SAR image is usually

impeded due to the noise produced from the imaging system. However, beamlet transform provides a solution to the images with noise.

Suppose we have a noisy image containing a line segment of unknown length, orientation and position. The problem of detecting the presence of linear features in a noisy picture (y_{i_1, i_2}) can be modeled as:

$$y_{i_1, i_2} = A \cdot \overline{\Phi}_{i_1, i_2} + \varepsilon \cdot z_{i_1, i_2}, \quad 0 \leq i_1, i_2 < n, \quad (4.5)$$

Where ε is the noise level, z_{i_1, i_2} is the Gaussian white noise, A is an unknown amplitude parameter and $\overline{\Phi}_{i_1, i_2} = \overline{\Phi}(i_1, i_2; \overline{v_0 v_1})$ is the observed effect at the sensor array of an unknown beam $\overline{v_0 v_1}$. Now the problem becomes testing the simple null hypothesis

$$H_0 : A = 0 \quad (4.6)$$

Against the composite alternative

$$H_1 : A > 0, \quad v_0, v_1 \in [0, 1]^2. \quad (4.7)$$

Because of the wide range of possible endpoint pairs, we consider the random field suggested by the standard ideas in hypothesis testing

$$Y[v_0, v_1] = \langle \psi_{\overline{v_0 v_1}}, y \rangle, \quad (4.8)$$

Where $\psi_{\overline{v_0 v_1}}$ is the so-called filter matched to $H_{1, \overline{v_0 v_1}}$:

$$\psi_{\overline{v_0 v_1}}(i_1, i_2) = \overline{\Phi}(i_1, i_2; \overline{v_0 v_1}) / \|\overline{\Phi}(\cdot, \cdot; \overline{v_0 v_1})\|_2 \quad (4.9)$$

Consider the maximum beam statistic

$$Y^* = \max \{Y[v_0, v_1] : v_0, v_1 \in [0, 1]^2\}, \quad (4.10)$$

Y^* takes over all the beams $\overline{v_0 v_1}$ with endpoints in $[0, 1]^2$. In the Generalized Likelihood Ratio Test (GLRT), H_0 is rejected if Y^* exceeds a certain threshold. Then all the choices of $H_{1, \overline{v_0 v_1}}$ are optimized to test the composite alternative.

However, there are at least $O(n^4)$ beams to consider, and $O(n^4)$ evaluation costs $O(n)$ flops each, which is $O(n^5)$ in total. Obviously, this is very time-consuming and expensive for large n .

Because beamlets can be used as a substitute for any single beam, a test based on the maximum beamlet statistic which is over all beamlets is considered rather than over all beams. The maximum beamlet statistic is defined below:

$$Y^+ = \max\{T_y(b)/\sqrt{L(b)} : b \in B_{1/n,n}\}. \quad (4.11)$$

Where $T_y(b)$ is the beamlet transform of data y and $L(b)$ is the Euclidean length of the beamlet b . Thus, the problem of detecting the existence of linear features can be simplified as: whenever (4.11) exceeds a certain threshold, H_0 is rejected.

A conclusion has been drawn in [30] that by using beamlets, we can have a low complexity algorithm yielding a good approximation to the GLRT. Figure 4-4 illustrates the nearly identical sensitivity of the maximum beam statistic and the maximum beamlet statistic [30]. The test image of 128×128 contains a very faint line segment embedded in a noisy array. We can see that when the pixel signal-to-noise ratio (SNR) varies from -30 to -1, no pixel-level analysis will detect this feature. Figure 4-4 also gives median values of the maximum score of the classical GLRT based on the collection of line segments, a threshold for the GLRT, and a median maximum standardized beamlet coefficient from the beamlet transform. Both methods start to work when the SNR is around -24. In contrast, the pixel-based detectors fail to identify the underlying object until the SNR is around -10. It is evident that the largest standardized beamlet coefficient is almost as large as the GLRT, yet the largest

beamlet coefficient can be computed approximately using an algorithm of order $O(n^2 \log_2(n))$ as compared to $O(n^5)$ of the GLRT.

Figure 4-4: Nearly identical sensitivity of the maximum beam statistic and the maximum beamlet statistic [30]

4.4 Implementation of Linear Feature Extraction Based on Beamlet Transform

It is explained in the previous section that there are $O(n^2 \log_2(n))$ beamlets if multi-scaling is involved. However, only single scaling is considered in this thesis, in other words, there are $O(n^2)$ beamlets.

The algorithm to extract the linear features is described in the following sub-sections.

4.4.1 Image Partitioning

The first stage is partitioning the image into dyadic squares, marking the boundaries of the squares with equi-spaced vertices at pixel spacing. Although the large-sized window is robust to noise, it cannot provide detailed information. Besides, the longer beamlets of the larger window increase the computational time. In this thesis, the size of the input image is 256×256 , and the size of the partitioned window is 16×16 .

4.4.2 Build Beamlet Dictionary

The beamlet dictionary is built by connecting pairs of vertices within a dyadic square. The Beamlet dictionary is a dyadically-organized library of line segments at a range of locations, orientations, and scales, which gives a multi-scale approximation to the collection of all line segments. Since the single scale beamlet transform is considered in this thesis, the beamlet dictionary needs to be calculated only once and applied to all sub-squares. And the following information for each beamlet is recorded:

- (1) The coordinates of the pixels that are considered to be on the beamlet.
- (2) The length of the beamlet.

4.4.3 Beamlet Transform

In this stage, the discrete beamlet transform is applied to the beamlets obtained from the last stage, and the beamlet transform coefficients are normalized as:

$$T'_f(b) = T_f(b) / \sqrt{L(b)} \quad (4.12)$$

Where $T_f(b)$ is the beamlet transform coefficient and $L(b)$ is the length of the beamlet. In each sub-square, the beamlet with the maximum $T'_f(b)$ value is saved.

4.4.4. Threshold

The threshold value T is defined as:

$$T = [\max(T'_f(b)) - \min(T'_f(b))] / 2 \quad (4.13)$$

Then the normalized beamlet transform coefficient in each dyadic square is compared with the threshold value. If $T'_f(b)$ is greater than T , a line segment is considered to exist in the sub-square; if $T'_f(b)$ is less than the threshold value, the beamlet is discarded. In this thesis, threshold value $T=1.0$.

4.4.5 Gradient Test

In this stage, the gradient of each pixel on the beamlet which is saved from the threshold is tested. If the gradient of the pixel is not consistent with the beamlet, the pixel is discarded.

The gradient operator calculates the gradient of the image intensity at each point, since the gradient is a vector whose components measure how rapidly pixel values are changing with distance in the x and y directions, the components of the gradient can

be found using the following approximations:

$$\frac{\partial f(x, y)}{\partial x} = \Delta_x = \frac{f(x + d_x, y) - f(x, y)}{d_x} \quad (4.14)$$

$$\frac{\partial f(x, y)}{\partial y} = \Delta_y = \frac{f(x, y + d_y) - f(x, y)}{d_y} \quad (4.15)$$

Where d_x and d_y measure distance along the x and y directions respectively.

The gradient magnitude measure and direction are given by:

$$M = \sqrt{\Delta_x^2 + \Delta_y^2} \quad (4.16)$$

$$\theta = \arctan\left[\frac{\Delta_y}{\Delta_x}\right] \quad (4.17)$$

Mathematically, the gradient operators use kernels which are convolved with the original image to calculate approximations of the derivatives: one for horizontal changes, and one for vertical. In this thesis, the kernel of odd size is selected, because the operators with kernel of odd size are centered and can therefore provide an estimate that is biased towards the center pixel. Table 4.1 lists the kernels of two gradient operators.

Table 4.1: Kernels of two gradient operators

Gradient Operator	Horizontal Kernel	Vertical Kernel
Sobel	$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
Prewitt	$\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 2 & 1 \end{bmatrix}$	$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$

4.4.6 Summary of the Algorithm Based on Beamlet Transform for Linear Feature Extraction

The process of extracting linear feature using beamlet transform can be summarized in the diagram shown in Figure 4-5.

Figure 4-5: Flow chart of linear feature extraction using beamlet transform

4.5 Example of Applying the Algorithm Based on Beamlet Transform for Linear Feature Extraction

Figure 4-6 presents an example of applying the method described in Section 4.4 to Figure 3-7 (a). The linear features of Figure 3-7 (a) are successfully extracted, but some discontinuities still exist. Chapter 5 will discuss the post-processing algorithm which deals with those discontinuities.

Figure 4-6: Result of applying beamlet based method to Figure 3-7(a)

Chapter 5

Post-Processing Techniques

Generally, the linear features extracted from beamlet transform possess discontinuities, as can be seen from Figure 4-6. In order to further enhance the extracted linear features, a post-processing algorithm is proposed to link the discontinuities.

5.1 Description of Post-Processing Algorithm

5.1.1 Initialization of Data Structure

To link the discontinuities, a data structure as listed in Table 5.1 is generated. Each sub-square is initially marked with status ‘empty’ or ‘unchecked’. The sub-squares without a beamlet are marked with status ‘empty’.

Table 5.1: Data structure for post-processing

	Description	Initialize	Value
Status	Record the status of each sub-square	‘unchecked’ or ‘empty’	‘unchecked’, ‘checked’ or ‘empty’
Orientation	Record the orientation of each beamlet	0	The orientation of each beamlet

5.1.2 Calculation of Beamlet Orientation

This stage begins with finding the first sub-square with ‘unchecked’ status and calculating the orientation of the beamlet with its distance changes in the x and y directions respectively. The value of the orientation is obtained by the following equation:

$$\alpha = \arctan\left[\frac{\Delta_y}{\Delta_x}\right]. \quad (5.1)$$

The status of the corresponding sub-square is then modified to ‘checked’, and the value of its orientation is saved in the data structure. The calculation proceeds to all eight neighboring sub-squares with ‘unchecked’ status.

5.1.3 Calculation of Orientation Difference Between Beamlets

Having the orientation values of a sub-square and its eight neighboring sub-squares, the value of the orientation difference θ between two neighboring beamlets can be obtained by applying the following equations:

$$\Delta \sin \theta = \sin \alpha \cos \beta - \cos \alpha \sin \beta \quad (5.2)$$

$$\Delta \cos \theta = \cos \alpha \cos \beta + \sin \alpha \sin \beta \quad (5.3)$$

$$\Delta \theta = \left| \arctan\left(\frac{\Delta \sin \theta}{\Delta \cos \theta}\right) \right| \quad (5.4)$$

Where α, β are the orientation values of beamlets derived from the last stage.

5.1.4 Connectivity Analysis using Angle and Distance Measures

If the orientation difference between the two neighboring beamlets is less than the pre-set angle tolerance, then the point proximity is tested to determine linking the discontinuity that exists between the two beamlets. The procedure to compute the point proximity is explained as follows:

(1) Let B_{1P_1}, B_{1P_2} denote the beginning point and the end point of beamlet $\overrightarrow{B_1}$, B_{2P_1}, B_{2P_2} denote the beginning and the end point of beamlet $\overrightarrow{B_2}$. The vector dot product D of $\overrightarrow{B_1}$ and $\overrightarrow{B_2}$ is then calculated. The results are used for further analysis as shown in Table 5.2.

Table 5.2: Results of vector dot product

$D \geq 0$	Beamlets $\overrightarrow{B_1}$ and $\overrightarrow{B_2}$ are in the ‘same’ direction, continue to next step.
$D < 0$	Beamlets $\overrightarrow{B_1}$ and $\overrightarrow{B_2}$ are not in the ‘same’ direction, reverse the order of start point and end point of $\overrightarrow{B_2}$ so that both beamlets are in the ‘same’ direction.

(2) The point proximity N can be derived from the following equation:

$$N = \| B_{1P_2} - B_{2P_1} \|. \quad (5.5)$$

In this thesis, the pre-set angle tolerance is chosen to be $(\pi/180)$ rad, and the maximum distance tolerance is two pixels. The process continues until all the eight-neighborhood sub-squares have been considered.

The post-processing algorithm proceeds to all sub-squares until all the sub-squares are marked with status ‘checked’ or ‘empty’.

5.2 Summary of the Post-Processing Algorithm

The post-processing algorithm is performed by calculating the orientation difference and point proximity. The neighboring beamlets are linked if they meet two conditions: first, the orientation difference is less than the angle tolerance; second, the ends of the neighboring beamlets are within the maximum distance constraint of each other. The flow chart of the algorithm is summarized in Figure 5-1.

Figure 5-1: Flow chart of post- processing techniques

5.3 Example of Applying the Post-Processing Algorithm

Figure 5-2 presents the result of applying the post-processing algorithm to Figure 4-6. The discontinuities in Figure 4-6 are linked when they meet the requirements of orientation and distance tolerance.

Figure 5-2: Result of applying the post-processing algorithm to Figure 4-6

Chapter 6

Experimental Results and Analysis

In this chapter, several SAR images that contain linear features are processed with the proposed method, and the resulting images are shown. These images are processed by pre-processing, beamlet transform and post-processing.

In the following, different SAR images of road networks and the corresponding results from the proposed method are shown. The performance and analysis of the experimental results are discussed. In each experiment, the original image, the image in each stage of pre-processing, the image after beamlet transform and thresholding, and the image after post-processing are presented. For the purpose of comparison, Canny's edge detection technique [12] is implemented to extract the linear features, and the results are shown as well.

6.1 Experimental Results

6.1.1 Experiment 1

(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 6-1: Results of Experimental 1

(a) Original image (b) Gray-scale image (c) Image after histogram equalization (d) Image after median filtering (e) Image after beamlet transform (f) Image after post-processing (g) Result of applying Canny's edge detector

6.1.2 Experimental 2

The original SAR image used in Experiment 2 contains the straight lines only, as shown in Figure 6-1 (a.)

(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 6-2: Results of Experimental 2

(a) Original image (b) Gray-scale image (c) Image after histogram equalization (d) Image after median filtering (e) Image after beamlet transform (f) Image after post-processing (g) Result of applying Canny's edge detector

6.1.3 Experimental 3

The original SAR image of Experiment 3 contains the curve features as well as the linear features, as shown in Figure 6-3 (a).

(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 6-3: Results of Experimental 3

(a) Original image (b) Gray-scale image (c) Image after histogram equalization (d) Image after median filtering (e) Image after beamlet transform (f) Image after post-processing (g) Result of applying Canny's edge detector

6.2 Analysis of Experimental Results

The experimental results in Section 6.1 have demonstrated that the beamlet transform is able to identify the linear features effectively from the noisy SAR images

with low signal-to-noise ratio (SNR). The discontinuities that satisfy the constraints in Chapter 5 are linked. In Figure 6-4, an example is given to highlight the effect of post-processing algorithm, the linked discontinuities of Figure 6-1(f) are marked with the rectangles.

Figure 6-4: An example of Figure 6-1(f) with linked area marked

It is evident that the method based on beamlet transform performs much better than the Canny's edge detection scheme when processing the noisy SAR images. In Figure 6-1 (g), Figure 6-2(g) and Figure 6-3(g), we can see that the Canny's edge detection is unable to separate the linear features from the noisy environment. In addition, other information detected by Canny's edge detector also interferes with the linear features.

Chapter 7

Conclusion and Future Work

This thesis presents a new beamlet transform based technique to extract the linear features from satellite images. Beamlet transform can be viewed as an extension of the wavelet transform because of its multi-resolution framework. Due to the property that the line segments in the beamlet transform play a role analogous to the role played by points in wavelet analysis, the beamlet transform appears to be very effective in extraction of linear feature in noisy environment. The method proposed in this thesis includes three main steps: the first step is pre-processing which reduces the noise by histogram equalization; the next step is beamlet transform to extract the linear features followed by post-processing procedures to eliminate the discontinuities.

Experimental results in Chapter 6 show that beamlet transform provides an effective and robust approach for linear feature extraction in the SAR images with noise and low SNR, compared with the traditional edge detection methods.

The beamlet transform algorithm stated in this thesis could be enhanced by reducing its complexity and speeding up the computational process, and the algorithm can also be improved by introducing the multi-scale pyramid analysis that can

represent the linear features more efficiently. Furthermore, the performance of the proposed method will be affected by the choice of pre-processing and post-processing algorithms, which could be further investigated in the future.

References

- [1] C. Oliver and S. Quegan, “Understanding Synthetic Aperture Radar Images,” Norwood, MA: Artech House, 1998.
- [2] B.Wessel, “Road Network Extraction from SAR Imagery Supported by Context Information,” Proceedings of ISPRS congress “Geoinformation Bridging Continents”, Istanbul, International Archives of Photogrammetry, Remote Sensing and Spatial Information Science, vol.35, pp.360-365, 2004.
- [3] Eshan Nadernejad, Sara Sharifzadeh and Hamid Hassanpour, “Edge Detection Techniques: Evaluations and Comparisons,” Applied Mathematical Sciences, vol.2, no.31, pp.1507-1520, 2008.
- [4] R. Owens, “Lecture 6,” Computer Vision IT412, [http:// homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/ LECT6 /node2.html](http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT6/node2.html), Oct. 1997.
- [5] Min Dai, Cheng Peng, Andrew K.Chan and Dmitri Loguinov, “Bayesian Wavelet Shrinkage With Edge Detection for SAR Image Despeckling,” IEEE Transactions on Geoscience and Remote Sensing, vol.42, no.8, pp.1642-1648, Aug. 2004.
- [6] D.L.Donoho and X. Huo, “Beamlets and Multiscale Image Analysis,” Multiscale and Multiresolution Methods, Springer Lecture Notes in Computational Science and Engineering, vol.20, pp.149-196, 2002.

- [7] D.L Donoho and X.Huo, "Beamlet Pyramids: A New Form of Multiresolution Analysis, suited for Extracting Lines, Curves and Objects from Very Noisy Image Data," Proceedings of SPIE 2000, vol.4119, pp.434-444, Jul. 2000.
- [8] X.Huo, "Beamlets," WIREs Computational Statistics, vol.2, pp.116-119, Feb. 2010.
- [9] R. Gonzalez and R.Woods, Digital Image Processing. Prentice Hall, 3rd edition, Jan. 2008.
- [10] Raman Maini and Dr.Himanshu Aggarwal, "Study and Comparison of Various Image Edge Detection Techniques," International Journal of Image Processing, vol.3, pp.1-11, 2009.
- [11] D.Marr and E.Hildreth, "Theory of Edge Detection," Proceedings of the Royal Society of London, Series B, Biological Sciences, vol.207, no.1167, pp.187-217, Feb. 1980.
- [12] John Canny, "A computational approach to edge detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.8, no.6, pp 679-698, Nov. 1986.
- [13] Richard O. Duda and Peter E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," Communications of Association for Computing Machinery, vol.15, no.1, pp.11-15, Jan. 1972.
- [14] J.Illingworth and J.Kittler, "The Adaptive Hough Transform," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.9, no.5, pp.690-698, 1987.
- [15] Jiqiang Song, Min Cai, Michael R.Lyu and Shijie Cai, "A New Approach for

- Line Recognition in Large-size Images Using Hough Transform,” Proceedings of 16th International Conference on Pattern Recognition, vol.1, pp.33-36, 2002.
- [16] J. Illingworth and J. Kittler, “A Survey of the Hough Transform,” Computer Vision Graphics and Image Processing, vol.44, pp.87-116, 1988.
- [17] Lesley M. Murphy, “Linear Features Detection and Enhancement in Noisy Images via the Radon Transform,” Pattern Recognition Letters, vol.4, no.4, pp.279-284, Sept.1986.
- [18] J. Randon, “On the Determination of Functions from Their Integral Values along Certain Manifolds,” IEEE Transactions on Medical Imaging, vol.5, no.4, pp.170-176, Dec.1986.
- [19] Ronald Bracewell, “The Fourier Transform and Its Applications,” McGraw-Hill, second edition revised, 1986.
- [20] Stephane Mallat and Sifen Zhong, “Characterization of signals from multiscale edges,” IEEE Transactions on Pattern Analysis and Machine Intelligence., vol.14, no.7, pp. 710-732, 1992.
- [21] Stephane Mallat and W. L. Hwang, “Singularity detection and processing with wavelets,” IEEE Transactions on Information Theory, vol.38, no.2, pp. 617-643, 1992.
- [22] Stephane Mallat, “Multi-frequency Channel Decomposition of Images and Wavelet Models,” IEEE Transactions on Acoustic, Speech and Signal Processing, vol.37, no.12, pp.2091-2110, 1989.
- [23] Michael Kass, Andrew Witkin and Demetri Terzopoulos, “Snake: Active

- Contour Models,” International Journal of Computer Vision, pp.321-331, 1988.
- [24] M.R.Della Rocca, M.Fiani, A.Fortunato and P.Pistillo, “Active Contour Model to Detect Linear Features in Satellite Images,” International Archives of Photogrammetry Remote Sensing and Spatial Information Science, vol.35, pp.446-450, 2004.
- [25] Nikolas P.Tiilikainen, “A Comparative Study of Active Contour Snakes,” Copenhagen University, Danmark, 2007.
- [26] Chenyang Xu and Jerry L.Prince, “Snakes, Shapes and Gradient Vector Flow,” IEEE Transactions on Image Processing, vol.7, no.3, pp.359-369, 1998.
- [27] J. Serra, “Image Analysis and Mathematical Morphology,” Academic Press, New York, 1982.
- [28] “GIS Image Processing and Analysis,” <http://www.gisdevelopment.net/tutorials/tuman005pf.htm>.
- [29] D.L. Donoho, “Wedgelets: Nearly minimax estimation of edges,” Annals of Statistics, vol.27, no.3, pp.859–897, 1999.
- [30] D.L.Donoho and X.Huo, “Applications of Beamlets to Detection and Extraction of Lines, Curves, and Objects in Very Noisy Images,” Proceedings of Nonlinear Signal and Image Processing, Baltimore, MD, Jun. 2001.