TEXTURE CLASSIFICATION OF SAR SEA ICE USING THE WAVELET TRANSFORM

QIYAO YU

Texture Classification of SAR Sea Ice Using the Wavelet Transform

By

Qiyao Yu

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 Sciences

Memorial University of Newfoundland

December 2001

Abstract

Sea ice types and concentrations are of great importance for ship navigation in or near the ice. The evaluation of ice types and properties using synthetic aperture radar (SAR) imagery has attracted much attention in recent years. SAR sea ice images usually have consistent textures that can be utilized for sea ice description and classification. Therefore, methods based on texture discrimination could be designed to identify ice types and evaluate ice properties by machine without human intervention.

This thesis contributes to the ice identification problem mainly by investigating the feature extraction phase in a texture classification process. A review is given of several different approaches including Gray Level Co-occurrence Matrices and Gabor filtering, while the emphasis is on those based on the wavelet transform techniques. Comparative studies have been conducted on both the selection of wavelet band signatures and of wavelet kernels.

A new wavelet band signature, named *wavelet entropy*, is proposed and applied to texture classification with encouraging results. This technique extracts features from wavelet band histograms. A promising aspect of this new technique is that it provides estimates of probability measures of the texture

memberships. These membership probabilities have been used in a ship navigation application with interesting results presented in the thesis.

Texture orientation issues are also addressed in this thesis. Because of the oriented structures apparent in some SAR sea ice textures, it is desirable to extract rotation invariant features. Some new work is presented that has achieved this goal to some degree by DFT encoding on the features of different orientations, obtained via the complex wavelet transform instead of the traditional discrete wavelet transform to separate the mixed diagonal directions.

Acknowledgement

I would like to express my sincere gratitude to my supervisors, Dr. Cecilia Moloney and Dr. F. M. Williams for their involvement, patience and insight. Without their invaluable help and support, this work would not have been possible. I would also like to thank Dr. D. A. Clausi of the Univerisity of Waterloo for providing me his thesis, and Dr. Nick Kingsbury of the Cambridge University for the Matlab code of DTCWT.

I am grateful to the School of Graduate Studies, the Faculty of Engineering, the Associate Dean of Engineering and his office for granting me admission to the graduate program and providing the necessary financial assistance. The Radarsat SAR image is used with the courtesy of Canadian Ice Service, Environment Canada and is copyright Canadian Space Agency 1997.

I also thank all my fellow graduate students for the good time and the help. And finally, I would like to thank my parents and my fiancée for their care, encouragement and moral support during the two years.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 General

Ship navigation in cold ocean regions is often greatly influenced by the presence of sea ice and the properties of any such ice. The evaluation of ice types and their properties is a very challenging task since onsite investigation is often impractical or expensive, and the timing for gathering and interpreting ice information is often critical. During the last decade, much attention has been given to utilizing Synthetic Aperture Radar (SAR) to do this job. Being an active microwave imaging system, SAR is an efficient tool for sea ice monitoring for the following three reasons:

- As a satellite or aircraft mounted system, SAR can provide regular imaging of ice fields over extended areas.
- SAR use active microwave sensors, which have the ability to obtain measurements anytime regardless of time or season, and which can penetrate through cloud cover under most weather conditions.

- The portion of microwave energy returned is largely dependent on the target surface roughness, moisture content and electrical properties. Those properties are significantly different among different sea ice types.

Hence the problem arises of how to interprete SAR sea ice image accurately. It is most desired to obtain quantitive relationships between the physical properties of ice and image pixel value patterns. However this seems almost impossible because there are too many environmental factors and noise. Classifying the ice types in the image and then associating them with the corresponding physical properties is a feasible solution. A human operator can clearly and easily distinguish different ice types and land visually from a SAR image, based not only on the relative texture appearance of the different ice types, but on tonal and structral differences as well [1]. However, automatic classification of SAR sea ice images by machine is difficult and challenging for several reasons. There are often significant intensity variations among SAR images and even across a single image, making analysis techniques based on intensity values unreliable. More stable are those based on texture discrimination. However, no robust recognition method is available yet for textures because of their random nature. To make it worse, the speckle noise usual in SAR images will mask or mix with those textures. And the last, some sea ice types are more or less like each other and often form mixture areas that have no distinct texture appearance in the corresponding SAR image.

All these add to the difficulties in the design of a successful SAR sea ice image classification system. As mentioned earlier, textures give a more consistent description of the ice than does intensity, so it is better to base the classification system on the capability of identifying the textures of interest.

The design of a texture classification system typically consists of four steps listed in Table 1.1: Feature extraction, classifier design, classifier training and performance evaluation.

Step	Function
1.Feature extraction	Select texture properties that best destinguish the texture types and decide how to extract and measure these properties.
2.Classifier design	Design the classification algorithm to be used to get the classifier parameters (i.e. how to separate the feature space)
3.Classifier training	Determine the classifier parameters (decision boundaries, etc.) for each candidate type.
4.Performance evaluation	Estimate the classification accuracy.

Table 1.1: Classification System Design

The feature extraction is perhaps the most important part. It produces a set of suitable features that represent the information needed for subsequent training or classification. In the training stage, those features belonging to the same class are grouped and will be used as the reference for future classification. Then in the classification stage, the same kind of features are extracted and compared with the references obtained in the previous training stage. The performance of the classification system therefore depends largely on how well those features are extracted. Many different texture feature extraction approaches have been proposed in the literature, including Gray Level Co-occurrence Matrix (GLCM), Gabor filtering, Markov Random Model, and Wavelet filtering. This

thesis places an emphasis on wavelet filtering approaches to texture feature extractions.

Besides those difficulties already mentioned above in designing a SAR sea ice texture classification system, orientation is another factor which needs to be considered very carefully. Although many types of sea ice appear to be isotropic, some ice types, such as rubble ice, have obvious oriented structures. If features corresponding to those orientational structures are rotation variant, they will cause significant degradation of the classification system when the training samples and the classification targets do not have the same orientation. Discarding the anisotropic features, however, is not a good choice since the discarded features can provide unique information for ice identification. A better solution is to have a rotation invariant representation of those anisotropic features.

Under circumstances where the ice types are not distinctly different or there is no clear boundary between ice types, fuzzy classification is more appropriate than traditional classification (i.e. crisp classification). The result should be not just an answer to "what ice type is it", but "how much does it look like a given ice type". Made up of a set of membership probabilities, those results are expected to be able to provide information on the composition percentage of different ice types and hence on the physical properties.

1.2 Thesis Objectives

The goals of this thesis are to study how to efficiently extract SAR sea ice texture features for classification; then given the classfication results to evaluate

the corresponding ice physical properties: and hence to develop a simple ship navigation application. Our particular interests are with the ice types that have not experienced winter growth. for example new ice and young ice. These ice types are in some sense alike. and are often mixed together. As mentioned earlier. fuzzy classfication may be more effective here. but first crisp methods of classification will be investigated because they are well defined and their performance are much easier to evaluate.

Since well-oriented features like leads and ridges often exist in those ices. another thesis objective is to find a solution to the orientation problem mentioned earlier. Efforts are directed to looking for a rotation invariant representation of the oriented features. with which the classification system will be less sensitive to the object orientations.

1.3 Data Set and Software Environment

Our research uses two data sets for the classification experiments. The first one consists of nine Brodatz textures (selected from the Brodatz's album [2]) and three SAR textures. as shown in Appendix A. The Brodatz's texture images are of dimension of 512 × 512 pixels. For feature extraction. each was split into sixty-four nonoverlapping 64 × 64 pixels regions. Half of them were used for training and half were used for testing. The SAR texture images are areas of perceptually uniform texture and of dimension 1024 × 1024 pixels extracted from the large Radarsat image of Figure 2.3. They are histogram equalized since the original SAR image is very dark with a limited effective dynamic range. Note that histogram equalization may enhance noise and destroy subtle

sea ice texture information unpredictably. However, we first aimed at distinguishing different ice textures to see if there are visual differences among these textures. Each of these SAR images was divided into sixty-four nonoverlapping 128×128 pixels subimages. Here a larger dimension was selected to capture the larger scale features in land and rubble ice textures. We name this composite data set **MBS** (Mixed Brodatz and SAR) for future reference.

Our second data set consists of four SAR textures of Pack ice, Rubble ice, Landfast ice, and Land. Unlike in MBS, no histogram equalization is performed but a logarithm step is taken, the reason for which will be presented later in Chapter 2. Therefore we call this data set **PLS** (Pure Logarithmed SAR). This set is shown in Appendix B (please note the values after the logarithm are very small and therefore these images have been scaled for better visualization).

Most of the Brodatz textures in MBS come from the USC-SIPI image Database [3] except the stone and canvas which are scanned from our Brodatz album.

Considering that classification and segmentation experiments are usually very time consuming, we use C++ to implement all the algorithms used for experiments in Chapter 4, 5, and 6. The programs are developed with Microsoft Visual C++ under the Windows system. They are listed in Appendix C.

1.4 Organization of Thesis

The organization of this work is as follows. Chapter 2 gives an introduction to the application environment, covering topics on SAR imaging system as well as the backscattering characteristics of sea ice. Chapter 3 continues the literature review with some popular texture feature extraction methods. Here methods based on Gray Level Cooccurrence Matrix (GLCM), Gabor filtering, and the Wavelet Transfrom are reviewed. Following that in Chapter 4, detailed studies are performed on feature extraction methods based on the wavelet transform. Some previous research in the literature is experimentally repeated, and different wavelet kernels and signatures are investigated and compared. Chapter 5 deals with texture orientation issues. A new feature extraction method that can partially solve the rotation invariant recognition problem is discussed. Chapter 6 proposes a new approach for fuzzy classification, which is used in a simple application of ship navigation. A summary and recommendations for future work comprise the final chapter (Chapter 7).

Chapter 2

Application Environment

2.1 Introduction

Modern remote sensing techniques make it possible to acquire information about the earth surface in real time without actually being in contact with it. Synthetic Aperture Radar (SAR), a remote sensing technology, is especially useful for our research objective of identifying sea ice types because of its active microwave sensors, which are characterized by their all-weather and all-time operating ability. SAR images also have high resolution relative to other non-optical remote sensing technologies. The resolutions could be up to $3m$ for aircraft mounted SAR and $10m$ for space craft mounted SAR. A brief introduction of the fundamental theory of SAR system is presented in the following section, which draws heavily on [4]. Then the next section deals with electrical properties and SAR signatures of sea ice. The last section of this chapter discusses the multiplicative speckle noise that can significantly influence the performance of a texture classification system.

2.2 Synthetic Aperture Radar (SAR)

Like other radar systems, a SAR transmits a sequence of electromagnetic pulses and records the returned waveforms. This make it advantageous over passive optical sensing because the microwaves used can penetrate cloud covers and data accquisition can be performed at night. However due to the much longer wavelength of microwaves (see Figure 2.1), it seems difficult to achieve resolution comparable to optical sensors with a microwave system, because the resolution is on the order of $\lambda R/D$, where λ is the wavelength of the illuminating source, R is the target range, and D is the diameter of the antenna aperture or lens [5]. This implies that a conventional radar would need an impossibly long antenna to achieve the same level of resolution as that of an optical system.

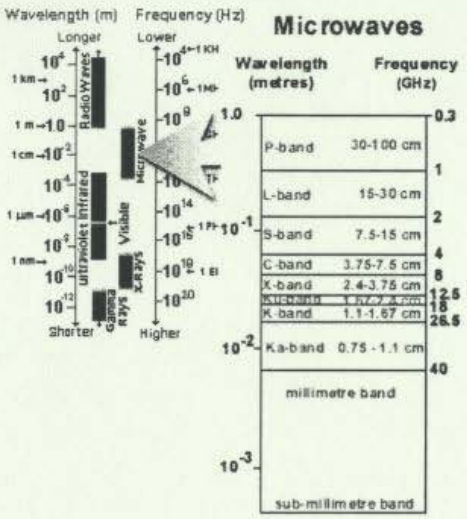


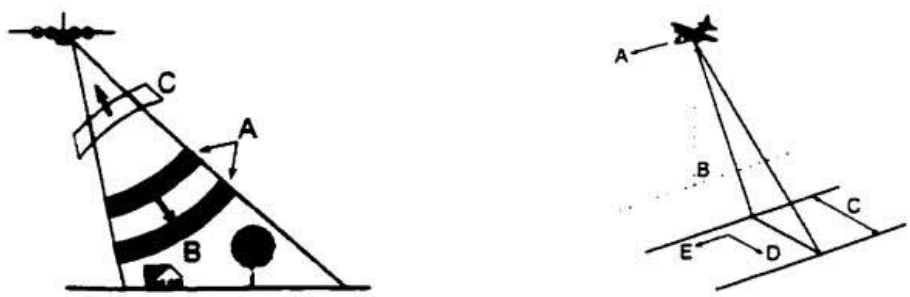
Figure 2.1: Microwave Spectrum (taken from [4])

By using the motion of its platform (i.e. aircraft or satellite) to synthesize

the effect of a very long antenna. a SAR is able to overcome the antenna size limitation. Today the resolution provided by RADARSAT (a spaceborn SAR system) is in the range of 10m-100m. which is sufficient for the study of the structures of sea ice.

2.2.1 SAR Basics

Figure 2.2(a) shows the basic mechanism of a side-looking SAR. The Transmitter generates pulses (A) at regular intervals. The pulses are focused by the antenna into a beam (B). which illuminates the ground surface obliquely. A portion of the energy (C) is then backscattered and received by the antenna. By measuring the time delay. the distance to a target and hence its location can be determined. With the sensor platform moving forward a 2-dimensional image can be obtained by signal processing of the pulse returns over time and the flight path.



(a) Basics of SAR Imaging (b)Imaging Geometry

Figure 2.2: SAR Basics (taken from [4])

2.2.2 Viewing Geometry and Spatial Resolution

The imaging geometry of a SAR system is shown in Figure 2.2(b) in which (A), (B), (C), (D), (E) correspond to the **flight direction**, **nadir**, **swath**, **range**, and **azimuth** respectively. The range resolution is dependent on the length of the radar pulse, while the azimuth resolution depends on the width of the illumination and the speed of the flight. Detail explanations of the resolution dependency can be found in [4].

It seems that fine range resolution could be achieved by using a very short pulse length. However, this is usually not practical since the peak power may exceed the limitation of the transmitter in providing a good signal-to-noise ratio (SNR). A technique known as pulse compression is therefore used to solve the problem. Wider pulses are transmitted instead of impulse-like waveforms, and the received signal is processed so as to compress the energy into a much narrower pulse [5].

Determined by the width of the illumination, azimuth resolution is actually influenced by radar beam angle and the slant range distance. The radar beam angle is approximately $2\arctan(\lambda/2D)$, where λ is the wavelength and D is the antenna length. Finer azimuth resolution therefore seemingly could be achieved by increasing the antenna length. However, the actual length of an antenna has to be limited by the platform carrying it. To overcome this, the motion of the platform and special signal processing of the echos are used to simulate the effect of a very long antenna. For further details, the reader is referred to [5].

2.3 SAR Sea Ice Imagery

Our particular interest is the appearance of sea ice in SAR images. It is influenced by the SAR system parameters (wavelength, incident angle, polarization, etc.) and by the physical properties of the sea ice types being imaged.

2.3.1 Parameter Dependency

Whether a surface will appear smooth or rough under the illumination can be indicated by the Rayleigh criterion: $h < \lambda/(8\cos(\theta))$, where h is the root mean square height of the surface, λ is the signal wavelength and θ is the angle of incidence with the surface [1]. If the Rayleigh criterion holds true, the surface will be smooth and reflect away most of the signal like a mirror, resulting in a low returned energy. Similarly, a surface will be considered rough if the Rayleigh criterion is false.

Since the smoothness is influenced by the incidence angle, the tone values for areas of the same ice type will decrease as the incidence angle increases across the image. This justifies the statement that classification and segmentation methods based on tone alone are not stable for SAR images.

A suitable microwave band needs to be selected to capture the small scale surface roughness while being able to penetrate well through the atmosphere. The microwave region ranges from $1\mu m$ to $1m$, from the shorter wavelength near the thermal infrared region to the longer wavelengths approaching those used for radio broadcasts, as shown in Figure 2.1. For our research, the SAR

data is obtained by RADARSAT using C-band which is suitable for sea ice imaging. Shorter wavelengths are more attenuated by the atmosphere, while longer wavelengths are less suited to discriminating the small scale roughness of new or young ice.

Polarization, another SAR system parameter, is also very important in influencing the appearance of objects in SAR images. In most radar systems, microwave is transmitted in either horizontal (H) or vertical (V) planes. Similiarly the antenna receives either horizontal or vertically polarized backscatters. Thus, there are four combinations of the mode of transmitting and receiving signals. Depending on the mode, the signals will interact with the surface and be backscattered differently, thus greatly affecting the appearance of the surface.

2.3.2 SAR Signature of Sea Ice

Sea ice is a mixture of freshwater ice, brine and air. The SAR signature of sea ice is influenced by the constituents' dielectric properties, the volume fraction of each constituent and the geometry (shape, size and orientation) of brine pockets in the ice as well as by the surface conditions of the sea ice. As these factors are determined by many natural environmental variables including temperature, salinity, wind condition, ocean currents and rate of freezing, studying and utilizing the SAR signature of sea ice is really a complex task.

Before investigating the electrical properties of sea ice, several electromagnetic quantities need to be defined as follows [6]:

- Relative permittivity (At frequencies lower than about 1MHz, the domi-

nant electrical property is the bulk conductivity, while at higher frequencies, the complex permittivity is used).

$\epsilon = \dot{\epsilon} - j\ddot{\epsilon}$

where $\dot{\epsilon}$ denotes the dielectric constant and $\ddot{\epsilon}$ is the loss factor giving the electromagnetic loss in the materials.

- Propagation, absorption, and phase constant.

  The intensity of the electric field at position z can be expressed as

  $E(z) = E_0 e^{-\gamma z}$

  where $\gamma = \alpha + j\beta$, α is the absorption constant and β is the phase constant, and z represents the depth from the surface. They are related to the complex permittivity by

  $\alpha = \frac{2\pi}{\lambda} |\mathrm{Im}\{\sqrt{\epsilon}\}|$, $\beta = \frac{2\pi}{\lambda} |\mathrm{Re}\{\sqrt{\epsilon}\}|$

- Extinction and scattering coefficient.

  The total electromagnetic loss consists of absorption loss (electromagnetic power transformed into other forms of energy, such as heat) and scattering loss (energy travel in directions other than that of the incident radiation).

  $K_e = K_a + K_s$ and $K_a = 2\alpha$

- Penetration depth.

  When across the boundary into the medium, the penetration depth is defined as the depth δ_p at which

  $\int_0^{\delta_p} K_e(z)dz = 1$

  If scattering is ignored, α does not depend on z, and $\ddot{\epsilon}$ is much smaller than $\dot{\epsilon}$,

  $\delta_p = \frac{1}{K_a} = \frac{\lambda\sqrt{\dot{\epsilon}}}{2\pi\ddot{\epsilon}}$

The returned signals of sea ice consist of contributions from both surface

scattering and volume scattering (scattering by lower layers inside). The surface scattering is mainly influenced by the roughness of the surface, and the volume scattering by the complex permittivity as it determines how far the microwaves will penetrate into the ice.

Because salt increases the dielectric loss by adding free charge carriers, the dielectric loss factor of brine is much higher than that of other constituents of sea ice. In a microwave frequency range from 1GHz to 10GHz for example, the value of loss factor $\ddot{\epsilon}$ is greater than 20 for brine, while for freshwater ice it is in the order of 10^{-3}. Since the dielectric constant of brine (> 10) is also fairly high compared with that of freshwater ice which is around 3.17, the volume fraction of brine plays an important role in determining the overall electrical properties of sea ice. The following equations give an approximation of the relationships between the permittivity of sea ice and relative brine volume V_b [6].

$$\begin{aligned} 1GHz: &\quad \dot{\epsilon} = 3.12 + 0.009V_b, \quad \ddot{\epsilon} = 0.04 + 0.005V_b \\ 4GHz: &\quad \dot{\epsilon} = 3.05 + 0.007 2V_b, \quad \ddot{\epsilon} = 0.02 + 0.003 3V_b \\ 10GHz: &\quad \dot{\epsilon} = 3.0 + 0.012V_b, \quad \ddot{\epsilon} = 0.0 + 0.010V_b \end{aligned} \tag{2.1}$$

Categories of different sea ice have different SAR signatures. These categories are defined by the World Meteorological Organization (WMO) [7] and are summarized below:

> **New Ice**: A general term for recently formed ice. These types of ice are composed of ice crystals which are only weakly frozen together (if at all) and have a definite form only while they are afloat.

Grey: Young ice 10-15 cm thick.

Grey-white: Young ice 15-30 cm thick.

Thin first-year: First-year ice of not more than one winter's growth. 30-70 cm thick.

Medium first-year: First-year. ice 70-120 cm thick.

Thick first-year: First-year ice over 120 cm thick.

Old Ice: Ice which has survived at least one summer melt. Often sub-divided into either second year ice or multi-year ice.

Since first year ice has a high brine volume and therefore is not easily penetrable by microwaves. surface scattering dominates. For first year smooth ice. most of the signal energy is reflected away and only a small part is returned in the direction of the radar. As the result the tone values of such an area are very low. making it looks like a black silhouette. For first year rough ice. more returns can be obtained and the corresponding area appears brighter.

Multi-year ice has a much lower salinity. Therefore besides the surface scattering. the backscatters are also determined by volume scattering, which is influenced by particle (air bubble and ice crystal) size and density. and the extinction coefficient.

Clausi [1] believes it is uncertain whether or not new ice can be distinguished from first year smooth ice. But he also argued that new ice often has leads. which are narrow irregular cracks in the thick ice and consist of thin ice and sometimes open water. Therefore new ice with leads can be characterized

by well-oriented features parallel to the lead axis or polynya boundary.

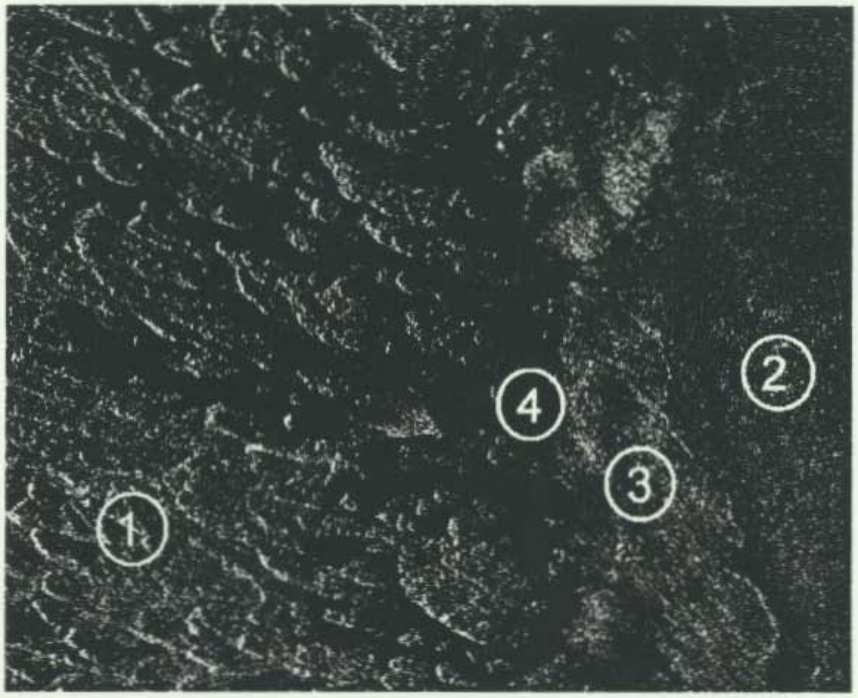


Figure 2.3: An example SAR sea ice image of the Nain/Voisey's Bay area

An example SAR sea ice image is shown in Figure 2.3. This image was obtained via RADARSAT operating in the Standard 2 beam mode, providing a resolution of $12.5m \times 12.5m$, over an area of approximately $113km \times 92km$. On the left of the image (area 1) is the land area and on the right (area 2) is pack ice containing new ice or open water. Between them are rubble ice (area 3) with a brighter appearance and landfast ice (area 4) with a dark appearance. From the image, orientational characteristics can clearly be observed in the rubble ice area. Those features can provide critical information for classification.

2.4 Speckle Noise

The presence of the notorious speckle noise in SAR images adds considerable difficulty to SAR sea ice texture classification. SAR speckle noise is caused by the coherent nature of the radar imaging system. When a medium containing many elementary sub-resolution scatterers is illuminated by a radar, the return waves will add either constructively or destructively, forming a random, interference pattern. It has been experimentally shown that speckle noise is predominantly multiplicative [8], as expressed by the following model:

$$y(i,j) = x(i,j) \cdot n(i,j) \tag{2.2}$$

where $y(i,j)$ is the intensity or amplitude of a SAR image pixel at (i,j), $x(i,j)$ is the noise-free quantity at (i,j) and $n(i,j)$ is the speckle noise characterized by a distribution with a unity mean ($E[n] = 1$) [12]. Although some others [13] argued that this model should take into account the correlation of the speckle as well as its overall probability distribution, in most cases this simple model is accurate enough.

Many speckle noise filters have been devised to suppress the noise before doing any further processing. However for our texture analysis, such filtering would probably break down the texture details and therefore is not adopted. Instead, we apply a logarithm to our SAR texture images, which approximately converts the muliplicative speckle noise to additive noise with zero mean value. By this process the PLS data set mentioned above is obtained.

2.5 Summary

In this chapter we have reviewed topics concerning SAR imaging systems, SAR sea ice backscattering, and speckle noise. Understanding these is important for proper interpretation of SAR sea ice imagery. The following three chapters will deal with texture analysis methods that can be used for sea ice classification.

Chapter 3

A Review of Texture Feature Extraction Methods

3.1 Introduction

For SAR sea ice, intensities often provide good visual discriminations. However, they are not very reliable for classification purposes because of intensity variations caused by numerous environmental factors. Although some previous work such as that of Kwok [9] has shown success in using intensities for classification, strict limitations exist in terms of the environment including the imaging parameters and the target sites. Texture is expected to provide a more stable description about SAR sea ice, and hence our research uses texture information for SAR sea ice classification.

Many different definitions exist for texture. A formal one given by the IEEE is that texture is "an attribute representing the spatial arrangement of the gray levels of the pixels in a region" [10].

As already introduced in the first chapter, the design of a texture classification system consists of four stages: texture feature extraction; classifier design; classifier training; and performance evaluation. Perhaps the most important stage is the feature extraction. "A texture feature is a value, computed from the image of an object, that quantifies some characteristic of the gray-level variation within the object. Normally, a texture feature is independent of the object's position, orientation, size, shape and average gray level (brightness)" [11, p.499]. The feature extraction step produces a set of features representing the information needed for subsequent classification. Compared to the original image, the feature vector contains much less data, but enough information to allow the unique and correct identification of the texture type. Good features should have significantly different values on textures belonging to different types, but similiar values on the same type of textures. Morever the various features should be uncorrelated with each other, to ensure that the feature vector has been reduced to the fewest possible dimensions.

Approaches to texture feature extraction span a wide range of methods. Most of them can be categorized as statistical, structural, and model based. Statistical approaches attempt to characterize textures in a probabilistic sense based on definitions such as smooth, coarse, grainy, regular, directional, etc [1]. These characteristics can be measured either spatially or spectrally. Simple statistical measures include the standard deviation, variance, skewness, and kurtosis of the gray levels [11]. More complex measures are those based on the GLCM and the wavelet transform, which are presented later in this work.

The structural approach assumes that the texture is a spatial arrangement of basic primitives. The texture feature extraction can then be done by obtaining measurements of the primitives and their spatial arrangments. Since no predictable, consistently repeating patterns seem to exist in SAR sea ice textures, Clausi excluded structural methods for use in sea ice discrimination [1].

The model based approaches are based on random fields and fractal parameters. They usually fit specific models to the textures. The model parameters thus obtained are then used for texture description. A commonly adopted model is the Markov Random Field (MRF) [14] [15]. In a more recent work [16] Clausi did a comparative study of the MRF approach and several statistical approaches for their ability to interpret SAR sea ice imagery. He concluded that the MRF results significantly lag others.

In this chapter we review previous research on statistical feature extraction methods including those based on the GLCM, Gabor filtering, and the Discrete Wavelet Transform. Some other related approaches are Power Spectrum [17] and Independent Component Analysis [18]. They are less pertinent and are not discussed in this thesis.

Strictly speaking, texture analysis by filtering cannot completely be categorized as statistical, since techniques other than statistical, for example the Markov Random Field, can also be applied to the filtering responses to retrieve the features. However as statistical measures are the ones most commonly adopted, we include filtering based methods in this category.

3.2 Gray Level Coocurrence Matrix (GLCM)

3.2.1 Definition of Cooccurence Matrice

Perhaps the most popular method of feature extraction for remotely sensed sea ice imagery is the cooccurrence matrix (GLCM). The following example illustrates how to calculate the GLCM.

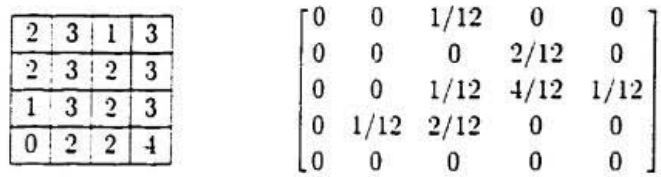


Figure 3.1: Calculation of GLCM

The left part in Figure 3.1 is a 4 × 4 pixel area of a texture image (i.e. window size 4). The image has five different gray levels and the integers in each element indexes the gray value of the corresponding pixel. Suppose we want to calculate the GLCM **C** given a direction (0° in this example) and an interpixel displacement (one pixel in this example). Then the (i, j) element of the GLCM is the number of times the gray level pair i and j occur together with the chosen displacement and direction, normalized by the total number of pixel pairs. For example, the combination of gray level 1 neighboured by gray level 3 to its right occurrs two times, therefore $\mathbf{C}(1, 3)$ (note the gray level index begins with 0) is 2 divided by the total number of pixel pairs 12. The dimension of this GLCM **C** is $G \times G$, where G is the total number of gray levels.

Different GLCMs can be obtained for different combinations of the four parameters (direction θ, displacement δ, number of gray levels G, and window size

w). Since G determines the dimension of the GLCM and since G can be large, a quantization is often performed to reduce the computation complexity. However too few quantization levels may corrupt or destroy important texture characteristics. Window size is another parameter that influences feature extraction by GLCM. The window should be large enough to include sufficient texture elements without incurring too high a computational burden. Like the window size parameter, selection of interpixel displacement is also texture dependent. A small displacement is efficient for fine textures and a large displacement is prefered for smooth textures. Directional features can be extracted well if the direction parameter is properly selected. However, those feature directions should be consistently same in both the training samples and the target texture to be classified. Otherwise such directional information may be a deterrent to correct classification. The average of multiple directional GLCM is often performed for isotropic textures, or to reduce directional variations for anisotropic textures.

Maximum Probability (MAX)	$max\{C_{ij}\}\forall(i,j)$
Uniformity (UNI)	$\sum_{i=1}^{G}\sum_{j=1}^{G}C_{ij}^2$
Entropy (ENT)	$-\sum_{i=1}^{G}\sum_{j=1}^{G}C_{ij}logC_{ij}$
Dissimilarity (DIS)	$\sum_{i=1}^{G}\sum_{j=1}^{G}C_{ij}\|i-j\|$
Contrast (CON)	$\sum_{i=1}^{G}\sum_{j=1}^{G}C_{ij}(i-j)^2$
Inverse Difference (INV)	$\sum_{i=1}^{G}\sum_{j=1}^{G}\frac{1}{1+\|i-j\|/G}C_{ij}$
Inverse Difference Moment (IDM)	$\sum_{i=1}^{G}\sum_{j=1}^{G}\frac{1}{1+(i-j)^2/G^2}C_{ij}$
Correlation (COR)	$\sum_{i=1}^{G}\sum_{j=1}^{G}\frac{(i-\mu_x)(j-\mu_y)}{\sigma_x\sigma_y}C_{ij}$

where in all cases (μ_x, μ_y) and (σ_x, σ_y) are means and standard deviations of (row i, column j) respectively

Table 3.1: GLCM texture statistics defined

For feature extraction purposes, statistical information is then generated from the GLCM (MAX, UNI, ENT, DIS, CON, INV, IDM, COR, etc), as summarized in Table 3.1, which together with the following quotation is taken from Clausi [1, p.26-27]. "The statistics extract three fundamental characteristics from the cooccurence matrix. Moments about the main diagonal indicate the degree of smoothness of the texture. The closer the entries to main diagonal, the smoother the texture. The statistics dissimilarity (DIS), contrast (CON), inverse difference (INV), and inverse difference moment (IDM) are statistics of this type. Another fundamental characteristic of the cooccurence matrix is the uniformity of its entries. If the gray levels in the window tend to be homogeneous, then only a few gray level pairs represent the texture. The statistics maximum probability (MAX), uniformity (UNI), and entropy (ENT) describe homogeneity. The final statistic, correlation (COR), describes the correlation between the gray level pairs".

3.2.2 GLCM on SAR Sea Ice Imagery

Research using the GLCM for SAR sea ice texture analysis can be found in several papers, where discriminations are made between distinct ice categories such as first year ice and multi year ice. In [19] Shoker investigated the GLCM statistics and found that they are highly correlated. Thus he argued that combining gray tone with only one GLCM texture statistic is a good enough choice for SAR sea ice texture classification. Another discovery of his experiment is that when a significant overlap in gray tone exists between two ice types, texture will not be capable of separating the two ice types. His experiments also showed that IDM was the best statistic, followed by entropy (ENT) and uniformity (UNI). In selecting GLCM parameters he tried several quantization levels.

window sizes and displacements. He concluded that 16 levels is appropriate for quantization, and with respect to both window size and interpixel displacement the results were almost insensitive.

In [20] the authors did studies on sea ice classification with both GLCM texture statistics and standard statistics. They investigated the issue of directionality in a sea ice GLCM and concluded that multiple GLCMs should be averaged together to reduce the directional variance. The displacement issue was also addressed. They found that a length of four pixels was appropriate (the image data they used is of $16m$ resolution) and that the statistics were almost invariant for length over four pixels. The emphasis of their study was on the comparison between the performance of GLCM statistics and that of standard statistics. The results showed that among the three top statistics are two GLCM statistics (ENT, which describes homogeneity and IDM, which describes smoothness) and one standard statistic (range). They also found that standard statistics have greater classification accuracy than texture statistics in general (87% vs. 75% for their data). The combination of standard statistics and texture statistics gives only marginally better results (by 3%) than using the standard statistics alone. These results suggested that using standard statistics alone is as good as any other complex combinations of statistics for the purpose of SAR sea ice classification.

The authors of [21] also investigated the direction and displacement parameters, as well as the quantization factor. They gave the same argument as [20] for directionality issues (i.e. average matrix of different directions). For displacement, unlike [20] in which the conclusion was specific to the image tested.

the authors of [21] concluded that a single displacement value for GLCMs to represent sea ice textural contexts is not advisable. For the quantization issue, two conclusions were drawn: 1) Dissimilarity between two ice types varies very little with the number of quantization levels. 2) GLCM statistics are more consistent when using a higher number of quantization levels. Then the authors did experiments on three cooccurence matrices - MDMO (the average of multi-displacement and multi-orientation matrices), ODMO (optimal-displacement and multi-orientation) and ODOO (optimal-displacement and optimal-orientation). Results showed that the MDMO approach is significantly better. Therefore they concluded that the orientation factor is probabily not important in SAR sea ice analysis and a range of displacement is more representative than a single displacement value.

3.3 Multi-channel Filtering

Most filtering based classification approaches assume that textures can be identified by their energy distribution in the spatial frequency domain. Depicted in Figure 3.2 is a typical filtering based classification system. The input image passes through a series of filters and the output responses usually correspond to frequency subband components. They are further processed nonlinearly to estimate the local energies. Features vectors can then be generated with these energies using statistical measures. In many cases, the nonlinear function performs rectifying operations, transforming both negative and positive amplitude to positive amplitude [22], and the statistical function is just an averaging function. Some systems do not directly use the subband energies, and therefore

do not have the nonlinear processing step but use a more complex statistical method to extract the features.

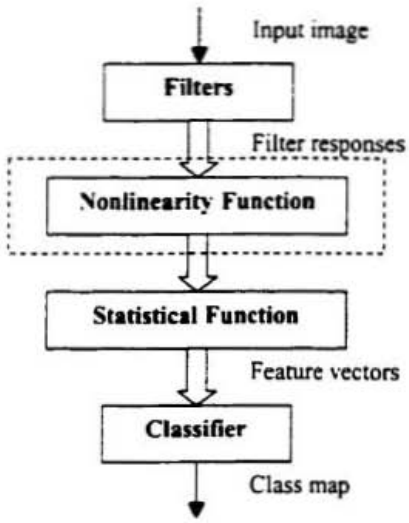


Figure 3.2: A typical filtering based classification system

Two early filtering approaches are the Laws filter masks [23], and Ring and Wedge filters [24]. Laws suggested using 25 separable filters for the two dimension image, 5 in each dimension. Thus, the spatial frequency domain is almost evenly split into 25 subbands by these filters. Coggins and Jain [24] designed another bank of filters made up of seven dyadically spaced ring filters and four wedge-shaped orientation filters, which extract the frequency and orientation information separately.

Some later approaches, Gabor filtering and the wavelet transform, have attracted more attentions in recent years since they agree well with research on human visual system (HVS). It has been found that the HVS decomposes im-

ages by several relative narrow-band filters tuned to different spatial frequency ranges and orientations, and the filtered images are then processed by subsequent detectors [25]. Experiments have shown that the frequency bandwidth of each filter is about one octave [26] and the orientation range is not more than 30^o.

Both Gabor and wavelet filter banks can be designed to be tuned to octave frequency bands and different orientations. Although the Gabor function is not a wavelet in a strict sense, it can be implemented in a similar manner to that of wavelets. This section introduces the Gabor filtering and the wavelet transform.

3.3.1 Gabor Filtering

In determining how the filter should shape an input image, it is desirable that it have high spatial resolution for object localization, and also high spatial frequency resolution for object identification. Unfortunately, these are two conflicting goals. The Gabor filter has the optimum joint spatial and spatial-frequency localization, and therefore is often used to construct the filter bank.

A Gabor function is a Gaussian modulated sinusoid function. It is a band-pass filter with impulse response

$$h(x,y) = \frac{1}{2\pi\sigma_x\sigma_y} e^{-(\frac{x^2}{2\sigma_x^2}+\frac{y^2}{2\sigma_y^2})} e^{-j2\pi(Ux+Vy)} \tag{3.1}$$

where (U,V) is the center frequency of the filter. The corresponding function in the spatial frequency domain is

$$H(u,v) = e^{-2\pi^2[(u-U)^2\sigma_x^2+(v-V)^2\sigma_y^2]} \tag{3.2}$$

The values of σ_x and σ_y determine both the spatial resolution and the spatial frequency resolution of the filtered image. Low values of σ_x and σ_y give high spatial resolution but low spatial frequency resolution, and vice versa. Care must be taken in selecting (σ_x, σ_y) and the center frequency (U, V) for each filter in the filter bank. There are two kinds of approaches to determining these filter parameters. The first one tries to design the optimal Gabor filters adaptively for the separation of the features of given textures [28][30], while the second one deals with a more general case and uses a fixed filter bank for all textures. Unlike the first approach in which the filters may cover only a part of the spatial frequency domain, the second one usually provides coverage of the entire spatial frequency plane.

Motivated by the similarities between octave band decompostions and the HVS, most of the research using the fixed filter bank idea have designed their filters to be dyadic. Jain and Farrokhnia [27] suggest a bank of Gabor filters tuned to five radial frequencies and four orientations, as shown in Figure 3.3.1. Clausi [1] argued that the 45° angular bandwidth of the filters is not in agreement with the HVS and selected a smaller angular bandwidth of 30° for his filters.

The filtered image is processed nonlinearly (usually by rectification) and smoothed. The local energies of regions surrounding each pixel in the image can then be estimated with these responses. These energies comprise the corresponding feature vector for future classification or segmentation. Smoothing is required here because the rectified output often has non-negligible variations within the same texture area. Among possible smoothing filters, the Gaussian

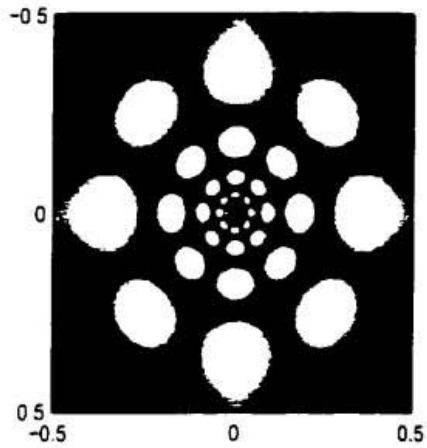


Figure 3.3: The frequency response of the dyadic bank of Gabor filters. The axes are in normalized spatial frequencies (taken from [22]).

shaped filter seems to be a good candidate. Bovik [28] uses the Gaussian filter that has the same parameters as that of the corresponding Gabor filters, but with larger spatial extent. This can reduce the spectral leakage and also preserve fairly good spatial localization.

There are not many references available in the literatures about using Gabor filtering for SAR sea ice image classification. Clausi [1] gives a comparative study on Gabor filtering and the GLCM for the SAR sea ice classification and segmentation. His experiments showed that "in general Gabor features are more clusterable than the cooccurrence features" [1, p.151], which makes it advantageous in unsupervised segmentation. He also concluded that the cooccurrence features can capture fine boundary details well with a small window size. Since smoothing is often required for the Gabor filtering output, the Gabor technique

tends to blur texture boundaries belonging to small regions.

3.3.2 Discrete Wavelet Transform (DWT)

Wavelets are families of basis functions generated by dilations and translations of a basic filter function. The wavelet functions construct an orthogonal basis and the discrete wavelet transform is thus a decomposition of the original signal in terms of these basis functions:

$$f(x) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} C_n^m \upsilon_{m,n}(x),$$

where the $\upsilon_{m,n}(x) = 2^{-m/2}\upsilon(2^{-m}x - n)$ are dilations and translations of the basic filter function $\upsilon(x)$. Unlike Fourier bases which are composed of sines and cosines that have infinite length, wavelet basis functions are of finite duration. The discrete wavelet transform coefficients C_n^m are the estimation of signal components centered at $(2^m n, 2^{-m})$ in the Time-Frequency plane, and can be calculated by the inner products of $\upsilon_{m,n}(x)$ and $f(x)$. It is obvious that the wavelet transform is an octave frequency band decomposition of the original signal, which agrees well with HVS. The narrow-band signals then can be further downsampled and provide a multi-resolution representation of the original signal.

The discrete wavelet coefficients C_n^m can be efficiently computed with a pyramid transform scheme using a pair of filters (a low-pass filter and a high-pass filter), as depicted in Figure 3.4 [31]. As we can see from the figure, the orthogonal multi-resolution analysis of the DWT is characterized by a resolution factor of 2 between two consecutive scale levels.

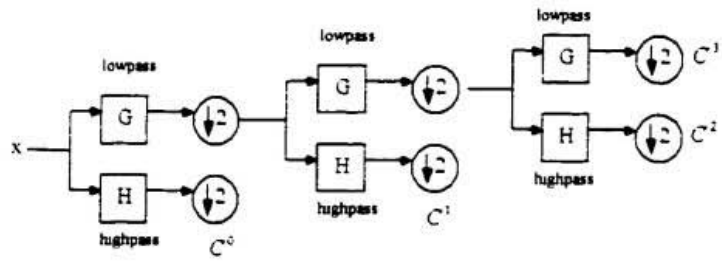


Figure 3.4: 3-level DWT of 1-D signals

For images which have two dimensions, the filtering and downsampling steps will be repeated in rows and columns respectively. The procedure for two levels is shown in Figure 3.5. A sample image and its wavelet transform obtained in this manner are shown in Figure 3.6.

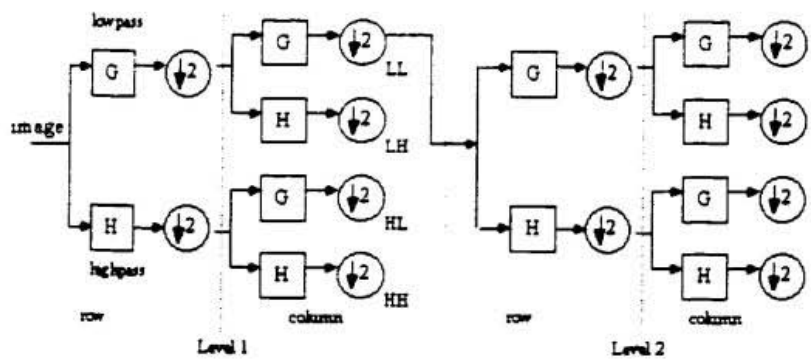


Figure 3.5: 2-level DWT of images

At each level the image can be transformed into four sub-images: LL (both horizontal and vertical directions have low frequencies), LH (the vertical direction has low frequencies and the horizontal has high frequencies), HL (the vertical direction has high frequencies and the horizontal has low frequencies) and HH (both horizontal and vertical directions have high frequencies).

(a) original image (b) band image of 2-level DWT (c) bands of 2-level DWT

Figure 3.6: 2-level DWT example

Statistical information can be calculated from the resulting sub-images. These statistics will represent the characteristics of the original image at different resolution levels and directions. Typically, a simple energy statistic is computed. More complex techniques such as histogram statistics and the GLCM can also be applied on the wavelet bands. These are presented and discussed in chapter 4.

3.4 Summary

In this chapter we have reviewed three kinds of approaches (GLCM, Gabor filtering, DWT) for texture classification. They have all been categorized as statistical methods. The GLCM usually works at the pixel level, while the Gabor filtering and DWT can provide a multi-resolution analysis similar to what is effectively performed by the HVS. Texture classification using the DWT is studied in detail in the next chapter.

Chapter 4

Texture Feature Extractions with Wavelet Transform

4.1 Introduction

Texture features that are useful for classification usually exist at various scales. For sea-ice textures this also seems to be true as suggested by [21] in the context of an investigation of the selection of interpixel displacements for GLCMs. A weakness shared by many texture analysis schemes, including the GLCM and MRF, is that the image cannot be efficiently analyzed at multiple scales. Thus, we based our research on the Discrete Wavelet Transform (DWT), which provides an efficient way to obtain a multi-resolution representation of the texture image.

The output of the DWT is a collection of images, each of which represents the component of the original image at specific directions and resolutions. Statistical information can then be computed for those image bands to complete

the feature extraction.

This chapter focuses on texture feature extraction methods related to the Wavelet Transform. In Section 4.2, a simple comparative study is presented on the selection of different wavelet kernel functions and different signatures. Section 4.3 investigates feature extractions using non-dyadic wavelets. Then applications of these classifiers on SAR textures are presented in Section 4.4. Section 4.5 deals with texture segmentation, which provide visual evaluations of classifiers based on these feature extraction methods.

4.2 DWT Band Signatures

4.2.1 First Order Signatures

Energies

It is typical for filtering approaches, including the DWT, to use the response energies as the signatures. Energies can be measured by either the magnitude $|\cdot|$ or by squaring $(\cdot)^2$. Before being used as features, the response energies have to be averaged or smoothed within a local area when doing texture segmentation, or within the whole image when doing texture classification. This is because the responses often have non-negligble variations, especially for the DWT where different downsample offsets will cause different DWT responses (known as the shift variant property of the DWT). This variation will be significantly reduced when averaging over a whole subband is performed, due to the random nature of textures. And hence we use the average energies of the subbands as the texture features for our classification experiments.

The classification procedure is as follows:

- Training:

  Decompose the sample images in the training set with the DWT and calculate the average energy of each band. That is, if the coefficients of a band are $C(i,j)$, with $1 \leq i \leq H$ and $1 \leq j \leq W$, where H and W are the height and width of the band, then the energy is

  $$\overline{E} = \frac{1}{HW} \sum_{i=1}^{H} \sum_{j=1}^{W} |C(i,j)|.$$

  This process produces a feature vector for each image. Vectors belonging to the same class can be further processed (averaged here) to represent features of that class type.

- Classification:

  Decompose the image to be classified with the DWT and calculate the feature vector as in the training step. Compare the feature vector with that of each class type and classify the image with a minimum distance method. Here for simplicity, the Euclean distance is used.

Performance evaluation of the classifier is based on experiments with data set MBS (see Appendix A). A 3-level DWT is performed. Thus the feature vector consists of the energies of nine bands when excluding the lowest frequency band which is sensitive to illuminations, or ten bands when including it. The confusion matrices in Table 4.1 and Table 4.2 show respectively the performances of the classifiers including and excluding the lowest frequency band, using the 6-order Daubechies [32] wavelet. The row labels correspond to the texture types of the input test samples, while the column labels indicate the classified types.

To save space for the table, instead of printing the whole name of each texture only the first three charaters are used (i.e. bar, bub, can, gra, lea, san, sto, str, woo, pac, rub, and lan representing bark, bubble, canvas, grass, leather, sand, stone, straw, woolen, pack ice, rubble ice, and land respectively).

	bar	bub	can	gra	lea	san	sto	str	woo	pac	rub	lan
bar	25	4									3	
bub	6	22					1	1			2	
can			32									
gra	1			23	1	3		3		1		
lea				1	31							
san						32						
sto							30					2
str		1		2	6	3		18		2		
woo	2								27	2	2	1
pac									1	27	4	
rub									1	16	15	
lan											1	31

Table 4.1: Classification result by 3-level Daubechies(6) DWT energy classifier (including lowest frequency band). Overall accuracy: 81.51%

The classifier that utilizes the lowest frequency band energy has a much poorer overall performance. This is because textures are often more characterized by mid or high frequency bands, and the lowest frequency band is usually sensitive to illumination variation. Excluding the coarsest approximation band, however, might discard useful information corresponding to slow variations of large scale texture features. A feasible way to avoid this problem is to include the variance of the approximation band. In fact, variances of all subbands form an orthogonal decompostion of the overall variance of the original signal [35]. But for simplicity, we just exclude the lowest frequency band in all our experiments presented subsequently in the thesis.

	bar	bub	can	gra	lea	san	sto	str	woo	pac	rub	lan
bar	30	2										
bub	1	29						1				1
can			32									
gra	1			24	1	3		3				
lea				1	31							
san						32						
sto							32					
str				2	7	4		19				
woo	2								29			1
pac										29	3	
rub										15	17	
lan												32

Table 4.2: Classification result by 3-level Daubechies(6) DWT energy classifier (excluding lowest frequency band). Overall accuracy: 87.5%

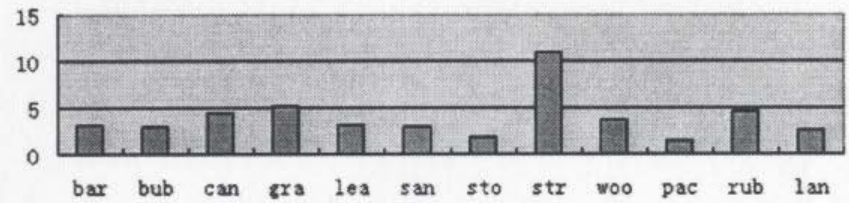


Figure 4.1: Standard Deviation of Features

A troublesome Brodatz texture is that of straw, for which the classification accuracy rate for Table 4.2 is only 59.4%. Figure 4.1 shows that the straw feature vectors have a high variance. This is most probably because of the directional variations within and among the straw samples. Since the DWT has some degree of directional selectivity (HL band for horizontal, LH for vertical, and HH for diagonal), those texture directional variations will cause the energy to be distributed among different subbands inconsistently, leading to a high variance.

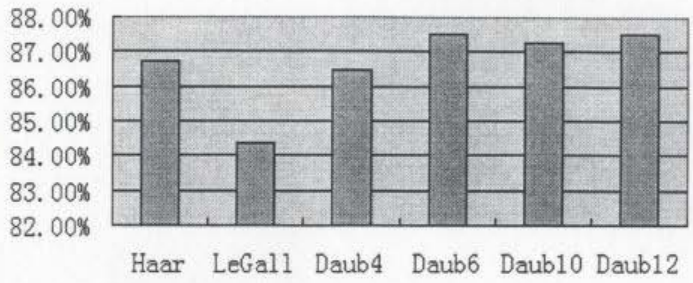


Figure 4.2: Performances of DWT classifiers of different wavelet basis

Figure 4.2 shows the overall performances of classifiers based on different wavelets. In selecting a wavelet basis careful consideration must be paid to the shape of the waveform, the order of the wavelet filters, and whether it is symmetric or not. If the waveshape of the chosen wavelet can match image components well, the DWT will provide a very efficient and compact representation for identification purpose. There has been some work in designing signal-similar wavelets for specific applications, especially in defect detection [34]. However, this kind of method seems not to be advantageous for applications involving multiple textures, or textures that do not have regular primitives which is the case for many natural textures (probably including SAR sea ice). The symmetry of the wavelet filters affects the amount of phase shift in the decompostion and thus the localization of image components. For image segmentation, where exact localization of features is required, biorthogonal bases are preferred since orthonormal basis functions lack the desired symmetry. But for classification this symmetry property is not so necessary. High order wavelets usually give better approximations to highpass filters with less spectral leakage. As a result, high order wavelets can extract the different frequency bands of components better than low order wavelets, and therefore are also more sensitive to direc-

tional variations of the training and validation samples, as implied by Figure 4.3. However, two disadvantages exist for high order wavelets: they are more involved in boundary effects and have higher computation complexities. Therefore a trade off needs to be made and we have chosen the 6-order Daubechies wavelet for most of our experiments.

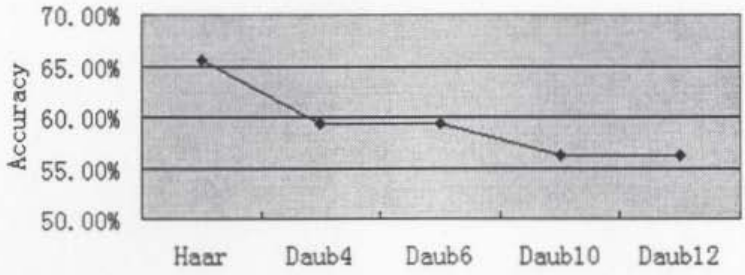


Figure 4.3: Classification Accuracy of Straw with Wavelets of Different Order

Directional sensitivity can be reduced by averaging the energy of bands at the same level. However, useful orientation information is also eliminated at the same time. This information loss may cause poor classification results, especially for textures that have significant and consistent orientation features crucial for classification. A new method that utilizes orientation features for rotation invariant classification will be presented later in Chapter 6.

Histogram Parameters

Mallat [31] observed experimentally that the detail histograms of many natural texture images can be modeled by a family of exponential functions.

$$h(u) = Ke^{-(|u|/\alpha)^{\beta}} \tag{4.1}$$

The variance α is proportional to the width of the histogram, while β models the decreasing rate of the peak. K is a scaling factor determined by both the total count of elements and the distribution of the histogram.

Van de Wouwer et al. [43] utilized the parameters of these histogram models as features in their classification experiments and concluded that these parameters are better features than the energies. Energy information, whether it is measured by magnitude or variance, is only part of all the information contained in histograms and therefore an accurate histogram model is expected to characterize the texture better than energy does.

A special case of this model is the Gaussian function (β is 2), where the squared energy signature is the square of the corresponding model parameter α. Thus in this case the two kinds of features, energy signatures and histogram parameter signatures, are equivalent. After comparing the curve fitting using the Gaussian model and the general exponential function of Equation 4.1, as shown respectively in Figure 4.4 and Figure 4.5, we can conclude that both the Gaussian and general exponential functions give good approximations of the wavelet band histogram, although the general exponential function may be a slightly more accurate model.

Quantization is needed to obtain the histogram. The selection of the quantization bin could influence the exponential model parameters. For this consideration, the Gaussian model is advantageous since its only parameter can be simply and consistently computed by the squared energy.

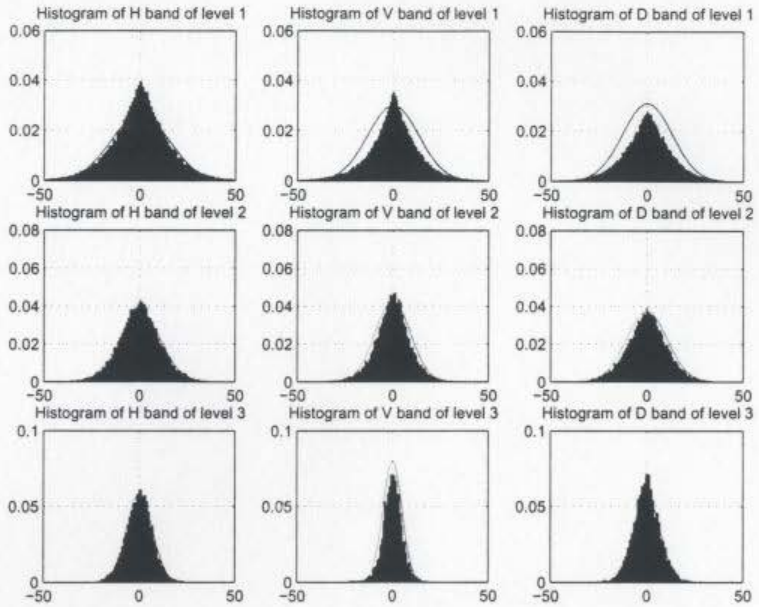


Figure 4.4: Curve fitting using Gaussian function for the 3 level wavelet band histogram of bark texture

4.2.2 GLCM for Second Order Signature

If first order signatures do not suffice in characterizing the texture wavelet bands, higher order signatures could be included. The GLCM is an obvious choice for obtaining second order statistics.

Experimental results of classification using DWT-GLCM signatures are shown in Table 4.3. Although the overall performance is lower than that of the simple DWT energy classifier, some textures such as straw are identified more accurately. Unlike the DWT wavelet which can only extract mixed diagonal (45^o and 135^o) features, the GLCM can obtain separate diagonal features, and hence can be expected to perform better for textures like straw in which the dominant

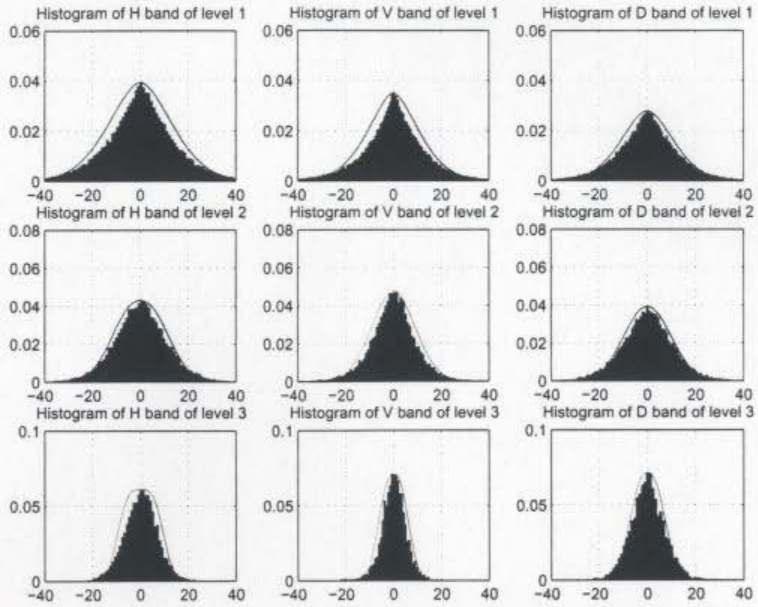


Figure 4.5: Curve fitting using the general exponential function for the 3 level wavelet band histogram of bark texture

energies are in the diagonal directions.

Whether a first order statistic or a second order statistic is more suitable depends on the given textures. For the textures which have higher energies in the mid or high frequency bands, second order statistics are preferred since they will characterize the mid or high frequency bands in more detail than first order signatures. This is more or less like the non-dyadic wavelet transform which will be introduced later. Also, for textures which have dominant energies in the diagonal directions, second order signatures are better since they can further decompose the mixed diagonal directions separately.

	bar	bub	can	gra	lea	san	sto	str	woo	pac	rub	lan
bar	31					1						
bub	3	23				3		1				2
can			32									
gra	2			13	5	8				2	1	1
lea					29	2			1			
san				1		30			1			
sto							32					
str		1			2	1		28				
woo	1	1				2			28			
pac										27	5	
rub										9	23	
lan				2								30

Table 4.3: Classification result by 3-level Daubechies(6) DWT-GLCM Classifier. Four directions $(0°, 45°, 90°, 135°)$ and DIS statistics are used here . Overall accuracy: 84.90%

4.3 Classification with Non-dyadic DWT

4.3.1 Tree-structured Wavelet Transform (TSW)

Chang and Kuo [33] utilized a non-dyadic Wavelet Transform for texture classification and did experiments on Brodatz textures. Unlike the traditional DWT that keeps decomposing the band of the lowest frequency, they only decompose those bands in which the energy is not negligible. This approach is motivated by the observation that many natural textures have dominant frequencies in the middle frequency bands (for SAR sea ice textures, we need to see if the argument is still true and if the method is applicable). Their decomposition algorithm is described below and is called the Tree-structured Wavelet Transform(TSW), or Wavelet Packets:

1. Decompose a given texture image with a 2-D wavelet transform into 4 subimages, which can be viewed as the four child nodes to the node rep-

resenting the input image. Calculate the energy of each band. using the same formula used in previous DWT classifier.

2. If the energy of a subimage is significantly lower than that of others. the decomposition for this region is terminated since it contains little information. This step can be achieved by comparing the energy with the largest energy value in the same scale. That is. if $E < CE_{max}$. stop decomposing this region. where 0.15 is used for C as in [33].

3. If the energy of a subimage is large. the above decomposition procedure is applied again to that subimage.

4.3.2 Classification with TSW

Textures can be identified based on the corresponding decomposition tree patterns. A progressive TSW based classification scheme is detailed as follows [33]:

- Training:

  1. Decompose the images in a training set with the TSW.
  2. Each class maintains a list of all the tree patterns that have occurred. together with the mean energy tree of each tree pattern.

- Classification:

  1. Decompose the image in the testing set with the TSW.
  2. Arrange the bands (i.e. tree leaf nodes) in decreasing energy order. i.e.. $x_1 > x_2 > ... > x_{j-1} > x_j > x_{j+1} > ....$

3. Place all possible texture types into a candidate list, and perform the following iteration from the first feature (i.e., $x_m, m = 1$).

4. Remove texture types from the candidate list if they do not have the same leaf node as the mth dominant channel of the unknown texture.

5. For the remaining texture types, calculate the distance between the image and all candidate textures, using Euclidean distance for simplicity. Let $D_{min} = minD_i$. If $D_i > KD_{min}$, where K is a constant greater than one, remove texture type i from the list.

6. If there is only one texture left, assign the unknown texture to this texture. Otherwise, repeat from 4 the next iteration by increasing the value of m by one.

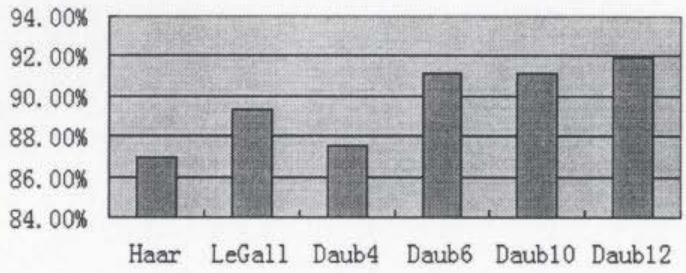


Figure 4.6: Performances of TSW classifiers of different wavelet basis

Figure 4.6 gives the performance of this TSW classifier using different wavelet basis functions. Compared to the performance of the DWT classifiers shown in Figure 4.2, the TSW classifiers have significantly better classification accuracy. The improvement may be explained by either the finer frequency resolution achieved in mid bands, or the reduction of sensitivity to orientational variation (since the TSW scheme maintains for each texture type a list of all tree patterns which occur, orientational variation sensitivity is reduced because the distance

to a class is measured by distance to the several nearest neighbours in the class). As an indication of this characteristic, the confusion matrix in Table 4.4 shows that the TSW performs significantly better for orientation-varying textures like straw and grass. However, this improvement is not robust because the training samples have to be carefully selected to include all possible orientational variations. It is not a solution to the orientation invariant classification problem.

	bar	bub	can	gra	lea	san	sto	str	woo	pac	rub	lan
bar	27	5										
bub	1	31										
can			32									
gra	2			27		3						
lea				1	31							
san						32						
sto							32					
str				1	2			28		1		
woo		1							31			
pac					1					30	1	
rub										15	17	
lan												32

Table 4.4: Classification result by 3-level Daubechies(6) TSW classifier. Overall accuracy: 91.15%

It appears that the Brodatz and SAR textures are quite different. Very few SAR textures and Brodatz textures are confused with each other. Usually SAR textures have more fine features (probably because of the speckle noise) and thus have more energy in the mid and high frequency bands. When examining the tree patterns of all textures obtained by the TSW, it is found that for most of the Brodatz's textures only the lowest band at each level had been decomposed, whereas for all three SAR textures the trees are quite balanced. However, SAR textures classification accuracy has not improved much even though the

TSW has achieved a better separation in the middle and high frequency area. A possible explanation is that the speckle noise energy in the middle and high frequency area has masked the real energy of the ice textures themselves.

One thing which should be noted is that when the TSW is applied to texture segmentation the computational complexity could be very high. As the classification window moves across the image, each subimage in the window area may have different decomposition patterns. The transform can no longer be applied to the whole image only once as in the DWT. Each pixel is involved in the TSW computation as many times as it appears in any window.

4.3.3 Comparision between TSW and DWT-GLCM classifiers

Before comparing the TSW classifier and the DWT-GLCM classifier, we first study the relationship between DWT energy and GLCM statistics. Figure 4.7 compares the average magnitude of one level Haar wavelet LH subband and a GLCM statistic (DIS, 0^o, and one pixel displacement) using bark, canvas and straw textures. The resulting graph indicates very strong correlations between DWT band energies and GLCM statistics (i.e. the DWT magnitude and the DIS are almost linearly proportional).

Suppose we have calculated the GLCM of an image g of size $M \times N$ pixels with direction $\theta = 0^o$ and interpixel displacement $\delta = 1$. The Dissimilarity(DIS) statistic, given in the formula $\sum_{i=1}^{G} \sum_{j=1}^{G} C_{ij}|i-j|$ can be rewritten as

$$DIS = \sum_{i=1}^{G} \sum_{j=1}^{G} \sum_{m=1}^{M} \sum_{n=1}^{N-1} \frac{\delta(g(m,n+1)-i, g(m,n)-j)}{M(N-1)} |g(m,n+1)-g(m,n)|$$

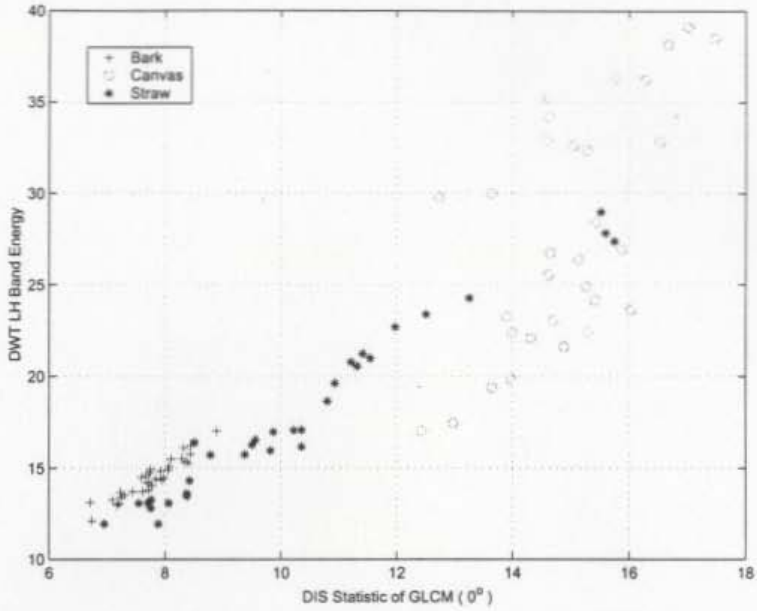


Figure 4.7: Correlation between DWT filtering and GLCM statistics

$$= \frac{\sum_{m=1}^{M}\sum_{n=1}^{N-1}|g(m,n+1)-g(m,n)|\sum_{i=1}^{G}\sum_{j=1}^{G}\delta(i-g(m,n+1),j-g(m,n))}{M(N-1)}$$

$$= \frac{\sum_{m=1}^{M}\sum_{n=1}^{N-1}|g(m,n+1)-g(m,n)|}{M(N-1)} \tag{4.2}$$

Thus the Dissimilarity of the GLCM is very close to the average magnitude of the first level LH band of the Haar wavelet transform. Similarly, the Contrast (CON) corresponds to the variance of Haar wavelet bands. However, there are still some differences. For example, the GLCM of $\theta = 0^o$ lacks a smoothing step in the vertical direction which is performed in obtaining the Haar wavelet LH band. This smoothing step can reduce the amount of jitter and noise, thus improving the feature extraction of that specific direction. Another difference is the downsampling step in the wavelet transform. Shift variance of band en-

ergy is introduced by downsampling. But this variance is expected to be small because of the random spatial arrangement of primitives in textures. The last difference is that the GLCM often has a quantization step for the original image while DWT does not. Quantizing the image will probably remove the subtle differences between two visually similiar textures. especially for SAR sea ice. Therefore Clausi [1] argued that it should be much safer and more consistent to utilize the full dynamic range. However on the other hand. the quantizations will also help to reduce noise. It is hard to say which effect will dominate. We can also apply quantizations to the wavelet coefficients. There has been a large amount of work in the literature on noise reduction using wavelet coefficient thresholding. which is in some sense similar to quantizing the wavelet coefficients. For more information on denoising with wavelet coefficients thresholding. the reader is referred to [42].

Because high correlations exist among GLCM statistics as stated in [19]. it seems not to be very advantageous to combine two or more GLCM statistics. Thus. the DWT classifier is comparable to that based on GLCM (for textures that do not have dominant information in diagonal directions). even if only a one level transform is applied. A similar comparison can be applied to the DWT-GLCM methods and the TSW. GLCM Disimilarity and Contrast calculated from wavelet bands can also be approximated with energies of the bands obtained by further decomposition. Therefore we can conclude that the same level of performance as that of the DWT-GLCM can be achieved by a non-dyadic wavelet energy method (like TSW). which has a significantly lower computational requirement than GLCM has. Please note that the DWT-GLCM excludes the relatively lower frequency part. while the TSW utilizes this infor-

mation.

4.4 SAR Texture Classification

In this section we apply the three kinds of feature extractions (DWT energy, DWT-GLCM, and TSW) to the logarithm processed SAR data set PLS (see Appendix B), which includes pack ice, rubble ice, landfast ice, and land (labeled as pac, rub, lfn, and lan respectively).

	pac	rub	lfn	lan
pac	27		5	
rub	4	27	1	
lfn		1	29	2
lan				32

(a)DWT energy. Accuracy: 89.84%

	pac	rub	lfn	lan
pac	30	2		
rub	1	31		
lfn	4	3	25	
lan			2	30

(b)TSW. Accuracy: 90.63%

	pac	rub	lfn	lan
pac	19	4	9	
rub	2	21	8	1
lfn	7	13	11	1
lan			4	28

(c)DWT-GLCM. Four directions and DIS. Accuracy: 61.72%

Figure 4.8: SAR texture classification results by DWT energy, TSW, and DWT-GLCM classifiers. For all three experiments the 3 level Daubechies wavelets are used

Experimental results are shown in Figure 4.8. Obviously, it seems not to be attractive to further explore spatial correlations within wavelet detail bands, as indicated by the poorer performance of DWT-GLCM. Nor is it of much improvement to have higher mid frequency resolutions using TSW. A possible explanation for these is that though SAR textures have many fine details cor-

responding to middle and high frequency bands. these details are more likely contributed or masked by speckle noise. A dyadic decomposition therefore seems to be adequate for characterizing these SAR texture frequencies.

It also seems that the DWT-GLCM's capability for separating mixed diagonal features (i.e. better directional selectivity) does not have much effect. SAR sea ice textures are typically treated as isotropic [20] [21]. However in our SAR texture image of Figure 2.3. the rubble ice has visually obvious diagonal structures at a very large scale.

To utilize these orientational features. no matter if in the horizontal. vertical. or diagonal directions. we have to consider carefully the effect of the orientational variations in the training and validation samples. As we have seen in our previous experiments. the orientational variations can cause significant degradation of the classification system. Discussions and solutions are presented in Chapter 5.

4.5 Application to Texture Segmentations

Texture segmentation can be achieved by either grouping regions with similar texture features or separating regions that have significant dissimilarities. Texture classification measures the similarities between a region and a set of predefined textures and therefore its application to segmentation belongs to the first kind. When utilizing the classifiers to segment textures. a window is usually slid across the image. A texture classification is performed on each window area. and the center pixel of the window is labeled as the corresponding texture

type.

The second category of approaches determines the boundaries to segment the image. They usually measure the texture dissimilarities between neighbouring areas. Here, gray level segmentation schemes are often applied, where the texture feature values are viewed as gray values.

It is hard to say which is better. For SAR sea ice, in cases where collisions of ices generate obvious boundaries, the second kind of method is perhaps better. But for mixture areas of different ice type resulting only in a fuzzy boundary, the first approach is preferred. Since our specific research interests are those young and new ices in different stages which are more or less alike, we apply texture classification to achieve the segmentation task.

Figure 4.9 gives the segmentation results using DWT energy, DWT-GLCM, and TSW feature extraction methods. An original mosaic image (see Figure 4.9(a)) has a dimension of 512 × 512 pixels, and is made up of 5 textures of bark, bubble, grass, leather and sand. Consistent with the classification experiment, TSW is a little better than DWT energy, while the DWT-GLCM method is much worse than the other two. Their segmentation results on the SAR image of Figure 2.3 are shown respectively in Figure 4.10, 4.11, and 4.12. As in PLS set, this large SAR image is pre-processed with a logarithm before segmentation. Because the SAR image is very large, the window is shifted by 8 pixels for classification to classification to save both computation time and storage space. From the segmentation result, we can obtain the same conclusion as with the Brodatz experiments.

The window size is an important parameter that has to be judiciously chosen. To utilize the large scale features, like those orientational features in rubble ice, a large window has to be selected. However, using a large window increases the likelihood that the window contains more than one texture type, resulting in more classification errors and poor localization of segmentation boundaries. For this reason keeping the window size as small as possible is desirable. A possible solution to this contradiction is a coarse to fine approach. The image is first classified pixel by pixel at the coarsest scale, making use of large scale features. For boundary areas or clutters lacking large scale features, the pixel classification is deferred and finer scale classification is performed. This process is iterated until the finest scale is reached. But this kind of approach also has a problem. Small classification errors in the coarser scales will be propagated to finer scales, generating large blocks of errors. Unless we can guarantee that the classifications on the coarse scales are close to one hundred percent correct, or we do not use deterministic classifications, the problem seems to be inevitable. Due to the difficulty inherent in the approach, further investigation is out of the scope of the thesis and we leave it for possible future work.

4.6 Summary

In this chapter we have introduced several feature extraction methods based on the wavelet transform, and made a comparative study of them. Among the three approaches, the TSW is the best but is impractical because of its high computational complexity. We have shown that for SAR sea ice, the simple DWT energy is a good choice as it has performance close to that of the best

coming from the TSW.

We also noticed that directional variations will cause a lot of trouble in the design of a robust classification system. One of textures prominent in our research, namely rubble ice, often exhibits obvious oriented structures. Although the research presented so far has not explored the large scale features of rubble ice (the experiments are carried out with 3 level wavelet transform, of which the corresponding scales are all much smaller than that of the oriented structures), it is ideal if we can design a rotation invariant classification system to make use of the oriented features for the identification of rubble ice. Chapter 5 deals with the topic of designing a rotation invariant classification system.

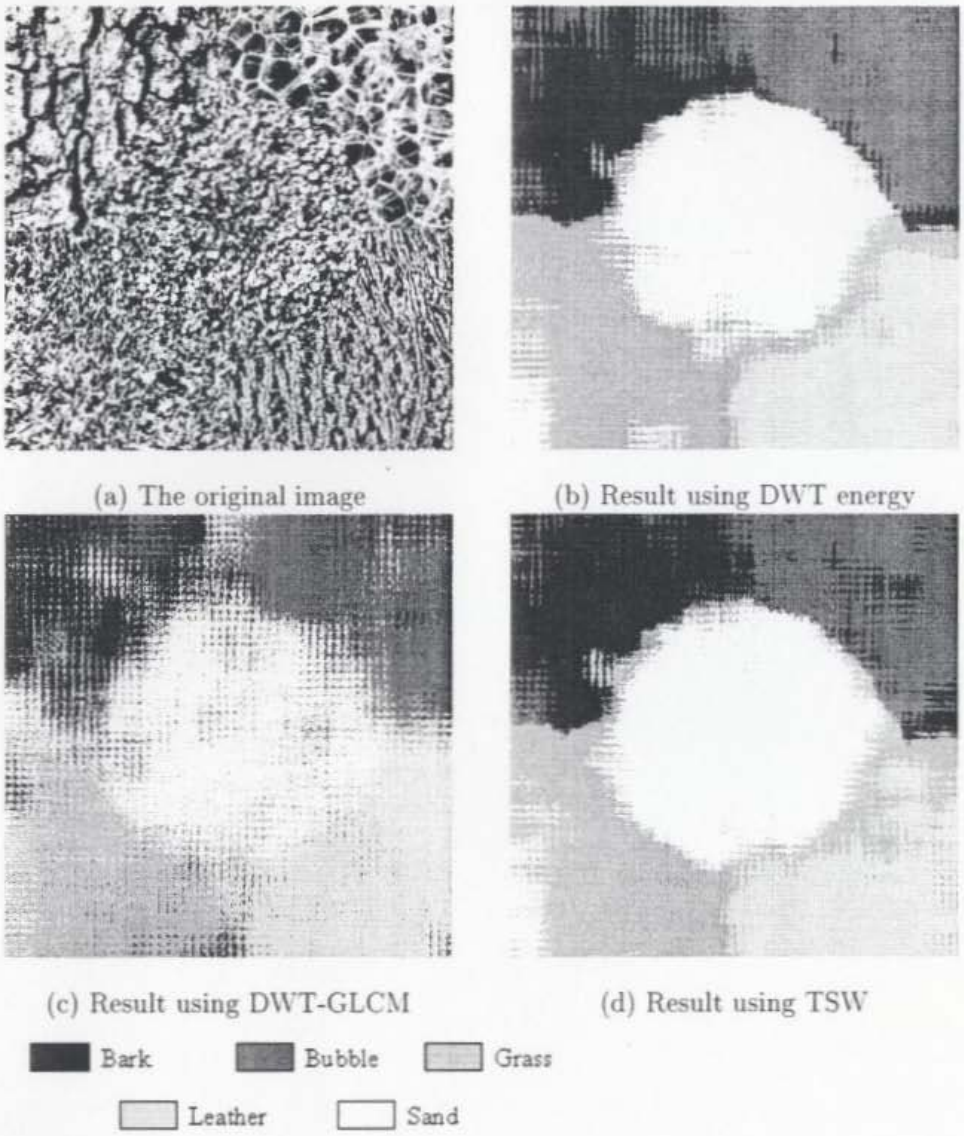


Figure 4.9: Texture segmentation using DWT energy, DWT-GLCM, and TSW methods. A mosaic image of 5 textures (bark, bubble, grass, leather, sand) is used. The window is of 64×64 pixels size and shifted by 1 pixel.

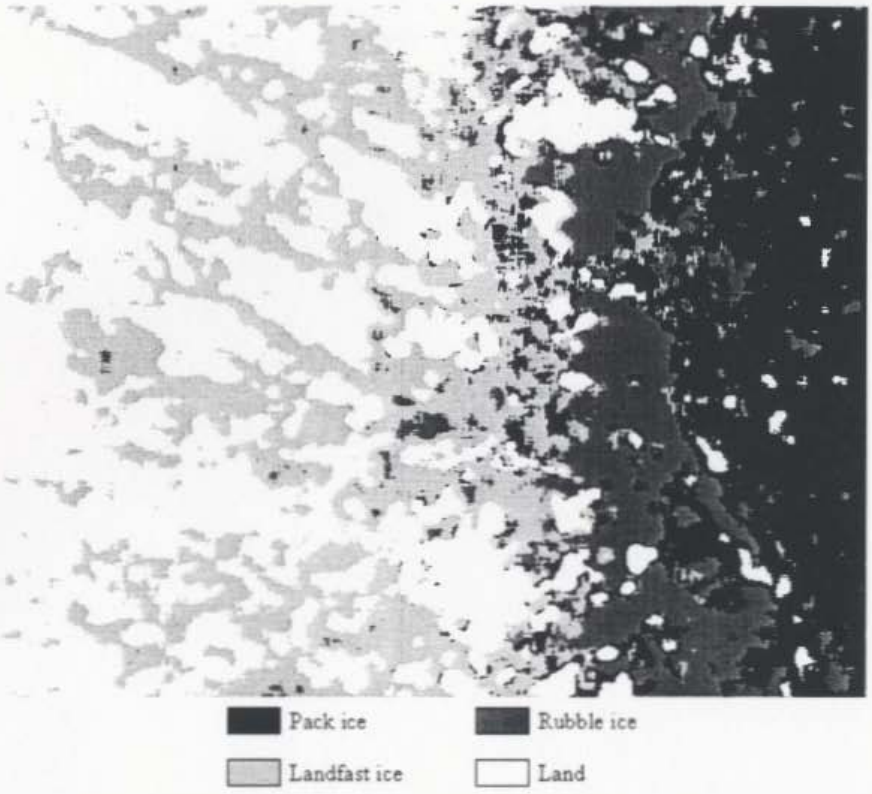


Figure 4.10: SAR Texture segmentation using DWT energies. Original image pre-processed with logarithm. The window is of 128×128 pixels size and shifted by 8 pixels.

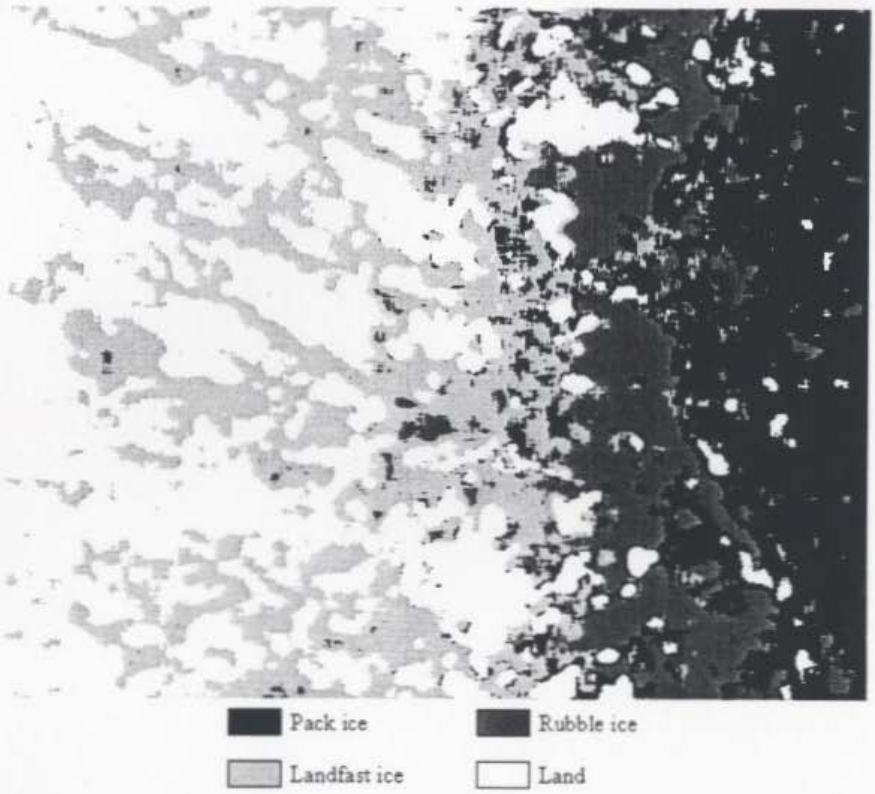


Figure 4.11: SAR Texture segmentation using the TSW. Original image pre-processed with logarithm. The window is of 128 × 128 pixels size and shifted by 8 pixels.

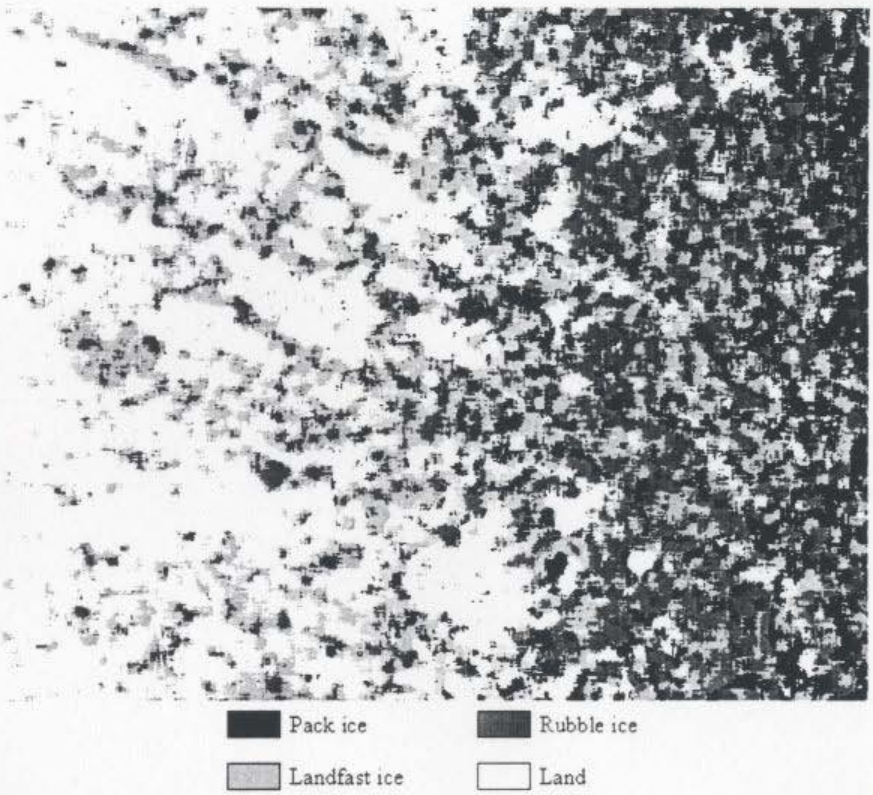


Figure 4.12: SAR Texture segmentation using the DWT-GLCM. Original image pre-processed with logarithm. The window is of 128×128 pixels size and shifted by 8 pixels.

Chapter 5

Rotation Invariant Classification

5.1 Introduction

Most existing classification methods assume that the training and validation samples are carefully selected to have the same orientation and scales. However these two factors, especially the orientations, are often unpredictable and therefore can make the classifiers not robust at all. For our research target of SAR sea ice, oriented features, whose orientations are determined by the coastline, wind, currents and other environment factors, often exist. The ice against the coast in Figure 2.3 is an example.

Rotation invariant texture classification can be obtained by eliminating the orientation information. For example, averaging the GLCMs of different θ are adopted by [20] and [21] in their experiments. Using the energies of Laplacian Pyramid [36] bands instead of wavelet bands as features is another example. A common drawback of these algorithms is that the loss of orientation information will probably cause the classifier to perform poorly when it is trying to classify

oriented textures.

A promising method using oriented Laplacian pyramid filtering was proposed in [37]. Rotation invariance features are extracted with DFT encoding of the output of the filtering. Since our method is motivated by this work, a brief introduction is given in the following section on it.

5.2 Invariant Texture Recognition Using Steerable Laplacian Pyramid

5.2.1 Algorithm

The idea of using a Laplacian Pyramid to extract components of different spatial frequency bands for image analysis can be found in [36]. The original image is decomposed into sets of octave-spaced lowpass and bandpass components by the following steps:

1. The given image is lowpass filtered first. A bandpass filtered version of the input image can then be obtained by the subtraction of the input image and the lowpass filtered one.

2. The lowpass filtered version of the input image is downsampled. Thus a lower resolution image is obtained, to which the above procedure is applied again.

The image pyramid formed is not orientationally tuned. In order to extract orientational components, each level of the pyramid is modulated with a set of oriented complex sinusoids, followed by another lowpass filtering (LPF) opera-

tion using a separable filter, and then downsampling, as defined in equation 5.1 [37].

$$O_{n\alpha} = LPF\left[e^{j\vec{k}_\alpha \cdot \vec{r}} L_n[x, y]\right]. \tag{5.1}$$

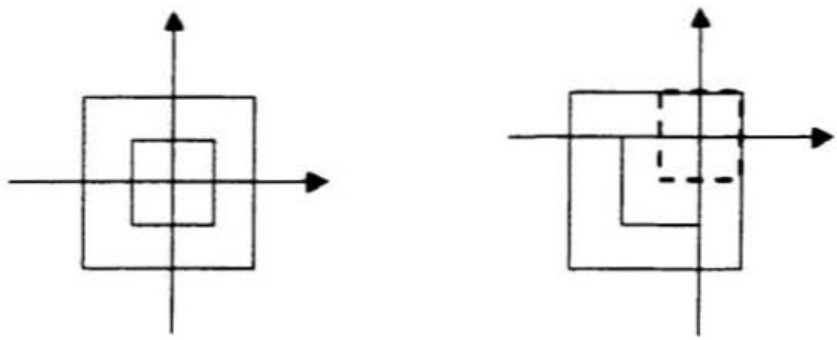
where $O_{n\alpha}$ is the oriented image at scale n and orientation α. L_n is the Laplacian image at scale n. $\vec{r} = x\vec{i} + y\vec{j}$ (x and y are the spatial coordinates of the Laplacian image). $\vec{k}_\alpha = (\pi/2)[\cos\theta_\alpha \vec{i} + \sin\theta_\alpha \vec{j}]$ and $\theta_\alpha = (\pi/N)(\alpha - 1)$, $(\alpha = 1 \ldots N)$.

A feature curve (per scale) can be defined across orientation space, as the textures' response to the oriented filters above in the 360° space. Rotation of the input texture will cause a shift of the feature curve across the orientation axis. Rotation invariant features can then be extracted with the Fourier Transform, where magnitudes of the transform outputs are used as features, since the shift on the orientation axis will only influence the phase.

It is impractical to extract image components of a continual set of orientations on the interval $[0, 2\pi)$. A discrete set of orientations is used instead. In [37], the whole orientation space was spanned by interpolations of the selected oriented kernels. The authors proved that a set of eight orientations with 45° bandwidth is sufficient to span the 360° of orientation space with more than 99% accuracy. This spanning, however, may not be necessary if only limited dominant orientations exist per scale for the given texture. For example, 4 kernels spaced 45° apart is sufficient for the representation of a single dominant orientation.

5.2.2 Discussions

A problem of [37]'s method is that the oriented Laplacian pyramid is not efficient in representing components of different orientations. A lot of redundancy exists among orientation images of the same scale. This is clear if we look in the spatial-frequency domain. Figure 5.1 shows the spatial-frequency representation of a one level Laplacian Pyramid. The bandpass filtered version of the input image corresponds to the shaded area in (a). Suppose we want to extract components of 45°. After modulation it is shifted down and left by $\pi/2$ so that the upper right part of the original image is now in the center as shown in (b). and will be extracted when lowpass filtered. Therefore. the ouputs for orientation $0°, 45°, 90°$ and $135°$ correspond to the four parts in figure 5.2. Significant redundancy exists between neighbouring orientations. That's a possible reason for the high accuracy in spanning the orientation space with only a small discrete set of oriented kernels.



(a) Laplacian Pyramid in Spatial-frequency Domain

(b) Modulation and Lowpass Filtering

Figure 5.1: Oriented Laplacian Pyramid in Spatial-frequency Domain

A more accurate orientation feature extraction method is Gabor filtering.

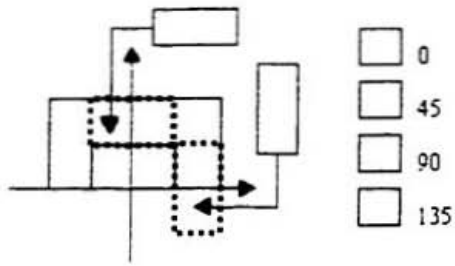


Figure 5.2: Components in 0°, 45°, 90°, 135° of Oriented Laplacian Pyramid

but it is computationally costly. The Common DWT is incapable of this task because orientations of 45° and 135° are mixed together as shown in figure 5.3. To extract these two orientation parts separately, the ability of separate positive or negative frequency filtering is required. That is, the wavelet filters have to be constructed by complex coefficients instead of real coefficients. This motivated our choice to use the complex wavelet transform for oriented feature extraction.

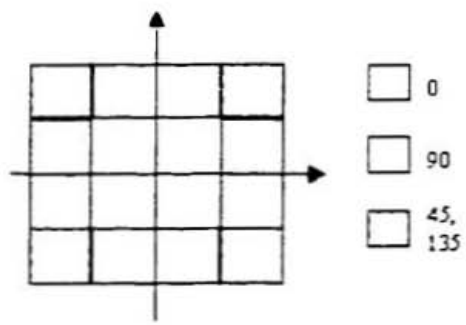


Figure 5.3: Components in Horizontal,Vertical,Diagonal Direction of 1-level DWT

5.3 Complex Wavelet Transform

5.3.1 Definition

The Complex Wavelet Transform (CWT) was proposed to overcome the two disadvantages of real wavelet transform: lack of shift invariance and poor directional selectivity, which has been mentioned in the previous section. Lack of shift invariance can be solved by designing the complex filters in such a way that the magnitudes of their responses vary slowly with input shift - only the phases vary rapidly. Good directional selectivity can be achieved if only either the positive or negative frequencies are filtered through. Therefore, all of the complex filters should emphasise positive frequencies and reject negative frequencies or vice-versa. Detail explanations can be found in [38].

The Complex wavelet transform has the same structure as that of the real wavelet transform except that the filter coefficients are complex instead of real, as described in figure 5.4. The inputs of all level filters are made up of two parts (real and imaginary), except for the filters of the first level whose input is the real signal x. Since each coefficient contains a real and imaginary part, a 2:1 redundancy is introduced.

Like in DWT, separate filtering in row and column directions is performed to extend CWT to two dimensions. Two adjacent quadrants of the spectrum are needed to fully represent the original image without any information loss (note that opposite quadrants are conjugate pairs). Therefore a 4:1 redundancy is required, which is achieved by additional filtering with complex conjugates of either the row or column filters. The directional selectivity therefore is oriented

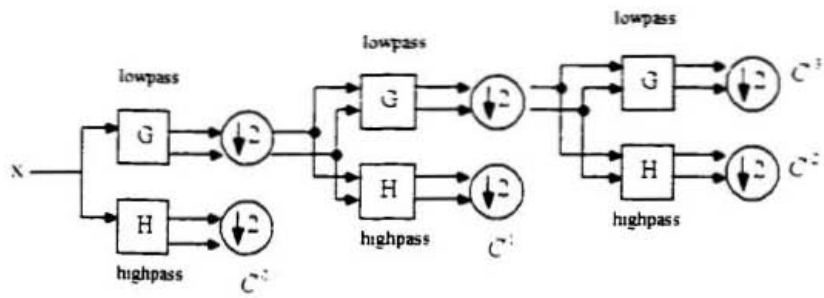


Figure 5.4: Three levels of the complex wavelet transform for a real 1-D signal x. The wavelet coefficients C consist of the real and imaginary parts.

at $\pm 15°$, $\pm 45°$ and $\pm 75°$, as shown in Figure 5.5. Since the human visual system (HVS) is known to be made up of a set of 'filters' that have bandwidth of about one octave and are tuned to directions no more than 30° apart [1], the complex wavelet transform agrees with the HVS well.

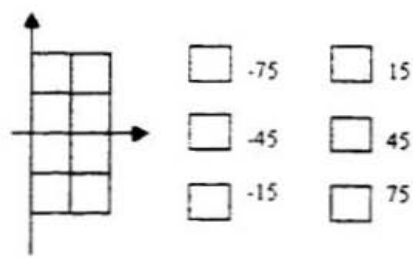


Figure 5.5: Different Directional Components with 1-level CWT

5.3.2 Dual-Tree Complex Wavelet Transform (DTCWT)

Calculating and manipulating complex data with complex filters is really a troublesome task. A method named Dual-tree CWT (DTCWT) was proposed

in [39] to approximate complex wavelet transform. Figure 3.6 gives the one dimension diagram of the algorithm. It employs a dual tree of wavelet filters to obtain the real and imaginary parts of complex wavelet coefficients. The output of the upper tree provides the real part of the complex coefficients while the lower tree generates the imaginary part. For details the readers are referred to [39].

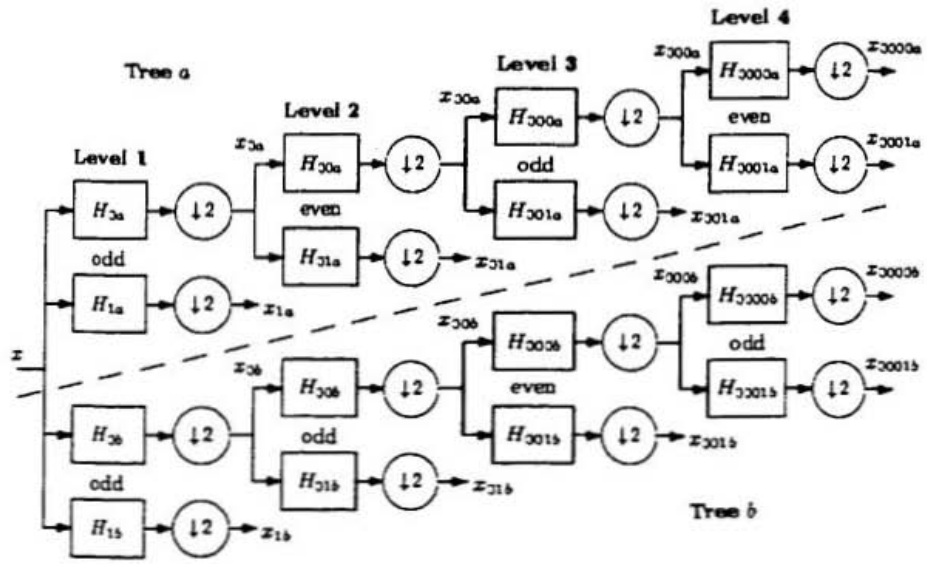


Figure 3.6: Dual tree of filters for the complex wavelet transform (taken from [39])

5.4 Rotation Invariant Classification Using Complex Wavelet Transform

5.4.1 Classifications with DTCWT

Before applying Discrete Fourier Transform (DFT) encoding on the DTCWT output to get an rotation invariant representation for the features, we will first do an experiment on the classification with DTCWT alone. The test was performed on the MBS set. The selected wavelet kernel is the 10-order QShift filter provided in [40], which gives improved versions of the filters presented in [39]. Like the typical Daubechies wavelet filters, the QShift filters are orthonormal and not symmetric. An encouraging result has been obtained, and is shown in Table 5.1. The separation of 45° and 135° directions is the reason for the improvement in the overall performance. This is clearly demonstrated by the impressive accuracy obtained for straw texture, which has important orientational features existing around the 45° direction.

The classification accuracies of SAR sea ice have not improved. Orientation features obvious in rubble ice image seems have not played a role in the process. Those orientation features are of so large a scale that even the coarsest of the three decompsition levels is insufficient to capture them. Further decomposition beyond three levels is tested on a rubble ice image of a larger dimension of 512×512, and the result is shown in Figure 5.7(a). It seems that in the 5th and 6th decomposition levels, the energies are skewed, having maximum value in the directions of 75 degree. Since the 75° band measures the variations in that direction, it actually corresponds to the linear features stretching along $-15°$ apparent in the rubble ice image. The 5th and 6th levels represent scales be-

	bar	bub	can	gra	lea	san	sto	str	woo	pac	rub	lan
bar	30	2										
bub	2	30										
can			32									
gra	2			26		4						
lea					32							
san						32						
sto							32					
str	1				1			30				
woo		1							30			1
pac										28	4	
rub										15	17	
lan												32

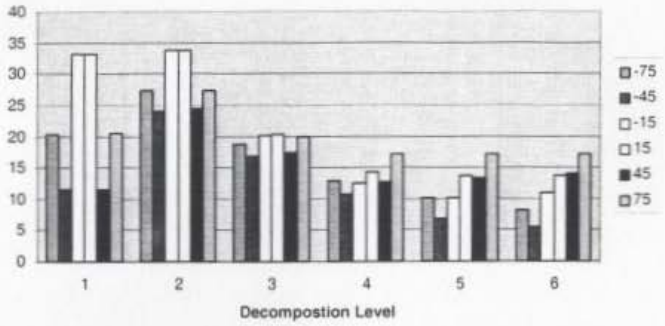
Table 5.1: Classification result by 3-level QShift(10) DTCWT classifier. Overall accuracy: 91.41%

tween 16 pixels and 32 pixels (i.e. $200m$ and $400m$). For comparision, subband energies of pack ice and land are presented in figure 5.7(b) and (c).

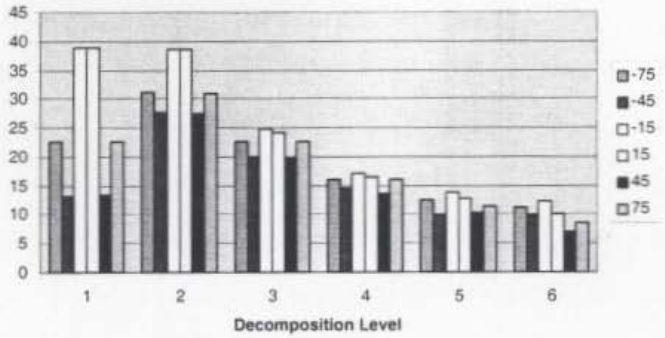
The figure also indicated that for SAR sea ice the first two decomposition levels have relatively higher energies. This might contribute to the argument in [20] that length of four pixels was appropriate for GLCM statistic calculation. However for high resolution levels, the speckle noise will influence significantly [41].

5.4.2 Rotation Invariant Classification with DFT Encoding on CWT

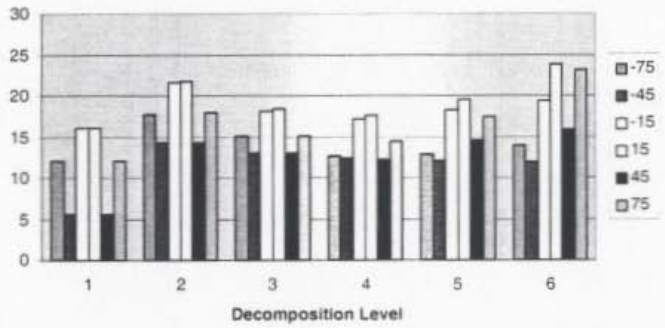
As explained earlier when introducing the method of [37], Discrete Fourier Transform (DFT) encoding provides a feasible way to obtain rotation invariant representations for oriented features. We applied DFT encoding on the



(a) Rubble ice



(b) Pack ice



(c) Land

Figure 5.7: Mean Energies of Different Orientations on Different Scales

six complex wavelet bands per scale and we call this method DFT-DTCWT. The magnitudes of the DFT coefficients obtained can then be used for distance calculations, while the phases indicate the rotation angle of the input image. Obviously, neither training nor test samples need to have the same orientation.

In our experiments the training samples are of the same orientation within each class, but the test samples are rotated with different angles. Figure 5.8 gives performances of a non rotation-invariant classifier (DTCWT) and two rotation-invariant classifiers (DFT-DTCWT and classifier based on the energy of Laplacian Pyramid bands), with respect to rotation angle of the test sample textures. The non rotation-invariant classifier (DTCWT) has a strong decline in performance when rotation angle increases, resulting in poor performances beyond 20°, while the rotation-invariant classifiers are much more stable over the angle axis. For the Laplacian Pyramid classifier, the orientation features are all merged together within a single non-oriented band per scale. Therefore, its overall classification performance is much lower than that of DFT-DTCWT, which utilized seperate orientation features in the feature extraction step.

Compared to the non rotation-invariant DTCWT classifier, the DFT-DTCWT classifier is slightly worse if the input textures are not rotated. This is because we characterize the feature curves by only the magnitude of the DFT coefficients without phase. As suggested by [37], this could be improved by preserving the relative phase information across scale.

We noted that the performance curve of the DFT-DTCWT classifier reached its minimum at 40°. The reason for that is that our extraction of different

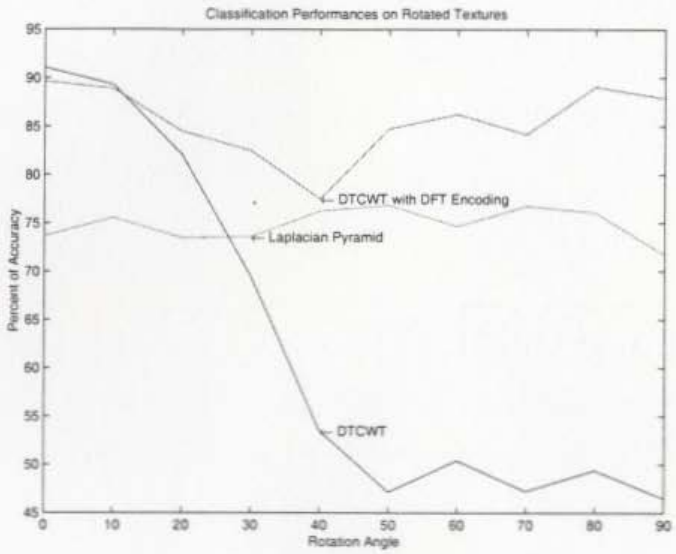


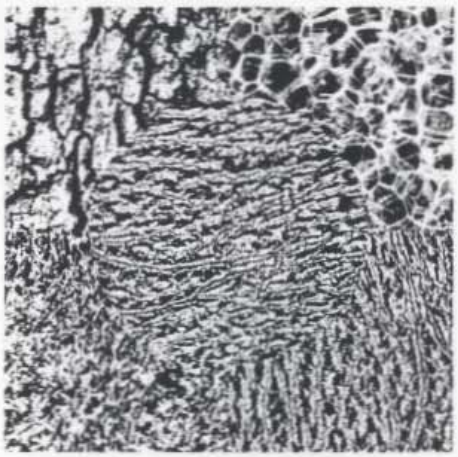
Figure 5.8: Overall classification performances with respect to rotation angle of the input textures. The decomposition levels are all 3. The wavelet kernel used in DTCWT and DFT-DTCWT is QShift(10). The data set tested is MBS.

orientation features within the same decompostion level is, strictly speaking, not on the same scale. This is clear if we look at Figure 5.5. The components namely on the same scale do not have equal distances to the origin point. The sampling distance of the image grid in the diagonal direction is $\sqrt{2}$ times of that in the horizontal and vertical directions, and thus the scale of the diagonal features extracted are actually $\sqrt{2}$ times that of horizontal and vertical directions within the same decompostion level. Possible solutions could be found in non-separable wavelets [45]. However, considering its low computation requirement and good performance, DFT-DTCWT is an acceptable choice.

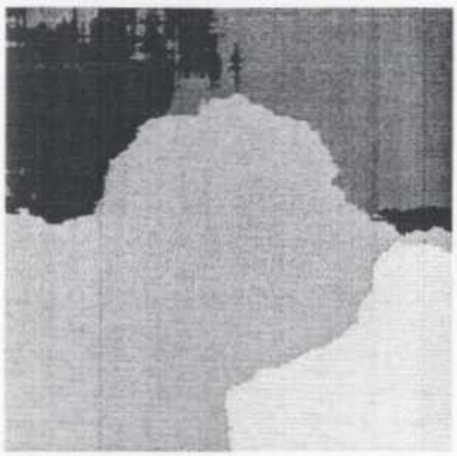
5.4.3 Segmentation Results

Again we apply the DTCWT and the DFT-DTCWT methods to texture segmentations for a visual evaluation. This time the central circular area of the Brodatz mosaic image is replaced with a rotated leather texture (see Figure 5.9(a)). We train the two classifiers with samples that have particular single orientations. The orientation of the leather training samples is consistent with that of the lower right part of the mosaic image.

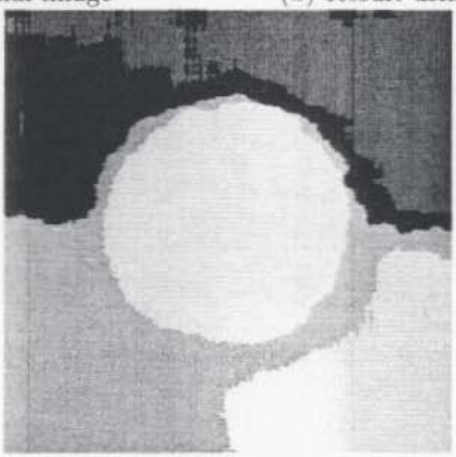
Segmentation results are shown in Figure 5.9. Obviously, the DFT-DTCWT identified the leather textures with high accuray despite the rotation. This justifies our using DFT-DTCWT methods for rotation invariant classifications and segmentations. Compared to the segmentation results in Figure 4.9, both DTCWT and DFT-DTCWT have poorer performances in locating boundaries, which is the result of using higher order filters for the complex wavelet transform.



(a) The original image



(b) Result using DTCWT energy



(c) Result using DFT-DTCWT energy

Figure 5.9: Texture segmentation using DTCWT energy and DFT-DTCWT methods. A mosaic image of 4 textures (bark, bubble, grass, leather(rotated), leather) is used.

Chapter 6

Fuzzy Classification and its Application to Ship Navigation

6.1 Introduction

So far all our efforts are directed to finding a good feature extraction method for deterministic classification (i.e. crisp classification), the task of which is to assign class labels for classification objects. Sometimes the class labels are not appropriate or sufficient representations. Users may want to know "how much the object is like a given class type". Fuzzy classification has therefore been proposed to deal with these degrees of similarities. Because our research targets of young and new ice are alike and continuous stages exist between the ice types, fuzzy classification is more appropriate and is covered in this chapter.

6.2 Definition of Fuzzy Classification

Let X be a vector in an n-dimensional real space $\Re^n$ (the feature space), and let k be the total number of classes. A fuzzy classification is the mapping

$$X \rightarrow \{\mu_1, \mu_2, \ldots, \mu_k\}$$

where $\mu_i \in [0, 1]$ for all $1 \leq i \leq k$ and $\sum_{i=1}^{k} \mu_i = 1$. [46]. Obviously a crisp classification is a special case of the fuzzy classifications where for some j, $\mu_j = 1$ and $\mu_i = 0$ if $i \neq j$.

Fuzzy classification is usually more complex than crisp classification. There is often difficulty in defining and measuring the degrees of similarities. And therefore the classifier parameters are hard to estimate and the performances are hard to evaluate.

6.3 Wavelet Entropy - A Fuzzy Feature

In this section, we propose a new feature extraction approach. The method is also based on the wavelet transform and uses a new wavelet signature called *wavelet entropy*. An interesting aspect of this method is that it can provide a fuzzy feature set for the purpose of fuzzy classification.

6.3.1 Wavelet Entropy Signature

In Chapter 4 we have introduced work by Wouwer et al. [43] using wavelet histogram model parameters as features. To test the validity of the exponential model in SAR textures, we investigated the histograms of the pack ice, rubble ice and land samples. The samples choosen are the large images that have been

split to generate the PLS set. Of course, they have been pre-processed with a logarithm. The good fit of the exponential curves to the histograms shown in Figure 6.1, 6.2, and 6.3 indicated that this kind of model provides an acceptable approximation of the histograms and can be useful for characterizing the SAR textures.

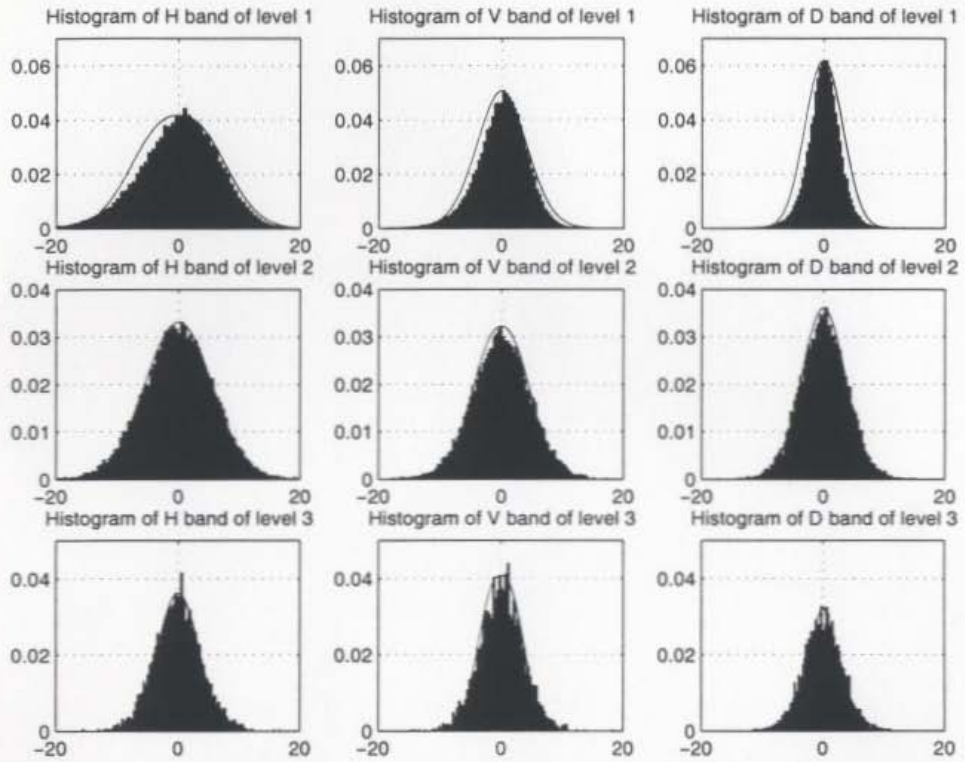


Figure 6.1: Histogram of DWT detail bands of pack ice. The wavelet kernel used is Daubechies(6). 'H', 'V', and 'D' represents the horizontal, vertical, and Diagonal bands respectively. Level 1 corresponds to the finest scale while level 3 corresponds to the coarsest.

The exponential models work well in level 2, while for the other two levels

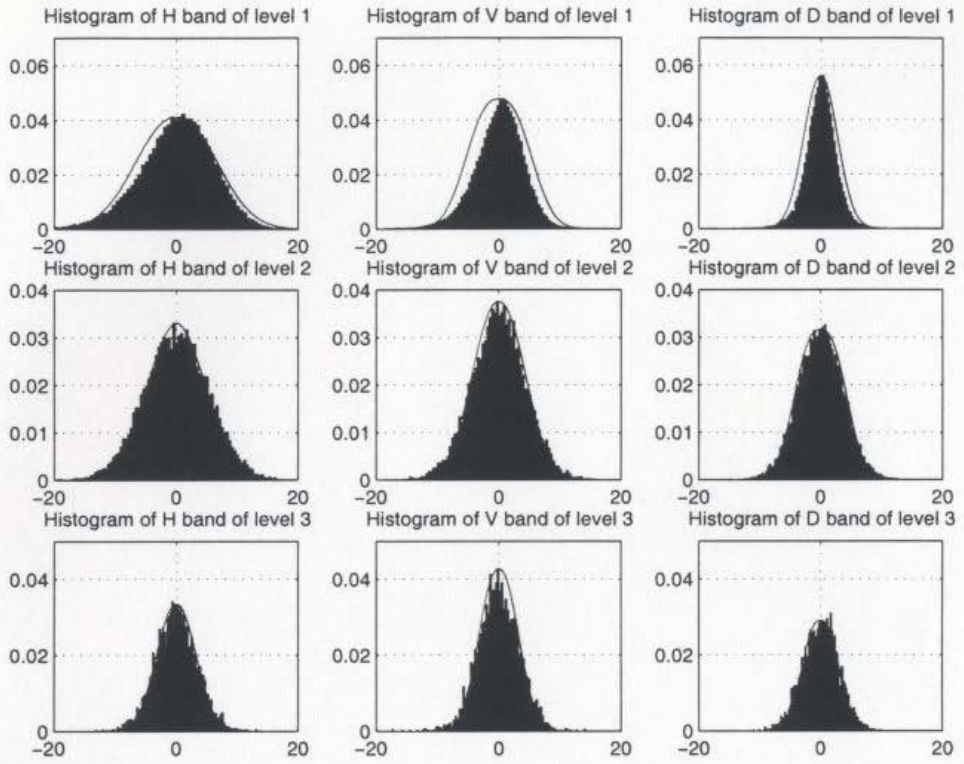


Figure 6.2: Histogram of DWT detail bands of rubble ice. The wavelet kernel used is Daubechies(6). 'H', 'V', and 'D' represents the horizontal, vertical, and Diagonal bands respectively. Level 1 corresponds to the finest scale while level 3 corresponds to the coarsest.

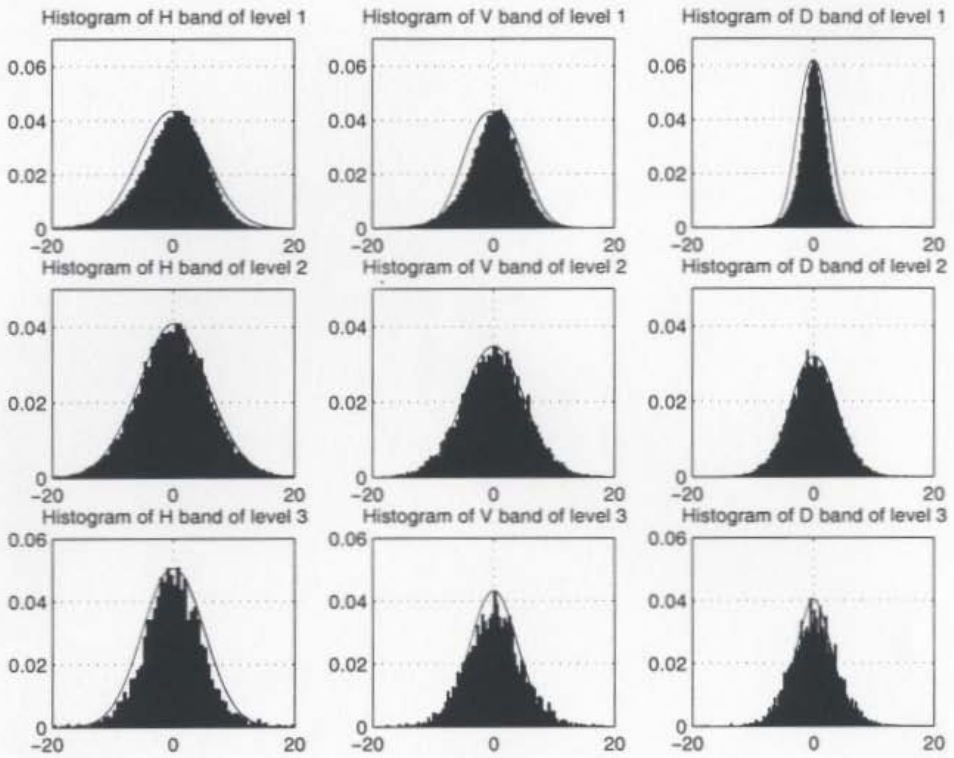


Figure 6.3: Histogram of DWT detail bands of land. The wavelet kernel used is Daubechies(6). 'H', 'V', and 'D' represents the horizontal, vertical, and Diagonal bands respectively. Level 1 corresponds to the finest scale while level 3 corresponds to the coarsest.

errors are obvious. It is probably because level 1 is more influenced by speckle noise and level 3 does not have a large enough number of pixels for the calculation of histograms. Comparing the histograms of pack ice and rubble ice, we found that the most significant differences exist in level 2. Interestingly, this agrees well with the conclusion of Nystuen [20] in their GLCM experiment that the length of four pixels was appropriate for feature extraction.

Now consider the problem of classifying a texture image to one of the K class types $(T_1, T_2, \ldots, T_K)$, using a wavelet band (B) obtained with the original image. The likelihood of the image belonging to class i is given by

$$P(T_i|B_1, B_2, \ldots, B_N)$$

where N is the total count of pixels in B and $B_1, B_2, \ldots, B_N$ are the pixel values. According to Bayes formula:

$$P(T_i|B_1, B_2, \ldots, B_N) = \frac{P(B_1, B_2, \ldots, B_N|T_i)P(T_i)}{P(B_1, B_2, \ldots, B_N)}$$

The *class a priori probability* $p(T_i)$ is usually unknown, so we assume they are equal for all classes. Therefore $P(T_i|B_1, B_2, \ldots, B_N)$ can be easily calculated by normalizing $P(B_1, B_2, \ldots, B_N|T_i)$.

To obtain $P(B_1, B_2, \ldots, B_N|T_i)$, we can simply use the following formula:

$$P(B_1, B_2, \ldots, B_N|T_i) = \prod_{n=1}^{N} P(B_n|T_i) \tag{6.1}$$

where $P(B_n|T_i)$ is the corresponding normalized histogram value. Here we assume $B_1, B_2, \ldots, B_N$ are independent to each other. The computational burden of a product can be eased by applying a logarithm to Equation 6.1, so that the

product is changed to sum:

$$log[P(B_1, B_2, \ldots, B_N|T_i)] = \sum_{n=1}^{N} log[P(B_n|T_i)]$$

Obviously, the value of this probability respresentation is largely dependent upon the dimension of the image. To remove the dimension factor, we use the average instead of the sum. Suppose G is the set of values in B after quantization. We can get

$$\begin{aligned}
-\frac{log[P(B_1, B_2, \ldots, B_N|T_i)]}{N} &= -\frac{\sum_{n=1}^{N} log[P(B_n|T_i)]}{N} \\
&= -\frac{\sum_{n=1}^{N} log[P(B_n|T_i)] \sum_{g \in G} \delta(g - B_n)}{N} \\
&= -\sum_{g \in G} \sum_{n=1}^{N} \frac{\delta(B_n - g)}{N} log[P(B_n|T_i)] \\
&= -\sum_{g \in G} log[P(g|T_i)] \sum_{n=1}^{N} \frac{\delta(B_n - g)}{N} \\
&= \sum_{g \in G} -log[P(g|T_i)]P(g) \qquad (6.2)
\end{aligned}$$

When there is an infinitely large number of samples (i.e. $N \to \infty$) and the quantization bin becomes infinitely small, the sum in formula (6.2) becomes the integral

$$\int_{-\infty}^{\infty} -log[\rho(g|T_i)]\rho(g)dg$$

where the $\rho(.)$ is the probability density function, and $\rho(g|T_i)$ corresponds to the histogram model function of class type i.

The crisp classification problem can then be viewed as trying to label the texture image with the class type whose output of formula (6.2) is minimum and therefore closest to the entropy of the image. For this reason we can call it the entropy signature.

Since the wavelet entropy signature represents the memberships in some degree for the given scales and directions, these features can be useful in fuzzy classification or segmentation. They can be linearly combined to obtain the final memberships.

6.3.2 Crisp Classification and Segmentation Results

As usual, we perform the crisp classification with wavelet entropy signatures on the two data sets: MBS and PLS. Instead of using the histogram model, we use the raw histogram for simplicity. Wavelet entropy values of different scales and directions are simply summed. Classification results are shown respectively in Table 6.1 and 6.2.

	bar	bub	can	gra	lea	san	sto	str	woo	pac	rub	lan
bar	31	1										
bub		32										
can			32									
gra	3			19	3	7						
lea					32							
san						32						
sto							32					
str	1	1		4	9	2		15				
woo	2	1							29			
pac										30	2	
rub										16	16	
lan												32

Table 6.1: Classification result by 3-level Daubechies(6) DWT Entropy classifier on MBS set. Overall accuracy: 86.4583%

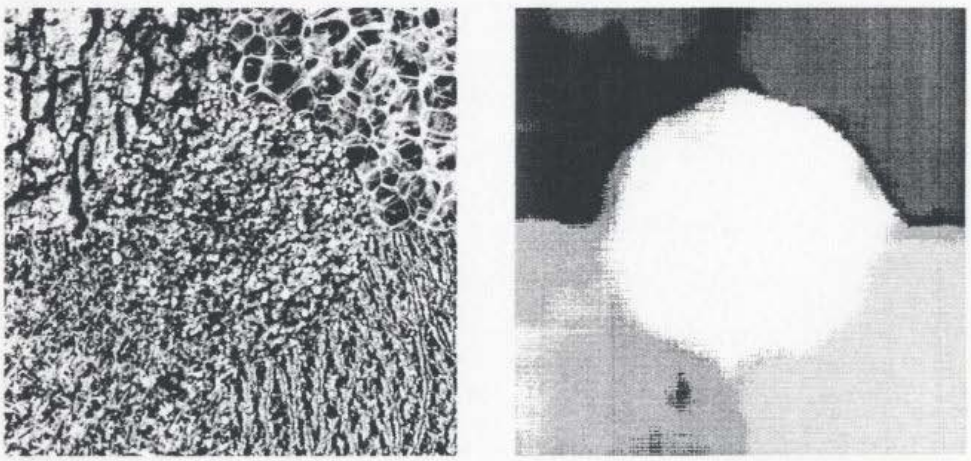
The DWT entropy features have lower performances than DWT energy features for both of the data sets. This is probably because we have used raw histograms which have some outliers that should have been excluded. And

	pac	rub	lfn	lan
pac	25	2	5	
rub	2	28	2	
lfn	5	6	21	
lan		1	2	29

Table 6.2: Classification result by 3-level Daubechies(6) DWT Entropy classifier on PLS set. Overall accuracy: 80.4688%

compared to the energy, the entropy signatures are sensitive to the anomaly. Therefore for textures that have high in-class variations, the performances of the DWT entropy features can probably be worse.

Segmentation results are given in Figure 6.4 and Figure 6.5. Interestingly, although the classification performance of DWT entropy method is a little worse than that of DWT energy, the segmentation result seems to be the reverse. The DWT entropy method gives more uniform regions and better boundaries.



(a) The original image (b) Result using DWT entropy

Figure 6.4: Texture segmentation using DWT entropy methods. A mosaic image of 5 textures (bark, bubble, grass, leather, sand) is used

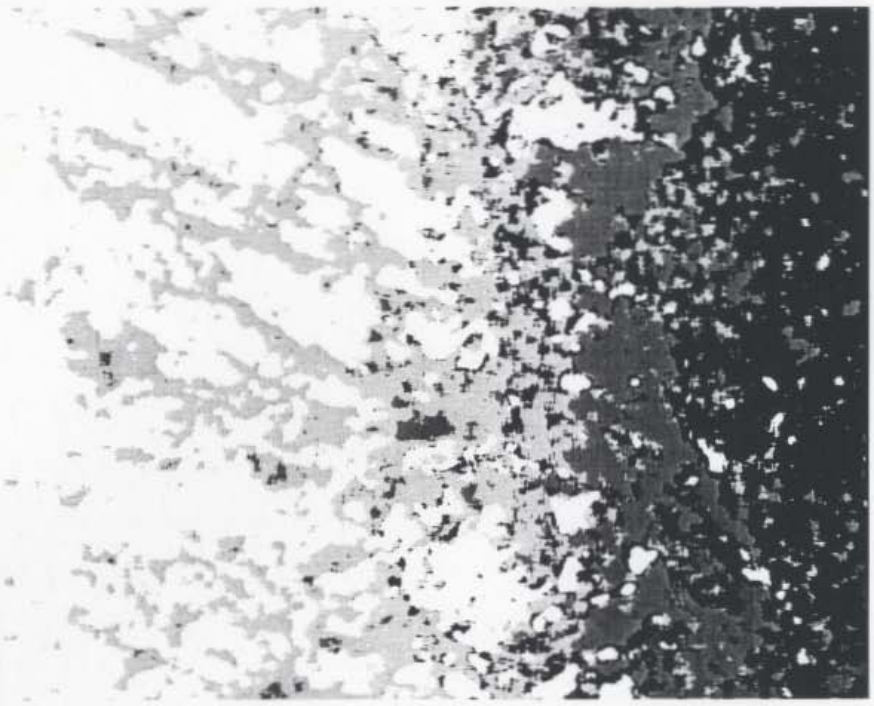


Figure 6.5: SAR Texture segmentation using DWT entropy methods. The original image is Figure 2.3

6.3.3 Fuzzy classification with DWT entropy

Using wavelet entropy signatures, we have obtained logarithm representations for memberships of each scale and direction. Their weighted sum can represent the overall memberships. To estimate the weights, the least square error method could be used. Because of the difficulty in acquiring knowledge of the sea ice information on the corresponding sites, we have not performed precise fuzzy classification and segmentation experiments for SAR sea ice textures. We leave it for future work.

6.4 Application to Ship Navigation

6.4.1 Definition of the Problem

Once we have obtained the membership maps, it is possible to calculate the optimal path for ships going through sea ice such that the ship has the minimum probability of being beset, that is unable to proceed.

First, we need to define a pass-through probability model for each texture type existing in Figure 2.3, as listed in Table 6.3. In practice, these numbers depend on the characteristics of a particular ship. The values in Table 6.3 are for illustration purpose only. And suppose we have obtained the membership maps such as the examples shown in Figure 6.6. The pass-through probability of a given pixel is calculated by the formula:

$$P_p(x, y) = \sum_{i=1}^{4} P_i(x, y) P_p(i)$$

where (x, y) is the position of the given pixel. $P_i, i = 1, \ldots, 4$ are the memberships of the four texture types respectively, and $P_p(i)$ is the pass-through probability of texture type i.

A pass-through probability map can hence be computed from the pass-through probability model and membership maps. Having processed the pass-through probability map with a logarithm, the ship navigation problem is now transformed to a typical **shortest path** problem, for which the path length between two neighbouring pixels is measured by the average of the two corresponding map values.

Pack ice	0.7
Rubble ice	0.4
Landfast ice	0.9
Land	0

Table 6.3: Pass-through probabilities of different SAR texture types

6.4.2 Shortest Path Routing

The shortest path routing problem has been extensively explored in the data networks field. We here use a well known routing method called Dijkstra's algorithm [47]. The algorithm maintains for each node (in our case a node is a pixel) its shortest distance from the source node, and also a pointer to the previous node (for tracing the route) along the shortest path. Initially no paths are known, so the distance values of all nodes are infinity. Then the algorithm updates these distance values as it proceeds, from the neighbouring nodes of the source node. The update rule is as follows:

- If a node finds that any of its neighboring nodes has an update, it will check if itself needs an update.
- The check compares the current distance value with the sum of the updated neighbour's distance and the path length between these two nodes. If the current distance is smaller, no update is performed. Otherwise, the distance of the node is updated with the sum value and the pointer is directed to the corresponding neighbour node.

For our experiment, the update is iterated until there is no further update for all the pixels. From the result, we can obtain the shortest path from a given source pixel to any pixel in the image.

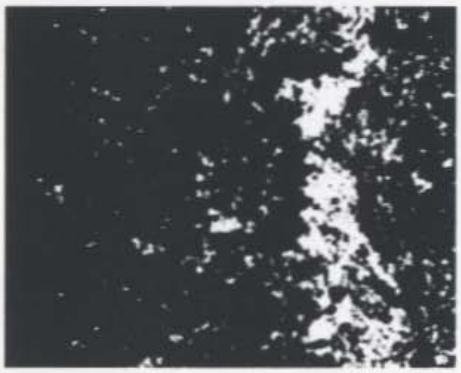
Figure 6.7 gives an example of ship navigation between two sites. The red, green and blue channels of the pseudo color image are respectively the membership maps of pack ice, rubble ice and landfast ice, which have already been shown in Figure 6.6. The path obtained has the minimum beset probability, as we can see intuitively from the figure that the path tries to circumvent rubble ice and select landfast ice sites wherever possible.

6.4.3 Reduce the computation

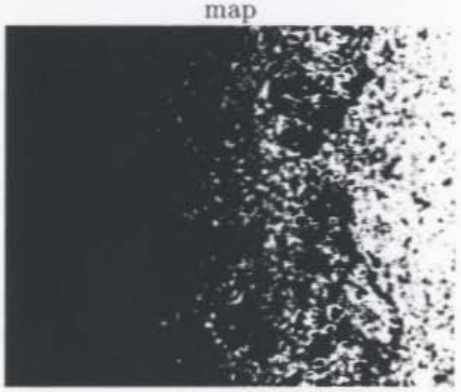
The computation complexity of the Dijkstra's algorithm is very high. To reduce it, we also use a multiscale model. The pass-through probability map is first transformed to a Gaussian pyramid representation [36]. The shortest path is calculated at the coarsest scale. Then we refine it as we proceed to finer scales. The computation complexity of Dijkstra algorithm applied to our routings in image is in the order of $n^{3/2}$, where n is the total number of the pixels in the image. Therefore, our method can dramatically reduce the computation since only the coarsest scale (much smaller than the original image) will do the Dijkstra routing and the refinement of the path has negligible computations compared to the Dijkstra routing. A path calculation result of a 3-level model is given in Figure 6.8. It seems that the reduction of computations is not without cost. Sudden changes of directions along the route are obvious, which the ship's captain may not be glad to see. Errors may also occur for narrow straits, which may be seen as impenetrable at coarse scales.

(a) The land membership map

(b) The landfast ice membership map



(c) The rubble ice membership map



(b) The pack ice membership map

Figure 6.6: Membership maps of the land, landfast ice, rubble ice, and the pack ice

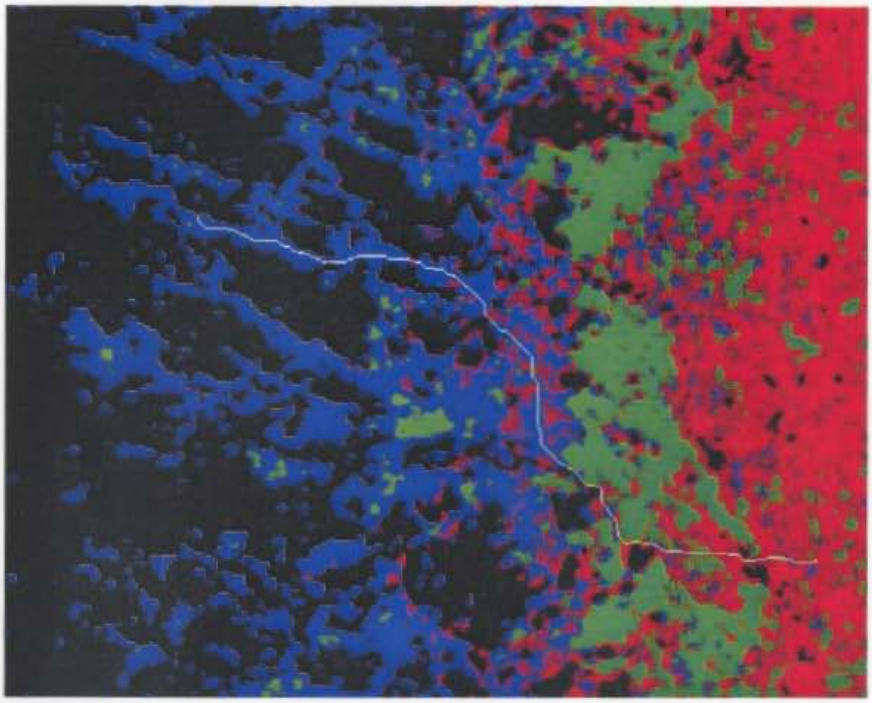


Figure 6.7: An example optimal path

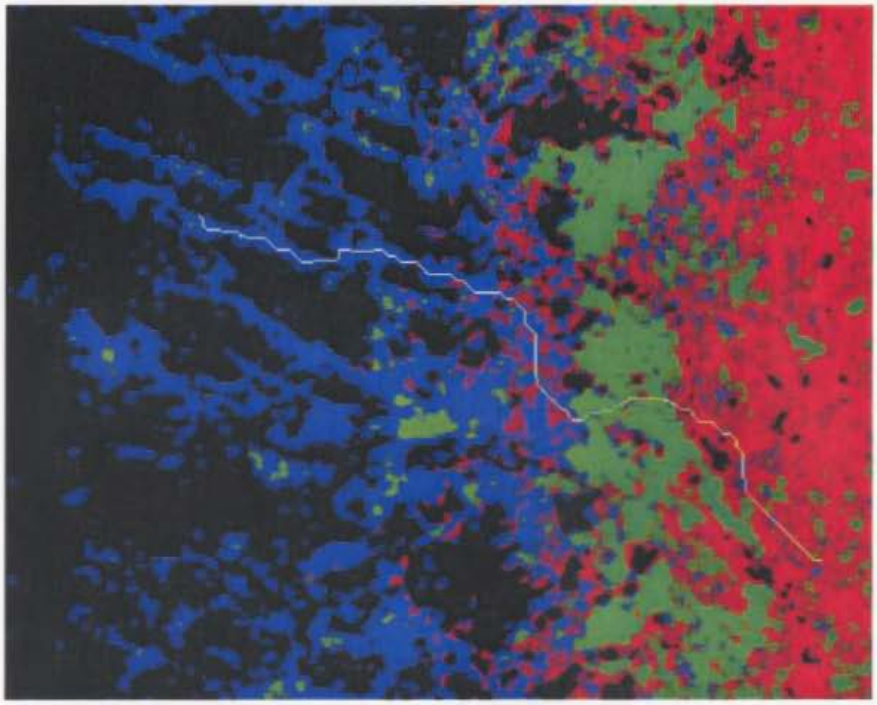


Figure 6.8: An example optimal path obtained with a 3-level multiscale model

Chapter 7

Summary and Conclusions

7.1 Summary of contributions and conclusions

- In this thesis we have presented a comparative study on feature extraction methods using different wavelet signatures, for both the Brodatz textures and SAR sea ice textures. Unlike most previous research in literature that focus on distinct ice types such as first year ice and old ice, our SAR sea ice samples are ice types in young stages, which are more difficult to discriminate. We conclude that feature extraction methods based on the wavelet transform have good performance because of the multiscale nature of the wavelet transform. Results of the comparisons among different wavelet signatures indicate that first order signatures are sufficient for identification of most textures (including our SAR sea ice textures), though combining higher order signatures is probably slightly better.

- Directional variations of the texture features can cause significant degrations of the classification system. Discarding the directional features is not advisable since these features can help in the classification. A good solu-

tion is to find a rotation invariant representation for these features. We propose a new rotation invariant feature extraction method that combines DFT encoding with the Complex Wavelet Transform. The experiments have yielded promising results and hence show the method is feasible for the rotation invariant texture classification purpose.

- For some sea ice such as new and young ice, fuzzy classification is more appropriate. Our new approach provides a fuzzy feature set corresponding to various scales and directions. Linear combinations of these fuzzy features will give represention of the final memberships. By integrating the classification with a shortest path routing module, we have implemented a simple but novel ship navigation system. We have shown with the system how SAR sea ice classification can be utilized to assist in ship navigation in difficult ice-infested northern waters in coastal areas.

7.2 Future work suggestions

- Our fuzzy features represent the memberships of different scales and directions, and their weighted sums are the final memberships. Because of the current difficulty in accquiring the membership knowledge, we cannot give an accurate estimation of those weights and therefore the fuzzy classifiation is not well validated. Future work is needed to acquire the information and perform LSE estimations of the weights.

- We haven't utilized the oriented structure features obvious in rubble ice during the classification experiments. Its scale is too large to make use of, unless we use a much larger window and perform more levels of wavelet decomposition. This also implies a loss in the accuracy of the boundary

localization. A possible way to solve the contradiction is by a coarse-to-fine approach. Further work can be directed to find a feasible coarse-to-fine approach for texture segmentations.

- This research was conducted on a single SAR image. yet this image is large and has a wide diversity of ice types against an irregular coastline. As such. this one image contains reasonable SAR sea-ice image variability to assist the initial development of ice type classifications techniques as presented in this thesis. However. future research is recommended to verify these results on different images.

Bibliography

[1] David Anthony Clausi. "Texture Segmentation of SAR Sea Ice Imagery". Ph.D. thesis. University of Waterloo. 1996.

[2] P. Brodatz. *Textures: A Photographic Album*. New York: Dover. 1966.

[3] USC-SIPI Image Database.
http://sipi.usc.edu/services/database/Database.html

[4] Canada Center for Remote Sensing web site. "A Tutorial of Remote Sensing". http://www.ccrs.nrcan.gc.ca/ccrs/eduref/tutorial/indexe.html

[5] David C. Munson. Jr.. Robert L. Visentin. "A Signal Processing View of Strip-Mapping Synthetic Aperture Radar". *IEEE Trans. Acoust. Speech. Signal Processing*. vol. 37. pp. 2131-2147. Dec. 1989.

[6] Frank D. Carsey *et al.*. *Microwave Remote Sensing of Sea Ice*. American Geophysical Union. Washington DC. 1992.

[7] Canadian Ice Service web site. "Ice Terminology".
http://www.cis.ec.gc.ca/about/term.html

[8] J.S.Lee. "Speckle Analysis and Smoothing of SAR images". *Computer Graphics and Image Processing*. vol. 17. pp. 24-32. 1981.

[9] R. Kwok. E. Rignot. B. Holt. and R. Onstott. "Identification of sea ice types in spaceborne synthetic aperture radar data". *Journal of Geophysical Research*. vol. 97. no. C2. pp. 2391-2402. 1992

[10] IEEE Standard 610.4-1990. *IEEE Standard Glossary of Image Processing and Pattern Recognition Terminology*. IEEE Press. New York. 1990.

[11] Kenneth R. Castleman. *Digital Image Processing*. Prentice Hall. 1996.

[12] Ying Li. "Synthetic Aperture Radar (SAR) Image Compression Using the Wavelet Transform". M.Eng. thesis. Memorial University of Newfoundland. 1997.

[13] M. Tur. K.C. Chin and J.W. Goodman. "When Is Speckle Noise Multiplicative". *Applied Optics*. vol. 21. no. 7. pp. 1157-1159. 1982.

[14] Richard C. Dubes and Anil K. Jain. "Random Field Models in Image Analysis". *Journal of Applied Statistics*. vol. 16. no. 2. pp. 131-164. 1989.

[15] J. Besag. "Spatial Interaction and The Statistical Analysis of lattice Systems". *Journal of the Royal Statistical Society. Series B*. vol. 6. pp. 192-236. 1974.

[16] D.A. Clausi. "Comparison and Fusion of Co-occurrence. Gabor. and MRF Texture Features for Classification of SAR Sea Ice Imagery". *Atmosphere and Oceans*. vol. 39. no. 4. pp. 183-194. 2001.

[17] F. D'Astous and M.E. Jernigan. "Texture discrimination based on detailed measures of the power spectrum". *Proc. IEEE Comput. Society Conf. on Pattern Recognition and Image Processing*. pp. 83-86. 1984.

[18] Sea Ice Type Classification from SAR Images Using Independent Components.
http://www.hut.fi/ karvonen/karvonen.html

[19] Mohammed E. Shoker. "Evaluation of Second-Order Texture Parameters for Sea Ice Classification From Radar Images". *Journal of Geophysical Research*. vol. 96. no. C6. pp. 10625-10640. June 15. 1991.

[20] Jeffrey A. Nystuen and Frank W. Garcia. Jr.. "Sea Ice Classification Using SAR Backscatter Statistics". *IEEE Trans. on Geoscience and Remote Sensing*. vol. 30. no. 3. pp. 502-509. May 1992.

[21] Leen-Kiat Soh and Costas Tsatsoulis. "Texture Analysis of SAR Sea Ice Imagery Using Gray Level Co-Occurrence Matrices". *IEEE Trans. On Geoscience and Remote Sensing*. vol. 37. no. 2. pp. 780-795. March 1999.

[22] Trygve Randen. "Filter and Filter Bank Design for Image Texture Recognition". Ph.D. thesis. Norwegian University of Science and Technology. 1997.

[23] K. I. Laws. "Rapid Texture Identification". *Proc. of the SPIE Conference on Image Processing for Missile Guidance*. pp. 376-380. 1980.

[24] J. M. Coggins and A. K. Jain. "A Spatial Filtering Approach to Texture Analysis". *Pattern Recognition Letters*. vol. 3. no. 3. pp. 195-203. 1985.

[25] F. W. Campbell and J. G. Robson. "Application of Fourier Analysis to The Visibility of Gratings". *J. Physiology*. vol. 197. pp. 551-565. 1968.

[26] D. A. Pollen and S. F. Ronner. "Visual Cortical Neurons as Localized Spatial Frequency Filters". *IEEE Trans. Syst.. Man. Cybern.*. SMC-13. no. 5. pp. 907-916. Sep-Oct 1983.

[27] A. K. Jain and F. Farrokhnia. "Unsupervised Texture Segmentation Using Gabor Filters". *Pattern Recognition*. vol. 24. no. 12. pp. 1167-1186. 1991.

[28] A. C. Bovik. "Analysis of Multichannel Narrow-band Filters for Image Texture Segmentation". *IEEE Trans. Signal Processing*. vol. 39. no. 9. pp. 2025-2043. Sept.. 1991.

[29] A. C. Bovik. M. Clark. and W. S. Geisler. "Multichannel Texture Analysis Using Localized Spatial Filters". *IEEE Trans. Pattern Anal. Machine Intell.*. vol. 12. no. 1. pp. 55-73. Jan.. 1990.

[30] D. F. Dunn and W. E. Higgins. "Optimal Gabor Filters for Texture Segmentation". *IEEE Trans. Image Processing*. vol. 4. no. 7. pp. 947-964. July. 1995.

[31] S. G. Mallat."A Theory for Multi-resolution Signal Decomposition: The Wavelet Representation". *IEEE Trans. PAMI*. vol. 11. no. 7. pp. 674-693. 1989.

[32] I. Daubechies."Ten Lectures on Wavelets". *Commun. Pure Appl. Math.*. vol. 44. pp. 909-996. 1988.

[33] T.Chang and C.Kuo. "Texture Analysis and Classification with Tree-structured Wavelet Transform". *IEEE Trans. Image Processing*. vol. 2. pp. 429-441. Oct. 1993.

[34] Warren J. Jasper. Stephen J. Garnier and Harsh Potlapalli. "Texture characterization and defect detection using adaptive wavelets". *Optical Engineering*. vol. 35. no. 11. pp. 3140-3149. Nov. 1996.

[35] R.W.Linsay. D.B.Percival. and D.A.Rothrock. "The Discrete Wavelet Transform and the Scale Analysis of the Surface Properties of Sea Ice". *IEEE Trans. Geosci. Remote Sensing*. vol. 34. pp. 771-787. 1996.

[36] P.J.Burt and E.A.Adelson. "The Laplacian pyramid as a compact image code". *IEEE Trans. on Communications*. vol. 31. pp. 532-540. 1983.

[37] H.Greenspan. S.Belongie and P.Perona. "Invariant Texture Recognition Using a Steerable Pyramid". California Institute of Technology. 1994. http://www.cs.berkeley.edu/~ hayit/

[38] Nick Kingsbury. "Image Processing with Complex Wavelets". *Phil. Trans. Royal Society London A*. on a Discussion Meeting on "Wavelets: the key to intermittent information". London. Feb. 24-25. 1999. http://www-sigproc.eng.cam.ac.uk/~ ngk/

[39] Nick Kingsbury. "The Dual-tree Complex Wavelet Transform: A New Technique for Shift Invariance and Directional Filters". IEEE Digital Signal Processing Workshop. DSP 98. Bryce Canyon. August 1998. paper 86.

[40] Nick Kingsbury. "A Dual-Tree Complex Wavelet Transform with improved orthogonality and symmetry properties". *Proc. IEEE Conf. on Image Processing*. Vancouver. September 11-13. 2000. paper 1429.

[41] F.Ulaby. F.Kouyate. B.Brisco. and T.Williams. "Texture Information in SAR Images". *IEEE Trans. Geosci. Remote Sensing*. vol. 24. pp. 235-245. Mar. 1986.

[42] S.G.Chang. B.Yu. and M.Vetterli. "Adaptive Wavelet Thresholding for Image Denoising and Compression". *IEEE Trans. on Image Processing*. vol. 9. pp. 1532-1546. 2000.

[43] G. Van de Wouwer. P. Scheunders and D. Van Dyck. "Statistical texture characterization from discrete wavelet representations". *IEEE Transactions on Image Processing*. vol. 8. pp. 592-598. 1999.

[44] H.Xiong. T.Zhang. and Y.S.Moon. "A Translation- and Scale-Invariant Adaptive Wavelet Transform". *IEEE Transcations on Image Processing*. vol. 9. pp. 2100-2108. 2000.

[45] A Karasaridis and E Simoncelli. "A Filter Design Technique for Steerable Pyramid Image Transforms". Int'l Conf. Acoustics Speech and Signal Processing. Atlanta GA. May 1996.
http://www.cis.upenn.edu/~eero/ABSTRACTS/karasaridis96-abstract.html

[46] Ludmila I. Kuncheva. *Fuzzy Classifier Design*. Physica-Verlag Heidelberg. 2000.

[47] E. W. Dijkstra. "A Note on Two Problems in Connexion with Graphs". *Numer. Math.*. vol. 1. pp. 269-271. Oct. 1959.

[48] Charles Robertson. "Design of an Object-Oriented Image Processing Library". Technical Report. Memorial University of Newfoundland. August. 1994.

Appendix A

Mixed Brodatz and SAR Images (MBS)

Figure A.1: Bark (D12)

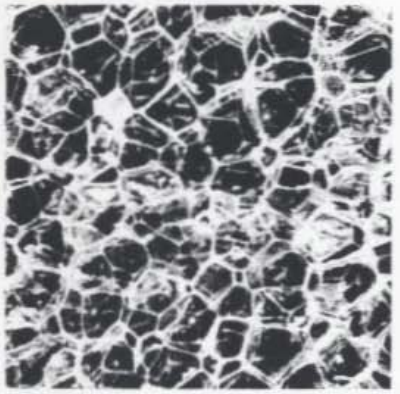


Figure A.2: Bubble (D112)

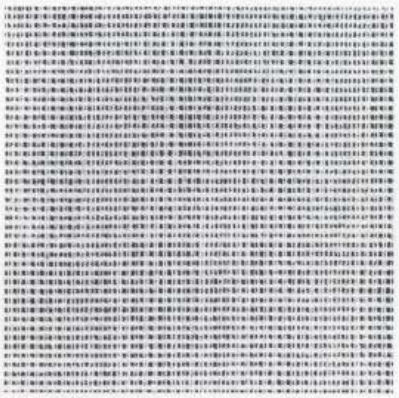


Figure A.3: Canvas (D21)

Figure A.4: Grass (D9)

Figure A.5: Leather (D24)

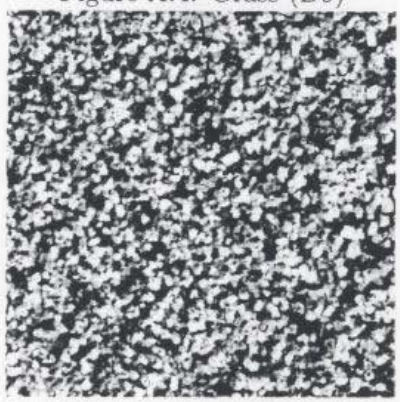


Figure A.6: Sand (D29)

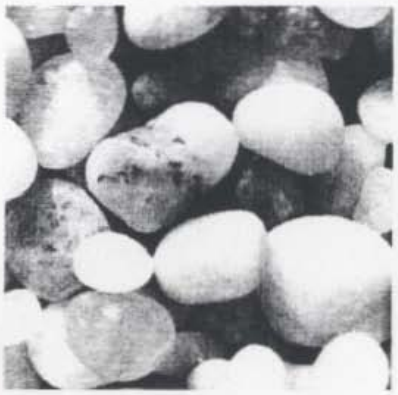


Figure A.7: Stone (D30)

Figure A.8: Straw (D15)

Figure A.9: Woolen (D19)

Figure A.10: Pack ice

Figure A.11: Rubble ice

Figure A.12: Land

Appendix B

Pure Logarithmed SAR Images (PLS)

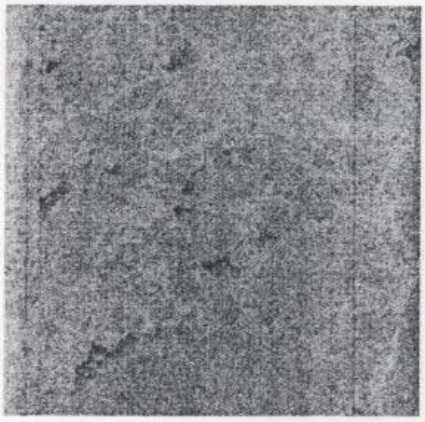


Figure B.1: Pack ice

Figure B.2: Rubble ice

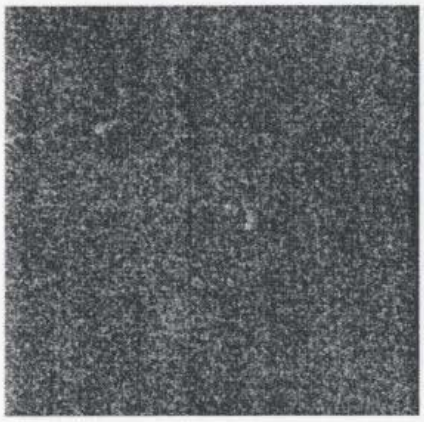


Figure B.3: Landfast ice

Figure B.4: Land

Appendix C

Software Documentation

Most of our experimental work was performed with Microsoft Visual C++ under Win95/winNT/win2000. The software package (named IPL) is adapted from a SunOS Unix version of IPL originally developed by Charles Robertson [48], modified by Ying Li, Chen Ju and Dr. Cecilia Moloney. The organization of the software packages and the corresponding disk directories are listed as follows.

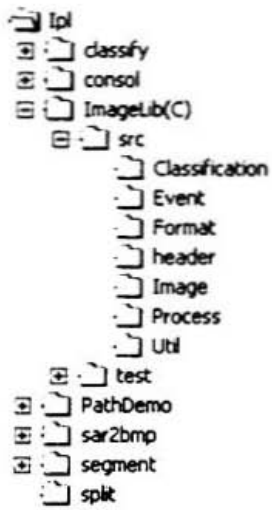


Figure C.1: IPL software packages

ImageLib(c)/src: package of image processing library functions

Classification: base classification functions

BaseClassifier.cpp	Gives the base class for all kinds of classifiers.
BaseFeatures.cpp	Gives the base class for all kinds of features.
BaseSegmenter.cpp	Gives the base class for all kinds of segmenters.
Class.cpp	Gives the class representing each texture type.
ClassifySegmenter.cpp	Segmenter based on sliding window classifications.
DistanceClassifier.cpp	Gives the base class for all classifiers that measure distances.
MahaFeatures.cpp	Mahalanobis distance features.
OtherSegmenter.cpp	Gives the base class for all segmenters exluding that based on sliding window classifications.
VectorFeatures.cpp	Gives the class for feature vector.

Event: events

Event.cpp	Gives the base class for all events.
ProgressEvent.cpp	Gives the class for progress event. It defines the interface that will be used in event handling.

Format: image file format

BaseImgFmt.cpp	Gives the base class for all image formats.
Bitmap.cpp	Windows bitmap format.
PicDes.cpp	Pic format.

Image: image objects

Image.cpp	Gives the base class for all kinds of image objects.
GrayImage.cpp	gray image object. It gives the parent class of that in CharGrayImage.cpp, FloatImage.cpp, and IntImage.cpp.
CharGrayImage.cpp	image object of 256 gray levels.
FloatImage.cpp	image object of float pixel values.
IntImage.cpp	image object of integer pixel values.
ColorImage.cpp	color image object. It gives the parent class of that in CharImage.cpp and TrueColorImage.cpp
CharImage.cpp	image object of 256 colors.
TrueColorImage.cpp	image object of true colors.

Process: miscellaneous image processing functions	
ProcObject.cpp	Gives the base class for all kinds of processings.
Algebra.cpp	Linear algebra functions.
Alu.cpp	Arithmetic and logic functions.
Statistic.cpp	Statistic functions.
AdptRationOp.cpp	Adaptive rational operator for denoising.
AnisoDiffEdge.cpp	Anisotropic diffusion.
Glcm.cpp	Gray level co-occurrence matix computation.
Filter.cpp	Filter functions.
LapPyramid.cpp	Laplacian Pyramid transform.
Wavelet.cpp	Discrete Wavelet Transform.
DualWav.cpp	Dual-tree Complex Wavelet Transform (translated from Kingsbury's Matlab code).
QShiftWav.cpp	Q-Shift Complex Wavelet Transform.
Util: utilities	
BaseInputStream.cpp	Gives the base class for all input streams.
BaseOutputStream.cpp	Gives the base class for all output streams.
BitInputStream.cpp	input stream bit by bit.
BitOutputStream.cpp	Output stream bit by bit.
HuffInputStream.cpp	input streams that have incoporate the adaptive huffman coding.
HuffOutputStream.cpp	Output streams that have incoporate the adaptive huffman coding.
BaseTree.cpp	Gives the base class for all tree representations.
HuffTree.cpp	Huffman tree.
ImageNode.cpp	tree node with data of image object.
ImageTree.cpp	tree representation for image objects.
Quantizer.cpp	Quantizer.
DiscreteFourier.cpp	Discrete Fourier transform.

ImageLib(c)/test (dependent on **ImageLib(c)**): windows GUI for miscellaneous image processings

DIBAPI.cpp	Windows bitmap functions.
test.cpp.	windows GUI functions. It accepts user input and calls functions of ImageLib(c)/src package to do miscellaneous image processings.
MainFrm.cpp.	
ChildFrm.cpp.	
testDoc.cpp.	
testView.cpp.	
dialog.cpp	

Consol (dependent on **ImageLib(c)**): console for miscellaneous image processings.

Consol.cpp	main function for miscellaneous image processings using ImageLib package.

Classify (dependent on **ImageLib(c)**): package for image classifications.

ClassifyBat.cpp	main function for both training and classification. It reads a setting file. which specify the classifier type and other parameters. and then do the corresponding process.
DwtClassifier.cpp	DWT energy classifier.
GlcmClassifier.cpp	GLCM classifier.
LppClassifier.cpp	Laplacian Pyramid energy classifier.
DtCwtClassifier.cpp	Dual-tree CWT energy classifier.
DtCwtDftClassifier.cpp	DFT-DTCWT classifier.
DwtProbClassifier.cpp	DWT entropy classifier.
EntropyFeatures.cpp	DWT histogram entropy features.
TswClassifier.cpp	TSW classifier.
TreeFeatures.cpp	Tree representation of features. It gives the parent class of that in TswFeatures.cpp.
TreeFeatureNode.cpp	Tree feature node. It gives the parent class of that in TswFeatureNode.cpp.
TswFeatures.cpp	TSW feature.
TswFeatureNode.cpp	Tree node of TSW feature.

segment (dependent on **ImageLib(c). classify**): package for texture segmentations.

segment.cpp	main function for texture image segmentations.
DwtSegmenter.cpp	coarse-to-fine segmenter using DWT entropy feature.

sar2bmp (dependent on **ImageLib(c). classify**): package for sar related applications.

sar2bmp.cpp	main function for extracting an area from SAR images and store it as windows bitmap image file.
sarsegment.cpp	main function for segmenting SAR texture image.
sarprob.cpp	main function for computing SAR texture membership maps.
findpath.cpp	main function for finding optimal path for ship navigation application.
findpathfunc.cpp	functions for finding optimal path for ship navigation application.

PathDemo (dependent on **ImageLib(c). sar2bmp**): package for demonstrating the path computations.

PathDemo.cpp.	windows GUI functions. It accepts user input and calls
MainFrm.cpp.	functions of sar2bmp package to find the optimum path
ChildFrm.cpp.	for ship navigation.
PathDemoDoc.cpp.	
PathDemoView.cpp	

Split (dependent on **ImageLib(c)**): package for slitting and cropping images.

split.cpp	main function for splitting a big image to smaller images.
crop.cpp	main function for cropping a image.