A VOLUMETRIC APPROACH TO SEGMENTATION AND TEXTURE CHARACTERISATION OF ULTRASOUND IMAGES

A Thesis Submitted to the College of

Graduate Studies and Research

in Partial Fulfillment of the Requirements

For the Degree of Doctor of Philosophy

in the Department of Computer Science

University of Saskatchewan

Saskatoon, Saskatchewan

by

Russell E. Muzzolini

Fall 1996

Permission To Use

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College in which my thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis.

Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science
University of Saskatchewan
Saskatoon, Saskatchewan, Canada
S7N 0W0

Abstract

Visual interpretation of noisy images is not an easy problem. This is certainly apparent with ultrasound images. Due to the noise inherent in the images, it is often the case that discrepancies as to location of object boundaries and detection of different tissues arise even among highly trained physicians. The relatively low cost and short image acquisition time, however, make ultrasound an attractive imaging modality.

Currently, diagnostic evaluation of ultrasound images is performed on two-dimensional (2D) cross-sections of the object of interest. No depth information is available and there is no way of viewing the outer surface of the object. The only way for a physician to visualise the entire object is by mentally reconstructing the object based on a series of a 2D images as well as prior expectations of the morpholoogy of the object. In the case of abnormal or diseased growth, the physician's expectations often do not correspond to the actual morphology of the object. However, the use of three-dimensional (3D) data acquisition and visualisation may be used to overcome these problems.

The present work addresses a number of difficulties in processing 3D ultrasound data. This includes special treatment of the volumetric ultrasound data obtained from a 3D probe, determination of 3D features of the different tissue types present in the ultrasound data and identification and localisation of objects (segmentation) in the volumetric ultrasound data. Experimental results obtained from synthesised and real ultrasound data demonstrate that the present work contributes significantly to the use of ultrasound imaging as a diagnostic tool. As well, the present work can be applied to different imaging modalities or different applications areas and is thus beneficial to the area of biomedical image processing, in general.

Acknowledgements

I thank Dr. Herb Yang for the time and effort he has devoted to me over the past several years. Herb has always been supportive academically, financially and personally. His continued guidance allowed me to obtain a goal, which if asked years ago, I thought I could never obtain. I will always be grateful for Herb's friendship and advice. I thank Dr. Roger Pierson for his instruction and continued financial support. I hope that in the future we can collaborate on another exciting project.

I thank committee members Dr. Jean Paul Tremblay and Dr. Derek Eager of the Department of Computer Science, Dr. Gregg Adams of the Department of Veterinary Anatomy and the external advisor, Dr. Xiaobo Li of the Department of Computing Science at the University of Alberta, for their close review of my thesis. I thank the chair of the advisory committee, Dr. John Cooke of the Department of Computer Science, for keeping the defence to around three hours and the required corrections to a weekend worth of editing. Thanks to the examining committee chair, Dr. D.K.J. Gorecki from the College of Graduate Studies and Research, for being the first to congratulate me with the good news. I also thank the staff and graduate students in the Department of Computer Science for keeping the machines and software running (ha, ha). Special thanks go out to Brian ven der Buhs for the many years of story telling, wine making and friendship.

I thank my friends and family for their support. My parents, Ennio and Janet Muzzolini have always been generous with their love and time. They have always supported my endeavours; they taught me how to work hard and how to make good decisions and they offered advice whenever it was asked for. I hope that my dad can learn to relax and that my mom finds the answers to her life long questions. I also thank my sister Tracey and my brother Blair. They, too, will someday earn the status of "Golden Child".

Most importantly, I thank my own family for their love, trust and support. My son Gabriel may be too young to know what has been accomplished, but he can enjoy the rewards that the future will provide. My wife Leslie has been quite patient over the past decade or so. We have seen many changes but could always count on one constant: a student's life at the U of S. Now that this part of our lives is over I look forward to what the future holds.

Table Of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Visual interpretation of ultrasound images is not an easy problem. Even among highly trained physicians, discrepancies as to location of object boundaries and detection of different tissues arise due to the noise inherent in the images. Also, since evaluation is performed visually, the interpretation is quite subjective and influenced by such factors as prior knowledge of the object being imaged, the physician's training, as well as the quality of the ultrasound imaging device. Advances in ultrasound technology have resulted in much improved images. The cost and short acquisition time of ultrasound imaging, as compared with other modalities such as computed tomography (CT) and magnetic resonance imaging (MRI), makes ultrasound imaging very attractive. However, these modalities have advantages over ultrasound due to their higher quality images and computer-based post processing of the raw data. To enhance the capabilities of ultrasound imaging as a diagnostic tool, it is necessary to develop advanced software for post processing of the data so that the resulting output has little subjective variability. Even more desirable is the capability of the software to extract information from the ultrasound data including the presence of tumours, probability of tissue types, and accurate localisation of objects of interest.

Currently, diagnostic evaluation of ultrasound images is performed on two-dimensional (2D) cross sections of the object of interest. No depth information is available and there is no way of viewing the outer surface of the object. The only way for an operator to visualise the entire object is by mentally reconstructing the object based on a series of 2D images. Prior expectations of the morphology of the object may influence interpretation and confound the operator's recognition of abnormal or diseased growth. However, the use of three-dimensional (3D) visualisation techniques, based on the 2D images, may be used to overcome these problems. By reconstructing a 3D model of the object it is possible to rotate the viewing angle, perform surface and volume measurements, and to make arbitrary portions of the object's outside surface transparent so that the details inside the object can be evaluated.

The increased success of 3D visualisation of CT and MRI data has resulted in the application of similar visualisation techniques to ultrasound data by researchers and physicians. However, because of the noise inherent in ultrasound images due to speckle and various artifacts, an accurate 3D representation of the object of interest is difficult [95]. Recently, a new approach to 3D modelling of objects in ultrasound images, based on a series of cross-sectional images has been demonstrated [77, 79, 80]. This approach uses the texture in ultrasound images to characterise different types of tissue present in the images. The characterisation provides an ability to produce a more accurate 3D model of the object being imaged. The main limitation of this approach is that it processes individual 2D images without consideration to the relationship among the images. Thus, the information in the third dimension (axial direction) is not used in characterising the different tissue types nor in determining a 3D model of the object. By processing 3D, rather than 2D, ultrasound data the potential increase in accurate characterisation and 3D modelling of ultrasound images will be substantial.

1.1 Motivation

There are a number of difficulties to overcome in order to process 3D ultrasound data. First, serial acquisition of the data must be modified so that motion of the patient and the ultrasound probe introduce as little alignment error as possible. This error can be reduced with the use of a motorised probe. By spinning the probe about an axis a 3D sector scan can be obtained. If the scan can be completed fast enough, then the alignment error introduced by patient movement can be minimised. Development of the 3D ultrasound probe is beyond the scope of this thesis. As a result, access to the volume data is limited. However, computer simulations of the probe are performed (see Appendix A) to obtain quantitative measurements of alignment error. Some experimentation is performed on a sample ultrasound volume dataset as well.

The resulting volume data produced by a 3D sector scan requires special treatment. Due to the rotation about an axis, points close to the axis will be oversampled while points further from the axis will be undersampled. The volume exists in a non-isometric "cylindrical" coordinate system. A method is presented in this thesis to transform the data to an isometric (equal resolution in each dimension) representation to simplify the task of processing. A 3D Cartesian coordinate system is a convenient isometric representation. The transformation must be performed such that any error in supersampling or interpolation is less than the resolution of the ultrasound transducer.

Another difficulty to overcome in processing 3D ultrasound data is characterising tissue. Most (if not all) approaches [80, 77, 82, 42, 14, 60, 137, 64, 63, 65, 66, 42, 103, 29] developed for tissue characterisation are based on 2D data. The use of 3D ultrasound data requires that segmentation (identification of objects) be performed in 3D. There are currently only a few 3D segmentation algorithms described in the literature [107, 108, 122, 46, 59]. Based on these reports, it is apparent that a new approach is required to accurately segment 3D ultrasound data. The difficulties in processing 3D ultrasound data described above are the motivation for the present work.

1.2 Goal of Thesis

Previous research demonstrates that 3D reconstruction of ovarian follicles *in vitro* is possible using a series of 2D ultrasound images [77]. In performing the reconstruction, it is assumed that each 2D image is perfectly aligned along the scanning axis.

For this assumption to be valid, mechanical placement of the ultrasound transducer is used to ensure proper alignment and orientation of the transducer. In most clinical situations (especially in gynaecology), however, precise control of the transducer orientation is difficult due to movement of the patient. One way to minimise the alignment error is to motorise the scanning method so that the series of 2D images can be obtained with little error. In effect, a 3D image of a region is produced. If 3D scanning can be performed fast enough, movement of the patient and operator will not introduce significant error into the images.

The main goal of this thesis is to determine methods for objective evaluation and visualisation of volume data. To satisfy this goal, the following tasks have been identified:

1. design a software system such that it can be applied to different types of images (other than ultrasound),
2. determine methods for analysing and representing the 3D structure of texture,
3. determine methods for characterising tissues in the ultrasound data,
4. using the information obtained in characterising tissues, determine a method for segmenting the ultrasound volume data,
5. determine a data structure for representing the ultrasound volume data which is appropriate for efficient processing,
6. determine methods for visualising the processed ultrasound data which provide a simple, meaningful representation of the object being imaged,
7. implement the above methods in software and evaluate their success/failure.

1.3 Overall System Design

The proposed software system, which is a realisation of the thesis goal, is comprised of five modules: the Tissue Characterisation module, the 3D Segmentation mod-

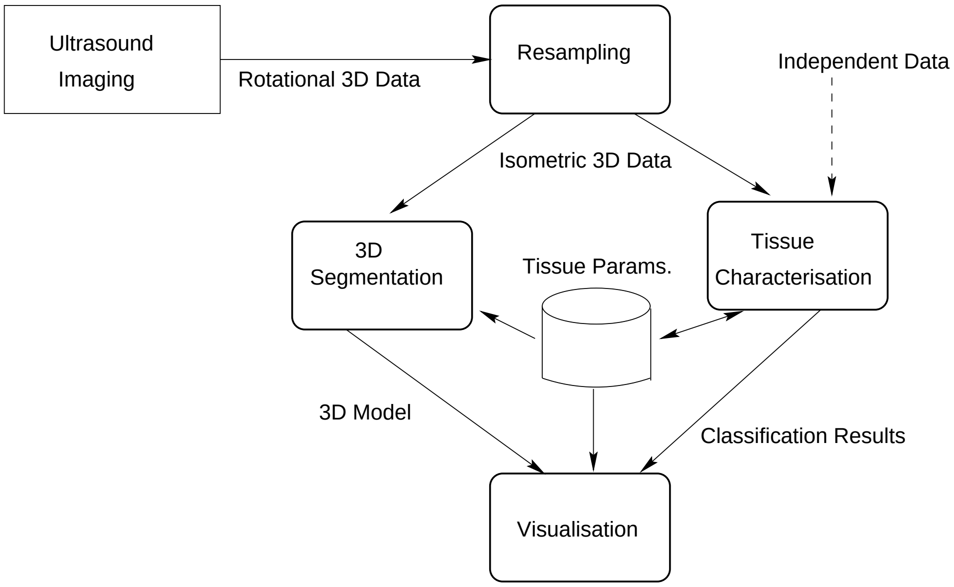


Figure 1.1: Overall system design.

ule, the Visualisation module, the Resampling module and the Ultrasound Imaging module.

Figure 1.1 illustrates the overall view of the modules that comprise the proposed software system. The function of the modules is summarised in the following paragraphs.

The purpose of the Tissue Characterisation module is to determine the parameters of representative samples of varying tissue types. A repository is used to store the tissue (texture) parameters which are calculated from training samples. These parameters are used by the 3D segmentation module to identify the tissue type of a block of pixels during segmentation of the volume data. As well, the Tissue Characterisation module can recall these parameters when performing classification of unknown patterns. This ability allows the system to have a more general application as classification can be performed on patterns independent of their source (i.e., the patterns can come from a source other than the ultrasound probe).

The representative samples are in the form of volume data which must be further processed to determine the tissue parameters. This processing involves the calculation of feature measures, based on the texture present in the data. The features provide a compact, vector representation of the volume data (training data) from which tissue parameters can be determined. Previous research used an *ad hoc* approach in analysing the features to determine the most effective parametric representation of the tissues [77, 78, 81]. Although good results were demonstrated for a small number of classes, the approach was not able to characterise tissue types which were very similar in appearance (as is the case with some tissues in ultrasound data). As a result, the Tissue Characterisation module uses a statistical pattern recognition approach in determining the tissue parameters. An advantage of this approach is that the theory behind statistical pattern recognition is well established. As well, the approach provides greater power in differentiating tissue types. Chapters 3 and 4 describe the details of the Tissue Characterisation module.

The purpose of the 3D Segmentation module is to identify objects in the ultrasound volume data. Objects are identified based on their tissue composition which is represented by texture patterns in the ultrasound data. The characteristics of different tissues are represented in the tissue parameters which are determined by the Tissue Characterisation module. The Multiresolution Texture Segmentation (MTS) algorithm [77, 81, 82, 78] demonstrated that a series of 2D ultrasound images could be segmented with good results. Since the ultrasound data is 3D, however, the MTS algorithm must be modified. The 3D MTS algorithm is presented as a method of segmenting ultrasound volume data. It has a number of advantages over the original MTS algorithm including the ability to segment isometric 3D data. Chapter 5 describes the 3D MTS algorithm in detail.

The purpose of the Visualisation module is to provide the ability to display both the segmentation results as well as the tissue parameters. The main purpose of this module is to allow the user to manipulate the data (through rotations, lighting effects, surface selection, etc.) so that visual analysis can be performed. Standard techniques (see Section 2.4) are employed in the Visualisation module to display the data. The publicly available software environment Khoros [147] provides enough functionality to meet the visualisation requirements of this research.

Both the 3D Segmentation and Tissue Characterisation modules require isomet-

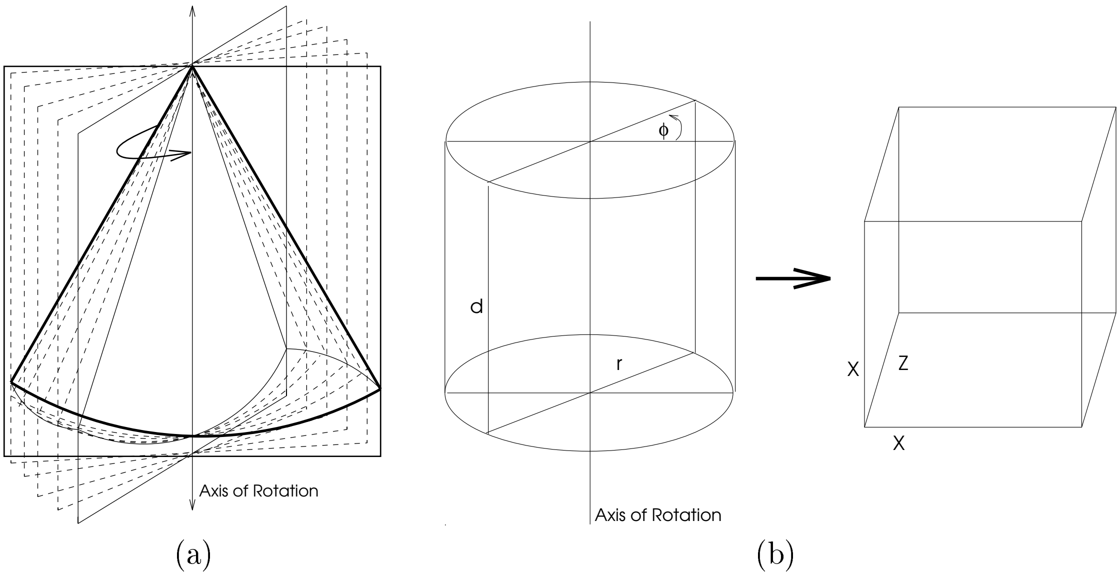


Figure 1.2: (a) Ultrasound data obtained as a volumetric sector scan using a 3D probe. (b) Non-isometric cylindrical coordinate system where d is the axial distance from the probe, r is the radial distance from the axis of rotation and θ is the angular increment between successive 2D images (slices).

ric data for efficient processing. The use of isometric data allows pixel values to be treated in the same manner, independent of their location within the volume. This has significant implications on the volumetric ultrasound data obtained from the Ultrasound Imaging module (see Section 2.1 for background on ultrasound imaging). Figure 1.2(a) shows how volumetric data is obtained from a motorised ultrasound probe which is rotated 180° about an axis at equal increments. By rotating the probe about an axis of rotation, a cylindrical volume is swept out. This design is attractive because of its simplicity and its relatively low cost. Steen and Olstad [119] also use this type of probe to obtain volumetric fetal scans during regular clinical examinations.

There are other scanning methods in which a conventional 2D ultrasound probe can be adapted to obtain volume data. A linear scan may be obtained by fixing the probe at a particular orientation and then translating it along a scanning axis parallel to the surface of the object. Since the orientation of the transducer is fixed, the slices are aligned and an isometric volume is swept out. The linear translation of the probe limits its use to primarily *in vitro* applications. Sakas et al. [107] obtain volumetric data with the use of a 2D probe which mechanically swivels about an axis. This is commonly referred to as a curvilinear scan and resembles a "donut-like" shape when rotated through 360°. The scanning technique is called curvilinear because it resembles a linear scan in which the translation of the probe is around the circumference of a circle. Gilja et al. [36] use a transducer controlled by a stepping motor to obtain volume scans of the abdominal organs. The transducer is tilted by the stepping motor through an angular displacement of up to 53°. Larger displacements result in increased error as the deviation of the transducer's beam increases.

There are a number of issues to consider when processing the volume data obtained from the Ultrasound Imaging module. It is usually assumed that the wobble about the axis of rotation is negligible (or alternatively, that the orientation of each slice can be registered) during image acquisition. It is also assumed that the motor does not induce any movement of the transducer when a slice is being obtained. Points close to the axis of rotation will be oversampled while points further from the axis will be undersampled. The rotation angle increment determines the sampling rate and effectively the amount of error introduced into the volumetric sector scan. If the angle is small enough, then the error introduced will be insignificant compared to the resolution of the ultrasound probe. Development of the Ultrasound Imaging module is beyond the scope of this thesis, however, experimentation is performed with the data obtained from the 3D ultrasound probe. Simulation of a 3D probe is also performed (see Appendix A).

Figure 1.2(b) shows the volumetric ultrasound data represented in a "cylindrical" coordinate system where d is the axial distance from the probe, r is the radial distance from the axis of rotation and θ is the angular increment between successive 2D images (slices). The purpose of the Resampling module is to transform the data to an isometric representation. A 3D Cartesian coordinate system is an appropriate target. It is vital that the transformation be performed without introducing significant sampling artifacts. Appendix A describes the details of the Resampling Module. Simulations of the ultrasound probe used in the Ultrasound Imaging module are also presented. Chapter 6 ties all the system modules together by demonstrating their application on a volumetric ultrasound dataset obtained with a 3D rotational probe.

The design of the system contributes new approaches for segmenting and classifying data. The most significant of these contributions is the explicit connection between classification and segmentation which are typically treated independently. In the system, segmentation is viewed as way of "classifying on the fly". That is, the 3D Segmentation module uses the class information obtained by the Tissue Characterisation module to determine the identity of different regions within the volume. The emphasis of the segmentation algorithm is on determining the appropriate regions (location and size) within the volume that are to be "classified" using the *a priori* class knowledge. In previous work [80, 79, 77], segmentation was based on an *ad hoc* characterisation of the classes, rather than using classification.

The success at which modelling of 3D objects is obtained using this approach is demonstrated on two applications: linearly scanned, bovine ultrasound volume data, and fetal ultrasound volume data obtained with an experimental, 3D rotational probe. The results from the bovine data are especially significant. Based on 3D texture analysis, three tissues types, corpus luteum, stroma and fluid were distinguished and modelled successfully. In previous work [80, 79, 77], it was not possible to distinguish corpus luteum from stroma, based on 2D texture analysis. As these tissues are visually very similar, differentiating between them represents a significant improvement in modelling objects in ultrasound volume data.

A number of original contributions are made by the work in this thesis. These contributions are summarised as follows:

- the design, implementation, and testing of a software system for characterising, classifying, and segmenting volumetric data,
- a unified multiresolution framework for segmentation and classification of (ultrasound) data in which segmentation is viewed as a way of "classifying on the fly",
- a new statistical decision rule, Inck (**Inc**omplete **K**nowledge), for use as a

criterion function in feature selection and a decision rule in classification of unknown patterns,

- the addition of a robust statistical parameter estimator, MVE, in the design of a classifier,
- the multiresolution design of a classifier,
- a new 3D segmentation algorithm, the 3D MTS algorithm, which provides the ability to segment noisy, isometric, ultrasound volume data,
- the development of the 3D cooccurrence matrix,
- the visualisation of ultrasound volume data obtained from a 3D ultrasound probe is made possible.

Many of the contributions listed above are independent of the type of data being processed. That is, the application of the system to data other than ultrasound is possible, provided that the textural information present is suitable for analysing the data.

1.4 Organisation of Thesis

The organisation of this thesis is as follows. Background material is provided in Chapter 2. This includes an overview of ultrasonic imaging, a survey of the different models and techniques used to characterise ultrasound tissues, a discussion of 2D and 3D segmentation methods as a way of identifying objects in 3D datasets, and finally, a description of techniques for visualising both segmented and unprocessed data. The details of the system modules are provided in Chapters 3 - 5. Chapters 3 and 4 describe the details of the Tissue Characterisation module. This includes a detailed description of main processes comprising the module as well as experimental evaluation of the module using both synthesised and real data. Chapter 5 presents the details of the 3D Segmentation module. Experimental evaluation of the 3D MTS algorithm and existing methods is provided. Chapter 6 demonstrates the application of the system to a volumetric ultrasound dataset of a 12 week old fetus. Chapter 7 provides a summary of the thesis as well as its contributions to the areas of image processing, pattern recognition and medical imaging. Future work for continued research of the thesis topic is also discussed. Appendix A describes the details of Resampling Module. Appendix B provides definitions for the probabilistic distance measures evaluated in the thesis. A glossary of terms is provided in Appendix C.

Chapter 2

Background

2.1 Ultrasonographic Imaging

In order to interpret the contents of an ultrasound image, it is necessary to understand how the interaction of a sound wave with physical structures (such as tissue) can be converted into an image. This section describes the organisation of a basic ultrasound scanner, how an image is formed from a sound wave, and the use of ultrasound images for diagnostic purposes.

2.1.1 The Ultrasonographic Scanner

This section describes the basic ultrasonographic scanner at a functional block level. A detailed description of each component is available in the literature [89, 10, 95, 21].

Ultrasonographic scanners in medical imaging produce a pulse with a frequency in the range of 3.5 to 10 MHz [77]. Better resolution can be achieved with a higher frequency, but the pulse can not penetrate as deeply into the tissue. Smaller structures which are closer to the surface are usually scanned with a high frequency while larger, deeper structures are scanned at a lower frequency. The ultrasound pulse rate is limited by the speed at which sound travels through tissue. Pulse rates of 100Hz to 1000's Hz are physically possible.

Figure 2.1 illustrates a basic ultrasonographic scanner. The directed arrows indicate the flow of information among the components in the scanner. Ultrasound pulses are transmitted and received by the *transducer*. The transducer emits a

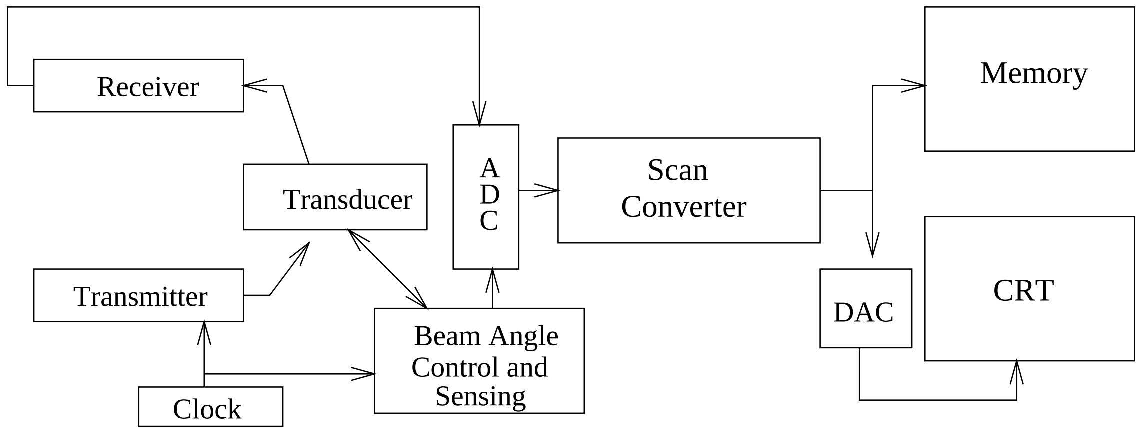


Figure 2.1: Block diagram of a typical ultrasonographic scanner.

constant rate of ultrasound pulses under the control of a clock and the *beam angle control and sensing circuitry*. The beam angle control circuitry tells the transducer where to aim the beam of ultrasound pulses. The sensing portion of this circuitry determines the angle at which the beam is directed and passes this information on to the *scan converter* [89]. Depth signals received by the transducer are also passed on to the scan converter, through the *receiver*. Since the depth signals are analog, they must pass through an analog to digital converter before reaching the scan converter. The scan converter uses the information it gets from the beam angle control and sensing circuitry to calculate the appropriate pixel address for the intensity value based on the transducer depth signal. Once the address calculations are performed, the intensity values can be displayed on a video display (CRT) and stored in memory for further processing.

The scan converter controls placement of each ultrasound echo to its appropriate pixel location in memory (or video display). Most scan converters locate pixel intensities on a 512x512 grid. The resolution of the pixel intensity ranges from 5 to 8 or more bits. In real time sector scanning the transducer consists of an array of elements which are electronically controlled to sweep the beam along the array. Each sweep produces a 512x512 image. As the transducer is moved along the object, a series of serial cross sections or images is produced.

The main purpose of the receiver is to amplify the incoming electrical signals

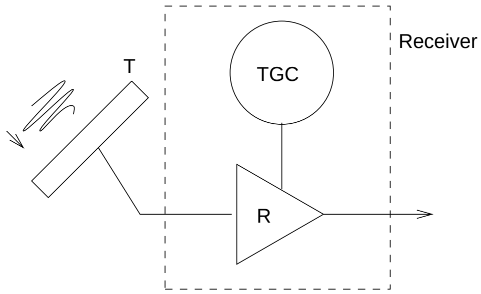


Figure 2.2: Simplified block diagram of the receiver.

from the transducer. In order to minimise the effects of attenuation of the sound wave, most ultrasound machines use time gain compensation (TGC) to increase the signal strength of farther echoes and decrease the strength of near-field echoes. Figure 2.2 shows a simplified block diagram of the receiver. Returning echoes are received by the transducer T and the force exerted upon the transducer is converted into an electrical signal. The electrical signal is passed to the radio frequency (RF) amplifier R (an RF amplifier is used since the input frequency range is within the RF spectrum) at which a signal gain determined by the TGC is achieved. The RF signal at the output of the RF amplifier represents the output signal for the corresponding received echoes. The TGC can be set such that echoes close to the transducer are amplified slightly while echoes farther from the transducer are amplified more. Note that the use of the TGC is to scale all electrical signals from the transducer so that variations in amplitude are independent of attenuation. Also, the TGC can be used to reduce the input dynamic range to the RF amplifier. This is desirable as an attenuated sound wave can have a corresponding dynamic range of up to 100 db. A typical linear RF amplifier only has a dynamic range of around 50 db. The TGC can prevent saturation of the RF signal by reducing the input dynamic range. The TGC is set manually by observing the intensity variations (from top to bottom) in the resulting ultrasound image. An ultrasound machine divides the image into a number of bands from top to bottom and the TGC setting for each band can be adjusted individually. Figure 2.3 shows how the TGC can be set to increase the intensity of the display for lower bands corresponding to deeper echoes.

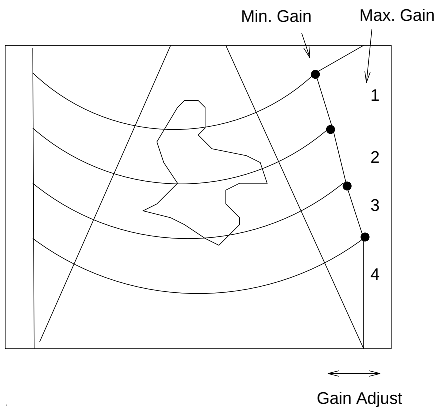


Figure 2.3: Adjusting TGC to subjectively equalise the intensity for each band in the image. An ultrasound machine will provide the user with the ability to adjust the gain for each band.

2.1.2 Image Formation

In order to interpret the contents of an ultrasonographic image, it is necessary to understand how a sound wave interacts with physical structures (such as tissue) and how the interaction can be converted into an image. This subsection describes how a sound wave is created, controlled and measured to produce a two-dimensional (2D) ultrasound image.

A complete discussion of acoustic wave theory is available in the literature [76, 93, 21]. Powis and Powis provide a high level summary of wave theory as applied to ultrasonographic imaging [95]. Background of discussions in this paper can be obtained from these sources.

Ultrasonic imaging is based on the process of echo-ranging [95]. In echo-ranging, a source (transducer) transmits a burst of energy (sound wave) into a medium. It is assumed that the velocity at which the sound wave travels through the medium is constant. Using this assumption, it is possible to determine the distance of the sound wave from the transducer at a particular time using the formula $d = vt$ where d is the distance, t is the elapsed time from the transmission of the sound wave, and v is the velocity of sound in the medium ($v \approx 1500 m/s$ in soft tissue). Since echo-ranging determines the location of interfaces by detecting the reflected echoes (of the sound wave) at the transducer, the time t represents the round trip time for

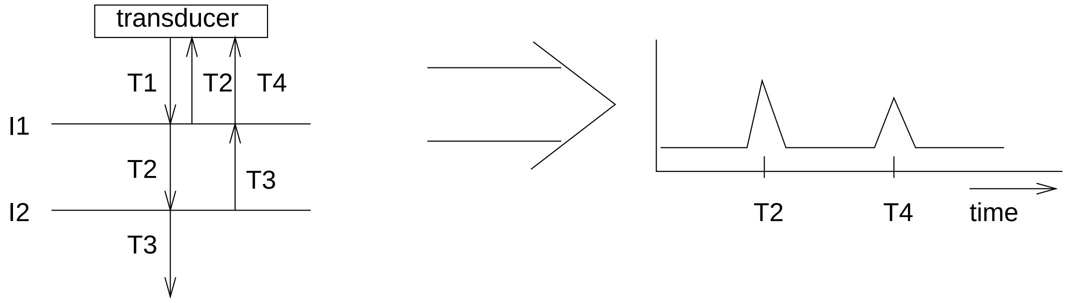


Figure 2.4: Propagation of echoes through tissue interfaces I_1 and I_2. Signal amplitude is proportional to echo strength at the transducer.

the sound wave to travel to an interface and for its echo to return to the transducer. Therefore, the actual distance of the interface to the transducer is $\frac{d}{2}$. Note that in calculating this distance it is implicitly assumed that the sound wave travels in a straight line between the transducer and the interface.

Now consider the process when a portion of the sound wave is reflected and a portion continues. Figure 2.4 shows the propagation of the echoes at successive time increments. At time T_1 the incident sound wave intercepts interface I_1 and is split into two components: the reflected echo (specular reflection) and the travelling (possibly diffracted) wave. At time T_2 the reflected echo is detected at the transducer, resulting in an output signal with amplitude proportional to the strength of the echo. Also at T_2, the continuing sound wave intercepts interface I_2 resulting in a second echo. At time T_3 the second echo reaches I_1 and, at time T_4, the second echo is detected at the transducer producing an output with amplitude proportional to the strength of the echo.

A number of factors greatly complicate this simple model. First, if the angle of the incident wave is not perpendicular to the interface then the echo will strike the transducer near its edge or will miss the transducer all together. The resulting output signal will be smaller in amplitude or non-existent. Thus, the orientation of the transducer with the interface has an effect on the resulting output signal. Figure 2.5 illustrates these cases.

Figure 2.6 illustrates the effects of diffraction and multiple specular reflections. Two interfaces cause successively smaller portions of the sound wave to be contin-

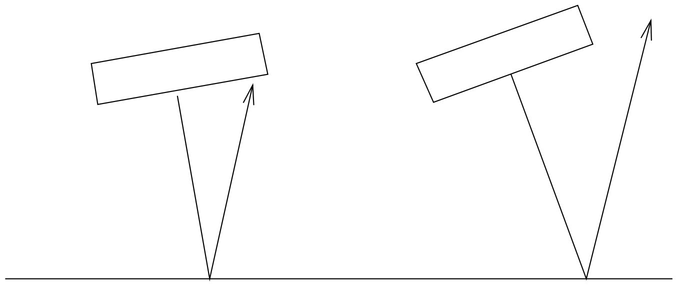


Figure 2.5: Echoes detected at the transducer are dependent on the orientation of the transducer with the interface.

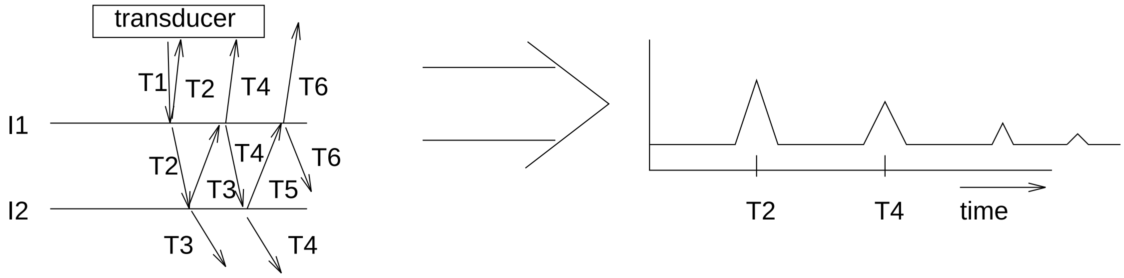


Figure 2.6: Effects of diffraction and multiple specular reflections. The presence of two reflective interfaces causes extra pulses to appear in the output signal.

ually reflected within the interfaces resulting in multiple echoes. Because the interfaces diffract the continuing portion of the sound wave, not all the echoes return to the transducer. It is also possible for some of the diffracted waves to return eventually to the transducer. Since it is assumed that all the waves travel in a straight line, these late arriving diffracted waves may be interpreted as echoes from a distant interface when they actually are not. Also, the diffracted waves may have little energy but their effect is to increase the noise in the image as their contributions are integrated into the output signal.

Another factor complicating the transmission of sound waves is that the medium itself attenuates the waves. Attenuation is the process of energy loss from a wave for any reason [95]. For a homogeneous soft tissue a common rate of attenuation assumed is 1 db/cm/MHz. By representing attenuation in this form it can be seen that attenuation is dependent on the frequency of the sound wave. As the frequency increases, the sound wave is increasingly attenuated. Also, the farther the sound wave travels, the more it is attenuated. For example, a sound wave with a frequency of 5 MHz will be attenuated by 5 db/cm. If the sound wave has an initial amplitude of 100 db then it can travel 20 cm before it has no energy left. This implies that interfaces at a distance of up to 10 cm from the transducer can be detected. It should be noted that this rate of attenuation will vary slightly, depending on the type of tissue through which the sound wave travels.

2.1.3 Diagnostic Imaging

One of the biggest challenges in ultrasonographic imaging is to present the acoustic data in a form which provides insight as to the structure of the tissues (objects) being imaged. It is also necessary for this presentation to be an accurate representation of the objects. The most common approach in achieving this is to represent the acoustic data as an image which can be visually (and subjectively) evaluated for diagnostic purposes. This section describes the various imaging modes used in ultrasonographic imaging. It also discusses some of the problems that can arise due to various artifacts in the image.

Imaging Modes

Christensen [21] provides a good review of the different types of imaging modes. The A-Mode or "amplitude" mode is a one-dimensional mapping of the tissue interfaces encountered along the line of propagation of the ultrasound beam. Figure 2.7(a) shows an example of an A-Mode scan. Each peak in the output represents an interface. The distance between the peaks indicates the distance between the interfaces. An advantage of the A-Mode is that it provides positional information quickly without requiring a significant amount of electronics and processing.

The B-mode or "brightness" mode is different from the A-Mode in that instead of displaying the echo strength as a vertical amplitude, the brightness of the output line is used to indicate the strength of the echoes along the line of beam propagation. Figure 2.7(b) shows an example of a B-mode scan for the equivalent A-mode scan of Figure 2.7(a). A real time B-mode image is simply the 2D arrangement of a lateral

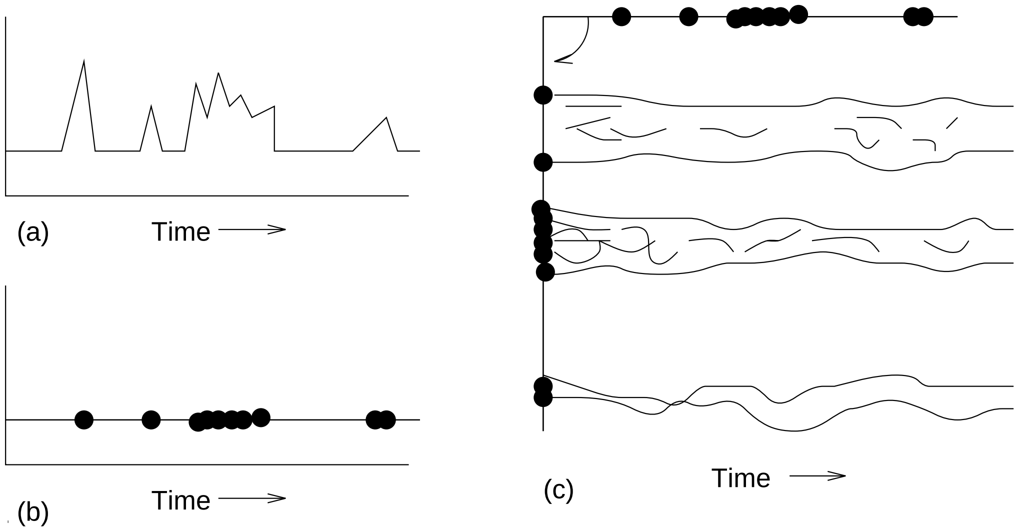


Figure 2.7: Various imaging modes; (a) A-Mode scan, (b) equivalent B-mode scan, (c) relationship between B-mode and M-mode.

set of B-mode scans. By combining the positional information of the transducer with the RF data (the 08 data is produced by sweeping the ultrasound beam over an angle θ) a 2D, real time B-mode ultrasound image is produced. Figures 2.10 and 2.11 provide examples of real time B-mode images. The real time B-mode is the most commonly used mode for diagnostic purposes.

The M-Mode or "motion" mode is used for examining the motion of the structures being imaged. Figure 2.7(c) shows the relationship between the B-mode and M-mode. An M-mode scan can be achieved by rotating the display of the B-mode scan by 90° and producing successive B-mode scans (with the same beam orientation). The M-mode scan will display the movement of tissue structures, with respect to a fixed point, over time. An M-mode scan is especially useful for studying the motion of structures such as the heart valves.

Another imaging mode that most modern ultrasonographic scanners provide is duplex imaging [101, 140, 10, 95, 21]. In the duplex mode, both the B-mode and Doppler information are combined in one image. The B-mode display is used to locate a desired region for Doppler interrogation. A line extending from the top to the bottom of the B-mode display is used to select the axial projection along which the Doppler pulse is to be transmitted. A cursor is used to select the depth along the line at which the sampling gate is to be positioned. Based on this positioning, the frequency shifted echoes from the reflected Doppler pulse are displayed, usually underneath or beside the B-mode display. This display shows the frequency spectrum of the echoes as a function of time. The direction of flow is indicated by the sign of the displayed frequencies; when the frequencies are positive, the blood flow is towards the transducer, while negative frequencies indicate that the flow is away from the transducer.

Another variant to displaying the Doppler information is to overlay the estimated flow velocities directly where they occur on the B-mode display. This is known as flow imaging [101]. A colouring scheme is used to indicate the direction of flow. Usually, red indicates flow in one direction while blue indicates flow in the opposite direction. Varying shades (scales) of each colour are used to portray different frequencies (velocities). This display provides the ability to visualise the blood flow characteristics within a region of tissue and thus can be quite useful for tissue characterisation (see Section 2.2).

Artifacts

Artifacts are undesirable effects in ultrasound images which can lead to erroneous diagnosis of the physiological structures being imaged. Furthermore, any automated processing of ultrasound images will be susceptible to artifacts. It is therefore necessary to identify the artifacts which occur in ultrasound images so that their effect can be minimised or eliminated.

The formation of an ultrasound image is based on the assumptions that the sound beam travels in a linear path and that the beam is infinitely thin. The resulting image will contain artifacts when these assumptions do not hold [40]. This section discusses the types of artifacts that can occur in an ultrasound image when these assumptions are violated.

Five categories of artifacts are discussed in [95]. These categories are:

- displaying of non-structural echo signals,

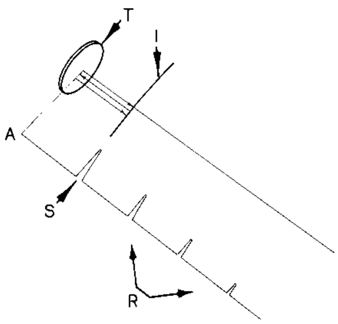


Figure 2.8: Formation of a non-structural artifact. Reverberation between the transducer, T, and a strongly reflecting surface, I, results in the appropriate output signal, S, but also the recurring echoes R; from Powis and Powis [95].

- removal of real structural echo signals,
- displacement of echo signals,
- distortion of the echo signal, and
- distortion of the organ dynamics.

Non-structural echo signals appear in an ultrasound image when reverberating echoes are incorrectly displayed as successive tissue interfaces. An ultrasound image displays the travel time of an echo as a distance from the transducer. Any reverberating echoes will appear as non-structural echoes over the distance of the display. Since these echoes do not represent actual tissue structures, they are undesirable. Figure 2.8 illustrates how a non-structural artifact is formed. A strongly reflecting surface near the transducer will cause repeated reflections (reverberation) between the transducer and the reflector. A similar effect will occur if the reverberation occurs between two highly reflective tissue interfaces as shown in Figure 2.9. These reflections will be interpreted as successive tissue interfaces along the axial direction of the ultrasound beam as shown in Figure 2.10. In this image a highly reflective interface close to the transducer causes reverberation resulting in a repeated pattern of high intensity reflections (indicated by the arrow).

Removal of real structural echo signals in the ultrasound image is primarily the result of acoustical shadowing. Shadowing occurs when the ultrasound beam can

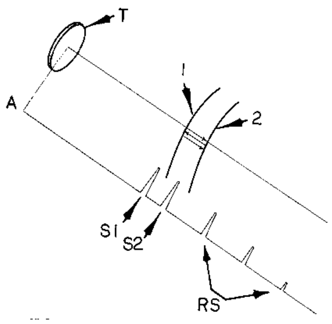


Figure 2.9: Formation of a non-structural artifact. Reverberation between two highly reflecting surfaces, 1 and 2, results in the appropriate output signals, S_1 and S_2, but also the recurring echoes RS; from Powis and Powis [95].

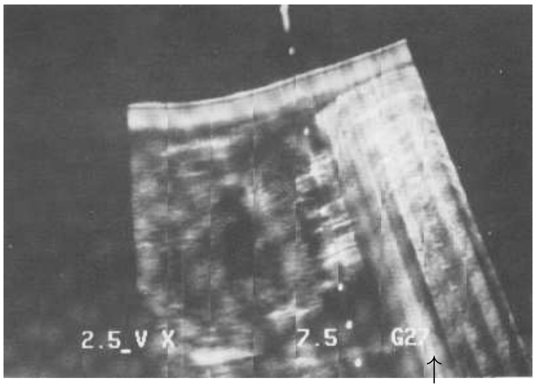


Figure 2.10: Non-structural echoes in a B-mode ultrasound image; from Powis and Powis [95].

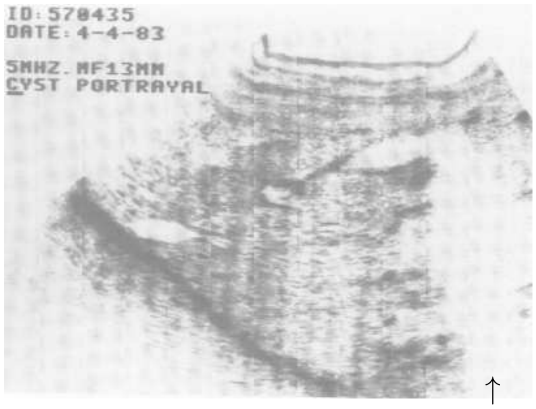


Figure 2.11: Acoustical shadowing in a B-mode ultrasound image; from Powis and Powis [95].

not penetrate beyond a certain tissue layer. As a result, tissue structures deeper than this tissue layer produce small or no echoes. The inability of the ultrasound beam to penetrate beyond a certain tissue layer is due to the presence of a strong reflector which reflects most of the ultrasound beam energy, to highly attenuating intervening layers which decrease the ultrasound energy, or to strongly scattering structures which disperse the ultrasound energy. Figure 2.11 shows an example of shadowing in which a highly attenuating structure results in the loss of echoes beyond the attenuating structure (area directly above the arrow).

Displacement of echo signals in the ultrasound image from their actual orientation is mainly due to the width (and shape) of the ultrasound beam. Since the beam is not infinitely thin, all information within the width of the beam will be incorporated into a single line of sight. Figure 2.12 illustrates how single point sources contribute to the echo amplitude as the ultrasound beam travels from left to right. It can be seen that the resulting echo produced for each point is dependent on the spatial resolution of the points and the width of the ultrasound beam.

A number of factors will cause distortion of the echo signal in the ultrasound image. If the TGC curve is set incorrectly then the amplitude of echoes at certain depths will be exaggerated or diminished in the resulting image. Re-adjusting the TGC curve may eliminate the distortion of the echo signal.

Distortion of the echo signal can also be caused by acoustical enhancement (oppo-

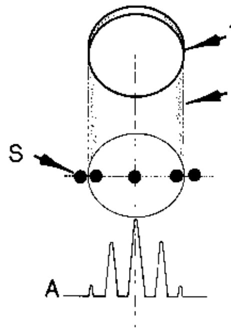


Figure 2.12: Displacement of echo signals. The echo from source, S, will be integrated into the signal, A, as long as S is within the width of the beam, B, produced by the transducer, T; from Powis and Powis [95].

site of acoustical shadowing). Acoustical enhancement occurs when the ultrasound beam passes through a low attenuating medium (such as fluid). This causes distortion of the echo signal as structures deeper than the low attenuating medium will have a higher than normal echo amplitude.

Another factor causing distortion of the echo signal is the orientation of the ultrasound beam with a tissue interface. When the ultrasound beam intercepts a tissue interface at an orientation other than 90°, the axial resolution will be decreased due to the width of the beam. Figure 2.13 shows how the axial resolution is affected by the orientation and width of the ultrasound beam. Note that the resulting echo corresponds to an increase in the width of the tissue interface (decrease in axial resolution) as the orientation of the beam strays farther from 90°.

Distortion of organ dynamics occurs in an ultrasound image when the structures being imaged move during formation of the image. It takes a finite amount of time to collect all the echoes which form the ultrasound image, therefore, any movement is sampled at a particular instant. If the movement is not sampled fast enough (i.e., the frame rate is too slow) the resulting image may be blurred. Also, motion aliasing could result if the sampling rate is less than twice the highest frequency of motion. For example, if a heart, beating at 60 beats per minute, is sampled at the same frequency (1Hz) and in phase then in the ultrasound images, it will appear that the heart is not beating at all.

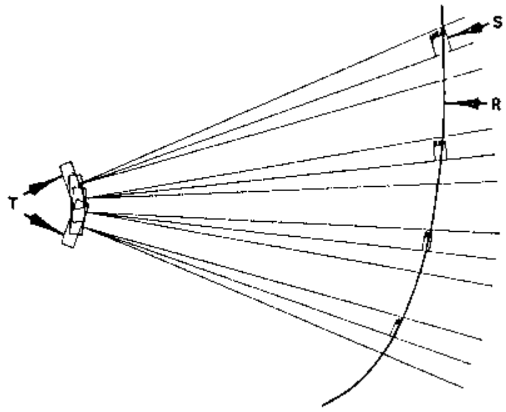


Figure 2.13: Effects of beam width and orientation on axial resolution. The signal, S, is wider for beam orientations other than 90° with surface, R. The artifact disappears when the transducer, T, orients the beam at 90° with R; from Powis and Powis [95].

Another type of artifact not covered in the above categorisation is speckle. Wagner *et al.* describe speckle in the ultrasound image as the fluctuation of the gray scale intensity about the mean for a given area [136]. The fluctuation is due to the accumulation of random scatterings whose phase components vary randomly from 0 to 2π. Powis and Powis describe speckle as the result of phase sensitive interference patterns of echoes arrive at the transducer at the same time as the primary echoes from tissue interfaces and scattering bodies [95]. Burckhardt describes speckle as an undesirable property of the ultrasound image which masks small differences in the gray scale intensities [11]. Based on these descriptions, it can be concluded that speckle is an undesirable property which contributes to the amount of noise present in the ultrasound image. Because of the inherent random scattering that occurs during formation of the ultrasound image, it is impossible to prevent speckle. However, by removing the system effects of the ultrasound machine (such as with a calibration spectrum [64, 65, 66]) from the ultrasound data, it is possible to reduce the effects of speckle.

Kling *et al.* [51] suggest a dual frequency method for reducing the speckle. This approach is based on the idea that random echoes produced by small scatters are frequency dependent while reflected echoes from flat boundaries are frequency independent. Scanning the same region at two different frequencies (5 MHz and

7.5 MHz) results in a pair of images which have different speckle patterns. Taking the difference between these images (several ways for calculating the difference are used) results in an image which contains only the pixel intensities due to differences in the speckle. Echoes from flat boundaries appear as dark regions within the difference image. Experimental results demonstrate that the approach is adequate at removing reverberation type echoes, however, there are some problems. First, since the specular echoes do not appear in the difference image, the task of identifying tissue interfaces (boundaries) will be made more difficult. Second, the difference operation (possibly) results in variation of the speckle pattern and a loss in signal-to-noise ratio. Kling *et al.* are not sure of the effect of this. Finally, the acquisition time of dual frequency images is longer than for a conventional scan. This increases the potential for alignment error of the images due to movement of the patient and transducer.

Texture of Tissue

Texture in an ultrasound image can be described as the pattern of intensity variations for a corresponding region of tissue. The ultrasound texture can be characterised by the shape, separation, and intensity variations in the image [95]. These characteristics vary depending on the following factors:

1. the physical properties of the tissue such as size and separation of scattering bodies,

2. the type of signal processing applied to the ultrasound signal prior to image formation (or scan conversion),

3. the extent to which spatial interference patterns (speckle) contribute to the detected signal at the transducer, and

4. the extent to which forces from the ultrasound beam modify the architecture of the scattering bodies comprising the tissue.

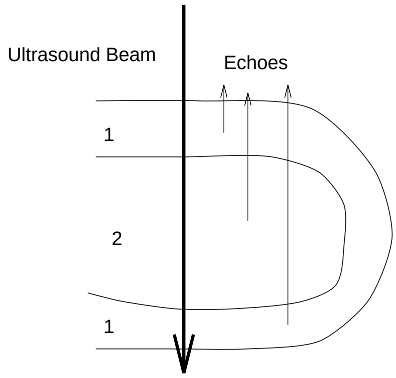


Figure 2.14: Inconsistency of texture at different depths. Intervening tissue layer 2 results in different textures for upper and lower portions of tissue 1.

It is desirable that the texture in an ultrasound image to have a consistent appearance for the same tissue. If the texture of the homogeneous tissue varies within the image, it is necessary to determine the factors which contribute to the variation of the texture. Once these factors are determined, it is possible to adjust the ultrasound image so that the texture is consistent for a homogeneous tissue over the entire image.

The biggest problem with texture in ultrasound images is its dependence on depth. The textural presentation is not the same from the top to the bottom of an image for the same tissue [95]. The main reason for the textural inconsistency is the intervening tissue layers between the transducer and the tissue layer of interest. Consider the tissue structures as shown in Figure 2.14. Since the ultrasound image is formed by the echoes of a transmitted ultrasound beam, the received echoes from the top portion of tissue 1 are not attenuated by any other tissue layers. However, the top portion of tissue 1 will attenuate echoes returned from deeper structures (tissue 2 and bottom portion of tissue 1). For this spatial arrangement, the texture of tissue 1 will appear different for the top and bottom portions. The intervention of tissue 1 causes the texture of tissue 2 to appear different than it would if the echoes were solely from tissue 2. Also, since the strength of the attenuated ultrasound signal decreases with depth, the returning echoes further from the transducer will be weaker than those closer to the transducer. Thus, the echoes from the bottom portion of tissue 1 may be too weak to be detected by the time they return to the transducer resulting in the loss of the original texture for tissue 1.

Despite the degradation of the ultrasound images caused by intervening tissue layers, it is possible to correct for some of the distortion. Carpenter *et al.* [13] suggest a forward-propagation technique in which refraction of the echoes at a fat-muscle interface can be compensated for by adjusting the focus and steering of the ultrasound beam. Experimental results obtained from phantom and clinical data demonstrate that gross artifacts and texture inconsistencies can be corrected. The major disadvantage of this approach is that it requires *a priori* knowledge of the geometry of the tissue layers in order to perform the correction. Furthermore, this knowledge is obtained from manual examination of the original distorted image.

2.2 Tissue Characterisation

The purpose of tissue characterisation is to estimate physical tissue characteristics by measuring features obtained from the ultrasound data. If the tissue characteristics can be estimated accurately then it is possible to identify tissues. The measured features can also provide the ability to differentiate among the tissues present in an ultrasound image. This makes it possible to segment the ultrasound image into homogeneous regions which are useful for subsequent 3D modelling of ultrasound images [77].

The usual approach taken in ultrasound tissue characterisation is to estimate a number of parameters based on the ultrasound data. This includes estimation of attenuation [42, 14, 60, 137], backscatter coefficients [53, 134], scatter sizes [64, 63, 65, 66, 42, 103, 29], and other effects such as machine characteristics [145, 133] and speckle [136, 11]. These estimations are then compared to known tissue parameters to determine the tissue type. A survey of these model-based approaches is provided in Section 2.2.2. An alternative approach to tissue characterisation is to evaluate the texture present in the ultrasound images. The next subsection describes this approach.

2.2.1 Image-Based Features

Texture in an ultrasound image can be characterised by the dot size, shape, separation and intensity variations in the image [95]. If the texture in an ultrasound image is unique for each different tissue, then it is possible to identify each tissue based on its texture in the ultrasound image. Previous work has used this approach to characterise [78, 81], segment [80, 79] and subsequently model [77] ovarian follicles using ultrasound images. A multiresolution framework is used in characterising a set of texture classes. Any number of features can be used to characterise each texture. The goal of the framework is to evaluate the performance of the features in characterising each texture as well as to provide a "feature table" which parameterises each texture according to the features used. Feature performance is evaluated using robust statistics and other criteria such as the class coverage and the outlier value.

There are a number of advantages with this approach:

- the multiresolution approach characterises textures from small (4×4 pixels) to large resolutions enabling a subsequent segmentation step to adapt to various levels of detail in the image,
- the framework provides the ability to evaluate any number of features,
- the framework can determine the minimal set of features required to uniquely characterise each texture class,
- it is possible to characterise a large number of texture classes.

Wu *et al.* [144] also use texture features in classifying ultrasound images of liver. In their approach a common set of texture features (Laws' energy, Cooccurrence, statistical, Fourier power spectrum based features) are used to classify three types of liver tissues. They also propose a new set of multiresolution fractal features to detect diseased liver tissues faster and more accurately than the above mentioned features. The Bayes classifier and Hotelling trace criteria are used to evaluate the performance of the features in classifying the three types of liver tissues. The main problem with this approach is that it classifies a given set of homogeneously textured images rather than characterising each of the given textures. This is an important distinction when further processing of the ultrasound images is desired. For example, the approach used in previous work [78, 81, 77, 82] builds a table of features characterising each texture. This table can then be used to classify a given set of homogeneously textured images or can be used in a segmentation step to segment any ultrasound image containing the characterised textures.

Another problem with this approach is that it is sensitive to the selection of the regions containing the homogeneous liver tissue. Unlike the approach in previous work which is robust to inconsistencies (i.e., outlier data values), any inconsistencies (such as blood vessels) contained in the homogeneous structure of the regions will result in a decrease of classification accuracy.

The advantage of the image-based approaches is that they can be applied to different types of textures and images. Characterisation and classification are not limited to those textures in ultrasound images because the features used do not depend on the physical process involved in forming the ultrasound image. It is possible for the image-based approaches to be applied to different imaging modalities such as CT scans or MRI (although the effectiveness demonstrated for ultrasound images may not be the same for other imaging modalities).

The general applicability of the image-based approaches can also be a weakness. By considering the relationships among tissue structures, the ultrasound process, and the resulting ultrasound images, it is possible to gain more information for use in characterising tissue. Since the image-based approaches use only the B-mode image, this extra information is not available. For example, a more effective and precise set of features may be determined by accounting for imaging artifacts, such as acoustical shadowing and enhancement or by knowing the relationship between scattering bodies and their corresponding backscattered echoes. By modelling these interactions, a set of ultrasound-specific features can be determined. The next subsection describes features based on models of the ultrasound process.

2.2.2 Model-Based Features

A model of the ultrasound beam-tissue interaction is used to predict the contribution of various tissue characteristics to the resulting ultrasound data. Based on the model, features can be identified which are useful in uniquely identifying the tissue in the ultrasound data. Some features, which have been extensively used in the literature, are the backscatter and attenuation coefficients [53, 134, 104]. The backscatter coefficient is a measure of the effect in which a tissue structure reflects the ultrasound beam. Scattering body size and separation are tissue characteristics which directly affect the backscatter coefficient. The attenuation coefficient is a measure of the effect in which a tissue structure modulates the ultrasound beam in both frequency and amplitude. Tissue density will directly affect the attenuation coefficient.

Models and their corresponding features are either time-based [145, 137, 116] or frequency-based [64, 63, 65, 66, 29, 28]. Time-based models typically use the amplitude and pulse separation of the RF data in estimating tissue features. Frequency-based models use the frequency spectrum to estimate appropriate features. An excellent review of spectrum analysis as applied to ultrasound RF data is provided by Lizzi *et al.* [28]. Basically, the time dependent RF data is transformed into the frequency domain using a Fourier transformation which decomposes the signal into a spectrum (or "tissue signature") of differing frequency sine waves, each of which has a specific amplitude and phase. The main advantage in using the frequency spectrum of the RF data is that system effects can be minimised with the use of a calibration spectrum. The calibration spectrum represents only the effects of ultrasound instrument. Normalisation of the original spectrum with the calibration spectrum leaves only those components which are a result of the tissue-ultrasound beam interaction.

Lizzi *et al.* [64, 65, 66, 28] have done extensive research in tissue characterisation using spectral analysis. A detailed analytical model is used which relates the spectrum of the RF signals to tissue microstructure. Along with the model, calibration techniques are described which are useful for reducing system effects arising from the transducer and other machine components. Based on the model, a number of relationships between tissue microstructure and parameters of the frequency spectrum are determined. The slope and intercept parameters of the spectrum can be used to predict characteristics of the tissue microstructure including size, concentrations, and impedance of the elements comprising the tissue. The results predicted by the model are compared to a database of clinical data and a fairly high correlation is found.

The advantage of Lizzi *et al.*'s approach is that it minimises the effects of the ultrasound machine by using a calibration spectrum to normalise the spectrum of the RF data. Also, access to a large database of clinical data allows for accurate verification of the model. The disadvantage of this approach lies in the use of the spectral slope for characterising the tissue structure. The slope is estimated from the spectrum of RF data and is sensitive to the method used. Also, if there is high variability in the amplitude of the spectral components (as is evident from the experimental results obtained by Lizzi *et al.*) there will be significant error in estimating the slope. The slope is also used to predict the spectral intercept, the value of which is also subject to error. Thus, the relationship between the spectral parameters (features) and the tissue microstructure predicted by the model will have a high degree of variability. The variability may be high enough to limit the effectiveness of the spectral features in uniquely characterising similar tissues.

The approach used by Shmulewitz *et al.* [137, 116] in characterising tissue is to estimate attenuation using time domain analysis of the echo amplitude. It is assumed that variations in attenuation reflect variations in the homogeneity of the medium. Thus, tissues having different structures should have different attenuation coefficients.

A simple exponential model of the echo signal is used in estimating the attenuation coefficient. By examining a small region of interest and assuming the region is homogeneous, the model can be simplified into an over-determined system of equations. An estimate for the average attenuation of the region can then be calculated such that a certain error measure is minimised.

The unique characteristic of this approach is that an attenuation map is created to display the local attenuation estimate for each pixel in the B-mode image. The attenuation map is an alternative to the traditional B-mode image for displaying ultrasound data. Its purpose is to provide useful diagnostic information which can lead to the differentiation of diseased tissues. Unfortunately, experimental results show that displaying the attenuation estimates do not increase the detectability of the diseased tissue types.

There are a few more disadvantages with Shmulewitz *et al.'s* approach. A fairly large data window (e.g. 17×65 pixels corresponding to 17 RF lines each of 65 points) is required to achieve reliable estimates of the attenuation. Using a large window reduces the amount of detail that can be achieved in the resulting attenuation map. The model of the echo signal does not account for the frequency dependency of attenuation. As a result, incident waves of differing frequency reflected by the same tissue structure will have different attenuation estimates. This makes it impossible to consistently identify the same tissue using attenuation.

Yao *et al.* [145] characterise tissue using the backscatter and the attenuation coefficients. In this method a reference phantom is used in estimating the coefficients. Using a time domain technique, frequency and depth dependent ratios are obtained between the echo data from a homogeneous sample and the echo data from the phantom whose backscatter and attenuation coefficients are known. From these ratios the backscatter and attenuation coefficients are estimated.

The advantage of this approach is that it requires no explicit knowledge of the transducer beam pattern or the instrument dependent factors in estimating the attenuation and backscatter coefficients. Also, since calculations are performed in the time domain as opposed to the frequency domain, coefficient estimation can be done in real time. A disadvantage of this approach is that it depends on the accuracy in which the backscatter and the attenuation coefficients of phantom are known. If the phantom construction does not adhere to its given coefficients then error will result in the estimated backscatter and attenuation coefficients. Also, the accuracy of the coefficient estimates are sensitive to the data window size used. If the window is too small, significant error in estimation will result.

Insana *et al.* [42, 74] use pattern recognition techniques to determine a set of ultrasound tissue features which are effective in classifying normal and diseased liver tissues. They categorise tissue based on structure into three main classes and determine a set of four features that they claim are effective in completely characterising tissues in each of the classes. The features are based on first and second order statistics of the backscattered intensity as well as the attenuation coefficient. These features are obtained from various tissue models which are based on spectral analysis of the RF data. Results show that there is information present in the second order statistical features which is not present in the ultrasound B-mode image.

The advantage of this approach is that a number of different models can be evaluated to determine how effective each model's features are in classifying a common set of tissues. The disadvantage of this approach is common to all classification approaches. Insana *et al.*'s approach does not retain the tissue signatures obtained during the classification process. This makes it difficult to segment an ultrasound image into homogeneous regions, based on tissue type, in an unsupervised manner.

Romijn *et al.* [104] also evaluate a number model-based features in differentiating among a number of intraocular melanomas. The attenuation and the backscatter coefficients as well as a set of image-based features are evaluated. Six different models are used to estimate the attenuation coefficient. These models are based on spectrum analysis and on time domain analysis of the ultrasound signal amplitude. The features are calculated from simulated data, actual data from clinical examinations, as well as data obtained from a tissue mimicking phantom.

Each feature is calculated from a number of samples of each type of melanoma. Evaluation of six different attenuation features is performed to determine which model is the most effective in estimating the attenuation coefficient. Evaluation of all features is performed to determine which features are effective in differentiating among the different melanomas.

There are a number of advantages with this approach:

- diffraction correction is applied to the ultrasound data to minimise effects of the ultrasound beam profile,
- a preprocessing step is used to automatically select a homogeneous region of interest,
- an objective evaluation of a number of features is performed using a common set of ultrasound data.

A disadvantage of this approach is that the accuracy of feature estimates is greatly reduced by inhomogeneities in the selected region of interest. This implies that a significant amount of confidence must be placed in the preprocessing step to choose a homogeneous region for feature estimation.

2.2.3 Problems

The general applicability of the image-based approaches can be a weakness. By considering the relationships among tissue structures, the ultrasound process, and the resulting ultrasound images, it is possible to gain more information for use in characterising tissue. Since the image-based approaches use only the B-mode image, this extra information is not available. For example, a more effective and precise set of features may be determined by accounting for imaging artifacts such as acoustical shadowing and enhancement or by knowing the relationship between scattering bodies and their corresponding backscattered echoes. By modelling these interactions, a set of ultrasound-specific features can be determined.

It is obvious that the model-based features described above can only be used for specific applications (i.e., ultrasound). Also, a major problem with the model-based features is that most, if not all of the approaches require the RF data as input. The reluctance of ultrasound vendors to disclose any information about internal data formats and other processing techniques applied to the RF data makes it apparent that a model-based approach is not feasible in this thesis research.

2.3 Image Segmentation

The goal of image segmentation is to identify homogeneous regions in an image. The homogeneity of regions can be based on a number of factors including, texture, colour, or distribution of pixel intensities. Most of the research in image segmentation has focused on segmenting 2D images. The result of the segmentation is either an image of labelled pixels (each pixel in a homogeneous region having the same label) or a set of contours describing region boundaries. If the data to be segmented is 3D, such as the data obtained from a series of cross-sectional ultrasound, CT, or MRI images, typically, each 2D image is segmented to obtain a set of regions or contours. Then each of these contours is "stacked" together to form a 3D representation of the homogeneous regions. Interpolation and surface fitting algorithms are then used to estimate the boundaries of the homogeneous regions. Recently, a few approaches have been proposed to segment 3D data by incorporating the third dimension. Thus, rather than segmenting one 2D image (slice) at a time, the entire volume of data is segmented. The resulting segmented data from this type of approach should have less error since surface interpolation in the third dimension is incorporated into the segmentation process, rather than being performed after segmentation.

The next two sections describe some of the approaches to 2D and 3D image segmentation. Section 2.3.1 highlights some of the approaches to 2D image segmentation. As well, previous work in 2D image segmentation is briefly described. Section 2.3.2 describes some of the approaches to 3D image segmentation. Problems associated with segmentation algorithms are discussed in Section 2.3.3.

2.3.1 2D Segmentation

Numerous approaches to 2D image segmentation have been proposed in the literature. This includes methods based on texture [80, 79, 77, 15, 22, 23, 141, 68, 109, 118, 146, 50, 123], mathematical morphology [124, 47], neural networks [126, 68], stochastic relaxation [34, 54], region growing [9, 43, 48, 122], and intensity extrema

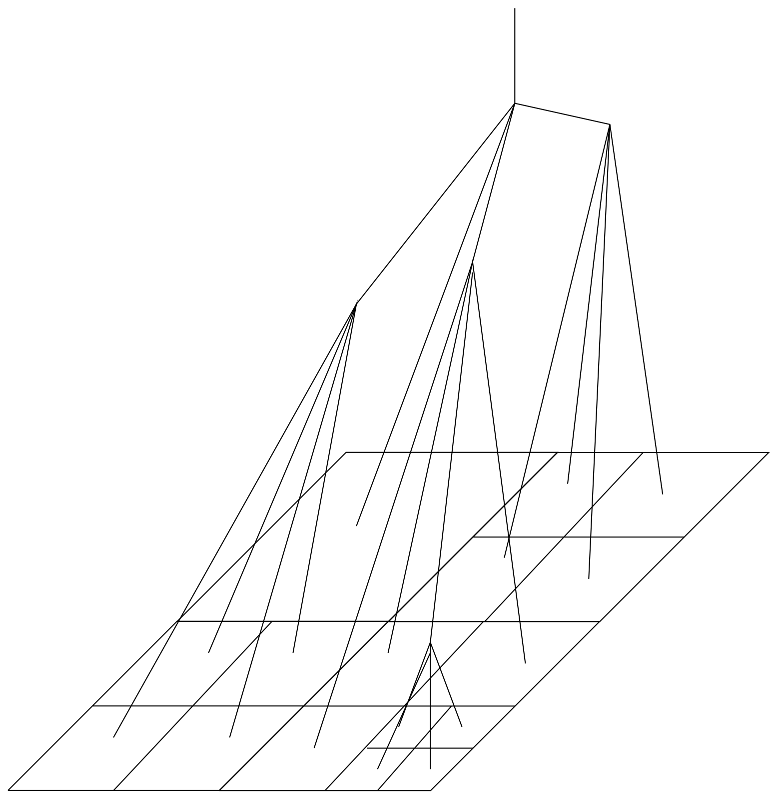


Figure 2.15: Quadtree structure.

[58]. Surveys of segmentation approaches can be found in [100, 90]. Two approaches to segmentation are described in this subsection. They are described because of their similarities as well as their inherent ability to be extended to 3D segmentation. The success of previous work in 2D segmentation is also motivation for a description of the second approach.

A split, merge and group (SMG) approach is proposed by Strasters and Gerbrands [122]. In this approach, an $M \times M$, $M = 2^n$, image is represented by a quadtree structure in which each leaf node corresponds to a block of pixels in the image [8, 110, 111, 112, 38, 41, 94, 3, 73, 12]. Figure 2.15 depicts a quadtree rooted at level 0. The quadtree is expanded to an initial level, s, in which each leaf node represents pixel blocks of size $2^{n-s} \times 2^{n-s}$ where $0 < s \leq n$. Next, a merging phase is performed in which nodes at level s', $0 \leq s' \leq s-1$, are examined. If the node's children at level s are homogeneous, then the children are removed from the tree and the node at level s' becomes a leaf node. The merging phase is repeated for each node at each level s'.

After merging, the splitting phase is performed. The inhomogeneous nodes at level s are split into 4 children at level $s+1$. Then, all the leaf nodes at level $s+1$ are examined and any inhomogeneous nodes are again split into 4 children. This process continues until no more splitting occurs or until level n (a single pixel) is reached. A grouping phase is next performed to group homogeneous nodes which are not direct descendants in the quadtree. Quadtree operations such as ADJ, REFLECT, and DIRECT [110, 111, 112] are used to determine nodes which are adjacent in the image and can be grouped into the same region.

There are a number of problems with this approach. First, it is apparent from the above description that the initial level, s, at which the quadtree is initialised influences the final segmented result. If s is too low then detail may be missed in the image. If s is too high then small regions will be produced. This is especially a problem for larger regions which contain a repetitive texture and appear to be homogeneous at larger resolution.

Another problem with this approach is that it is possible to get stuck in a local minimum solution. This is due to the rigid manner in which merging and splitting are performed. Also, since there is no probabilistic model used in controlling the merging and splitting of nodes, the order in which decisions for splitting or merging are made will have an effect on the resulting segmented image.

Finally, the homogeneity criteria used in determining the splitting or merging of nodes does not depend on the size (resolution) of the block of pixels. It was found in previous work that the resolution at which pixel blocks are examined has an effect on determining the homogeneity of the block [77, 81, 78]. This again emphasises the importance of selecting an appropriate initial level in the quadtree to initiate the segmentation process in the approach of Strasters and Gerbrands [122].

A similar approach to 2D image segmentation is the MTS algorithm proposed in previous work [80, 79, 77]. In the MTS algorithm, the image is stored in a quadtree structure. A split and merge (no grouping is required) approach is used to determine the most appropriate size (resolution) for a homogeneous block of pixels, based on a measure of the texture present in the block. A block is split if its 4 children produce a stronger measure than its own. A block is merged with its 3 neighbours if its parent produces a stronger measure than its own and its 3 neighbours. The best texture feature to be used at a given resolution is stored in a feature table, the design and construction of which is described in [77].

The MTS algorithm uses simulated annealing [131, 96, 1, 31] to control splitting and merging of blocks. Simulated annealing is a technique used for large scale optimisation problems. Typically, these problems involve combinatorial minimisation. There is an objective function (energy configuration) to minimise and the space in which the solution is obtained is a large, discrete configuration space such as all the possible allocations of labels to pixel blocks in an image. Image segmentation can be viewed as a combinatorial problem in which pixels or blocks of pixels are to be grouped into one of possibly many regions. This grouping is minimised when each region contains pixels having the same texture.

In the MTS algorithm, the energy for a configuration in the system is a function of the quadtree structure, x. Note that the structure of x is different for each new configuration due to the splitting and merging of nodes.

The energy for a configuration in the system is represented as

$$U(x) = \sum_{C \in S_x} \Phi_x(C) \tag{2.1}$$

where

x is the quadtree structure

C is a leaf node in the quadtree

S_x is the set of leaf nodes in x.

$\Phi_x(C)$ measures how well the current block fits the texture region in the image. If the current block is too small or too large (based on measures of the texture), then it must be split or merged. In this case, $\Phi_x(C) = 0$ and the energy in U remains the same. When the current block is split or merged independent of its texture measures (due to a random state change imposed by the simulated annealing approach), $\Phi_x(C) = 1$ and the energy in U is increased. When node C is not split or merged, $\Phi_x(C) = -1$ and the energy in U is decreased. The minimum energy in U occurs when no splitting or merging takes place for every leaf node in S_x.

Based on Equation (2.1) and the simulated annealing framework, the MTS algorithm performs a segmentation of an image by iterating until no more splitting and merging of nodes in the quadtree takes place or some maximum number of iterations is reached. In the first case the minimum energy state is reached. In the second case an approximation to the solution results. It is possible that this approximation does not provide a reliable estimate, but the approximation can decrease the time it takes to produce an initial segmentation.

Since blocks are labelled as they are examined, when iteration completes, all blocks should be labelled. In some cases (terminating the segmentation process at a maximum number of solutions, for example) it is possible that some unknown blocks (blocks having no texture class label) may exist. The identity of these blocks is determined by assigning each block to the texture class with the closest characteristics to the block.

The major difference between the MTS approach and the SMG approach is that MTS uses simulated annealing to control splitting and merging of blocks. Unlike the SMG approach, the decision to split or merge a block is performed dynamically. Also, the use of simulated annealing allows blocks to be examined numerous times and allows for random splitting and merging to occur. This minimises the chance of getting stuck in a local minimum solution.

Another important difference between the two approaches is that the MTS algorithm uses a texture characterisation component [77, 81, 82, 78] to determine the characteristics of the different types of homogeneous regions (texture classes) present in the image. This characterisation is dependent on the resolution of the blocks examined during the segmentation process. As well, numerous features are used in producing the characterisation, but only the most appropriate set of features for a particular resolution are used by the segmentation algorithm in deciding to split or merge a block of pixels. The characterisation of texture classes also provides the ability to identify (label) regions during the segmentation process.

In contrast, the SMG approach uses measures (or criteria) of homogeneity which do not take into account the resolution of the pixel block being examined. Also, since the SMG approach does not incorporate a texture characterisation component, it is quite difficult to determine the representative characteristics of the regions in the image. Thus, it is difficult to determine the values of the parameters required in the homogeneity criteria. It is also difficult to determine the identity of the regions during the splitting and merging process. Thus, the grouping phase is used in a "region growing" manner to determine the region identities.

2.3.2 3D Segmentation

Due to the increasing popularity of data visualisation (see Section 2.4), segmentation of 3D datasets is becoming more important. There are currently a few approaches to 3D segmentation reported in the literature [107, 108, 122, 46, 59]. This subsection describes some of these approaches.

The computationally easiest method for performing 3D segmentation is to use an "interactive" method. In this approach, a knowledgeable user must manually segment the data into regions based on their knowledge of the objects in the image as well as their ability to visually interpret the images. Sakas *et al.* [107] claim that the current segmentation methods offered by the image processing and computer vision communities "lack generality or require massive computational effort." Thus, these methods can not be practically used in any clinical system. They propose several methods for segmentation which basically clip regions within the volume. The methods differ in how the user must select the regions which are to be rendered or discarded. The big drawback of interactive segmentation is the effort required from the user. Each time the dataset is changed the user must perform the tasks required to segment the data. Reproducibility is also a concern. Diagnosis of clinical 2D ultrasound images is already subject to variability among physicians. Certainly, any segmentation results based on this subjective evaluation will suffer the same inconsistencies. Sakas and Walter [108] acknowledge these problems and propose a segmentation method, BLTP (binarize, low pass, threshold and propagate) which isolates a "region-of-interest" in the volume data. This method uses multiresolution filtering and a mathematical morphology like operation to produce a mask which identifies points in the original volume which are to be rendered. Since the mask can be computed fairly quickly (under one minute) it is possible to use the approach in (near) real time. However, the method does still require interaction on behalf of the user for selecting parameters such as threshold values and filter sizes.

Strasters and Gerbrands [122] extend the SMG approach to segment 3D images. 2D segmentation using the quadtree extended to 3D using an octree approach. With the octree (see Section 5.1 for more details), pixel blocks of size $N \times N \times N$ (as opposed to blocks of size $N \times N$ with the quadtree) are examined. The split, merge, and grouping phases are performed in basically the same manner as in the 2D approach. The biggest difference between the 2D and 3D versions is that in the latter the homogeneity criteria must be modified so as to incorporate the additional data in the third dimension.

Note the restriction on the size of the third dimension in the 3D version of the SMG approach. By the definition of the octree structure, the block of pixels (size $N \times N \times N$) examined is isometric. This implies that data being segmented should also be isometric (same resolution in each dimension). For data obtained as a series of cross-sectional images, it is often the case that the number of cross sections is less than the size of each cross section. It is therefore necessary to use interpolation so that the resolution along the axial direction (third dimension) is the same as that of cross sections. Alternatively, each cross section can be sub-sampled but a significant amount of data is lost.

Joliot and Mazoyer [46] propose a method for segmenting 3D MRI data of the brain. Segmentation of the 3D image is used to identify two different regions (gray matter, GM, and white matter, WM) and is performed in a series of steps. The first step involves selecting thresholds which can be used to identify GM and WM.

This is accomplished by manually sampling regions (from individual slices) within the 3D image to determine the range of intensity values for WM and GM. Once the thresholds are determined, they are used to create two new 3D images (one for GM and one for WM) from the original. The next step is to segment the WM image using a 3D connectivity algorithm. Basically, this involves selecting seed points within the volume and growing regions using connected voxels. The GM image is segmented in the same manner. Finally, both images are combined into a single 3D image. Mathematical morphology is used to perform interpolation between slices, prior to rendering.

The major problem with this approach is the first step used in selecting thresholds. Since threshold selection is based only on the average intensity within a desired region (GM and WM in this case), the process is very susceptible to noise. As well, the thresholds have to be determined for each new 3D dataset. Thus, if the level of noise in the image is high (as is the case with ultrasound images), it is very likely that the first step (and subsequently the remaining steps as well) of the segmentation process will fail.

Liou and Jain propose a parallel 3D segmentation algorithm which uses α-partitioning and volume filtering to identify regions in which the variation of intensities can be described by a regression model [59]. The use of α-partitioning eliminates the need to use region growing as a method for identifying homogeneous regions (volumes). The purpose of α-partitioning is to generate volume hypotheses based on the gradient information present in the image. Each of these hypotheses are then verified through volume filtering and invalid hypotheses are rejected. Invalid hypotheses are those which include volumes containing intensities from 2 or more distinct regions.

The biggest problem with this approach is that only the gradient information is used in determining possible volume hypotheses. Thus, the approach is only appropriate for images which have strong boundaries between regions. Also, in the presence of noise or regions having a highly varying texture (as is the case with ultrasound images), it is likely that α-partitioning will produce many small volumes and the resulting segmentation will be highly fragmented.

2.3.3 Problems

There are a number of problems associated with image segmentation approaches in the context of segmenting 3D datasets. With 2D segmentation, the potential increase in information which is available in the third dimension is not used since typical processing involves segmenting each 2D image and then combining the resulting segmented images in some manner. This also implies that the third dimension is not treated in the same manner as 2D images. Finally, when the 2D segmented images are "stacked" together, interpolation is used to determine the 3D surface representation of the segmented region (object). Since the interpolation is performed from the segmented images, there is no knowledge of corresponding points between successive images. Thus, error is introduced during interpolation.

The biggest problem in 3D segmentation is determining how the data in the third dimension can be effectively used. For segmentation approaches based on texture, such as the proposed 3D MTS algorithm in Section 5.1, the biggest challenge is to determine 3D texture features which can be used to characterise texture in 3D data. One of the contributions of the Ph.D. work is to determination of these 3D features.

2.4 Data Visualisation

The goal of data visualisation is to represent pictorially the relationships among variables within multidimensional datasets. Currently, visualisation of 3D datasets, especially those obtained by different imaging modalities (e.g., CT, MRI, and ultrasound) in the field of medical imaging, is achieving significant attention from researchers. Data visualisation is used as a tool by researchers and physicians to discover relationships within the data that previously could not be obtained by examining the data values themselves or by displaying 2D images (as in the case of medical imaging).

This section describes techniques for visualising data as well as some problems associated with these techniques.

2.4.1 Rendering

Rendering is the process of creating a visual representation of a scene based on a description of the scene. Typically, a 2D image is rendered from a 3D scene description. The scene can be described by mathematical equations or by data points distributed within a volume. Topics such as hidden surface removal, shading models, surface properties and shadowing are important for producing a rendered image which corresponds to our expectations of how the scene should appear.

Rendering techniques can be classified into two general categories:

- *surface rendering* techniques,
- *volume rendering* techniques.

In surface rendering [138, 127, 35, 125, 18, 128, 77, 139], a geometrical representation of the data such as edges (contours), mesh [113], polygons, triangles, pixels or voxels (3D representation of a pixel) is used to model the object or structure to be visualised. When a 3D dataset of voxels (assuming the dataset is an isometric grid of cubic voxels) is used, the surface is typically modelled as a thin "shell" with voxels either belonging to the surface (totally opaque) or not (totally transparent). A segmentation algorithm is used to classify each voxel and a surface representation is obtained using a surface fitting algorithm such as the marching cubes [139, 138]. Surface normals (which indicate the orientation of a local surface patch) are usually calculated from the geometric primitives representing the surface and in some methods the image gradient is also used [128]. The image gradient is an estimate of the change in surface orientation from one patch to another. It is calculated from the pixel intensities where a change in pixel intensity is assumed to represent a change in the surface orientation.

Volume rendering techniques [55, 85, 35, 56, 87, 84, 102, 5, 83, 30, 139] use the original data, rather than a geometrical representation based on the original data, for rendering. The entire volume of data is assumed to be semi-transparent and each voxel in the volume is assigned an opacity and colour value based on the intensity values of the data. This is commonly referred to as *classification*. The surface normals are estimated from the local image gradient since there is no geometric representation of the surface(s) to be rendered.

The selection of the mapping function from intensity to opacity has a profound influence on the appearance of surfaces in the rendered image. A single surface (isosurface) is obtained by selecting thresholds for the mapping function such that only intensity values within the specified range have an associated opacity value. All other intensities are classified as being totally transparent and do not contribute to the rendered image.

Renderings obtained directly from raw ultrasound data are typically noisy. To minimise the effects of noise a preprocessing step is commonly performed on the raw data prior to rendering. Filtering [108, 107, 49, 119] is commonly employed as a means of reducing the noise present in the raw data. There is a tradeoff between the amount of noise that is reduced and the amount of detail that is lost when filtering. Steen and Olstad [119] apply a Gaussian-like filter at different scales to smooth noise and small details in ultrasound volume data. They further process the data by performing a convolution operation to detect features (image gradients) from the filtered data. The resulting "feature map" is subsequently rendered using several volume rendering techniques.

Sakas and Walter [108] use a multiresolution approach to filter the opacity values subsequent to the mapping of the intensity values. The purpose of the (low pass) filtering is to suppress the effects of noise inherent in 3D ultrasound data. By applying the filtering at multiple resolutions, persistent regions in the data are determined. These regions are then used to create a "region-of-interest" (ROI) mask. The mask constrains the location of the isosurface to be within the ROI.

Karaman *et al.* [49] propose a filter which reduces the amount of speckle (undesirable noise degrading image quality) in ultrasound images. The filter adapts to localised regions based on the statistics collected within the region. A smoothing operation (mean or median) is performed within each region to reduce the speckle. Parameter selection is critical in order to maintain detail in the resulting filtered image.

Subsequent to identification of the surface(s) of interest and its associated normals, both surface and volume rendering techniques use a common method for rendering a 2D projection of the 3D dataset. The surface normal, colour and opacity values are used in conjunction with a shading model to produce the 2D projection. Typically, orthographic projection and the Phong shading model [30] are used. A procedure known as *compositing* incorporates the shading information and the opacity and colour values along each ray in the orthographic projection to calculate the intensity for each pixel in the viewing plane as follows:

$$I_o = C(x)\alpha(x) + I_i(1 - \alpha(x)) \tag{2.2}$$

where $\alpha(x)$, $0 \leq \alpha(x) \leq 1$, is the assigned opacity value for voxel x, I_o is the intensity of outgoing light, I_i is the intensity of light entering voxel x from the previous voxel along the ray (see Figure 2.16) and $C(x)$ is the intensity (colour) of voxel x calculated using the Phong shading model:

$$C(x) = k_a I_l + \frac{I_l}{k_1 + k_2 d(x)}(k_d(N(x) \cdot L_l) + k_s(N(x) \cdot H_l)^n) \tag{2.3}$$

where

I_l - intensity of light l,

L_l - unit normal for direction of light l,

$H_l = \frac{L_l + V}{2}$ - unit normal for direction of maximum specular highlight for light l and viewing position represented by the unit normal V,

$N(x)$ - surface normal for voxel x,

$d(x)$ - distance of voxel x from the viewing plane,

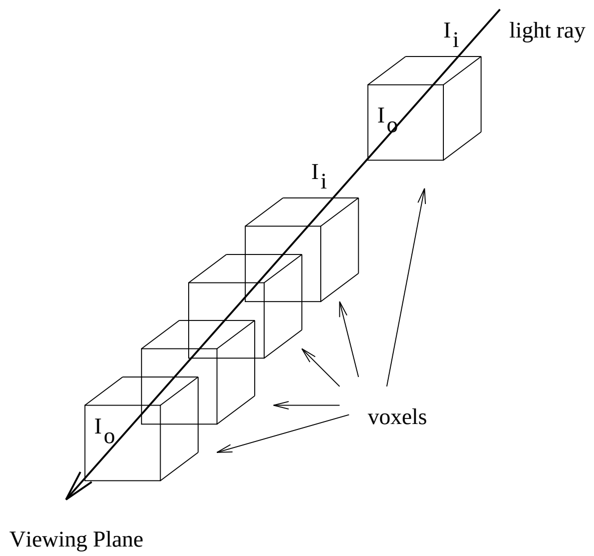


Figure 2.16: A single ray encounters a number of voxels as it passes through the volume on its way to the viewing plane.

n - exponent for approximating specular highlight,

k_a, k_d, k_s - ambient, diffuse and specular coefficients, respectively,

k_1, k_2 - constants approximating depth cueing.

It can be seen from Equation (2.2) that the contribution of the incoming light, I_i, entering voxel x depends on the transparency of the voxel $(1 - \alpha(x))$. I_o represents the value of the pixel on the viewing plane once all voxels along the ray have been processed. Figure 2.16 illustrates the case of a single ray encountering a number of voxels as it passes through the volume. The final I_o value represents the pixel value on the viewing plane. Note that compositing can be performed by projecting rays from the viewing plane into the volume (ray-casting) or by projecting rays though the volume onto the viewing plane (voxel-projection). In the latter case, it is possible that more than one pixel in the viewing plane is covered by the projection of a ray.

An alternative to the spatial domain volume rendering techniques described above is a frequency domain method proposed by Malzbender [69]. The advantages of frequency domain based volume rendering is a significant reduction in computational cost as well as additional assurances of image accuracy. The basic approach is to transform the 3D dataset into the frequency domain (using a Fourier or Hartley transform) in which all operations (e.g., filtering, 2D projection, etc.) are performed. An inverse 2D transform is applied to the 2D projection of the dataset onto the viewing plane in order to render an image for display. The major limitation of this approach is that the projection is order independent along the line of projection, and therefore, all depth information (hidden surface information) is lost. The resulting 2D images have a transparent look which resemble X-ray images.

2.4.2 Problems

There are a number of problems associated with the surface and volume rendering techniques. Surface rendering techniques require that a decision be made on whether or not each voxel (or pixel in the 2D case) in the dataset belongs to the object. This segmentation process is non-trivial and often small details are lost or artifacts are incorporated into the object. The presence of noise in the data severely affects the quality of the segmentation. Although volume rendering techniques do not require a binary classification of the data, it is precisely relaxation of this requirement that imposes a major problem in selecting a surface to be rendered. Since the identity of each voxel can only be determined from its intensity, the mapping function from intensity to opacity is crucial. Currently, *ad hoc* methods are used to determine this function. Furthermore, since most mapping functions are not based on any model (this is especially true for ultrasound data), the rendered images can vary significantly from physician to physician (or researcher to researcher). Noise in the dataset will also affect the quality of the rendered image since surface normals are estimated from the image gradient.

A problem common to both surface and volume rendering techniques is the computational expense and storage requirements. Since typical datasets are in the order of 30MB or greater, the expense of rendering methods makes interactive rendering difficult on current workstations. Although it is not the goal of this thesis to render ultrasound images at interactive rates (it is desirable, however), the issue of interactive rendering is quite important in the visualisation community. Of course, increasing the available resources (number of processors, speed of processors and memory) can substantially increase the rate of rendering. Alternatively, methods such as adaptive refinement [56], incremental rendering [87, 35, 127, 128], frequency domain rendering [69] and shell rendering [128] are ways of reducing the computational and storage requirements so that inexpensive workstations can approach interactive rates.

Another problem common to both rendering techniques involves resampling the original data. Often, the dataset to be visualised does not conform to the isometric 3D grid of voxels that is required by the rendering techniques (see the discussion of the 3D ultrasound probe in Section 1.3). Thus, when transforming the dataset to an isometric representation either data points are ignored or interpolation is required to "fill in" missing data values. Also, when the viewing plane is oriented at different locations with respect to the dataset, the orthographic projection is subject to sampling error due to the discretisation of the dataset.

Chapter 3

Tissue Characterisation

The purpose of this chapter is to describe the details of the Tissue Characterisation module. Previous research in characterising tissue in ultrasound images used an *ad hoc* approach in analysing the features to determine the most effective parametric representation of the tissues [77, 78, 81]. This approach provides good results for a small number of somewhat distinct tissue types. However, in this thesis it is desired to characterise tissues whose corresponding texture in the ultrasound data is quite similar. This requirement exceeds the capabilities of the previous research so a new approach is necessary. It is proposed that a statistical pattern recognition based approach be used to characterise (and classify) these types of tissues. This approach has a number of advantages. One advantage is that statistical pattern recognition is well established and thus provides a solid foundation on which tissue characterisation can be accomplished. As well, the improvements suggested in this chapter to a classical statistical approach provide the ability to combine both classification and segmentation in a unified, multiresolution framework. This is a significant contribution as classification and segmentation are almost always treated independently in the literature.

Pattern classification [67, 88, 52, 105, 25, 148, 32, 44, 6, 2, 132] is a well studied problem in which the identity of an unknown pattern is determined to be one of M classes. These M classes are either determined *a priori* or are determined by analysing the "clustering" of patterns within the pattern space. When classes are determined *a priori*, certain assumptions are necessary for unknown patterns to be identified accurately. First, the M classes span the entire pattern space. That is, the identity of each pattern must be one of the M classes. It is also usually assumed that there is no computational expense in obtaining the pattern space itself. For example, a sensor bank produces D dimensional patterns as output of a monitoring process. When any of these assumptions are violated, the accuracy and computational efficiency of the pattern classifier are reduced.

In statistical pattern recognition it is assumed that the classes can be described by a set of distributions. Once the distributions are determined, it is possible to use a statistical decision rule to determine the class to which a pattern belongs. It is usually the case that the actual parameters of the class distributions are not known, so the pattern space must be representatively sampled in order to estimate the parameters. When this sampling process is contaminated with outlier patterns, due to some sort of sampling error, the resulting parameter estimates will be biased and subsequent classifications based on these parameters will be inadequate. Recently, robust statistics [106, 17, 82, 81, 77] have been used to minimise the effects of outliers in parameter estimation. The minimum volume ellipsoid estimator (MVE) proposed by Rousseeuw and Leroy [106] can tolerate up to 50% contamination of the data and still provide an unbiased, affine equivariant estimate (in the case of a multivariate normal distribution, the mean and associated covariance matrix) describing a class distribution.

To increase the classifier's tolerance to outlier data, it is proposed that the MVE estimator be used to estimate the class distribution parameters. For distributions which exhibit elliptical symmetry, such as the multivariate normal distribution, the MVE estimator provides unbiased parameter estimation in the presence of up to 50% outliers. It is also shown experimentally that reasonable parameter estimates are possible for distributions which do not exhibit elliptical symmetry, provided that the sample size is large enough. Experiments are presented which examine the effects of dimension, sample size, outlier data and class distribution assumptions on the ability of MVE to provide unbiased parameter estimates.

Dubuisson and Masson [26] extended a statistical decision rule for M classes with a reject option, originally proposed by Chow [20], by redefining rejection with two options, *ambiguity reject* and *distance reject*, which relax the assumption that all patterns belong to one of M classes. This assumption must be relaxed when the *a priori* knowledge about the classes is incomplete. The ambiguity reject option reduces the probability of erroneously classifying a pattern into one of the M classes with the use of an ambiguous class. This class identifies those patterns which are classified as belonging to two or more classes with (near) equal probability. The distance reject option also reduces the probability of erroneous classification by removing the restriction that a pattern must be classified into one of the M classes. A distance reject class identifies those patterns which have little or no similarity to each of the prototypical representation of the M classes. Once patterns are classified into one of the reject classes, it remains that the true class identity of these patterns be determined.

When there is a computational expense associated with producing patterns, it is desirable to reduce the dimensionality of the patterns such that the computational overhead is minimised, while still maintaining accurate classification results. One common way to achieve this is to use feature selection to reduce the patterns from D to d dimensions. In doing so, only the most useful d dimensions need to be calculated and subsequently evaluated by the classifier. There are numerous algorithms for selecting features for the reduced feature space [25, 148, 32, 44, 98, 72, 130]. Each algorithm involves the evaluation of a criterion function to determine which is the best subset of d features. The choice for a criterion function depends on how separation among the classes is measured. Ideally, the value of the criterion function will be maximum when the separation among the classes is large and minimum when the separation among the classes is small. A theoretically ideal criterion is the Bayes error which indicates the probability in erroneously misclassifying patterns. As noted in the literature [25], calculation of the Bayes error is difficult as the intersection of the M class distributions in multi-dimensional space is required to determine the error probability. Determining the multi-dimensional intersection is a computationally complex operation.

A new criterion function, the Inck (**Inc**omplete **k**nowldege) criterion function, is proposed which approximates the error probability when the M classes do not span the entire pattern space. This criterion is based on the probabilistic measures obtained from a modified version of Dubuisson and Masson's statistical decision rule with reject. The error probability, or conversely the probability of correct classification, can be determined without having complete knowledge about the class distributions. The attractiveness of this criterion is that it is highly correlated with the statistical decision rule (with reject) which can be used by the classifier. Thus, the criterion provides a good indication of the classifier performance that can be expected with the reduced feature space. The modified version of Dubuisson and Masson's rule treats the distance reject class differently than that originally proposed. In its original form, the distance reject threshold has no direct relationship to the class distributions. This is especially problematic for the case of multivariate normal distributions with different covariances. In order to maintain an unbiased decision it is required that a reject threshold be chosen for each different covariance. The threshold selection is made difficult as each threshold must not add bias to the decision rule. The modified version of the rule relates the probability in which patterns are identified as belonging to the distance reject class to a reject threshold for each different covariance. That is, the threshold values are determined by fixing the probability in which patterns are classified as belonging to the distance reject class. This is equivalent to selecting a confidence interval at which patterns are classified into one of the M classes.

Finally, a classifier design is presented which incorporates the new criterion function. As well, it is described how the rejection classes of the statistical decision rule with reject can be used in a multiresolution classifier design to determine the class identity of patterns which can not be determined at a given resolution.

The organisation of this chapter is as follows. Section 3.1 describes the robust MVE estimator. Section 3.2 describes the statistical decision rule with reject proposed by Dubuisson and Masson. Section 3.3 describes how this rule can be modified which result in an improvement in rejection threshold selection. It is also described how this method can be used as a criterion function for feature selection. Section 3.4 describes various feature selection methods. Section 3.5 discusses classifier design and how the MVE estimator and the statistical decision rule with reject can be incorporated. Experimental objectives, procedures, and results are presented in Section 3.6. Section 3.7 provides a summary of the chapter.

3.1 Robust Parameter Fitting

When the decision has been made to use supervised training in the design of a classifier, it is necessary to obtain representative samples of each of the M classes on which the training is performed. This requires control over the sampling process such that when a pattern is obtained, its identity is known *a priori*. It is often assumed that control over the sampling process is adequate so that the identification is accomplished with a minimal amount error. However, if the sampling process is subject to error this assumption is violated. The performance of the classifier will certainly be degraded. One way to reduce the chances of this is to accept that sampling error exists. The sampling error results in patterns which do not represent the class to which they are assumed to belong. If these patterns could be "removed" from the training data then the performance of the classifier should be independent of the error in the sampling process.

Errors in the training data have a direct effect on how well the class parameters can be estimated. For example, the existence of one "outlier" pattern can severely bias the estimate of the sample mean for a class. An outlier is a pattern which is substantially different from the majority of the patterns belonging to the given class. The use of robust statistics [57, 120, 106, 17, 82, 81, 77] in parameter estimation is one way to minimise the effects of outliers. In the multivariate case, outlier detection can be difficult. Rousseeuw and Leroy [106] focus on robust methods for estimating the parameters of a multivariate point cloud which describe the centre of the cloud as well as the dispersion of the points about the centre. When the point cloud is assumed to have a multivariate normal distribution, these parameters are the multivariate mean, μ, and covariance, $\mathbf{\Sigma}$. The minimum volume ellipsoid estimator (MVE) is one method which estimates $\mathcal{N}(\mu, \mathbf{\Sigma})$ by fitting an ellipsoid to at least half, h ($h = n/2$), of the data points of X. The centre of the ellipsoid provides the estimate for μ while the ellipsoid itself provides an estimate of $\mathbf{\Sigma}$.

A brute force method to calculate the smallest volume ellipsoid is to consider all combinations of half the data points and calculate the volume of the ellipsoid surrounding each "half". This method is computationally infeasible, however, for any reasonable size n as the number of halves is large (C^n_h halves). It is possible to reduce the computational expense in two ways:

- reduce the number of ellipsoids that have to be calculated, and
- reduce the computation in calculating each ellipsoid.

Both of these are accomplished in the MVE.

Given a data set, X, containing n, p-dimensional vectors $\mathbf{x_i}$, $1 \leq i \leq n$, a subsample, J, of $(p+1)$ vectors are drawn from X, $1 \leq p < n$. It is assumed that $(p+1) \leq h$ and that there are at least h non-outlier data points. From J, the sample mean, $\mathbf{m_J}$, and covariance matrix, $\mathbf{C_J}$, are calculated. The ellipsoid corresponding to $\mathbf{m_J}$ and $\mathbf{C_J}$ must then be inflated or deflated to contain exactly h points. This is accomplished by considering the squared distance of each point, $\mathbf{x_i}, 1 \leq i \leq n$, from the centre of the ellipsoid:

$$d_i^2 = (\mathbf{x_i} - \mathbf{m_J})\mathbf{C_J^{-1}}(\mathbf{x_i} - \mathbf{m_J})^{\mathbf{t}}. \tag{3.1}$$

Half (h) of the population is above d_J^2 and half below, where

$$d_J^2 = median_i(d_i^2). \tag{3.2}$$

Every $\mathbf{x_i}$ whose corresponding $d_i^2 \leq d_J^2$ is closer to the centre of the ellipsoid than those $\mathbf{x_i}$ whose corresponding $d_i^2 > d_J^2$. Scaling the ellipsoid to fit these h points corresponds to scaling $\mathbf{C_J}$ by d_J^2. Figure 3.1 illustrates the fitting process for $p = 2$. The original ellipse is enlarged to include half the data points.

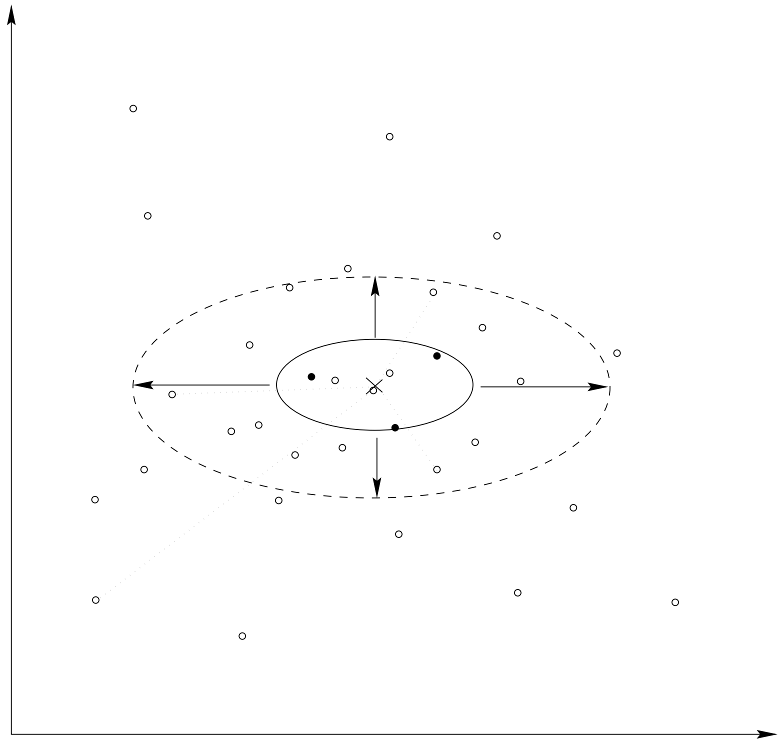


Figure 3.1: Fitting ellipsoid to h data points.

The volume of the ellipsoid, V_J, is proportional to the determinant of the scaled covariance matrix

$$V_J = (det(d_J^2\mathbf{C_J}))^{\frac{1}{2}} = (det(\mathbf{C_J}))\frac{1}{2}(d_J)^p. \tag{3.3}$$

Rather than computing Equation (3.3) for every possible C_{p+1}^n subsample, s subsamples are drawn and the one minimising Equation (3.3) determines which $\mathbf{m_J} = \mathbf{m}(X)$ and which $\mathbf{C_J} = \mathbf{C}(X)$. $\mathbf{m}(X)$ and $\mathbf{C}(X)$ are the mean and covariance parameter estimates for X. Note that $\mathbf{C_J}$ must be corrected by the factor

$$\mathbf{C(X)} = (\chi^2_{p,0.5})^{-1}\mathbf{C_J}$$

since X is multivariate normal. The choice for the number of subsamples, s, is determined by the probability that at least one of the subsamples consists of no outliers:

$$E = 1 - (1 - (1 - \varepsilon)^{p+1})^s \tag{3.4}$$

where ε is the percentage of outliers assumed to exist in the data. To ensure that there is good chance of selecting an uncorrupted subsample E is chosen to be some value close to 1 (0.95 or 0.99 for example). Rearranging the terms in Equation (3.4), the number of subsamples, m, is

$$s = log(1 - E)/log(1 - (1 - \varepsilon)^{p+1}).$$

The MVE parameter estimates can be used directly or they can be used as an initial solution for a refined version of the MVE. The reweighted MVE applies a weight to each $\mathbf{x_i}$ in X. The weight, w_i, is determined by the distance between the point, $\mathbf{x_i}$, and $\mathbf{m(X)}$ as follows:

$$w_i = \begin{cases} 1 & if \quad (\mathbf{x_i} - \mathbf{m(X)})\mathbf{C(X)^{-1}}(\mathbf{x_i} - \mathbf{m(X)})^{\mathbf{t}} \leq c \\ 0 & otherwise \end{cases} \tag{3.5}$$

where c is the maximum distance a point can be from $\mathbf{m}(\mathbf{X})$ without being considered an outlier. The choice for c depends on how tight a fit is desired. For example, $c = \chi^2_{p,0.99}$ includes 99% of the "good" data surrounding $\mathbf{m}(\mathbf{X})$. Thus, Equation (3.5) acts as a filter for removing outliers from X. The estimates $\mathbf{m}(\mathbf{X})$ and $\mathbf{C}(\mathbf{X})$ can be recalculated in the classical way using all the $\mathbf{x_i}$ in X with a corresponding $w_i > 0$.

The complexity of the MVE estimate is dependent on the calculation of the $p \times p$ matrix inversion (in Equation (3.1)) and calculation of the median from the n squared residuals (in Equation (3.2)). Each of these calculations is performed s times. The complexity of a single matrix inversion, in general, is $\Theta(p^3)$. The complexity of the median calculation is $\Theta(n^2)$. Note that as p increases linearly, s increases exponentially (c.f. Equation (3.1)). A limit must be placed on the size of s for even moderate values of p.

3.2 Statistical Decision Rule With Reject

Given M classes, $\omega_i, 1 \leq i \leq M$, each with a known distribution and a priori probability $P(\omega_i)$, $\sum_{i=1}^{M} P(\omega_i) = 1$, a vector, $\mathbf{x}$, in d dimensional pattern (vector) space, R^d, belongs to ω_i with probability $f(\mathbf{x}|\omega_\mathbf{i})$. Using the usual Bayes rule [32, 25], $\mathbf{x}$ is associated with the class which minimises the conditional error probability (error probability in classifying $\mathbf{x}$ into ω_i) $C_i(\mathbf{x})$,

$$C_i(\mathbf{x}) = \sum_{j=1, j\neq i}^{M} P(\omega_j|\mathbf{x}),$$

$$P(\omega_j|\mathbf{x}) = \frac{f(\mathbf{x}|\omega_j)P(\omega_j)}{\sum_{k=1}^{M} f(\mathbf{x}|\omega_k)P(\omega_k)}.$$

This optimal error probability is

$$e^*(\mathbf{x}) = Min_{i=1}^{M} C_i(\mathbf{x}) = 1 - Max_{i=1,M} P(\omega_i|\mathbf{x}). \tag{3.6}$$

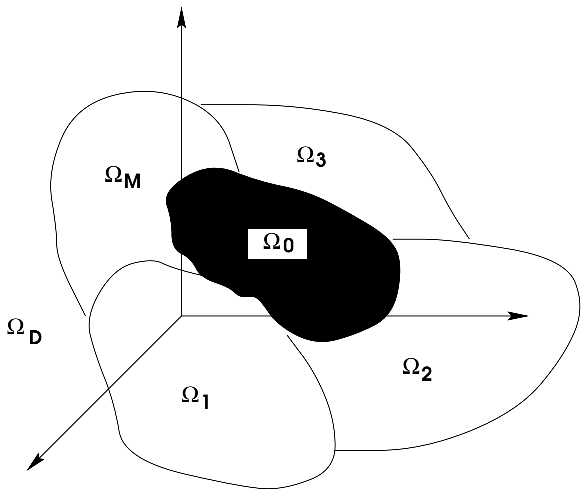


Figure 3.2: Using the statistical decision rule with both ambiguity and distance reject options results in the pattern space being partitioned into M+2 regions.

Chow [20] introduced an additional reject class, ω_0, to which $\mathbf{x}$ is assigned when the probability of $\mathbf{x}$ belonging to 2 or more of the M classes is (near) equal. This results in a decision rule with an ambiguity reject (as termed by Dubuisson and Masson [26]) option

$$\mathbf{x} \rightarrow \omega_0 \qquad \text{if } e^*(\mathbf{x}) > C_a \tag{3.7}$$

$$\mathbf{x} \rightarrow \omega_i \quad \text{if } P(\omega_i|\mathbf{x}) = Max_{j=1,M} P(\omega_j|\mathbf{x}) \geq 1 - C_a \tag{3.8}$$

where C_a is the *ambiguity reject threshold* and $0 \leq C_a \leq \frac{M-1}{M}$ in order for the reject option to exist.

The distance reject option proposed by Dubuisson and Masson decreases the probability of erroneously classifying $\mathbf{x}$ into one of the M classes when it is "far from" the prototypical representation of each of the M classes. This situation arises when the M classes do not actually span the entire pattern space and $\mathbf{x}$ comes from a region which is not spanned by the classes. Thus, the pattern space, R^d, is divided into M+2 regions as illustrated in figure 3.2. The following relationships hold among

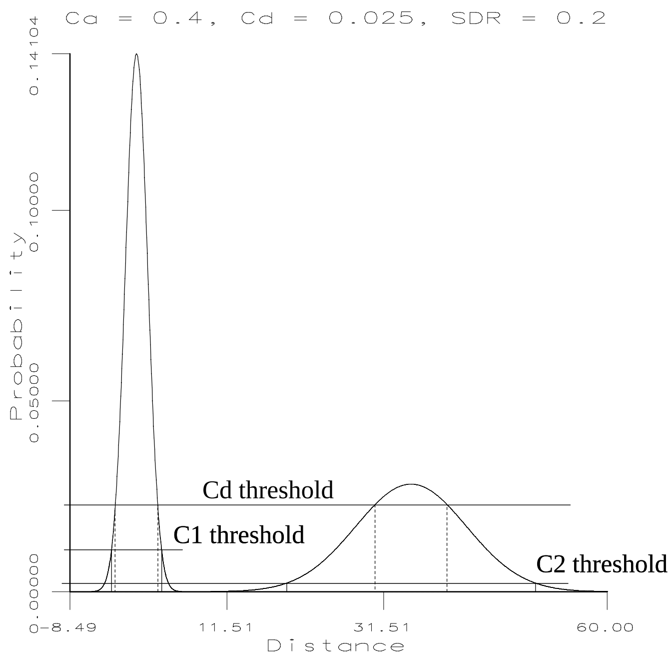


Figure 3.3: Mixture density for 2 classes with different variances.

the regions:

$$
\begin{aligned}
&R^d = \Omega_A \cup \Omega_R, \\
&\Omega_A = \Omega_1 \cup \Omega_2 \cup \ldots \cup \Omega_M, \\
&\Omega_R = \Omega_0 \cup \Omega_D, \text{and} \\
&\Omega_0 \cap \Omega_D = \emptyset.
\end{aligned}
$$

The reject regions for ambiguity and distance are Ω_0 and Ω_D, respectively.

Using the distance reject option in its original form, $\mathbf{x}$ is distance rejected if

$$f(\mathbf{x}) < C_d \tag{3.9}$$

where $f(\mathbf{x})$ is the mixture density and C_d is the *distance reject threshold*. Note that C_d is related to the mixture density and not directly to the class distributions. This is problematic when the distributions are parameterised with different values or different parameters. Figure 3.3 illustrates the mixture density for two classes having normal distributions with different means and variances. For $C_d = 0.025$, Equation (3.9) results in a bias favouring the distribution on the left (class 1). That is, the proportion of area in the tails of the distributions indicate that the probability of $\mathbf{x}$ being distance rejected when it belongs to the distribution on the right (class 2) is greater than the probability for class 1.

3.3 Modified Decision Rule

It is possible to modify the distance reject option so that there is no bias in the resulting decision rule. In this thesis multiple distance reject thresholds, rather than a single threshold, are used in determining the distance reject region in pattern space. In this modified decision rule, Inck, a distance reject threshold is chosen for each different class distribution. These thresholds are chosen according to the parameters of each class distribution. In Figure 3.3, two thresholds, C_1 and C_2, are chosen such that the area in each tail of the distributions is the same ($\frac{0.025}{2}$, which results in a distance reject probability of $0.025 = C_d$ for each class). By fixing the area in the tails of each class distribution, an upper bound for the probability of distance reject for the mixture density, P_{RD}^U, is

$$P_{RD}^U = \sum_{i=1}^{M} C_d P(\omega_i) = C_d \sum_{i=1}^{M} P(\omega_i) = C_d. \tag{3.10}$$

The upper bound in Equation (3.10) results when there is no overlap among the classes (i.e. $\Omega_0 = \emptyset$) or when there is total overlap among the classes (i.e. $\Omega_1 = \Omega_2 = \ldots = \Omega_M$).

A lower bound for the probability of distance reject for the mixture density, P_{RD}^L, can also be determined when each of the $P(\omega_i) = 1/M$ and there is overlap among the tails of each of the M class distributions as shown in Figure 3.4. In this case only the two *outermost* classes each have a tail which does not overlap with any other class. The resulting probability is

$$P_{RD}^L = 2(\frac{C_d}{2}\frac{1}{M}) = \frac{C_d}{M}. \tag{3.11}$$

In this modified decision rule there is now a direct relationship between C_d and the

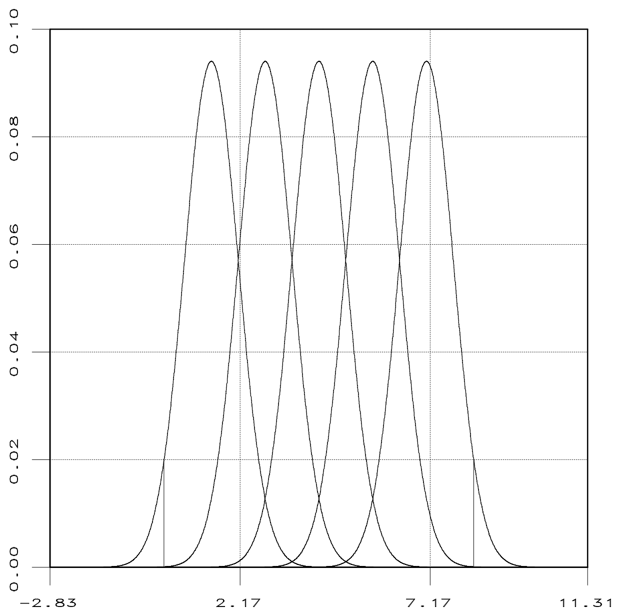


Figure 3.4: Overlapping multi-class distributions resulting in a lower bound for probability of distance reject.

probability of distance reject. The upper and lower bounds for the probability of distance reject are determined by the value of C_d.

In order to determine the values for $C_i, \quad 1 \leq i \leq M$, it is necessary to consider the parametric representation of each class. For the sake of simplicity it is assumed that each class has a multivariate normal distribution, $\mathcal{N}(\mu_{\mathbf{i}}, \Sigma_i)$. The distance between C_i and μ_i has a χ^2 (chi-square) distribution with d degrees of freedom. By letting C_d represent the confidence level it is possible to determine a single value, $\chi^2(d, 1-C_d)$, which can be used for each class as a rejection threshold. Thus, rather than calculating each C_i in terms of the mixture density, the distance between $\mathbf{x}$ and each class mean μ_i is calculated and compared to $\chi^2(d, 1-C_d)$. This is equivalent to selecting a confidence interval around each class and patterns outside this interval will be distance rejected. Specifically, $\mathbf{x}$ is distance rejected if for all $i, 1 \leq i \leq M$,

$$(\mathbf{x} - \mu_{\mathbf{i}})\Sigma_i^{-1}(\mathbf{x} - \mu_{\mathbf{i}})^t > c \tag{3.12}$$

where $c = \chi^2(d, 1 - C_d)$. Note that the l.h.s. of the inequality in Equation (3.12) is calculated in determining f($\mathbf{x}$) so no extra computational cost is incurred. If this does not hold then the pattern is classified according to Equation (3.7) and Equation (3.8). Also, the spatial relationship among the classes does not have an effect on the modified rule. This is different than the approach taken by Dubuisson and Masson in which the spatial relationship among the classes has to be considered in determining how the ambiguity reject and distance reject options are applied. In Section 3.6.2 simulations using Inck are presented which verify the results originally obtained by Dubuisson and Masson as well as show the advantages of the distance reject option.

The reject options in Inck allow patterns to be classified into M classes even if a complete representation of the classes is not known prior to training the classifier. Further, the reject options minimise the probability of misclassification error, P_E. This probability (as well as the probability of correct classification, P_C) is quantified as follows:

$$P_C(C_a, C_d) = 1 - P_E(C_a, C_d) - P_{RA}(C_a, C_d) - P_{RD}(C_d) \quad \text{correct classification} \tag{3.13}$$

$$P_E(C_a, C_d) = \sum_{i=1}^{M} P_{Ei}(C_a, C_d) \qquad \text{misclassification error} \tag{3.14}$$

$$P_{RA}(C_a, C_d) = \int_{\Omega_0} f(\mathbf{x})d\mathbf{x} \qquad \text{ambiguity reject} \tag{3.15}$$

$$P_{RD}(C_d) = \int_{\Omega_D} f(\mathbf{x})d\mathbf{x} \qquad \text{distance reject} \tag{3.16}$$

$$P_{Ei}(C_a, C_d) = \int_{\Omega_i} e^*(\mathbf{x})f(\mathbf{x})d\mathbf{x} \qquad \text{misclassification error for } \ d(\mathbf{x}) = i \tag{3.17}$$

$$f(\mathbf{x}) = \sum_{i=1}^{M} f(\mathbf{x}|\omega_i)P(\omega_i) \qquad \text{mixture density} \tag{3.18}$$

These probabilities are the same as that presented by Dubuisson and Masson [26] except for the regions of integration in Equation (3.15) and Equation (3.16). The differences are due to the reasons explained above. It can be seen from Equation (3.13) that P_C decreases as P_{RA} or P_{RD} increase. However, when patterns are rejected they can not be classified into one of the M classes and so P_E decreases which results in an increase in P_C. Thus, a tradeoff must be made between the classifier's ability to minimise classification error and maximise correct classification. For example, when there is high risk associated with misclassifying patterns it is desirable to choose C_a and C_d such that P_E is minimised.

To facilitate the selection of C_a and C_d an algorithm is presented which provides an intuitive method for selecting the parameters. The algorithm determines the value for C_a based on a given classification error tolerance, ϵ, at a given confidence level, C_d. The algorithm determines the value of C_a which maximises P_C while keeping $P_E \leq \epsilon$. C_d is the given confidence level (typically .05 or .01) at which P_C is maximised.

Based on the relationships among the probabilities in Equation (3.13), P_C is maximum when P_{RA}, P_{RD} and P_E are minimum. The reject probabilities are easily minimised by setting $C_d = 0$ and $C_a \geq (M-1)/M$. This will result in $P_{RA} = 0$ and $P_{RD} = 0$. In this case, however, P_E will almost certainly not be minimum as any class overlap or data outliers will result in an increase in P_E. Alternatively, P_E can be minimised by expanding the reject regions so that no error occurs. $C_d = 1$ and $C_a = 0$ will produce the desired effect. Obviously, these are not good parameter choices as all points are classified into one of the reject regions. However, if the restriction that $P_E \leq \epsilon$ is enforced then it is possible to formulate a criterion for maximising P_C:

$$\text{Choose } C_a \text{ s.t. } P_{RA} \text{ is minimum and } P_E \leq \epsilon. \tag{3.19}$$

With bounds placed on C_d (and thus P_{RD}) and P_E ($0 \leq P_E \leq \epsilon$), all that remains is to select the value for C_a from the range $[0, (M-1)/M]$ which satisfies Equation (3.19). Since P_E is a monotonically increasing function over this range (simulations are presented in Section 3.6.2 that verify this) a numerical method similar to root finding [97] can be used to find the value of C_a at which $P_E = \epsilon$. Given the confidence level, C_d, and the maximum classification error tolerance, ϵ, the algorithm is formulated as follows:

1. $C_a \leftarrow (M-1)/M,\ u \leftarrow (M-1)/M,\ l \leftarrow 0$, *done* $\leftarrow$ false

   Repeat steps 2 - 4 while not done

2. classify training data with C_a and C_d to obtain P_E, P_{RA}, P_{RD}, P_C

3. compare P_E with ϵ

   (a) if $|P_E - \epsilon| < \Delta$ then *done* $\leftarrow$ true (Δ a small number)
   (b) if $P_E < \epsilon$ *and* $C_a = (M-1)/M$ then *done* $\leftarrow$ true
   (c) if $P_E < \epsilon$ then $l \leftarrow C_a$
   (d) if $P_E > \epsilon$ then $u \leftarrow C_a$

4. if not *done* then $C_a \leftarrow (u-l)/2$

The resulting P_C will be a maximum for the calculated C_a, the given confidence interval and error tolerance.

Since both P_C and P_E provide a good indication of the performance that can be expected with a classifier using the proposed decision rule it is natural to extend the use of the rule to the evaluation of feature sets. In feature selection a subset of d features is chosen from the D available features. There are numerous algorithms for determining the "optimal" set of d features [25, 148, 32, 44, 98, 72, 130] and each algorithm must evaluate some criterion in which optimality can be quantitatively measured. Both P_C and P_E provide measures of optimality. The benefit of using either of these probabilities as a selection criterion is that they indicate the expected accuracy of the classifier using the feature subset. The subset of d features which maximises P_C or minimises P_E is then the most appropriate choice for the reduced feature set.

A potential problem in using Inck as a selection criterion is the expense of calculating a d-dimensional integral over the pattern space to determine the various probabilities in Equations (3.13) - (3.17). As d increases linearly, the complexity of the integral increases exponentially. Even the use of methods to approximate the integral [97] can not decrease the complexity to the point where evaluation of the criteria can be feasibly calculated. The *dimensional explosion* further exasperates the situation as the criterion must be evaluated many times during feature set evaluation. In fact, the complexity of the feature selection algorithm itself can grow exponentially [32, 44, 25, 148].

In order to use the proposed rule as a feature selection criterion, it is necessary to reduce the effort spent in performing integration. One way to accomplish this is to approximate the probabilities in Equations (3.13) - (3.17) using N patterns sampled from the pattern space (training set). Rather than integrating over the entire pattern space the mixture density, $f(\mathbf{x})$, can be approximated from these samples. Since it is usually the case that a training set, $S, S = \{\mathbf{x_j}\} \subset R^d, 1 \leq j \leq N$, is used to calculate the class densities, $f(\mathbf{x})$ can be evaluated at each $\mathbf{x_j}$. The probabilities in Equations (3.13) - (3.17) can then be approximated using Equations 3.7, 3.8 and 3.12 as follows:

$$D = \{\mathbf{x_j} |\ \exists i, 1 \leq i \leq M, (\mathbf{x_j} - \mu_\mathbf{i})\Sigma_i{}^{-1}(\mathbf{x_j} - \mu_\mathbf{i})^t > c\} \tag{3.20}$$

$$A = \{\mathbf{x_j} |\ d(\mathbf{x_j}) = 0\} \tag{3.21}$$

$$E = \{\mathbf{x_j} |\ d(\mathbf{x_j}) \neq i, \mathbf{x_j} \in \omega_i, 1 \leq i \leq M\} \tag{3.22}$$

$$P_{RD} = \frac{|D|}{N} \tag{3.23}$$

$$P_{RA} = \frac{|A|}{N} \tag{3.24}$$

$$P_E = \frac{|E|}{N} \tag{3.25}$$

where $D \cap A \cap E = \emptyset$ and $D \cup A \cup E = S$.

The error in approximating the above probabilities is dependent on the number of samples, N, the dimension of the pattern space, d, and the accuracy in estimating the class distribution parameters. Blayo *et al.* [7] suggest that a sample size which is exponentially proportional to d be used to estimate the class (normal) distribution parameters. This requires a huge sample size for even a modest value of d. Contrary to this, Rousseeuw and Leroy [106] suggest a robust method for parameter estimation in which the sample size grows linearly with d. Research by Wacker and El-Sheikh [135], Fukunaga and Hayes [33] and Aberhard *et al.* [4] also support the idea of using a sample size which is linearly related to the dimension of the pattern space ($N = k * d$ for some constant $k \geq 1$). Both analytical and experimental evidence indicate that the choice for k depends on the decision rule used by the classifier. For a linear decision rule (the "equal variance" case), a fixed k can be used. For a quadratic decision rule (non-equal class variances), k should increase with d in order to maintain classifier accuracy. In Section 3.6.2 experimental evidence is provided that supports the use of a training set of N patterns for approximating the probabilities of Inck.

3.4 Feature Selection

An important decision that must be made when designing a classifier is whether or not to incorporate feature selection. The purpose of feature selection is to reduce the patterns from D to d dimensions. In doing so, only the best d features need to be calculated and subsequently evaluated by the classifier. The use of feature selection is appropriate in a number of cases:

- there is a cost in obtaining the patterns,
- redundant information among dimensions,
- irrelevant information in dimensions.

In the proposed system, the first case is certainly a major concern. The generation of patterns is performed at the time of classification. Both the pattern generation and classification tasks are competing for the same computational resources. Using a smaller dimensional pattern space significantly decreases the resource demands. Fewer calculations are required as fewer features have to be computed. Also less features have to be processed during classification. The last cases are of concern in any classification task. The presence of redundant or irrelevant information may only degrade the performance of the classifier.

There are numerous algorithms for selecting features for the reduced feature space [25, 148, 32, 44, 98, 72, 130]. Devijver and Kittler [25] and Fukunaga [32] provide an excellent review of feature selection algorithms. The choice for a particular algorithm depends on the complexity of the criterion function, the dimension of the pattern space, the complexity of the algorithm itself, and the required performance of the classifier using the reduced pattern space. It is often the case the feature selection algorithm used will vary for each different application. Four different feature selection algorithms are evaluated in this thesis: branch and bound (BB), best features (BF), sequential forward selection (SFS) and sequential backward selection (SBS). The BB algorithm is optimal in the sense that all C_d^D subsets have the possibility of being evaluated in determining which is the best feature subset. The search is accomplished without performing an exhaustive search of every possible subset. The BF, SFS and SBS algorithms are all suboptimal algorithms in that they do not evaluate every possible feature subset. The BF algorithm is the simplest (and most unreliable) of these algorithms. The feature subset is constructed by choosing the d individually best features. This requires only D evaluations of the criterion function. However, this algorithm is not guaranteed to produce the best feature subset, even if the features are statistically independent.

The SFS and SBS algorithms are ranked between the BB and BF algorithms in terms of reliability and complexity. The SFS algorithm is a bottom-up approach that constructs the d feature subset starting with k features and incrementally adding features until d features are obtained. At each increment, the subset is evaluated with each of the remaining $D-k$ features (thus, $D-k$, subsets of $k+1$ features are evaluated). The subset producing the maximum value for criterion function becomes the new subset. The SBS algorithm is a top-down approach in which $D-k$ features compose the subset. Features are incrementally removed from the subset until d features remain. At each increment all combinations $(D-k)$ of $D-k-1$ features are evaluated. Again, the subset producing the maximum value for the criterion becomes the new subset. The main difference between SFS and SBS is that the SFS is less computationally expensive as the criterion function is always evaluated on subsets of dimension $\leq d$ and for SBS is evaluated on subsets of dimension $\geq d$.

The common ground which all feature selection algorithms share is the use of a criterion function, $J(D)$, in evaluating the various subsets of d features. The subset that is chosen is the one which maximises $J(D)$ and ideally minimises the probability of classification error when using the selected feature space. The criterion function provides a measure of the separability among classes. The classification error is minimum when the separability among classes is maximum. The best indicator of class separability is the Bayes error [25], however, its calculation is often difficult so other measures are used. In this thesis two categories of class separability measures are evaluated: interclass distance and probabilistic distance.

Interclass distance measures are based on the average pairwise distance between patterns of different classes. The higher the average distance, the greater class separation. The advantage of this type of measure is that no knowledge of class distributions is required as the measure can be calculated directly from the training data.

Probabilistic distance measures are based on the separation of the class conditional distributions, rather the individual patterns themselves. The proposed criterion function Inck described in Section 3.3 falls under this category. Probabilistic distance measures require knowledge about how each class is distributed in the pattern space as well as the *a priori* probability of each class. The training patterns are used to estimate the parameters of the class distributions. The estimated parameters are then used to calculate the class separability measures. The advantage of probabilistic distance measures is that they provide a true indication of the separability among the classes but the additional information is required.

3.5 Multiresolution Classifier Design

There are numerous factors to consider when designing a classifier. Of utmost importance is the decision to base the classifier on *a priori* information. This requires assumed knowledge of the pattern distributions which are to be studied by the classifier. It is often the case that a fixed number of classes are assumed to exist and all patterns examined by the classifier belong to one of these classes. Using supervised training, samples from each of M classes are selected and evaluated by the classifier. The results of the training are a decision for the feature space to be used in evaluating unknown patterns (unsupervised classification), the estimation of the parameters which define the assumed M classes, and the refinement of the decision rule used in determining the identify of unknown patterns. If the actual identity of an unknown pattern is not one of the M classes, then the a priori class assumption is violated and classification of the pattern will be incorrect. However, if the proposed decision rule is incorporated into the design of the classifier, it is possible that violation of the M class assumption will not have a severe effect on classifier accuracy.

When an unknown pattern does not actually belong to one of the assumed M classes, classification using the proposed decision rule will result in one of three possibilities:

1. the unknown pattern is erroneously classified into one of the M classes,

2. the unknown pattern is classified into the ambiguous class,

3. the unknown pattern is classified into the distance reject class.

For case 1, the only recourse is to re-train the classifier so that the misclassification error can be reduced. This requires either selecting a new feature space to represent the classes and/or expanding the assumption of M classes to $M + 1$ classes. Assuming the existence of another class requires re-training the classifier with the addition of known patterns belonging to the new class.

For cases 2 and 3, an unknown sample is "correctly" identified as belonging to none of the M known classes. Two alternatives are possible to determine the actual identity of the pattern: re-train the classifier as in case 1 or *re-calculate* the representation of the pattern itself. Re-calculation of the pattern is achieved by considering the sampling process in which the patterns are produced. For example, in a typical image classification problem, patterns are created by calculating features within a fixed window size in the image. If the identity of the pattern can not be determined to be one of the M classes then features can be re-calculated using a different window size. This results in the formation of a different pattern which may possibly belong to one of the M classes. Of course, it is necessary for the classifier to have been trained using samples from the M classes calculated in this way. Thus, a multiresolution approach to classification is achieved in which the identity of an unknown sample is determined when the appropriate resolution is selected such that the membership of the sample is one of the M classes. If a different classifier, ϕ_i is determined for each different resolution, β_i, $1 \leq i \leq r$, then the ϕ_i that classifies an unknown pattern into one of the M classes determines the resolution at which the pattern can be identified.

In this multiresolution approach it is possible that some of the M classes may not be represented at certain β_i. As well, more than one ϕ_i may classify an unknown pattern into one class and these classes may be different for each ϕ_i. In these cases it is necessary to choose some criteria such that the ϕ_i maximising the criteria determines the most appropriate resolution at which the identity of an unknown pattern is uniquely determined. If there is no ϕ_i which classifies an unknown pattern into one of the M classes then some alternative decision must be made. Re-training with additional a priori information is again an option. This is certainly necessary for those patterns that are distance rejected as they do not bear any similarity to the patterns classified into the known classes. Dubuisson and Masson [26] suggest that for an ambiguity rejected pattern, the patterns before and after this pattern be considered. If these patterns are classified into one of the M classes then it is most likely that the ambiguity rejected pattern belongs to the same class.

3.6 Experiments

Numerous experiments were performed to study the various parts of the Tissue Characterisation module described in this chapter. Both synthetic and real data is used in the experiments. The synthetic data is described and used in Sections 3.6.1

Table 3.1: 2D feature measures evaluated in experiments.

num	G	θ	d	features
0-7	8	0	1	ASM, CON, COR, ENT, CS, CP, IN, LH
8-15	8	135	1	ASM, CON, COR, ENT, CS, CP, IN, LH
16-23	8	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
24-31	8	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
32-39	64	0	1	ASM, CON, COR, ENT, CS, CP, IN, LH
40-47	64	135	1	ASM, CON, COR, ENT, CS, CP, IN, LH
48-55	64	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
56-63	64	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
64-71	32	0	1	ASM, CON, COR, ENT, CS, CP, IN, LH
72-79	32	135	1	ASM, CON, COR, ENT, CS, CP, IN, LH
80-87	32	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
88-95	32	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
96-98				MEAN, ADEV, SDEV

and 3.6.2 for quantitative analysis of the Inck decision rule and the MVE estimator. Subsequent analysis in the chapter is based on real ultrasound data.

Seven real datasets are evaluated in the experiments. The first dataset, 2C.2d, consists of $d = 99$, 2D feature measures collected from 2D samples of bovine ultrasound ovary data. Table 3.1 lists these features (see Chapter 4 for description of these features). The bovine data was obtained *in vitro* as a series of 72, 2D cross-sectional images (slices). The size of each image is 640×480 pixels with an inter-slice distance of 0.5 mm. Interpolation between slices was required to make the volume isometric as the resolution of a pixel in an image is 0.08 mm. The resulting volume had a size of $640 \times 480 \times 450$ points.

The bovine ovary data consists predominantly of three different structures; corpus luteum, stroma and fluid. Figure 3.5 illustrates a cross-sectional view of the structures in the bovine ovary data.

The corpus luteum (follicle) is roughly outlined in the figure. It has an ovular 3D shape. The stroma surrounds the corpus luteum which, in turn, is surrounded by fluid (dark region). Note that smaller follicles are also present (to the left of the large follicle) in this dataset. Some of these follicles are at an earlier developmental

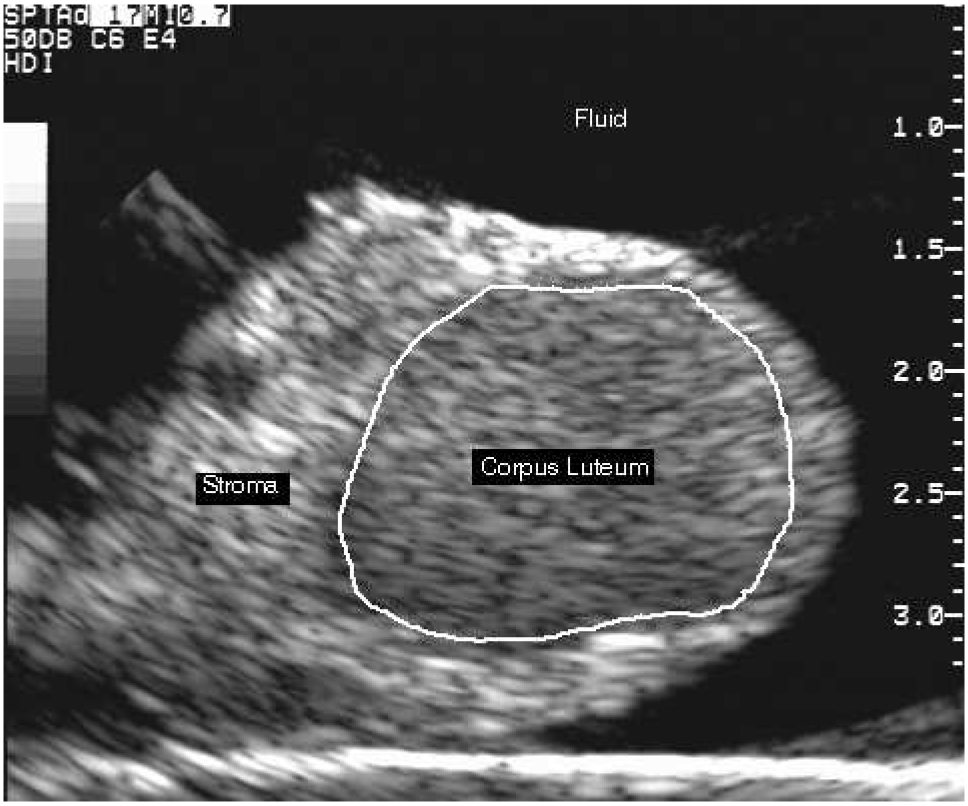


Figure 3.5: Cross-sectional view of major structures in bovine ovary data.

stage and filled with fluid. These follicles appear as dark regions, surrounded by stroma.

There are subtle textural difference between the corpus luteum and the stroma which are difficult to see with the untrained eye. Differentiation between these structures is even more difficult in the absence of the geometrical relationships described above (information on the geometry of the structures is not used in this thesis).

Samples were collected for two different classes: C1, corpus luteum, C2, stroma. Class C3, fluid, was not included in this dataset as it is significantly different from C1 and C2. Classes C1 and C2 are visually very similar and provided a significant challenge in the experiments. Actually, it was not possible to distinguish between the corpus luteum and stroma in previous work [80, 79, 77]. The exclusion of C3 reduced the processing requirements during the experiments. Based on expert knowledge of each class in the ultrasound data, 2D rectangular regions of 16×16 pixels were manually "cropped" out from various locations in the data. It is assumed that each region represents tissue from only one class. From these regions, 2D features were collected and subsequently used as training data.

The remaining datasets, 2C, 2C.cooc, 3C.4, 3C.8, 3C.16, and 3C.32 consist of 3D feature measures collected from 3D samples of the bovine ovary ultrasound volume data. Table 3.2 lists these features (see Chapter 4 for description of these features). Since the interpolated volume data is isometric, 3D blocks of size 16^3 points for datasets 2C and 2C.cooc were obtained by manually cropping out regions from series of consecutive 2D cross sections. Blocks of size 4^3, 8^3, 16^3 and 32^3 points were used for the 3C.i datasets. The difficulty in selecting the 3D blocks was that the 3D boundaries of a block had to be determined by analysing the 2D texture in the cross sections. This inevitably lead to some sampling error. Subsequently, $d = 99$, 3D feature measures were calculated and used as training data.

As with the 2C.2d dataset, datasets 2C and 2C.cooc contain samples from classes C1 and C2. Dataset 2C.cooc is a subset of 2C and contains $d = 24$ selected cooccurrence feature measures (ENT and LH for all parameters, see Table 3.2). The 3C.x datasets contain $d = 99$ features measures from all three classes. For all datasets,

Table 3.2: 3D feature measures evaluated in experiments ($d = -1$ indicates pixel distance).

num	G	θ	d	features
0-7	8	45	-1	ASM, CON, COR, ENT, CS, CP, IN, LH
8-15	8	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
16-23	8	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
24-31	64	45	-1	ASM, CON, COR, ENT, CS, CP, IN, LH
32-39	64	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
40-47	64	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
48-55	32	45	-1	ASM, CON, COR, ENT, CS, CP, IN, LH
56-63	32	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
64-71	32	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
72-79	128	45	-1	ASM, CON, COR, ENT, CS, CP, IN, LH
80-87	128	45	1	ASM, CON, COR, ENT, CS, CP, IN, LH
88-95	128	90	1	ASM, CON, COR, ENT, CS, CP, IN, LH
96-98				MEAN, ADEV, SDEV

$N = 1100$ points per class were used.

One major difficulty with the ultrasound data is the lack of normalisation among different scans. Various device-dependent parameters such as time-gain compensation and the zoom factor varied from scan to scan. With no calibration data available, the task of normalising the data from different scans was difficult. As a result, scans were chosen which appeared visually similar as far as the device-dependent parameters were concerned. Both the 2D and 3D datasets were obtained from these scans.

The experiments are presented as follows. Section 3.6.1 describes the goals, experimental environment and results obtained in evaluating the MVE estimator. Section 3.6.2 describes the evaluation of the Inck decision rule through the use of experiments and simulation. Section 3.6.3 describes the objectives, experimental environment and results obtained in evaluating various feature selection methods. As well the use of the Inck decision rule as a feature selection criterion is evaluated. Finally, Section 3.6.4 presents some experimental results obtained using the multiresolution classifier.

3.6.1 Parameter Fitting Using The MVE Estimator

The purpose of the experiments on the MVE estimator is to determine if it is successful at estimating the class distribution parameters in the presence of noise. Comparisons between the MVE estimator and the classical method of parameter estimation are performed. The assumption that every class distribution can be represented by a normal distribution with the parameters $(\mu, \mathbf{\Sigma})$, is also studied. μ is the mean (vector) and $\mathbf{\Sigma}$ is the covariance matrix. All experiments were performed on a 3-processor DEC AlphaServer 2100 4/200 with 256M of memory running OSF/1 V3.2. Timing information was obtained using the Unix "time" command. All times are given in CPU seconds.

For 2D and 3D data, it is possible to plot the points comprising a given class distribution. As well, the confidence intervals for a given parameter estimate can also be plotted. This provides the ability to visualise the fit of the parameters to the data. Some results obtained from the experiments using synthesised data are presented in this form. Quantitative analysis is also necessary in order to evaluate the performance of the MVE estimator. To accomplish this, a given class distribution is corrupted with the addition of outliers. The ability of the MVE estimator to reject these outliers is ascertained by comparing the number of outliers added to the data to the number of outliers rejected while estimating the class parameters. Outliers are identified as those points which are beyond the given confidence interval for the parameter estimates. When using the reweighted MVE, the value of c in Equation (3.5) determines the given confidence interval. Typically, a value of $c = \chi^2_{p,0.99}$ (p is the dimension of the data distribution) is chosen for a 99% confidence interval.

Data Sets

Numerous experiments are performed with synthesised data. The synthesised data is created from different distributions with varying distribution parameters. Table 3.3 lists the distributions and their associated parameters, used in the experiments. Different distributions (normal and non-normal) are used to determine the

Table 3.3: Data distributions and associated parameters used in creating synthesised datasets.

Class	Distribution	Parameters
C1	normal	mean (0,0), unit variance
C2	normal	mean (4,8), unit variance
C3	normal	mean (0,0,0), unit variance
C4	normal	mean (4,8,4), unit variance
C5	normal	mean (4,8,8), unit variance
C6	rayleigh	offset (0,0), unit variance
C7	rayleigh	offset (4,8), unit variance
C8	rayleigh	offset (0,0,0), unit variance
C9	exponential	offset (0,0), unit variance
C10	exponential	offset (4,8), unit variance
C11	uniform	min (1,5), max (7,11)

effect that the normal distribution assumption has on parameter estimation. 2D and 3D distributions are used to facilitate ease of visualisation. For each dataset, one of the eleven classes, consisting of 500 data points, is selected as the class for which the (normal) distribution parameters are to be estimated. A second class is then selected as a source for outlier data. The percentage of outliers points added to the dataset is increased from 0% (0 points) to 50% (500 points) of the total number of points in the dataset to determine the breakdown point of the MVE estimator. In all experiments on the synthesised data, the reweighted MVE estimator is used.

Results

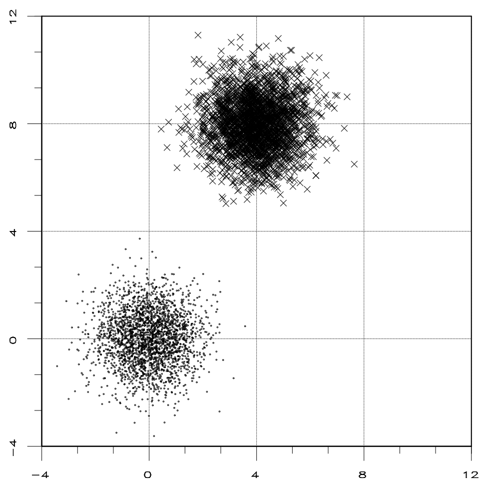
Figures 3.6 - 3.11 show the results obtained for parameter estimation using the MVE and classical methods. Various data source and outlier source combinations are presented. Figures 3.6(a), 3.7(a), 3.8(a), 3.9(a), 3.10(a) and 3.11(a) show the 2D datasets evaluated. In each of these figures, all 500 of the outlier points are shown (indicated by "x"). Figures 3.6(b), 3.7(b), 3.8(b), 3.9(b), 3.10(b) and 3.11(b) each show the corresponding 99% confidence interval ellipses for the classical parameter estimates with 0%, 10%, 20%, 30%, 40% and 50% outliers. With 0% outliers added, the parameter estimates result in a 99% confidence interval ellipse which fits tightly around the (non-outlier) class data. As the percentage of outliers increase, the confidence interval ellipse increases in size due to the accommodation of the outliers in the parameter estimation.

Figures 3.6(c), 3.7(c), 3.8(c), 3.9(c), 3.10(c) and 3.11(c) each show the resulting 99% confidence interval ellipses for the MVE parameter estimates with 0%, 10%, 20%, 30%, 40% and 50% outliers. In all cases, independent of the distributions, the corresponding 99% confidence ellipses fit tightly around the non-outlier data except when 50% outliers are added. This seems to verify the theoretical maximum tolerance of up to 50% outliers for MVE parameter estimation.

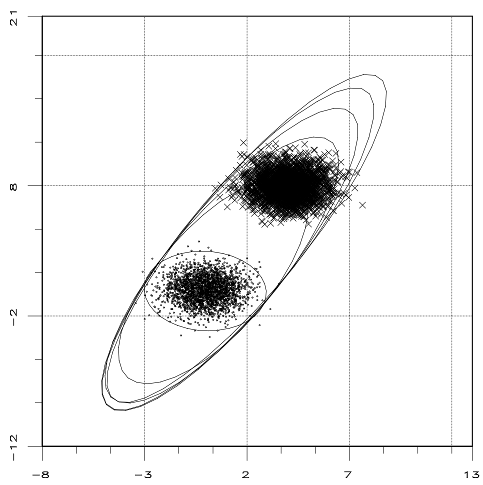
The above results obtained from the 2D datasets seems to support the conclusion that the MVE parameter estimator can tolerate the theoretical maximum of up to 50% outliers, independent of the type distribution from which the parameters are estimated. 3D datasets were evaluated to determine if this conclusion could be further supported. The results from two, 3D datasets are shown in Figures 3.12 and 3.13. The first dataset consisted of data from class C3 with outliers added from class C4. Experiments were performed with 0%, 10%, 20%, 30%, 40% and 50% outliers added to the dataset.

Figure 3.12(a) shows the dataset as well as the corresponding 99% confidence interval ellipsoids for the MVE and classical parameter estimates with 30% outliers in the dataset. There is a significant difference between the two ellipsoids. The figure shows that the ellipsoid for the MVE parameter estimates does not include outlier data while the ellipsoid for the classical parameter estimates includes all the outlier data. Figure 3.12(b) illustrates the same data from a different perspective (including a cut-away view of the ellipsoid for classical parameter estimates). With the addition of 50% outliers (not shown), the 99% confidence interval ellipsoids for both the MVE and the classical parameter estimates included the outliers points. This follows the results obtained for the 2D datasets.

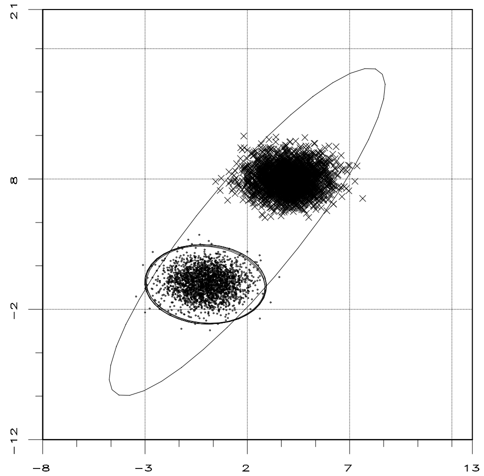
Figure 3.13(a) shows the second dataset as well as the corresponding 99% confidence interval ellipsoids for the MVE and classical parameter estimates with 30% outliers in the dataset. The results seems to follow those obtained for the first



(a) data with outliers

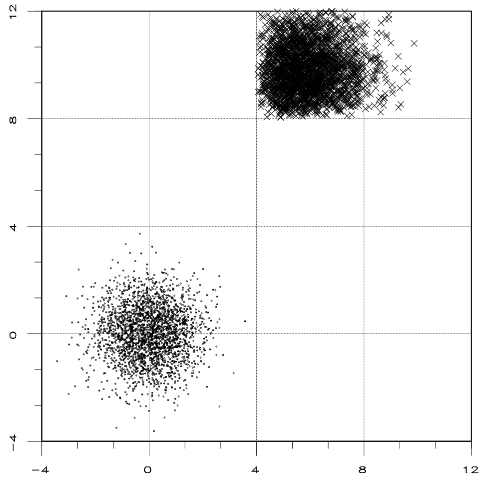


(b) classical CI ellipses

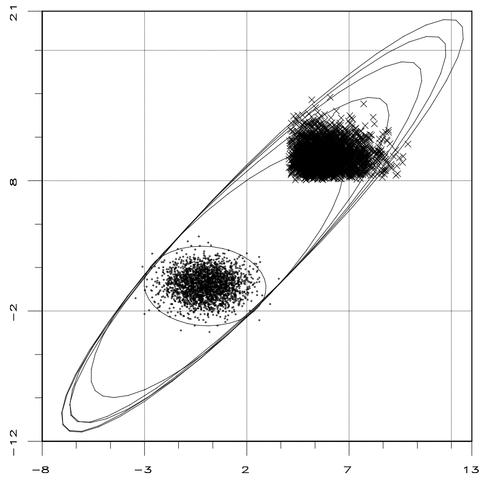


(c) MVE CI ellipses

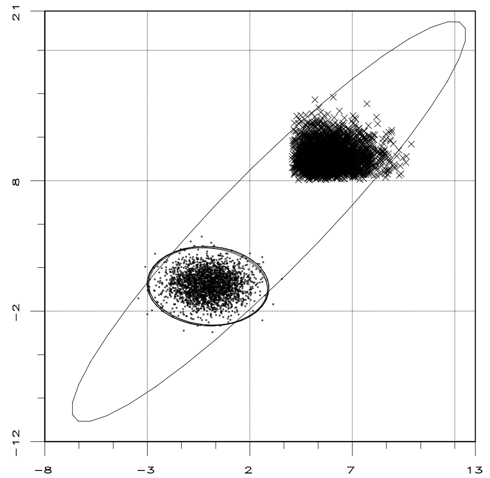
Figure 3.6: Data distributions and 99% confidence intervals (CI) for varying percentage of outliers; data source is C1, outlier source is C2.



(a) data with outliers

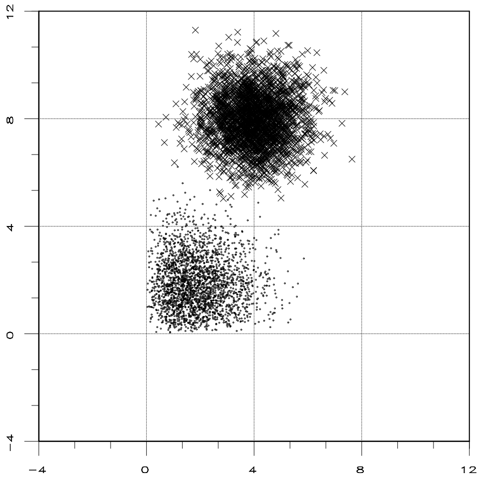


(b) classical CI ellipses

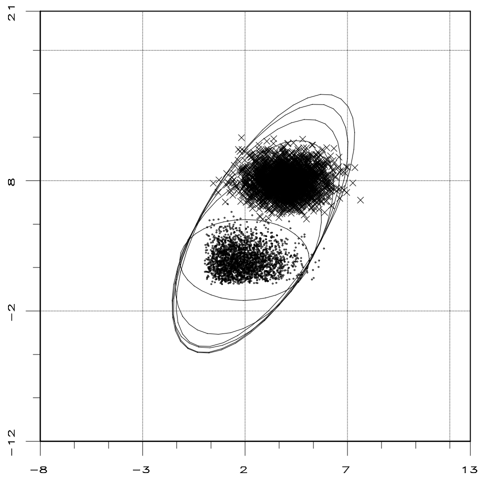


(c) MVE CI ellipses

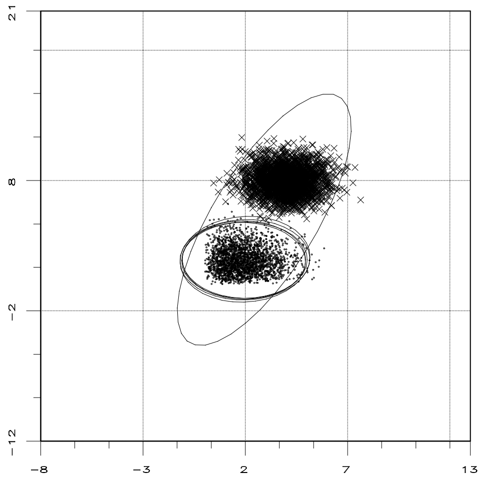
Figure 3.7: Data distributions and 99% confidence intervals (CI) for varying percentage of outliers; data source is C1, outlier source is C6.



(a) data with outliers

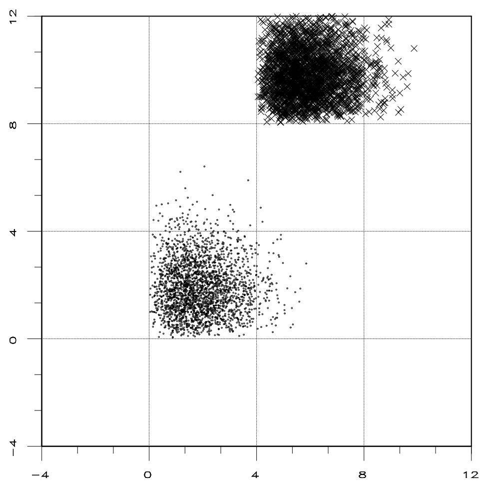


(b) classical CI ellipses

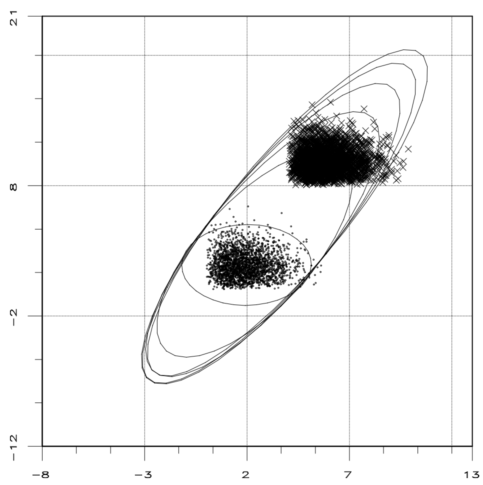


(c) MVE CI ellipses

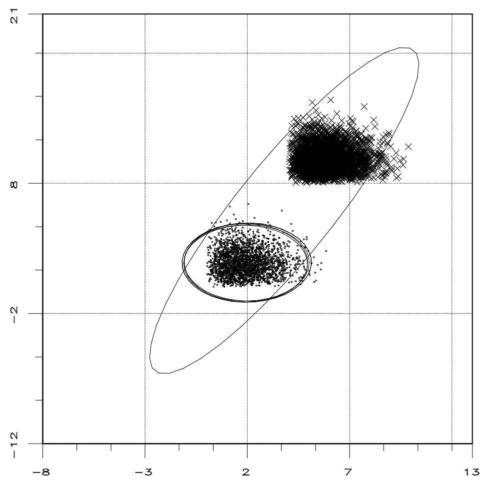
Figure 3.8: Data distributions and 99% confidence intervals (CI) for varying percentage of outliers; data source is C5, outlier source is C2.



(a) data with outliers

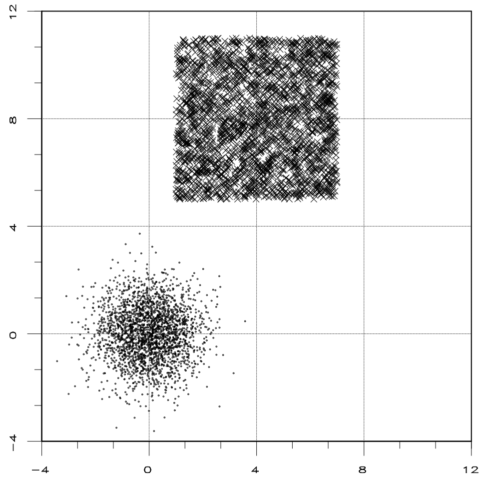


(b) classical CI ellipses

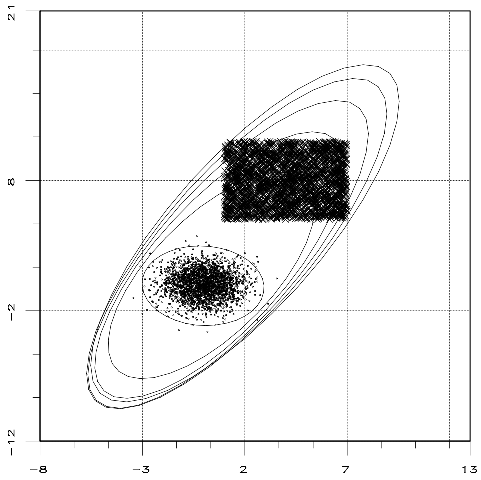


(c) MVE CI ellipses

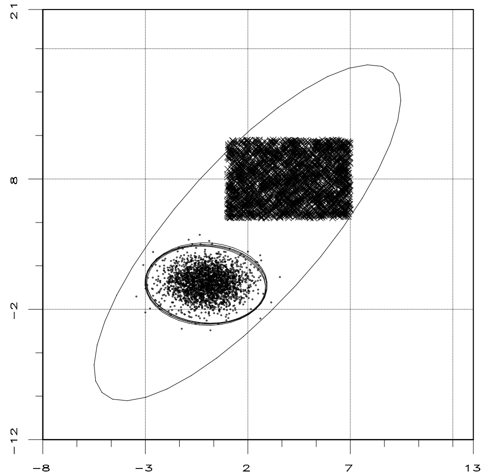
Figure 3.9: Data distributions and 99% confidence intervals (CI) for varying percentage of outliers; data source is C5, outlier source is C6.



(a) data with outliers

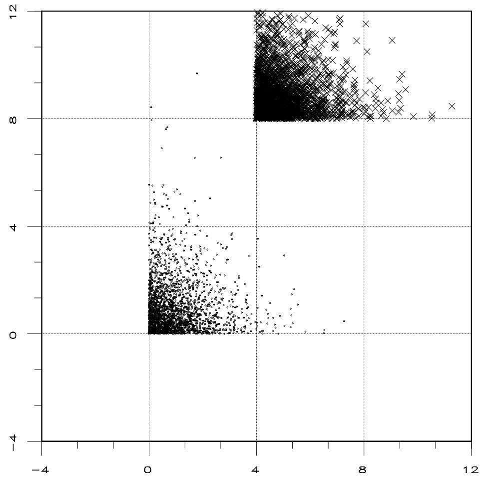


(b) classical CI ellipses

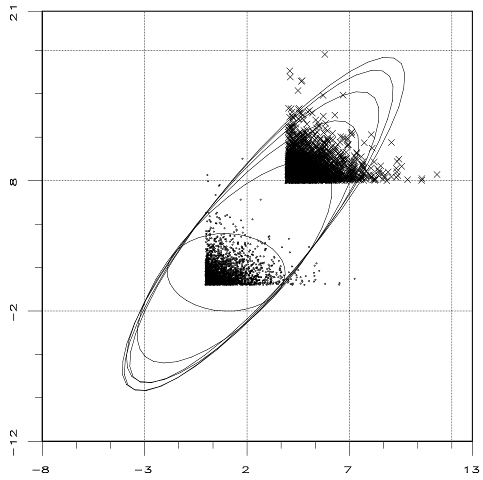


(c) MVE CI ellipses

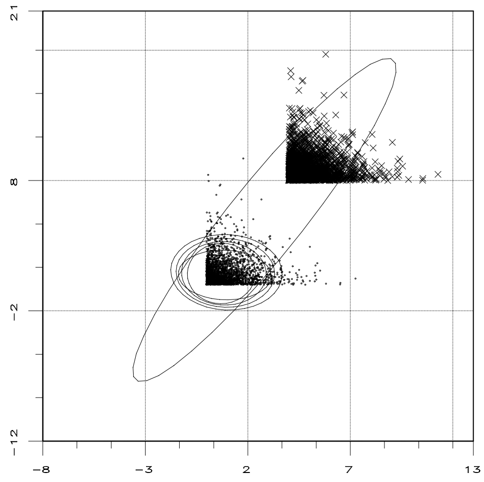
Figure 3.10: Data distributions and 99% confidence intervals (CI) for varying percentage of outliers; data source is C1, outlier source is C10.



(a) data with outliers

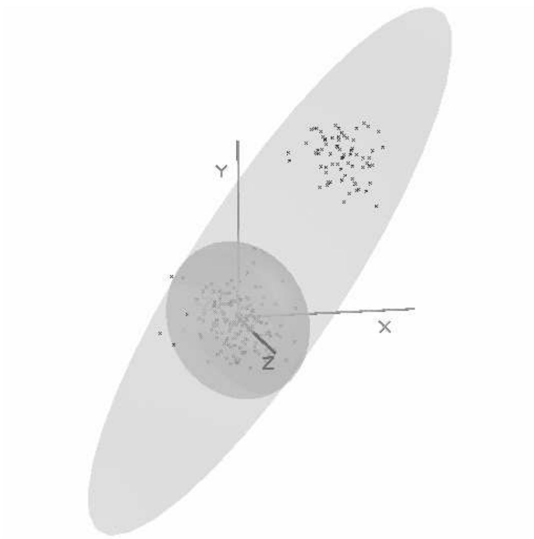


(b) classical CI ellipses

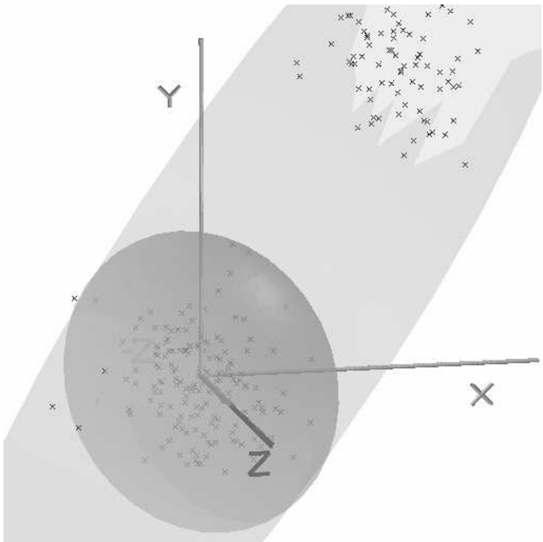


(c) MVE CI ellipses

Figure 3.11: Data distributions and 99% confidence intervals (CI) for varying percentage of outliers; data source is C8, outlier source is C9.



(a) MVE (small) and classical (large) ellipsoids

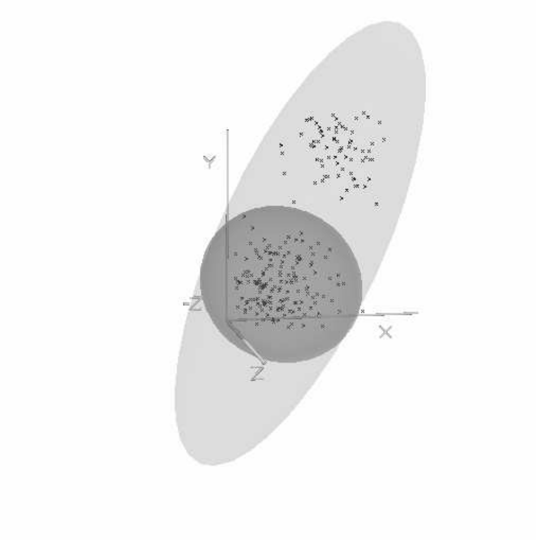


(b) different perspective

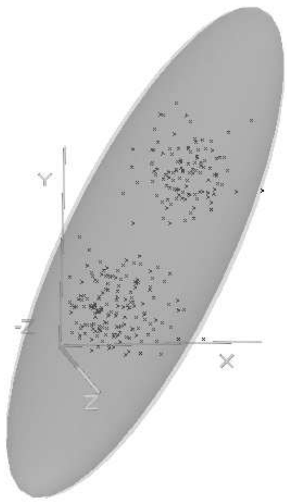
Figure 3.12: 99% confidence interval ellipsoids for 30% outliers; data source is C3, outlier source is C4.

dataset. However, Figures 3.13(b) and 3.13(c) show that with the addition of 40% outliers, the confidence ellipsoid for the MVE parameter estimates is approximately the same size as the confidence interval ellipsoid for the classical parameter estimates. This suggests the MVE estimator can not estimate the parameters of this dataset with 40% or more outlier points. This result contradicts those obtained in previous experiments. In the 2D case, the type of data distribution did not seem to affect the MVE estimator. In the 3D case, there is a difference in the results between the first dataset and this dataset. This difference is attributed to the different distributions of the non-outlier data comprising the datasets.

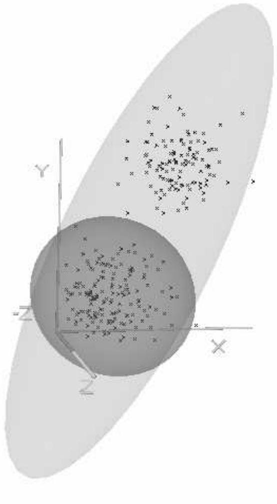
Because of this result, it was necessary to examine in more detail the relationship between the percentage of outliers added vs. the percentage of non-outlier data from which the MVE parameters are calculated (conversely, the percentage of outliers added vs. the percentage of outliers removed by the MVE estimator when calculating the parameters).



(a) CIs for 30% outliers

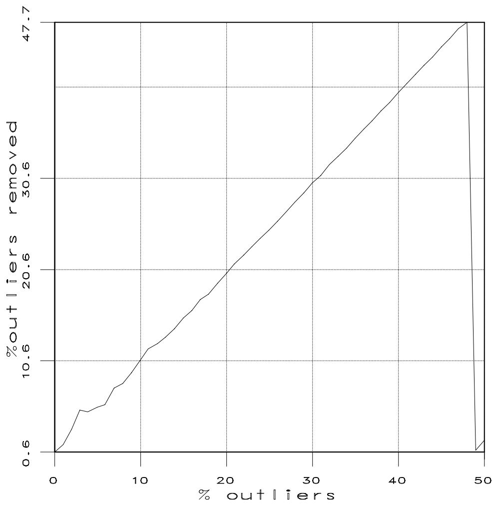


(b) CIs for 40% outliers

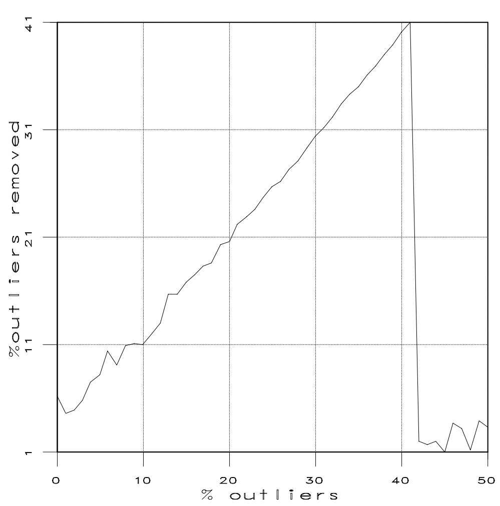


(c) MVE CIs for 30% and 40% outliers

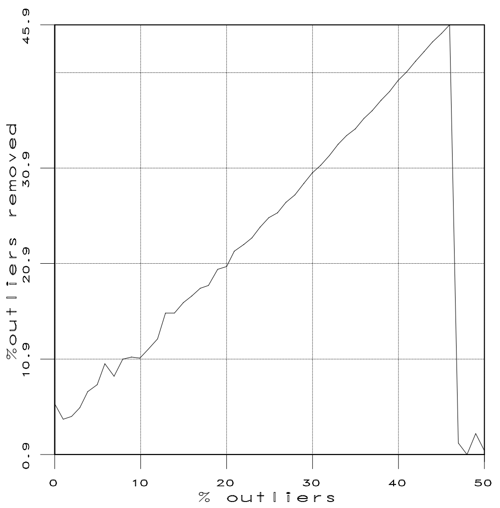
Figure 3.13: 99% confidence interval (CI) ellipsoids for various outlier percentages; data source is C7, outlier source is C4.



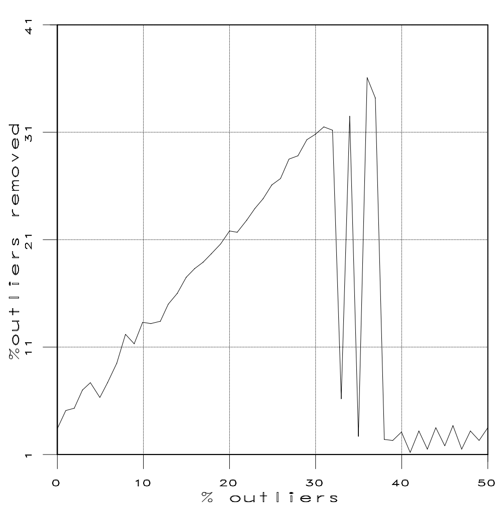
(a) data: C1, outliers: C2



(b) data: C3, outliers: C4



(c) data: C3, outliers: C5



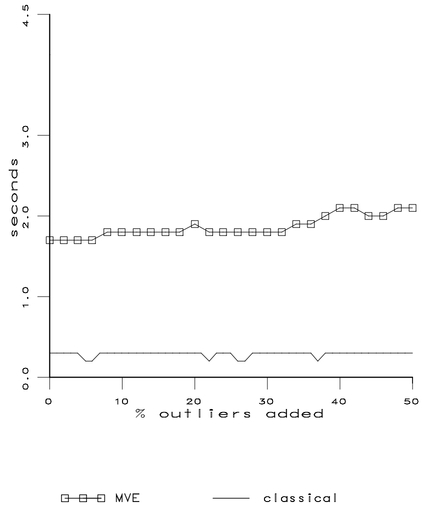
(d) data: C8, outliers: C4

Figure 3.14: Plots of percentage of outliers added vs. percentage of outliers removed for various data source and outlier source combinations.

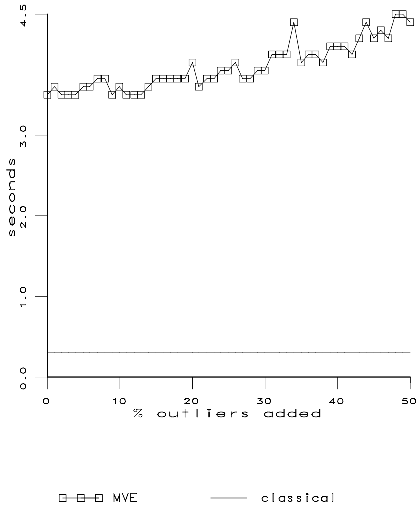
The graphs in Figure 3.14 show the percentage of outliers added vs. the percentage of outliers removed using the MVE parameter estimates, for various class data source and outlier source combinations. The MVE estimator fails to tolerate outliers when the percentage of outliers removed is significantly lower than the percentage of outliers added. It can be seen from these graphs that the estimator is tolerant to varying percentages of outliers.

From these results, the differences in outlier tolerance can be attributed to two factors: the data dimension and the number of samples used in MVE parameter estimation. For the 2D dataset in Figure 3.14(a), up to 48% outliers are tolerated before the parameter estimation fails. This is near the theoretical maximum of 50%. For the remaining 3D datasets in Figures 3.14(b) - 3.14(d), the outlier tolerance is less. 41% outliers are tolerated in Figure 3.14(b), 46% in Figure 3.14(c) and 36% in Figure 3.14(d). The two dips in Figure 3.14(d), at 33% and 35% outliers added, are attributed to random sampling error in selecting the s subsamples for MVE parameter estimation (see Section 3.1). This sampling error also contributes to the decrease in outlier tolerance as the dimension increases. Although s increases with the dimension, the number of subsamples does not increase enough to provide a subsample which is representative of the non-outlier data. Note that when the distributions are normal (Figures 3.14(a) - 3.14(c)), a higher percentage of outliers are tolerated. As well, as the outlier data is moved farther from the non-outlier data (Figure 3.14(c)), the percentage of outliers tolerated is higher.

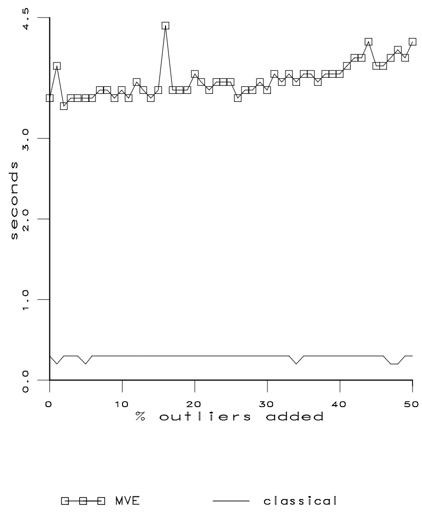
The graphs in Figure 3.15 show the resulting times for MVE and classical parameter estimation for various class data source and outlier source combinations. It can be seen from these graphs that time required for the MVE parameter estimation is significantly greater than for the classical parameter estimation. As well, the time required for MVE parameter estimation increases as the sample size increases. Both of these results are attributed to the matrix inversions and median calculations required in computing the MVE parameter estimates. The number of times these calculations are performed is dependant on s, the number of subsamples evaluated in estimating the parameters. s has an exponential distribution which is dependent



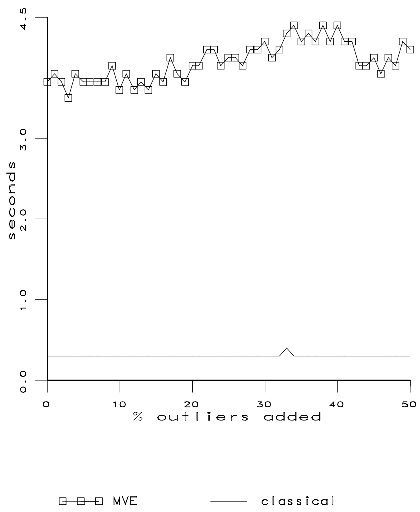
(a) data: C1, outliers: C2



(b) data: C3, outliers: C4



(c) data: C3, outliers: C5



(d) data: C8, outliers: C4

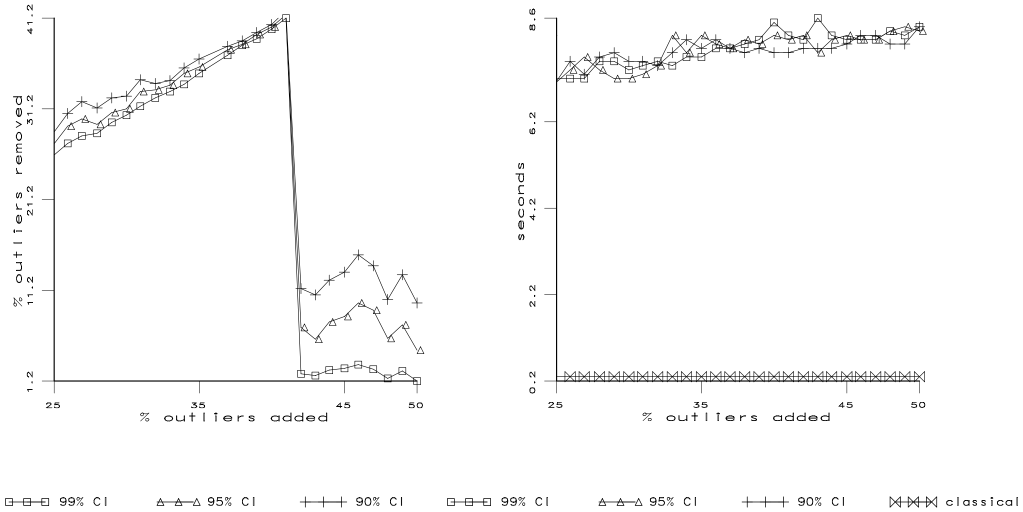
Figure 3.15: Plots of percentage of outliers added vs. time for parameter estimation for various data source and outlier source combinations.

on the dimension of the data so it follows that the time required for MVE parameter estimation has a similar distribution. In fact, to constrain the time for parameter estimation, a limit must be placed on s as the dimension of data gets large. This has an effect on the quality of the estimates as the probability of selecting a subsample with no outliers is reduced.

In order to examine, in more detail, the effect an increase in dimension has on the MVE parameter estimates, a final set of experiments was performed using 4D, unit variance normal distributions. The source for the non-outlier data consisted of 500 points having a mean of (0,0,0,0) and the source for the outlier data consisted of 500 points having a mean of (4,8,8,8). The percentage of outliers added was varied from 25% (250) to 50% (500) of the total number of points. As well, c, which is based on the confidence interval at which outliers are rejected (c.f. Equation (3.5)), was set for the 99%, 95% and 90% confidence intervals. The effect of this was to increase the number of outliers detected as well as decrease the number of non-outliers represented by the parameter estimates.

The graphs in Figure 3.16 show the results obtained for this set of experiments using the MVE estimator. Figure 3.16(a) plots the percentage of outliers added vs. the percentage of outliers removed. It can be seen that decreasing the confidence interval had little effect on increasing the outlier tolerance. The only noticeable effect was that more non-outlier data points were removed.

Figure 3.16(b) shows the percentage of outliers added vs. the time required for MVE parameter estimation. The time required for MVE parameter estimation is over double the time required for the 3D datasets in Figure 3.15. This seems to support the conclusion that as the data dimension increases linearly, the time required for MVE parameter estimation increases exponentially. The time required for classical parameter estimation of the 4D dataset is provided as a reference. For this method, the time required for parameter estimation was not influenced by the dimension or sample sizes evaluated in all the experiments described in this section.



(a) outliers added vs. outliers removed

(b) outliers added vs. time

Figure 3.16: Plots of time required for MVE parameter estimation and outliers removed for 4D, unit variance, normal distribution and varying c; data source has mean (0,0,0,0) and outliers source has mean (4,8,8,8).

3.6.2 Inck Decision Rule

There are two main goals in evaluating the Inck decision rule:

- verify the original results obtained by Dubuisson and Masson [26],
- determine if the modifications to the decision rule result in at least the same or better performance than the original version.

In order to satisfy the first goal, the original decision rule with reject was implemented as an extension to a software application called TOOLDIAG [99]. TOOLDIAG provides a set of tools for performing statistical pattern recognition. The Inck decision rule was also implemented as an extension to the software. All the simulations performed by Dubuisson and Masson [26] were recreated (as close as possible) and run using both the original and Inck decision rule. To satisfy the second goal, additional simulations were run. As well, experiments were performed using both synthetic and real data (see Section 3.6.3). All the simulations and experiments were performed on a 3-processor DEC AlphaServer 2100 4/200 with 256M of memory running OSF/1 V3.2.

Simulations

A substantial analysis of the original decision rule with reject was performed by Dubuisson and Masson [26]. Their assumption of two ($M = 2$), normally distributed, univariate classes with means m_1 and m_2 and standard deviations σ_1 and σ_2 are inherited, without loss of generality. For most of the simulations it is assumed that $\sigma_1 = \sigma_2 = \sigma$. When this is not the case, the class standard deviations are characterised by the ratio $sdr = \sigma_1/\sigma_2$. The two classes are separated by a distance, $d = (m_1 - m_2)/\sigma$. The parameters varied are C_a, C_d, d and sdr. The results of each simulation are the probabilities P_C, P_E, P_{RD} and P_{RA}. The probabilities are calculated using the integrals in Equation (3.15) - Equation (3.18) in the range (Rl, Ru) where $Rl \approx -\infty$ and $Ru \approx \infty$. In the simulations, $Rl = m_1 - 6\sigma$ and $Ru = m_2 + 6\sigma$. The integrals are approximated by summations of the mixture

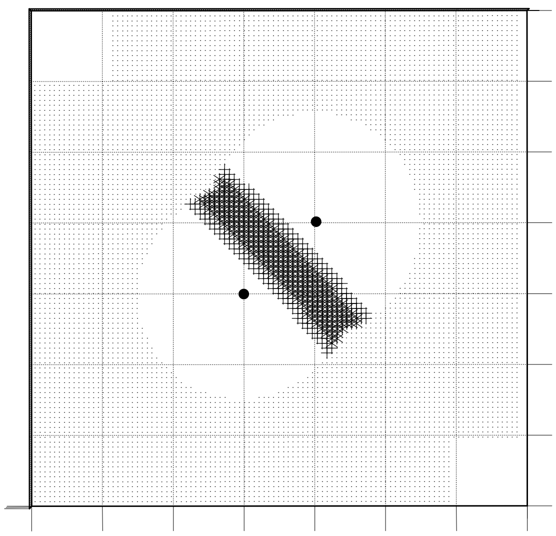


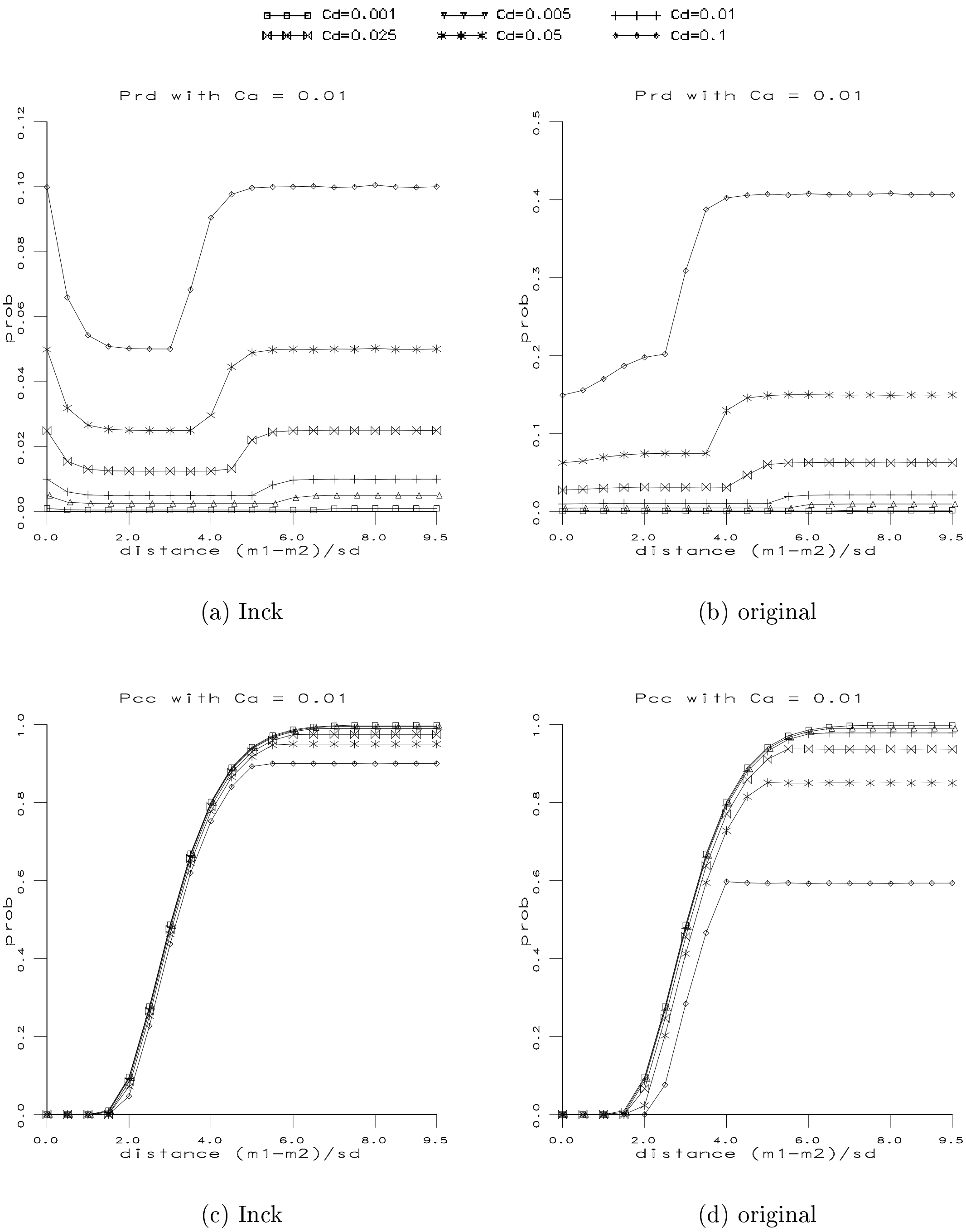
Figure 3.17: Integration regions for various probabilities.

density sampled at discrete, weighted grid points in the region (in the simulations, all weights are assumed equal).

Figure 3.17 depicts the integration regions for the various probabilities for a mixture density based on two, bivariate, normal classes with unit standard deviations in each dimension. Each sample point is evaluated and its probability contributes to either P_{RA} (dark middle strip separating the two classes), P_E (two adjacent lighter strips), P_C (circular white regions) or P_{RD} (grey surrounding dots).

Results For Varying Distance

Figures 3.18 and 3.19 show the resulting probabilities, for given values of C_d, when $C_a = 0.01$ and the between class distance, d, is varied from [0, 9.5]. The results obtained for the original decision rule, shown in Figures 3.18(b), 3.18(d), 3.19(b) and 3.19(d), closely resemble those originally obtained by Dubuisson and Masson [26], thus, verifying their results. There are similarities, but, also noticeable differences between these results and those obtained for the Inck decision rule in



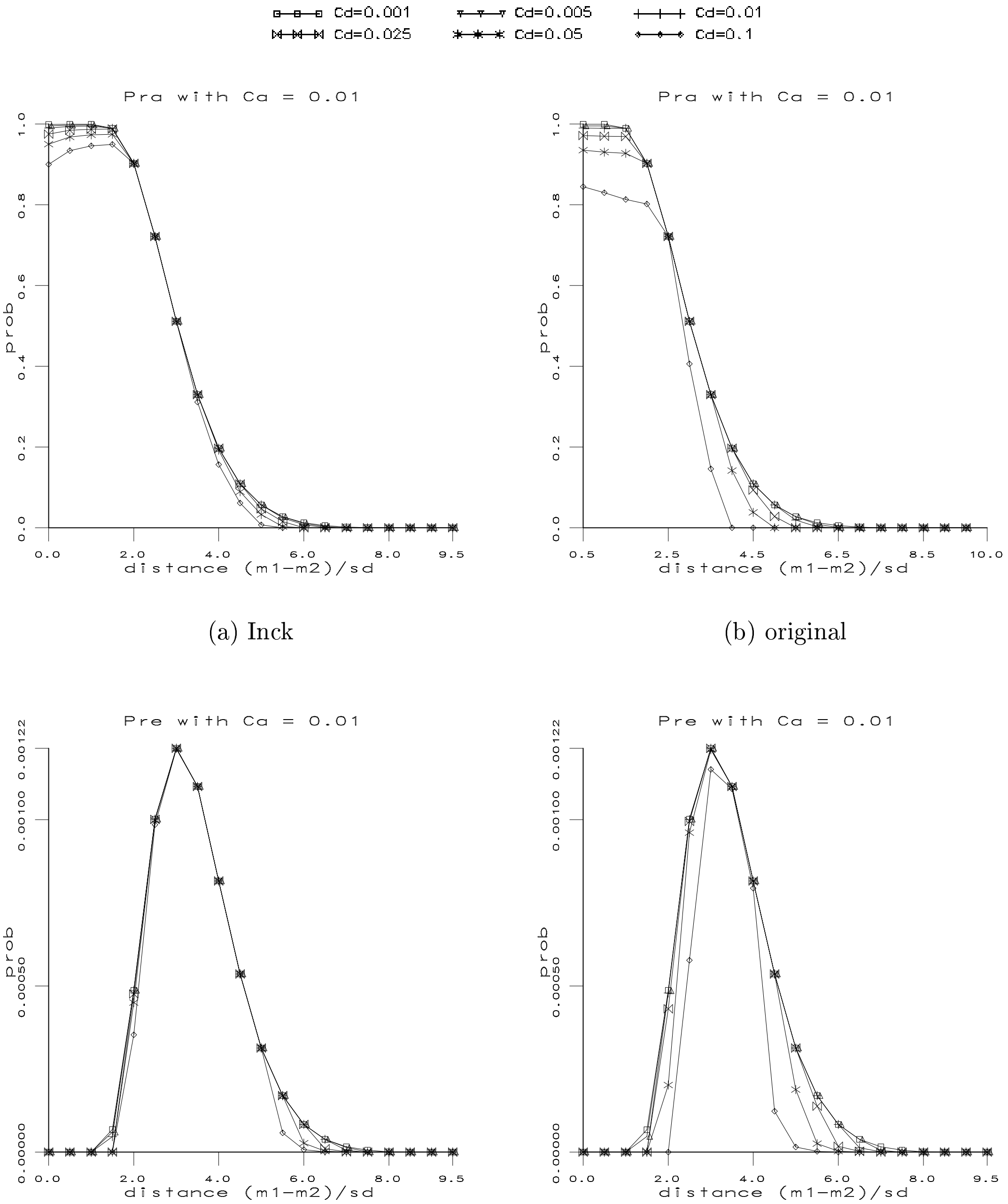
(a) Inck

(b) original

(c) Inck

(d) original

Figure 3.18: Probability of distance reject (Prd) and correct classification (Pcc) for varying between class distance, $C_a = 0.01$ and given values of C_d.



(a) Inck

(b) original

(c) Inck

(d) original

Figure 3.19: Probability of ambiguity reject (Pra) and error (Pre) for varying between class distance, $C_a = 0.01$ and given values of C_d.

shown in Figures 3.18(a), 3.18(c), 3.19(a) and 3.19(c).

The major difference between the rules is the probability of distance reject, P_{RD}. It can be seen from Figures 3.18(a) and 3.18(b) that the distribution of P_{RD} is quite different between the rules. For the Inck rule, $P_{RD} = C_d$ initially, decreases to $P_{RD} = C_d/2$, and then increases back to $P_{RD} = C_d$. This is predicted in theory. For the original rule, P_{RD} monotonically increases as the distance between classes increases. There is no obvious relationship between C_d and P_{RD} except that P_{RD} is larger for larger values of C_d. As well, P_{RD} is larger than it is for the Inck rule.

The difference between P_{RD} for the original and Inck rules influences the probability of correct classification, P_C. Figures 3.18(c) and 3.18(d) show the difference between P_C for the two rules. The maximum achievable P_C is significantly lower for the original rule, for larger values of C_d. Also, as C_d increases, P_C decreases significantly for the original rule. For the Inck rule, P_C only decreases by the value of C_d. The probability of ambiguity reject, P_{RA}, and probability classification error, P_E, for the rules are shown in Figure 3.19. The differences between the Inck and original rules is not very significant.

For a fixed C_a ($\leq (M-1)/M$), the following conclusions can be drawn from the above results:

- for both rules, as the distance between classes increases, the probability of correct classification increases and the probability of ambiguity reject decreases,
- the differences between the Inck and original decision rules are significant for the probability of distance reject and correct classification and are not significant for the probability of ambiguity reject and classification error, and
- the probability of classification error is 0 when the classes are far enough apart.

Results For Different Variances

To further exemplify the differences between the Inck and original decision rules, simulations were performed in which the ratio between the class standard deviations,

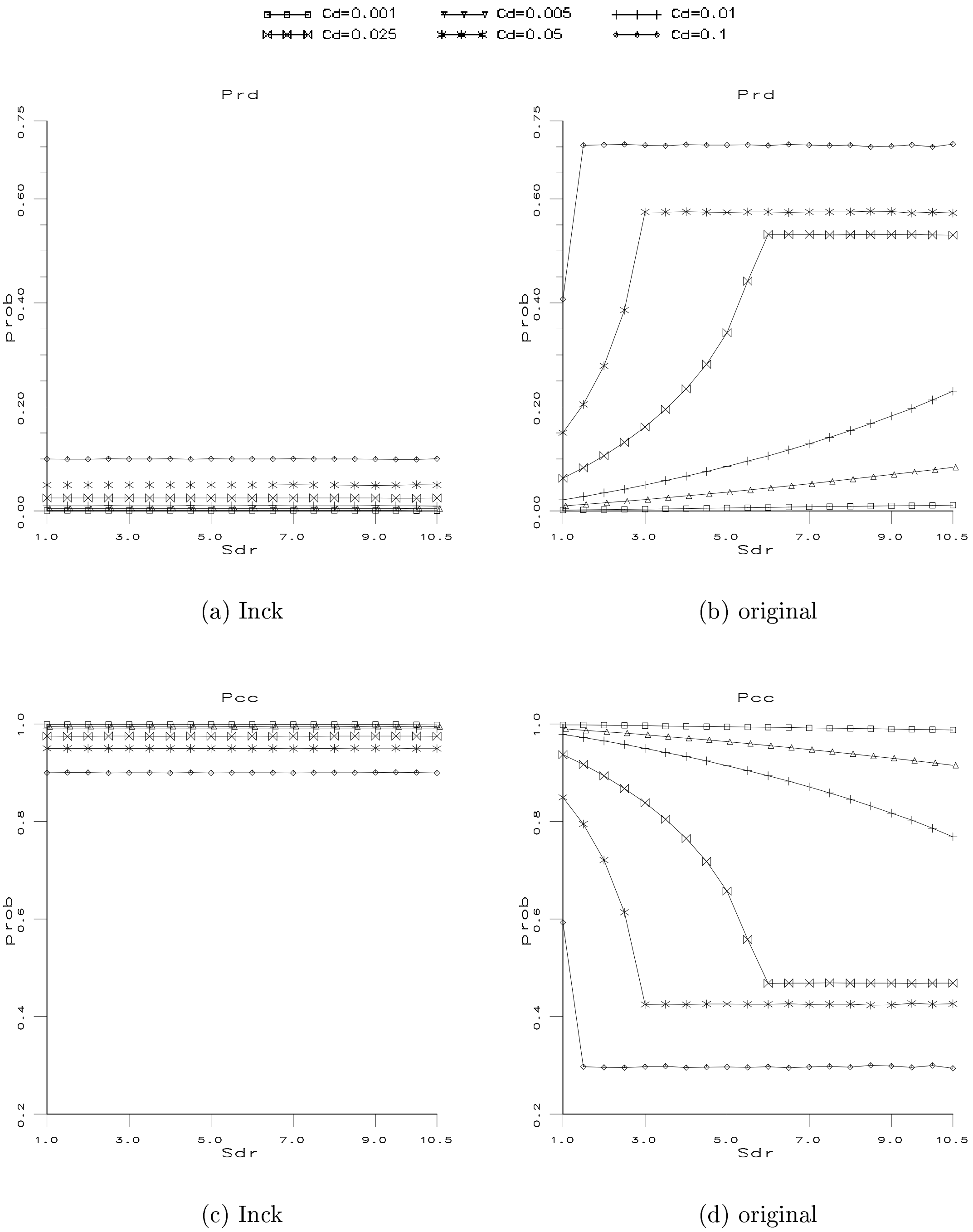


Figure 3.20: Probability of distance reject (Prd) and correct classification (Pcc) for varying class standard deviation ratios (Sdr), $Ca = 0.1$ and given value of C_d.

sdr, was varied from [1.0, 10.5]. That is, σ_2 was decreased in size while σ_1 was held constant. Results for $C_a = 0.1$ and given values of C_d are shown in Figure 3.20. The distance between classes is large enough such that there is no overlap between the classes. It is obvious from these results that the probability of distance reject and correct classification are influenced by the rule used in calculating the probabilities. Further, it can be seen for the Inck decision rule that these probabilities are independent of the ratio between the class standard deviations. As well, the value of C_d indicates the probability of distance rejection. For the original decision rule, the probability of distance reject increases substantially as sdr increases. This is certainly undesirable when classifying data from varying size standard deviations.

For the remainder of the simulations and experiments the Inck decision rule is used.

Results for Varying Ca

These simulations were performed to determine the effect that C_a has on the various probabilities. For a fixed C_d and distance between classes, simulations were run for varying C_a in the range [0.0, (M-1)/M = 0.5]. Figure 3.21 shows the resulting probability of correct classification, P_C, classification error, P_E, and ambiguity reject, P_{RA} for $C_d = 0.01$ and given between class distances. From Figure 3.21(a) it can be seen that $P_C = 0$ when $C_a = 0$ and P_C is maximum when $C_a = (M-1)/M$. The rate at which P_C (monotonically) approaches its maximum value depends on the distance between class means. As the distance increases, the rate at which P_C approaches its maximum increases. This implies that for larger distances, a smaller value of C_a can be used to maintain a high probability of correct classification, while guarding against ambiguous decisions. Figure 3.21(c) confirms this as the probability of ambiguity rejection (monotonically) decreases rapidly for larger between class distances. It can also be noted from this figure that $P_{RA} = 1$ when $C_a = 0$ (no class decision allowed) and $P_{RA} = 0$ when $C_a = (M-1)/M$ (uncertainty allowed in class decision). Figure 3.21(b) shows the probability of classification error. As expected, the maximum classification error occurs when $C_a = (M-1)/M$. It is

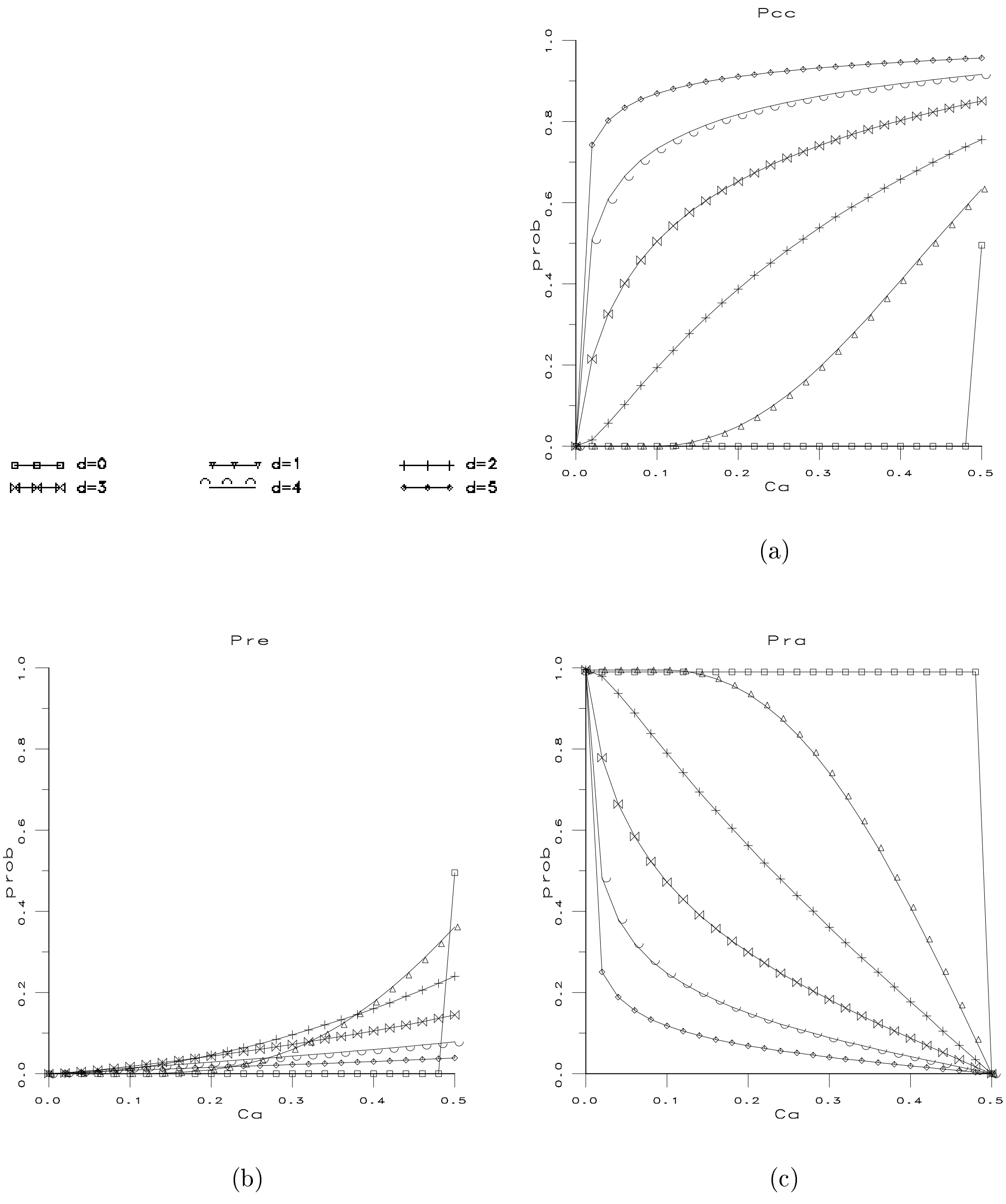


Figure 3.21: Probability of correct classification (pcc), classification error (pe) and ambiguity reject (pra) for varying Ca, $Cd = 0.01$ and given between class distances.

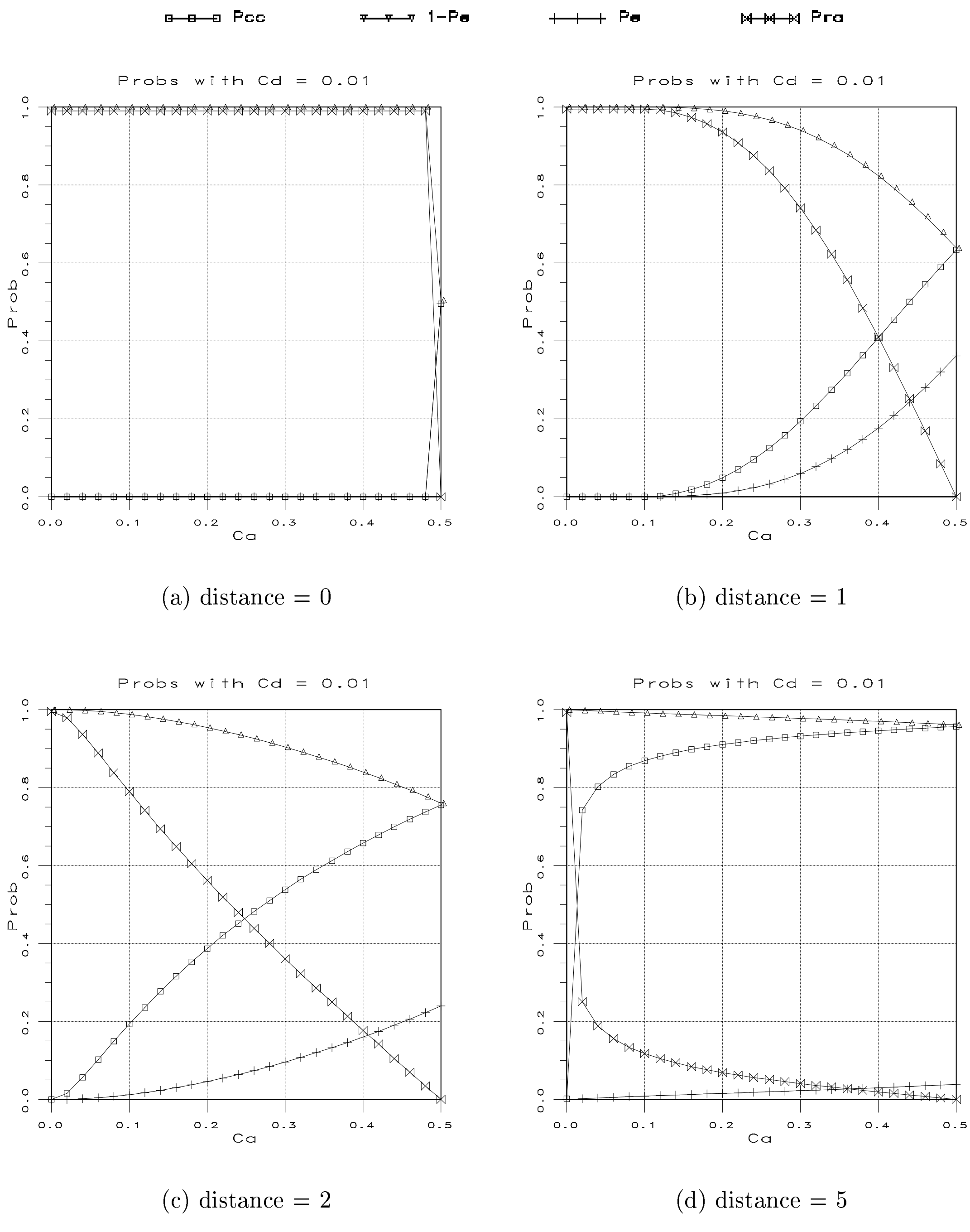


Figure 3.22: Probabilities of correct classification (pcc), classification error (pe), 1 - pe and ambiguity reject (pra) for varying Ca, $Cd = 0.01$ and given between class distances, d.

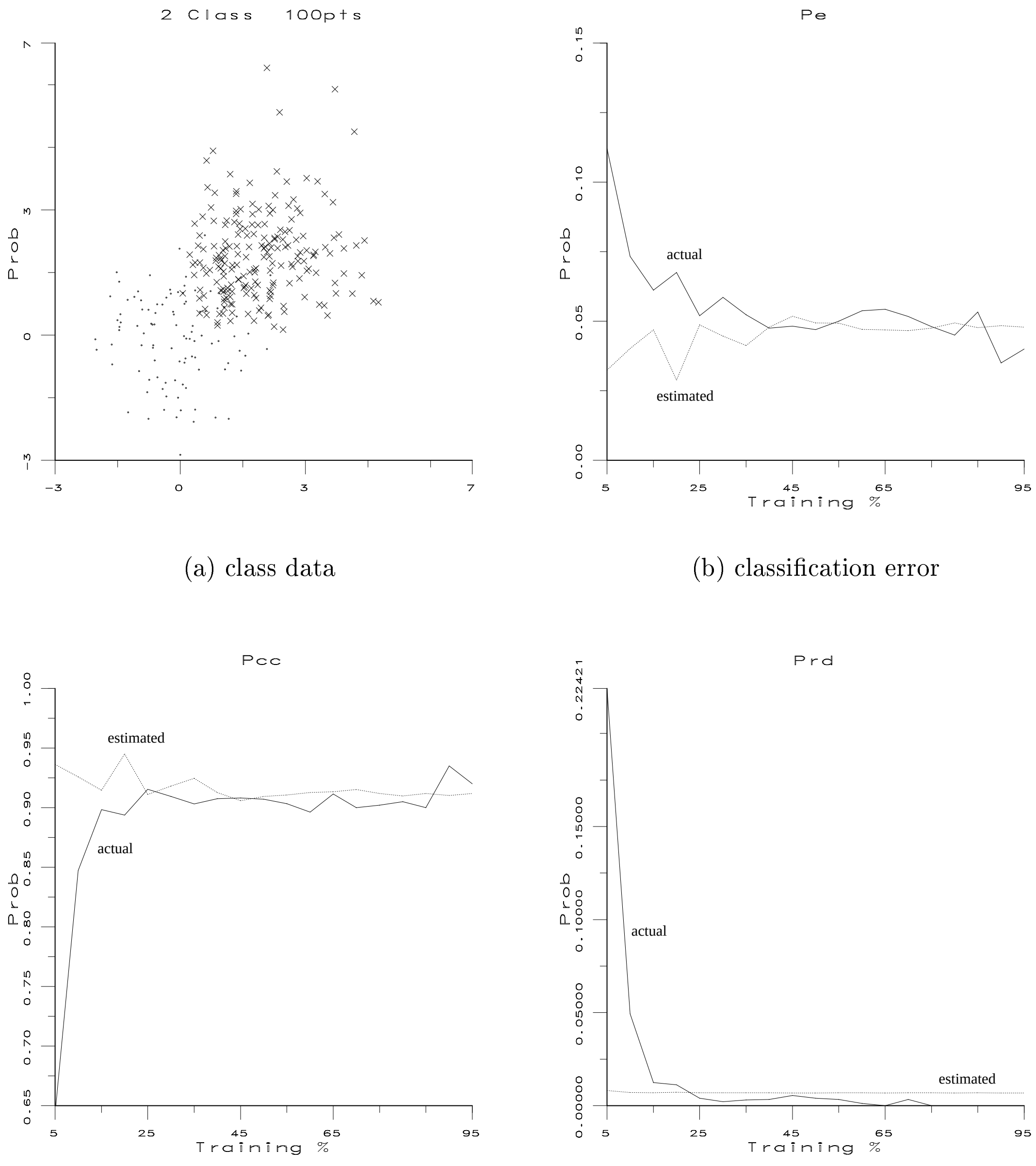
intuitive that the most errors occur for this value of C_a since the most uncertainty in a class decision is allowed.

Figure 3.22 displays the same probabilities in a different manner. The probabilities are displayed in the same graph, for a given distance between class means. These graphs illustrate the tradeoff between P_C and P_E when selecting a value for C_a. For example, it can be seen from Figure 3.22(c) that selecting a larger value for C_a results in a greater P_C, but also results in a greater P_E. The rate at which these probabilities increase, with increasing C_a, is quite dependent on the between class distance. For example, in Figure 3.22(d), P_E increases very little over a wide range of C_a while P_C increases substantially. Thus, when the distance between classes is large, it is appropriate to choose $C_a = (M-1)/M$ as this results the largest probability of correct classification while keeping the probability of classification error low.

Results for Varying Sample Size

These experiments were performed to determine the effect that the number of training samples had on the performance of the Inck decision rule. As well, estimation of the probabilities, given in Equations (3.20) - (3.25), were compared to the actual calculation based on the mixture density integrals, using Equations (3.13) - (3.17). This was done to determine if the probability estimates were a sufficient approximation of the actual probabilities.

Numerous experiments were performed. The results for two different synthetic datasets are presented. The first dataset consists of two normally distributed classes with means (0,0) and (2,2) and unit variance. Each class contains $N = 100$ points. Figure 3.23(a) depicts these class distributions. The training data was split into two sets: one used for estimating the class parameters, the other for estimating the class probabilities. The size of the parameter estimation set varied from 5% to 95% of N while the remaining points were used to estimate the class probabilities. Ten iterations at each size were performed and the results averaged. An equal number of samples from each class was drawn at random, for each iteration. The probabilities



(a) class data

(b) classification error

(c) correct classification

(d) distance reject

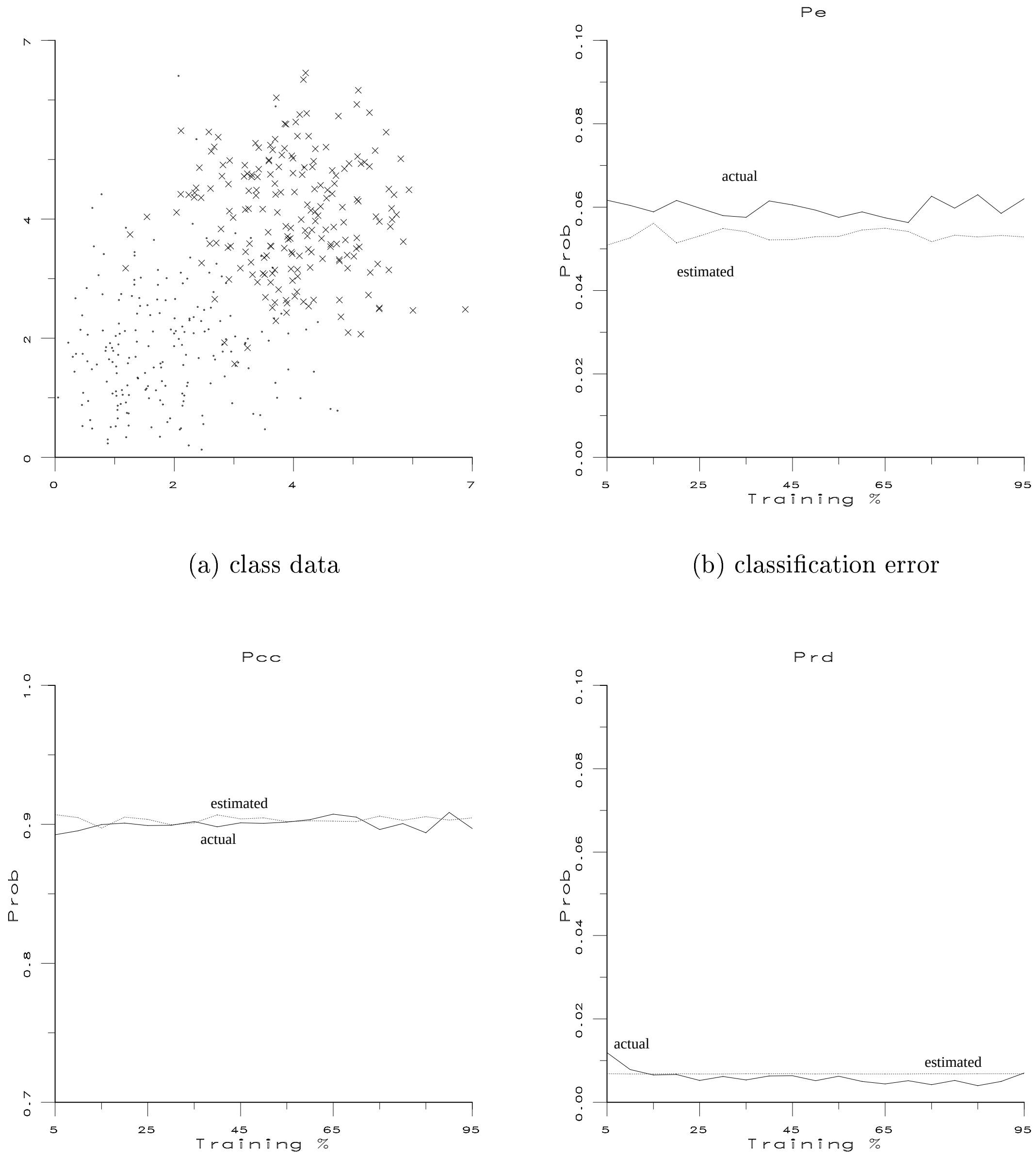
Figure 3.23: Estimated vs. actual probabilities for two normally distributed classes using varying size training data, $C_d = 0.01$ and $C_a = 0.4$.

of classification error, correct classification, and distance reject are shown in Figures 3.23(b) - 3.23(d), respectively. In each of these graphs, a large divergence between the estimated and actual probabilities is present for up to 25% (slightly higher for the probability of classification error) of the points used in class parameter estimation. As well, the probabilities remain fairly constant when over 40% of the points are used for class parameter estimation. Variations are attributed to random sampling and the fact that when the size of the class parameter estimation set is large (relative to N), the number of samples for probability estimation is small. Thus, the variability in the probability estimates is higher.

The results suggest that for (2D) normally distributed data, approximately 25 - 50 points are required for the accurate approximation of the class probabilities using the Inck decision rule. The number of data points mainly influences the estimation of the class parameters from which the probabilities are based. The results also suggest that the probability estimates given in Equations (3.13) - (3.17) are suitable approximations of the class probabilities.

The second dataset consists of one rayleigh distributed class with unit variance and one normally distributed class with mean (4,4) and unit variance. Each class contains $N = 1000$ points. Figure 3.24(a) depicts these class distributions. The training data was split into two sets in the same manner as for the first dataset. The same procedure was also used to obtain the averaged class probabilities.

The probabilities of classification error, correct classification, and distance reject are shown in Figures 3.24(b) - 3.24(d), respectively. In each of these graphs the probabilities remain fairly constant over the entire range of training set sizes used for parameter estimation. The (assumed normal) estimation of the class parameters for the rayleigh distributed class do not seem to be affected when the number of points used for the estimation is 5% (50) or greater. These results support those obtained for the first dataset.



(a) class data

(b) classification error

(c) correct classification

(d) distance reject

Figure 3.24: Estimated vs. actual probabilities for a rayleigh and normally distributed class using varying size training data, $C_d = 0.01$ and $C_a = 0.4$.

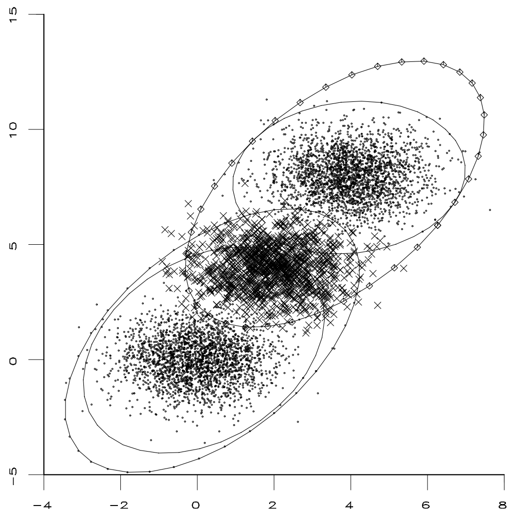


Figure 3.25: 2 class distributions (extreme clusters) with outlier (middle cluster) source (middle cluster) and associated 99 %confidence interval ellipses for MVE and classical parameter estimation, $C_d = 0.01$.

Results Using The MVE Estimator

These experiments were performed to determine the classification results when the MVE estimator is used in conjunction with the Inck decision rule. Two datasets were analysed. The first dataset consists of two normally distributed, bivariate classes. Each class contains 2500 points with an independent, unit variance in each dimension. The first class has a mean of (0,0) while the second class has a mean of (4,8). To each class, 625 (20% of total class population) outlier points were added. The outlier points were randomly selected from a third, normally distributed, bivariate class with an independent, unit variance in each dimension and a mean of (2,4). Figure 3.25 shows this dataset along with the corresponding 99% confidence interval ellipses for the MVE and classical parameter estimates of each class. In the figure, the MVE ellipses have a tighter fit around the non-outlier data than the ellipses for the classical parameter estimates. This implies that the MVE parameter estimates represent fewer of the outliers than the classical parameter estimates.

The Inck decision rule was used to classify the two classes. The *a priori* probabil-

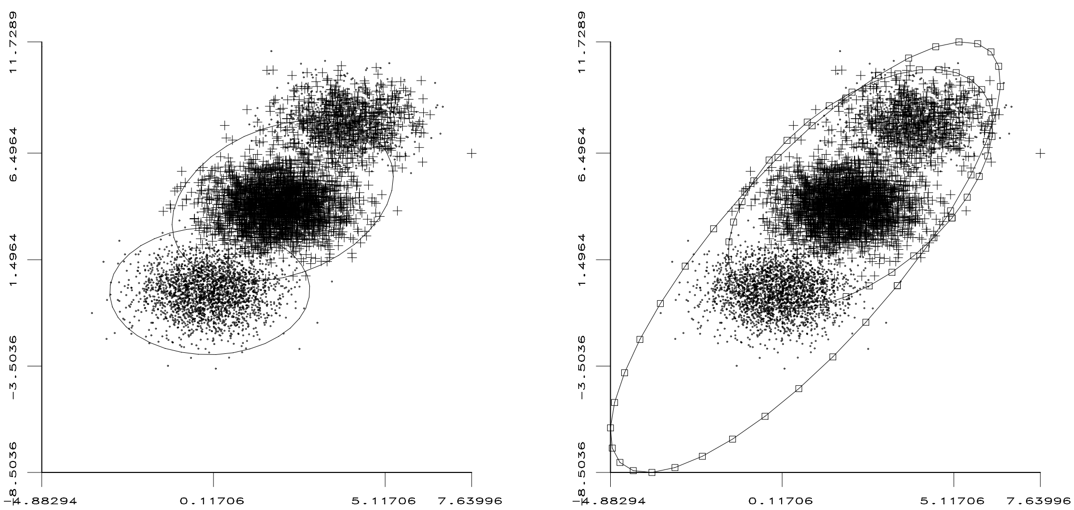
Table 3.4: Various probabilities using MVE and classical parameter estimates with $\epsilon = 1.0, 0.05, 0.02(\pm 0.01)$.

	$\epsilon = 1.0$		$\epsilon = 0.05$		$\epsilon = 0.02$	
	MVE	classical	MVE	classical	MVE	classical
C_a	0.5	0.5	0.5	0.188	0.5	0.125
P_C	0.813	0.894	0.810	0.822	0.810	0.836
P_E	0.026	0.101	0.023	0.052	0.023	0.020
P_{RD}	0.161	0.004	0.168	0.005	0.168	0.003
P_{RA}	0.000	0.000	0.000	0.121	0.000	0.141

ities of each class were assumed to be equal (0.5). The distance reject threshold was set to $C_d = 0.01$. The choice for C_a was determined using the algorithm presented in Section 3.3. A holdout method of training the classifier was used with 70% of the data used for parameter estimation and 30% of the data used for testing the Inck decision rule with the estimated parameters. Both the MVE and classical parameter estimates were used. The training process was repeated ten times for each estimator and the resulting probabilities averaged.

Table 3.4 shows the resulting probabilities and values of C_a for $\epsilon = 1.0$, 0.05, and 0.02 using both the MVE and classical parameter estimates. When $\epsilon = 1.0$, $C_a = 0.5$ and there is no ambiguity reject region ($P_{RA} = 0.0$). The remaining probabilities are different between the MVE and classical parameter estimators. For the probabilities resulting from the MVE parameter estimates, the different values of ϵ have no real effect. However, the probabilities resulting from the classical parameter estimates are affected as ϵ is decreased. Both estimators have approximately the same probability of correct classification (P_C) and same probability of classification error (P_E) when $\epsilon = 0.02$. There is a significant difference in the reject probabilities, however. $P_{RD} = 0.168$ and $P_{RA} = 0.000$ for the MVE estimator while $P_{RD} = 0.003$ and $P_{RA} = 0.141$ for the classical estimator.

From these results, a tradeoff between the MVE and classical estimators is identified. The use of the MVE estimator results in an increase in P_{RD} and a decrease in P_{RA} while the classical estimator results in a decrease in P_{RD} and an increase



(a) MVE parameter estimation

(b) classical parameter estimation

Figure 3.26: 2 class distributions each containing 20% outliers and associated 99 %confidence interval ellipses, $C_d = 0.01$.

in P_{RA}. For this dataset, approximately the same correct classification and classification error probabilities are obtained for either estimator. Another tradeoff is the additional time required in estimating the MVE parameters (see Section 3.6.1).

The second dataset consists of two normally distributed, bivariate classes. Each class contains 2500 points with an independent, unit variance in each dimension. The first class also has a mean of (0,0) while the second class has a mean of (2,4). To each class, 625 (20% of total class population) outlier points were added. The outlier points were randomly selected from a third, normally distributed, bivariate class with an independent, unit variance in each dimension and mean (4,8). Figure 3.26 shows this dataset along with the corresponding 99% confidence interval ellipses for the MVE and classical parameter estimates of each class. It can be seen from Figure 3.26(a) that the MVE ellipses have a tighter fit around the non-outlier data than the ellipses for the classical parameter estimates shown in Figure 3.26(b). In fact, for the first class, the ellipse for the classical estimator is quite large. It is

Table 3.5: Various probabilities using MVE and classical parameter estimates with $\epsilon = 1.0, 0.02, 0.01(\pm 0.01)$.

	$\epsilon = 1.0$		$\epsilon = 0.02$		$\epsilon = 0.01$	
	MVE	classical	MVE	classical	MVE	classical
C_a	0.5	0.5	0.5	0.234	0.250	0.250
P_C	0.830	0.863	0.785	0.429	0.776	0.473
P_E	0.032	0.132	0.021	0.024	0.016	0.019
P_{RD}	0.138	0.006	0.194	0.005	0.194	0.006
P_{RA}	0.000	0.000	0.000	0.543	0.015	0.502

obvious from these graphs that the MVE parameter estimates represent fewer of the outliers than the classical parameter estimates.

Table 3.5 shows the resulting probabilities and values of C_a for $\epsilon = 1.0$, 0.02, and 0.01 using both the MVE and classical parameter estimates. When $\epsilon = 1.0$, $C_a = 0.5$ and there is no ambiguity reject region ($P_{RA} = 0.0$). The correct classification probabilities (P_C) are similar for both estimators but the classification error (P_E) and distance reject (P_{RD}) probabilities are almost opposite. When ϵ is small ($\epsilon \leq 0.02$), there is a dramatic effect on the probabilities. For the classical estimator, P_C drops from 0.863 to 0.429, P_{RA} increases from 0.0 to 0.504 and P_E decreases from 0.132 to 0.024. The probabilities resulting from the use of the MVE estimator are affected substantially less. The biggest result to note is that when P_E is approximately the same for either estimator, P_C is much larger for the MVE estimator than for the classical estimator. This is mainly attributed to the fact that P_{RA} is much larger for the classical estimator, which results in a smaller P_C. For this dataset, it seems that there is an advantage in using MVE estimator, despite the additional time required for parameter estimation.

The results from both datasets suggest that the decision to use MVE parameter estimation should be weighed against the expected presence of outliers. If the data is considerably corrupted with outliers, then the added (time) expense of the MVE estimator will be justified with an increase in correct classification.

Table 3.6: Feature selection algorithms evaluated in experiments.

Branch and Bound (BB)
Best features (BF)
Sequential Forward Selection (SFS)
Sequential Backward Selection (SBS)

3.6.3 Feature Selection Methods using Ultrasound Volume Data

There are a number of objectives in performing the feature selection experiments:

- determine a feature selection algorithm which selects a reduced feature space that accurately represents the ultrasound data,
- determine if the Inck criterion function is appropriate for use in a feature selection algorithm,
- determine if the robust estimation of parameters using the MVE estimator outweighs its computational complexity when used in a feature selection algorithm,
- compare the performance achieved with the Inck criterion function to other functions, when used by a feature selection algorithm.

In order to satisfy these goals, a number of experiments are performed using real ultrasound data with various combinations of feature selection algorithms, criterion functions and parameter estimation methods.

Table 3.6 lists the feature selection algorithms evaluated in the experiments. For each algorithm, a number of criterion functions are evaluated as well as the Inck criterion function. These functions are listed in Table 3.7. Appendix B provides definitions for the probabilistic distance measures. Both probabilistic distance measures and interclass distance measures are used. Evaluation of the various feature selection, criterion function combinations is accomplished by comparing the classification accuracy achieved for a given set of training data. The classification accuracy

Table 3.7: Class distance measures evaluated in experiments.

Probabilistic Distance	Interclass Distance
Chernoff	Minkowski
Bhattacharyya distance	City block
Matusita distance	Euclidean distance
Divergence	Chebychev
Mahalanobis distance	Nonlinear
Patrick-Fisher	
Inck	

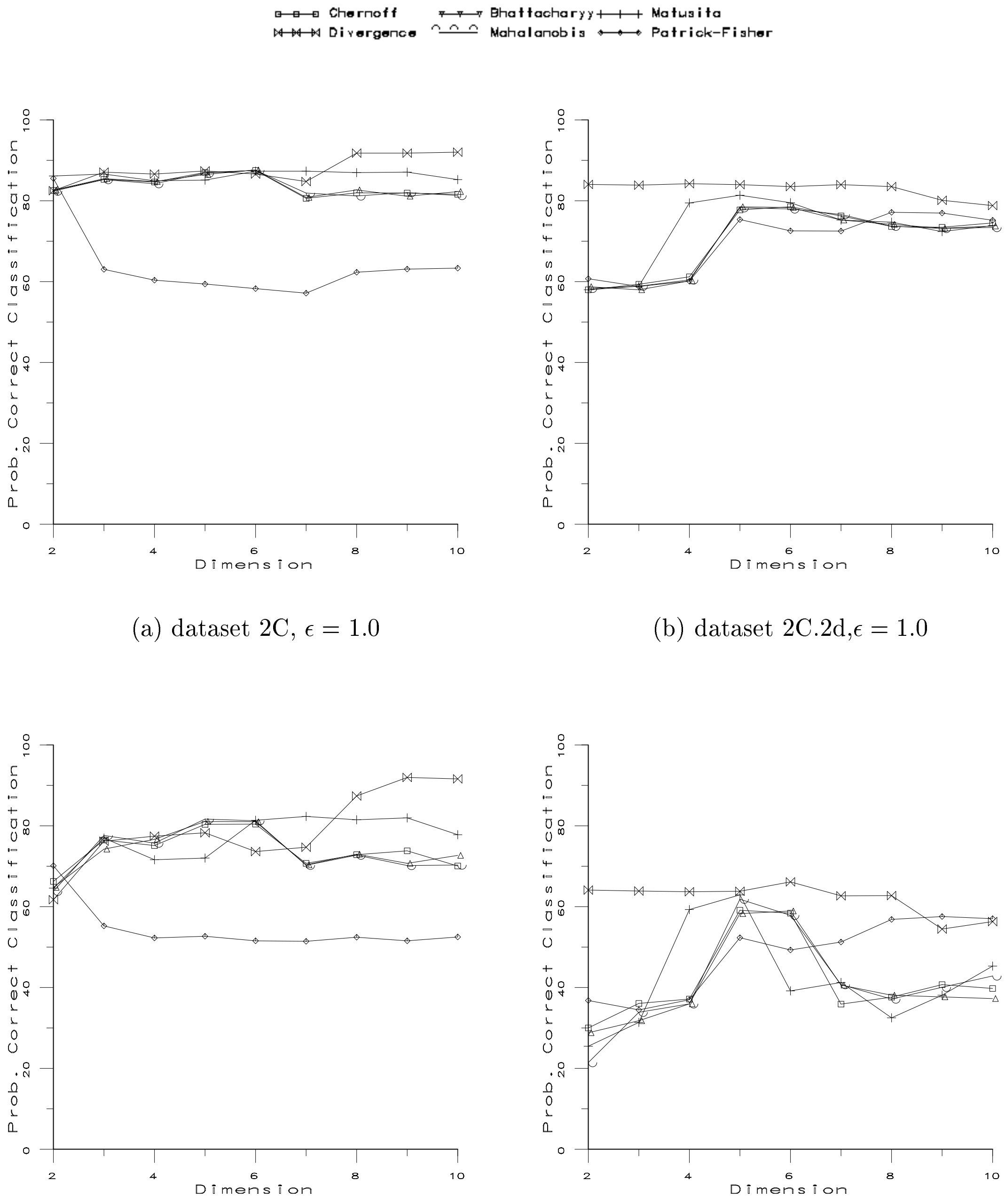
is determined using the same method as described in Section 3.6.2. The training data is split into two sets: one used for estimating the class parameters, the other for estimating the class probabilities. The classification accuracy is determined from both the probability of correct classification and the probability of classification error using the Inck decision rule. The classification accuracy is analysed as a function of the size of the feature space (number of selected features). That is, for a dataset consisting of D features, the feature space is evaluated with d, $1 \leq d < D$, features for each feature selection, criterion function combination. The 2C, 2C.2d, and 2C.cooc datasets created from the bovine ultrasound data (see Section 3.6) were evaluated in the experiments.

Results Using Distance Based Criteria

The large dimension of the the 2C and 2C.2d datasets placed a significant demand on the computing resources. As a result, the SBS and BB feature selection algorithms could not produce results in a reasonable time frame to be evaluated in conjunction with the BF and SFS algorithms. The results presented in this section are based on the analysis of the BF and SFS algorithms. It should also be noted that the Interclass distance measures are also computationally expensive. Since the complexity of these measures is dependent on the number of class samples, N, as N increases, the time required to compute the measures increases rapidly. For the given sample size in these experiments, $N = 1100$, the Interclass distance measures could not be computed in a reasonable time frame to be evaluated along with the probabilistic distance measures.

Figure 3.27(a) shows the probability of correct classification achieved by the BF feature selection algorithm with the various probabilistic criterion functions using both the 2C and 2C.2d datasets. Figures 3.27(a) and 3.27(b) show the probability of correct classification when $\epsilon = 1.0$ while 3.27(c) and 3.27(d) show the probability of correct classification when $\epsilon = 0.05$. In all the graphs, the probabilities are plotted as a function of the dimension, d, of the feature space. For $\epsilon = 1.0$, the various functions attain approximately the same probability (80%) at $d = 10$ for both the 2C and 2C.2d datasets. The exception is the Divergence function. It has consistently the highest probability of correct classification (over 90%) of all the functions, independent of d. For dataset 2C, this probability is a maximum (96%) when $d = 10$. The Patrick-Fisher criterion function has the lowest resulting probabilities. When $\epsilon = 0.05$, there is a significant difference between the probabilities attained for the 2C and 2C.2d datasets. For the 2C.2d dataset, the probabilities (in general) are minimum at $d = 2$, attain a maximum at $d = 5$, and then decrease as d increases. The probabilities are also lower for $\epsilon = 0.05$ as compared to the probabilities when $\epsilon = 1.0$. This decrease in the probability of correct classification suggests that there is a significant amount of ambiguity in classifying the 2C.2d data using resulting feature spaces (recall from Section 3.6.2 that as ϵ decreases, the probability of ambiguity reject increases while the probability of correct classification decreases). For the 2C dataset, the probabilities are relatively less affected when $\epsilon = 0.05$. The probabilities for all the criterion functions decrease, however, the Divergence function again attains its maximum probability (96%) when $d = 10$.

Figure 3.28(a) shows the probability of correct classification achieved by the SFS feature selection algorithm with the various probabilistic criterion functions using both the 2C and 2C.2d datasets. Figures 3.28(a) and 3.28(b) show the probability of correct classification when $\epsilon = 1.0$ while 3.28(c) and 3.28(d) show the probability of correct classification when $\epsilon = 0.05$. In all the graphs, the probabilities are plotted as a function of the dimension, d, of the feature space. For $\epsilon = 1.0$, the various



(a) dataset 2C, $\epsilon = 1.0$

(b) dataset 2C.2d,$\epsilon = 1.0$

(c) dataset 2C, $\epsilon = 0.05$

(d) dataset 2C.2d, $\epsilon = 0.05$

Figure 3.27: Probability of correct classification estimated for indicated ϵ using the BF feature selection algorithm and given criterion functions.

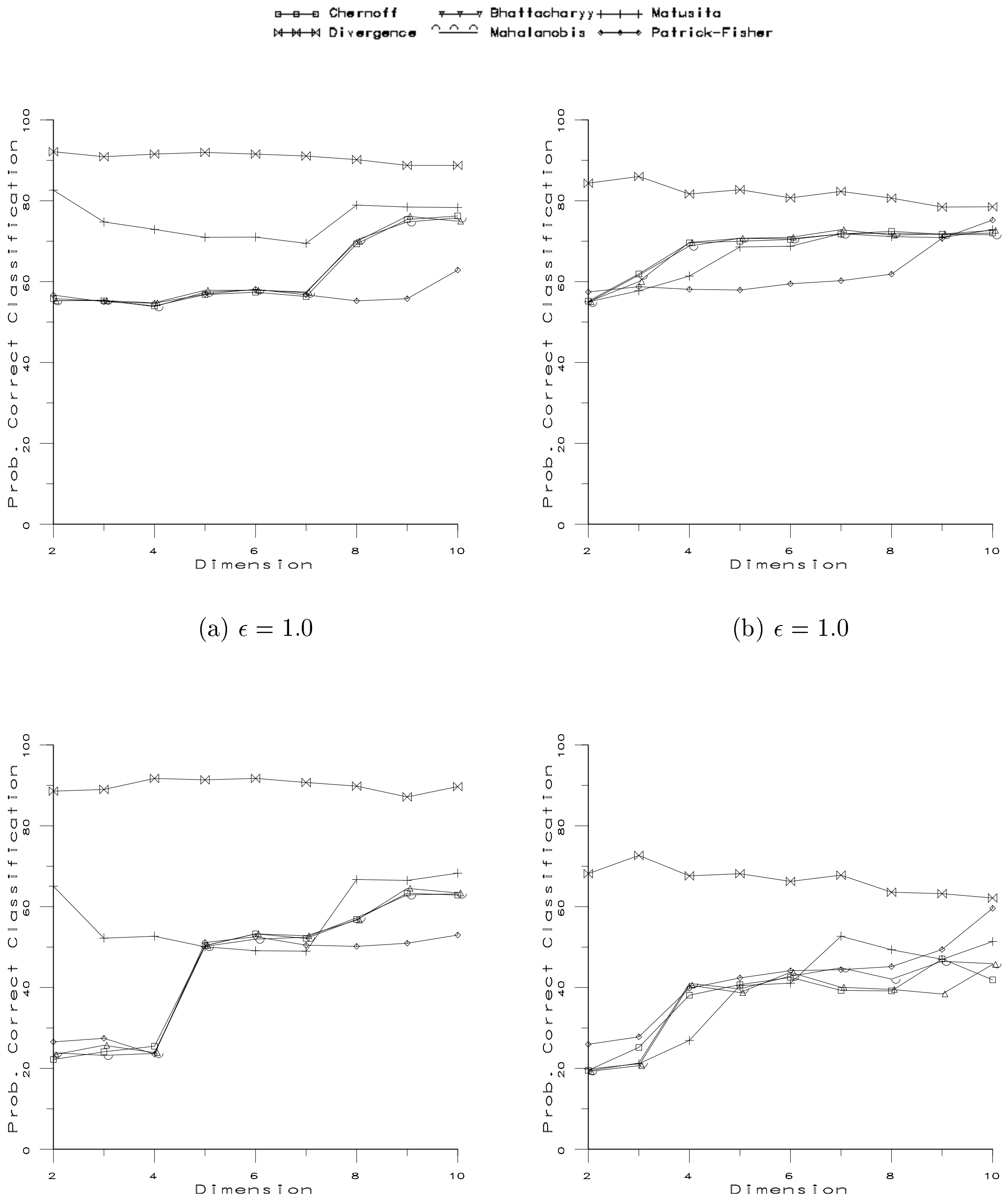
functions attain approximately the same probability (80%) at $d = 10$, for the 2C dataset, but not for the 2C.2d dataset. As with the BF algorithm, the Divergence function has consistently the highest probability of correct classification of all the functions, independent of d. Again, the Patrick-Fisher criterion function has the lowest resulting probabilities. When $\epsilon = 0.05$, there is a drastic decrease (20% - 40%) in all the probabilities achieved by the functions, except for the Divergence criterion function based on the 2C dataset. This decrease in the probability of correct classification suggests that there is a significant amount of ambiguity in classifying the data using the resulting feature spaces determined from the SFS algorithm. All subsequent experiments evaluate the probability of correct classification with $\epsilon = 0.05$.

The next section describes the performance of the Inck criterion function when used in conjunction with the BF and SFS feature selection algorithms.

Results Using the Inck Criterion Function

In order to objectively compare the performance of the Inck criterion function with the performance of the probabilistic distance functions, both the BF and SFS algorithms were used to select feature spaces up to $d = 10$ dimensions from the 2C and 2C.2d datasets.

Figure 3.29 shows the probability of correct classification using the Inck criterion function with the BF and SFS algorithms. Figure 3.29(a) shows the probabilities for the 2C and 2C.2d datasets using the BF algorithm. Figure 3.29(b) shows the probabilities for the 2C and 2C.2d datasets using the SFS algorithm. It can be seen from these graphs that the correct classification probabilities obtained from the 2C dataset are higher than those obtained from the 2C.2d dataset, independent of the feature selection algorithm used. Also, the feature spaces determined from the SFS algorithm result in a higher probability of correct classification than for the BF algorithm. Finally, the probabilities obtained using the Inck criterion function are in the same range as those achieved for the Divergence criterion function above, suggesting that the Inck criterion function is appropriate for use in a feature selection



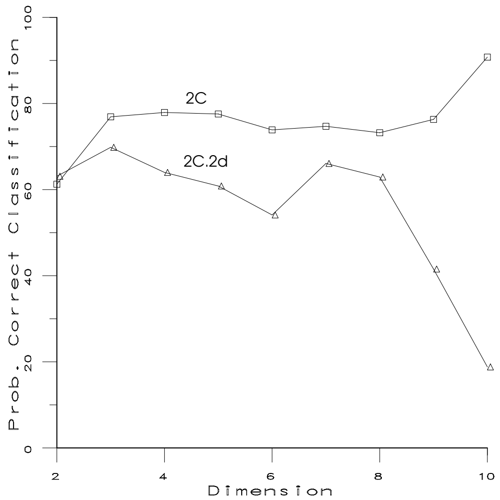
(a) $\epsilon = 1.0$

(b) $\epsilon = 1.0$

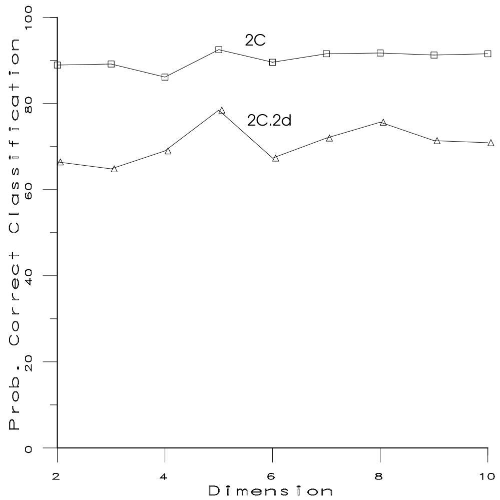
(c) $\epsilon = 0.05$

(d) $\epsilon = 0.05$

Figure 3.28: Probability of correct classification estimated for indicated ϵ using the SFS feature selection algorithm and given criterion functions.



(a) BF



(b) SFS

Figure 3.29: Probability of correct classification for given feature selection algorithm and dataset using the Inck criterion function.

Table 3.8: Resulting feature space using dataset 2C and the Inck criterion function.

	Dimension									
Algorithm	1	2	3	4	5	6	7	8	9	10
BF	31	79	55	7	51	27	75	48	3	24
SFS	31	87	79	47	95	83	23	33	43	62

algorithm.

Table 3.8 shows the features comprising the feature spaces (up to $d = 10$ dimensions) for the 2C dataset. Refer to Table 3.2 for an index of the features. The majority of selected features are the local homogeneity, LH, cooccurrence features (with various parameter values). In fact, the feature spaces determined with the SFS algorithm consist exclusively of the LH feature for $d <= 7$. Note that the statistical feature measures are not selected by either algorithm.

Table 3.9 shows the features comprising the feature spaces (up to $d = 10$ dimensions) for the 2C.2d dataset. Refer to Table 3.1 for an index of the features. Again,

Table 3.9: Resulting feature space using dataset 2C.2d and the Inck criterion function.

	Dimension									
Algorithm	1	2	3	4	5	6	7	8	9	10
BF	95	63	31	91	27	73	46	78	75	59
SFS	95	75	51	2	26	59	63	19	79	11

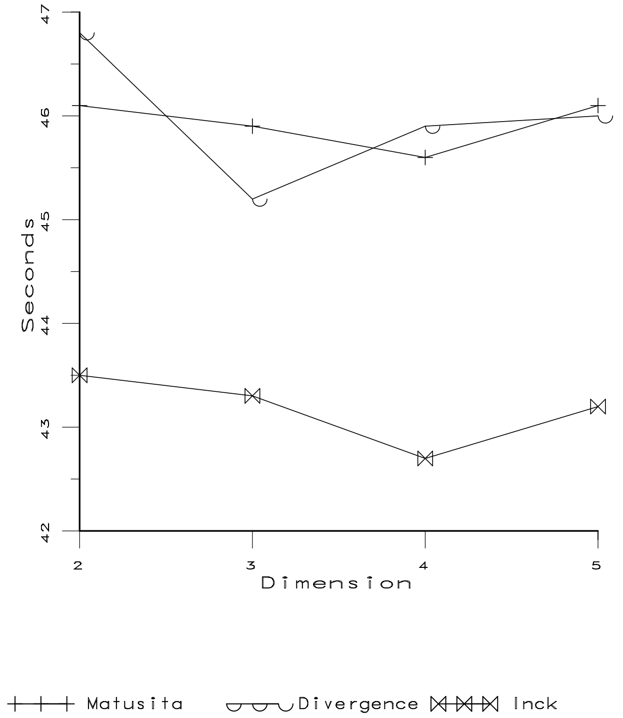
note that the statistical feature measures are not selected by either algorithm.

Results Using MVE Parameter Estimation

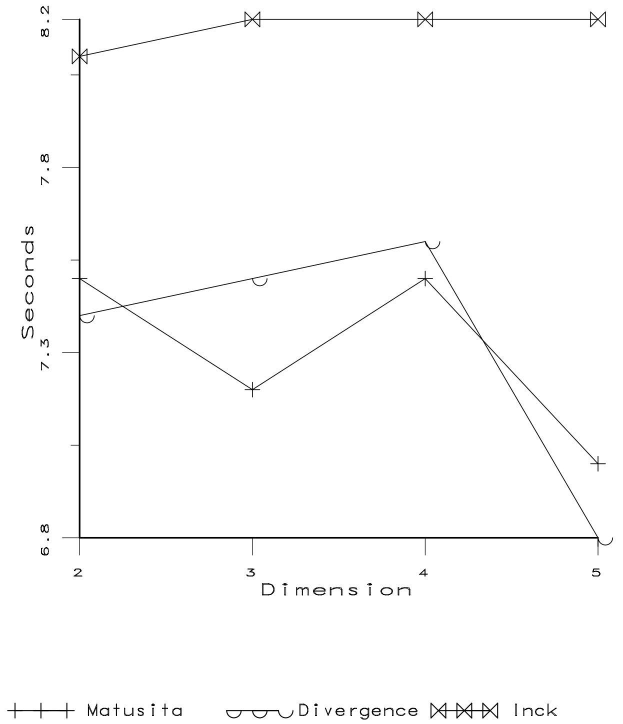
The purpose of this set of experiments is to determine if there is any advantage in using MVE parameter estimation to estimate the class parameters of the ultrasound data. Since the MVE estimator is computationally intensive, a subset of the features were selected from dataset 2C to construct the 2C.cooc dataset. The 2C.cooc dataset consists of 24 cooccurrence features (the 12 LH and 12 ENT features) which were most commonly selected to the feature spaces in previous experiments. The analysis of the MVE estimator was further constrained by limiting the size of the feature space to $d = 5$.

Figure 3.30 shows the CPU seconds required to determine the feature space using the BF algorithm with various criterion functions. Figure 3.30(a) shows the time required using the MVE estimator while Figure 3.30(b) shows the time required using classical parameter estimation. Since the BF algorithm only needs to calculate the class parameters in one dimension when creating the feature space, the times are relatively constant, independent of the dimension. The time required to select the feature spaces using the MVE estimator was approximately five times greater than that required using classical parameter estimation.

Figure 3.31 shows the CPU seconds required to determine the feature space using the SFS algorithm with various criterion functions. Figure 3.31(a) shows the time required using the MVE estimator while Figure 3.31(b) shows the time required using classical parameter estimation. There is a significant difference between the



(a) MVE



(b) Classical

Figure 3.30: Time for determining feature space for given criterion functions using the BF algorithm with MVE and classical parameter estimation.

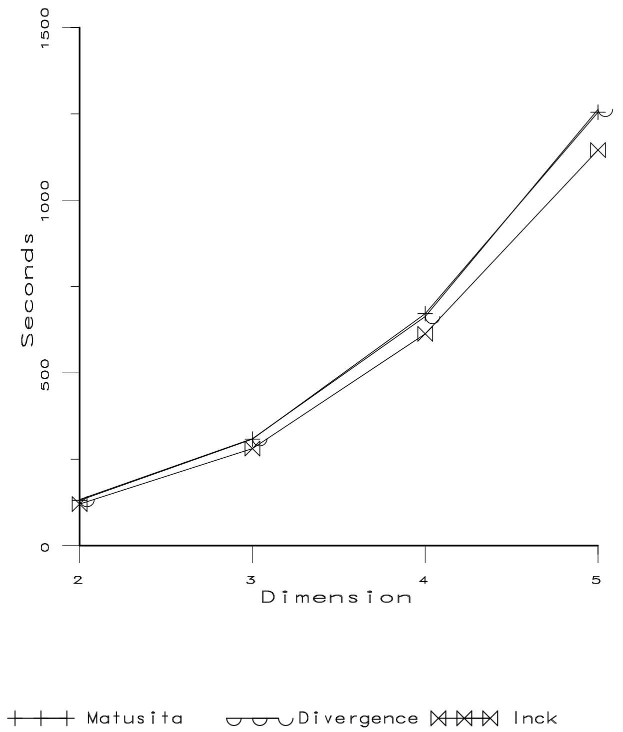
amount of time required to select the feature spaces using the MVE and classical parameter estimates. As the dimension of the feature space increases, the dimension of the class parameters also increase. The time required to compute the higher dimension parameters using the MVE estimator also increases significantly. In fact, for $d = 5$ the MVE estimates were not obtained after the amount of time shown. In comparison, the time required to select the feature spaces using classical parameter estimation increase linearly with the dimension and is at least an order of magnitude faster than using MVE parameter estimation.

Figure 3.32 shows the probability of correct classification achieved using MVE parameter estimation. Figure 3.32(a) shows the probability achieved with the BF feature selection algorithm using various criterion functions. Figure 3.32(b) shows the probability achieved with the SFS feature selection algorithm using the same functions. For the BF algorithm, the Inck criterion function performs poorly because the criterion is affected by the order in which features are added. This is in contrast with results obtained using classical parameter estimation. For the SFS algorithm, all the functions achieve a maximum probability of classification when $d = 3$. This, again, is in contrast with the results obtained using classical parameter estimation.

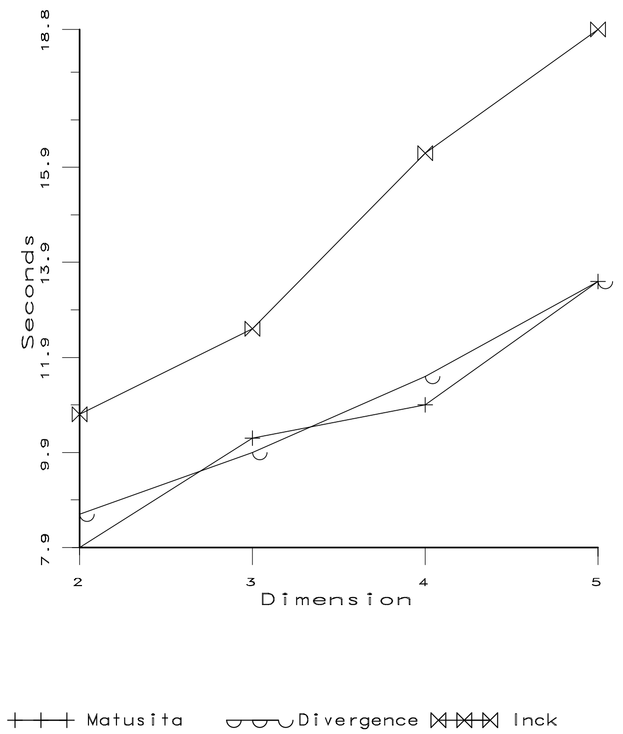
Based on these results it was decided that there was no benefit in using MVE parameter estimation in classifying the bovine ultrasound data. The reason for this is the absence of outliers in the ultrasound data so the MVE parameter estimates are quite similar to the classical parameter estimates. The added computational expense did not increase the performance of the feature selection algorithm used.

3.6.4 Multiresolution Classifier using Ultrasound Volume Data

The purpose of this set of experiments is to determine a multiresolution representation of the bovine ultrasound data. Four resolutions are evaluated: $\beta_2 = 4$, $\beta_3 = 8$, $\beta_4 = 16$ and $\beta_5 = 32$. In previous experiments it was found that the feature spaces based on the 3D features resulted in consistently higher correct classification prob-

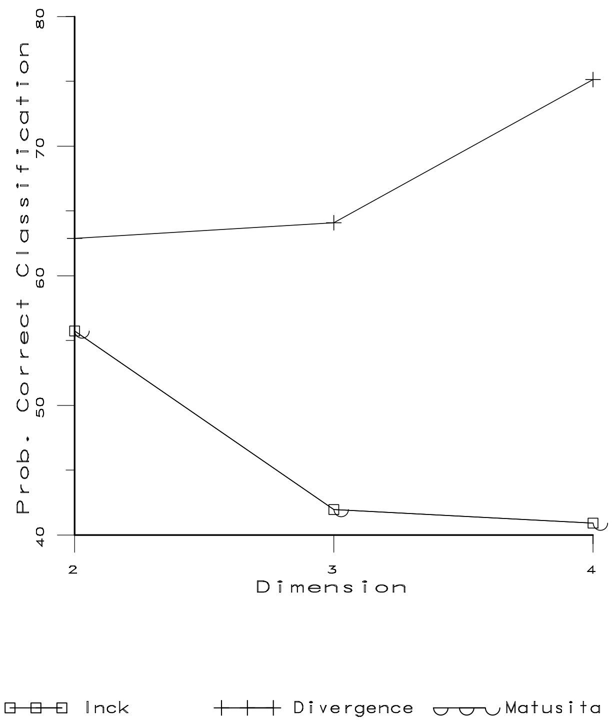


(a) MVE

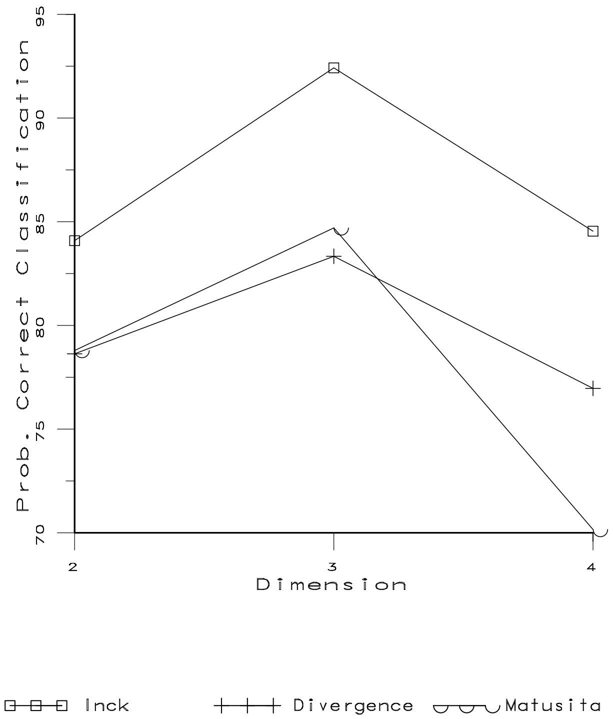


(b) Classical

Figure 3.31: Time for determining feature space for given criterion functions using the SFS algorithm with MVE and classical parameter estimation. Convergence of a solution for dimension 5 was not obtained using the MVE estimator.



(a) BF



(b) SFS

Figure 3.32: Probability of correct classification for given feature selection algorithms and criterion functions using MVE parameter estimation.

abilities as compared with those achieved for 2D features. Thus, the 3D based datasets 3C.x are used (see Section 3.6). Previous experiments also revealed that the feature selection algorithms using the Inck criterion function produced feature spaces with a high probability of correct classification. The Inck criterion function, along with the Divergence criterion function are used in conjunction with the SFS feature selection algorithm.

Results

Figure 3.33 shows the CPU seconds required to obtain the feature spaces using the Inck and Divergence criteria. Since the number of samples and classes for each of the datasets is the same, the time required to compute the feature space is the same for each β_i. The difference in times between the Inck and Divergence criteria is due to the way in which the Inck criterion is calculated. The criterion is estimated by randomly choosing a percentage of the data points for training and then classifying the remaining points to estimate the various probabilities required for computation of the criterion. In the experiments, this process was repeated 5 times and the results averaged. Thus, every time a feature is evaluated with the Inck criterion, 5 times the amount of computation is performed, as compared with the Divergence criterion.

Figure 3.34 shows the resulting probability of correct classification for the given resolutions. Figure 3.34(a) shows the probabilities for the Inck criterion function and Figure 3.34(b) shows the probabilities for the Divergence criterion function. It can be seen that as the β_i decreases, the probability of correct classification also decreases. This is due to the fact that as the resolution decreases, there are fewer points from which the feature measures are calculated from. As well, the within class variability increases resulting in higher probabilities of classification error, ambiguity and distance rejection. The graphs also show that the feature spaces based on the Inck criterion function result in higher probabilities of correct classification as compared with those obtained for the Divergence criterion. The dip in the curve for β_8 in Figure 3.34(a) is attributed to random sampling error.

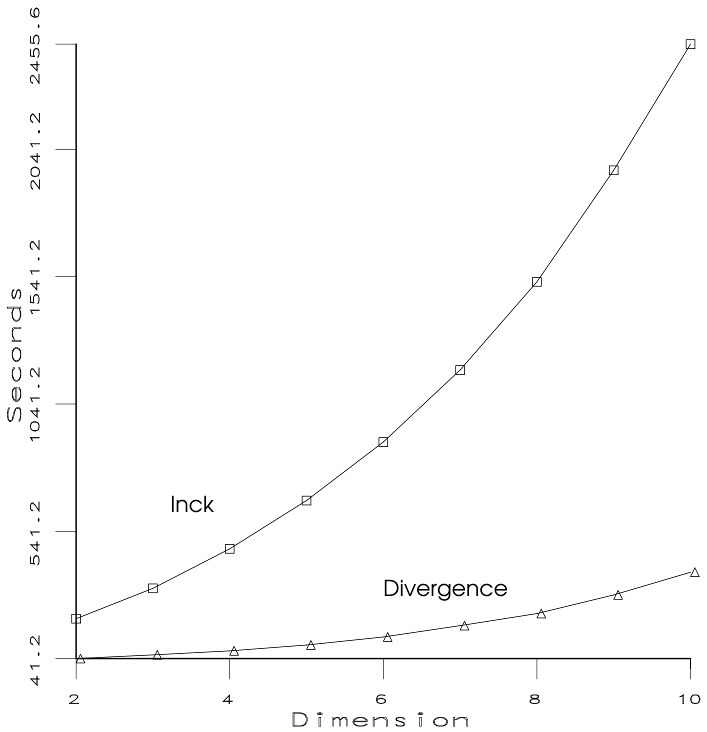
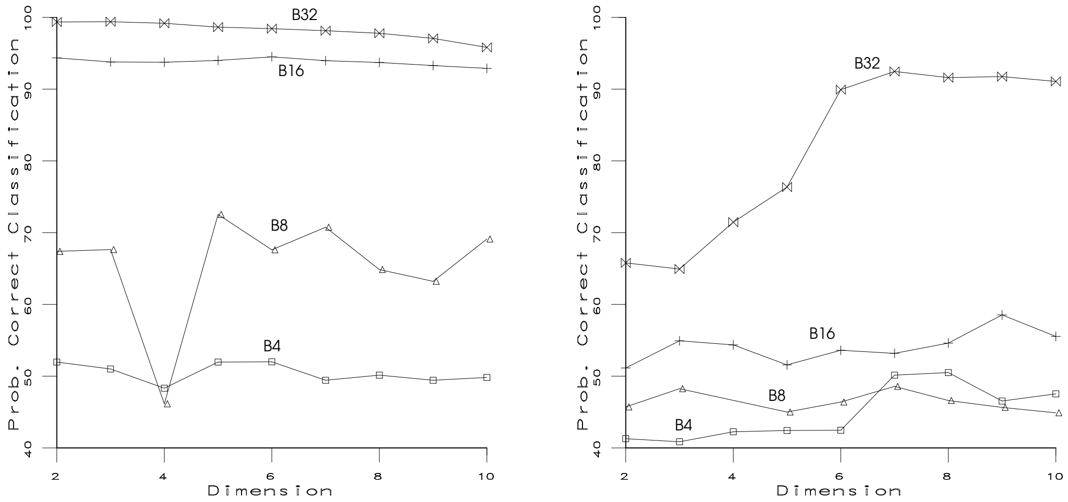


Figure 3.33: Time Required to obtain feature space of given dimension using the Inck and Divergence criteria.



(a) Inck (b) Divergence

Figure 3.34: Probability of correct classification of feature space at a given resolution β_i using SFS feature selection algorithm and indicated criterion function.

Table 3.10: Resulting feature space for given resolution β_i using SFS feature selection algorithm and the Inck criterion function.

	Dimension									
β_i	1	2	3	4	5	6	7	8	9	10
4	31	67	15	8	51	87	3	47	55	14
8	31	95	66	7	0	45	16	79	26	67
16	79	95	31	47	64	39	33	7	19	86
32	79	47	59	51	94	95	87	46	41	22

Table 3.10 shows the feature spaces for resolutions β_2, β_3, β_4 and β_5 using the Inck criterion function. Table 3.11 shows the feature spaces for resolutions β_2, β_3, β_4 and β_5 using the Divergence criterion function.

The above results suggest that a low dimension ($d = 2$) feature space can be used to classify the bovine ultrasound data with a high level of accuracy for resolutions $\beta_4 = 16$ and $\beta_5 = 32$. The advantage of this is that the 3D MTS algorithm, which uses this multiresolution information, does not have to perform a large number of

Table 3.11: Resulting feature space for given resolution β_i using SFS feature selection algorithm and the Divergence criterion function.

	Dimension									
β_i	1	2	3	4	5	6	7	8	9	10
4	93	29	84	77	85	45	90	42	56	20
8	61	84	60	76	85	5	77	36	8	72
16	61	84	60	13	85	44	76	77	11	26
32	69	58	53	84	5	66	65	68	64	71

calculations in segmenting the volume data. However, for resolutions $\beta_3 = 8$ and $\beta_2 = 4$, the probability of correct classification is lower (even if a higher dimension feature space is used). It can be expected that segmentation results based on this multiresolution classification will contain greater probabilities of classification error, distance and ambiguity reject when smaller details of the volume data are segmented.

3.7 Summary

The purpose of this chapter was to describe the details of the Tissue Characterisation module. It is based on a statistical pattern recognition approach which provides a solid foundation on which tissue characterisation can be accomplished. The approach used in the Tissue Characterisation module provides the ability to combine both classification and segmentation in a unified, multiresolution framework. This is significant as the topics are almost always treated independently in the literature.

A new criterion function, the Inck (**Inc**omplete **k**nowledge) criterion function, was presented which approximates the error probability when M classes do not span the entire pattern space. This criterion is based on the probabilistic measures obtained from a modified version of Dubuisson and Masson's [26] statistical decision rule with reject. The error probability, or conversely the probability of correct classification, can be determined without having complete knowledge about the class distributions. The attractiveness of this criterion is that it is highly correlated with the statistical decision rule (with reject) which can be used by the classifier.

Thus, the criterion provides a good indication of the classifier performance that can be expected with the reduced feature space.

It was also described how the robust statistical minimum volume ellipsoid estimator (MVE) proposed by Rousseeuw and Leroy [106] could be used to increase the classifier's tolerance to outlier data. The MVE estimator can theoretically provide an unbiased parameter estimation (in the case of a multivariate normal distribution, the mean and associated covariance matrix) describing a class distribution in the presence of up to 50% outlier data points.

A classifier design was presented which incorporates the new Inck criterion function. As well, it was described how the rejection classes of the modified statistical decision rule with reject could be used in a multiresolution classifier design to determine the class identity of patterns which can not be determined at a given resolution.

Numerous experiments were performed using both synthetic and real ultrasound volume data. Results obtained from these experiments were:

- The theoretical tolerance of up to 50% outliers using the MVE estimator could not be obtained in all experiments and was found to be dependent on the data dimension and the distribution of the data.

- The decision to use MVE parameter estimation should be weighed against the expected presence of outliers. If the data is considerably corrupted with outliers, then the added (time) expense of the MVE estimator will be justified with an increase in correct classification.

- Using a large dimensional pattern space ($D = 99$), the SBS and BB feature selection algorithms could not produce results in a reasonable time frame to be evaluated in conjunction with the BF and SFS algorithms.

- For the given sample size in the experiments, $N = 1100$, the Interclass distance measures could not be computed in a reasonable time frame to be evaluated along with the probabilistic distance measures and the Inck criterion function.

- The LH (local homogeneity) measures based on the 3D cooccurrence matrix (with various parameter settings) were consistently selected as the top features in discriminating among the corpus luteum (C1), stroma (C2) and fluid (C3) tissue classes in the bovine ultrasound volume data.

- The feature spaces based on the Inck criterion function resulted in higher probabilities of correct classification as compared with those obtained using probabilistic distance measures.

- For classification at a larger resolution , a low dimension ($d = 2$) feature space could be used to classify the bovine ultrasound volume data with a high level of accuracy.

- For classification at a smaller resolution, the probability of classification error, distance and ambiguity reject is high suggesting that segmentation of smaller details in the bovine ultrasound volume data, based on the classification data, will not be exceptional.

Chapter 4

3D Texture Features

This chapter describes the 3D texture features used in the Tissue Characterisation module of the system. Numerous feature measures have been proposed to represent texture [121, 115, 142, 143, 114, 71, 27, 129, 149, 92, 4, 70, 62, 45, 19, 86] in an image. A review of texture feature measures can be found in [77, 86, 19]. All of these features are based on the 2D grey level pattern present within a window or region of interest in the image. The processing of volume data by the Tissue Characterisation and 3D Segmentation modules of the proposed system necessitates the need for new texture feature measures which can provide measures of texture in 3D. Two classes of texture features are evaluated in this thesis: cooccurrence matrix based features and statistical features. Cooccurrence features are used because they have been found to consistently perform well in various texture analysis applications. Section 4.1 describes the cooccurrence matrix, its proposed definition in 3D, and the measures that are calculated from the matrix. Statistical features are used because of their simplicity and their common use in many classification applications. Section 4.2 describes the statistical feature measures.

4.1 3D Cooccurrence Matrix

One class of texture features which has been evaluated extensively are the cooccurrence matrix based features [39, 92, 37, 24, 86, 19]. Originally proposed by Haralick *et al.* [39], these features measure characteristics of the grey level (pixel) spatial dependencies. The cooccurrence matrix represents the frequencies at which pairs of pixels having grey levels i and j, $0 \leq i, j < G$ are separated by a distance d. The distance is usually quantised to discrete values of d and specific orientations θ ($\theta = 0°, 45°, 90°, 135°$). Thus, each matrix is parameterised by (G, d, θ), where G is the maximum gray level. Note that this parameterisation only allows frequencies to be gathered for points on a plane. In order to measure texture in a 3D volume the cooccurrence matrix must be redefined. The next two subsections define the 3D cooccurrence matrix as well as some feature measures based on the matrix.

4.1.1 Calculating the Matrix

Given an $M \times N$ image, I, with pixel intensities in the range $[0, G-1]$ the intensity of a pixel at location (m, n) is $f(m, n)$. Assuming the *pixel* distance d oriented at angle θ is represented by the vector $(\Delta x, \Delta y)$ the classical definition of a cooccurrence matrix, P, is given as follows:

$$P(i,j) = \frac{1}{2R} \sum_{m=1}^{M-\Delta x} \sum_{n=1}^{N-\Delta y} \delta(f(m,n) = i \textit{ and } f(m+\Delta x, n+\Delta y) = j) \tag{4.1}$$

where $R = (M - \Delta x)(N - \Delta y)$ and $0 \leq i, j < G$. P is a $G \times G$ matrix of probabilities which is symmetric about the diagonal. Typically, θ is quantised to $45°$ angles resulting in the following relationships between (d, θ) and $(\Delta x, \Delta y)$:

$$\begin{aligned}
(d, 0) &\rightarrow (d, 0), \\
(d, 45) &\rightarrow (d, d), \\
(d, 90) &\rightarrow (0, d), \\
(d, 135) &\rightarrow (-d, d).
\end{aligned}$$

Note that when $\Delta x < 0$ or $\Delta y < 0$ the range of the summations in Equation (4.1) must be shifted so that the image boundaries are not exceeded.

Each combination of (d, θ) results in a new cooccurrence matrix, $P'_{d\theta}$. Also, each matrix is orientation dependent which is not desirable in the intended application.

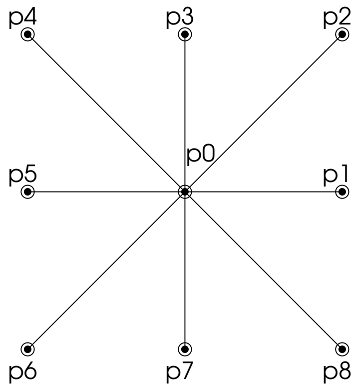


Figure 4.1: Neighbourhood of pixel p_0 for pixel distance d and $\theta = 45°$.

In this thesis the orientation dependency is removed by combining all matrices calculated for a given d into one matrix P_d

$$P_d = \sum_{\forall\theta} P'_{d\theta}. \tag{4.2}$$

P_d can be normalised by dividing its entries by the number of (d, θ) combinations over which the summation in Equation (4.2) occurs.

An alternative method of calculating P_d is to calculate it directly from the input image, I. Consider a point $p_0 = (m, n)$ in the image. Assuming a pixel distance d and θ quantised at 45° intervals, p_0 has 8 neighbouring pixels, $p1 \dots p8$, each d pixels away from p_0. Figure 4.1 illustrates the neighbourhood for p_0. Comparing p_0 to each of its neighbor provides 8 different pixel pairs from which P_d can be updated. Considering all such neighbourhoods in the image, P_d is calculated as follows:

$$P_d(i, j) = 1/R \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{k=1}^{8} \delta(f(p_0) = i \text{ and } f(p_k) = j \text{ and } p_k \in I), \tag{4.3}$$

where

$$R = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{k=1}^{8} \delta(p_k \in I).$$

The term $p_k \in I$ accounts for incomplete neighbourhoods at the borders of I by determining if p_k is within the boundaries of the image.

The definition of the cooccurrence matrix in Equation (4.3) provides the ability to consider many possible neighbourhood configurations. The dependence of the neighbourhood on θ is implicit in Equation (4.3). It is beneficial to explicitly state this dependence as follows:

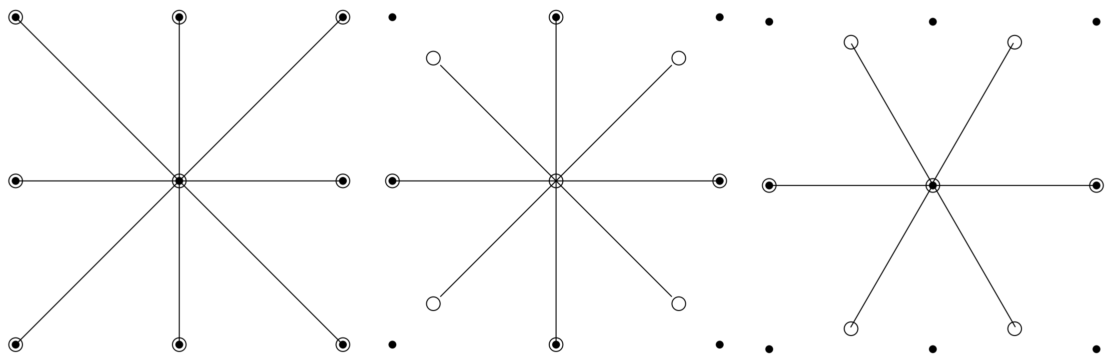
$$P_{d\theta}(i,j) = 1/R \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{k=1}^{360/\theta} \delta(f(p_0) = i \text{ and } f(p_k) = j \text{ and } p_k \in I) \tag{4.4}$$

where

$$R = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{k=1}^{360/\theta} \delta(p_k \in I)$$

and θ denotes the incremental angle between neighbours, rather than a specific orientation. Given the discrete locations of pixels in the image and the unit of distance measure, pixels, θ must be a multiple of $45°$, $45 \leq \theta \leq 360$. This limits the number of possible neighbourhood configurations. This limitation can be overcome by selecting a different unit of measure which is not based on discrete units. The distance between p_0 and any of its neighbours can be parameterised by the radial distance, r. With this new parameterisation of the cooccurrence matrix , $P_{r,\theta}$, many more neighbourhood configurations are possible. This provides the ability to collect more information about the frequency distributions in the image. When calculating the neighbourhood for a given (r, θ) it is possible that one or more locations may not fall at discrete grid intervals. The intensity of the points at these locations is (linearly) interpolated.

The added flexibility of the new parameterisation of the cooccurrence matrix comes at a cost. First, the location of neighbourhood points is performed using floating point operations rather than integer operations. Some savings can be accomplished by computing the pixel offsets from the centre of the neighbourhood for a given (r, θ). Second, interpolating pixel values incurs additional floating point operations. Interpolation also introduces data which is not in the original image. Finally, the symmetry that exists in a cooccurrence matrix using the pixel distance



(a) pixel distance, $\theta = 45°$ (b) radial distance, $\theta = 45°$ (c) radial distance, $\theta = 60°$

Figure 4.2: Neighbourhoods for p_0.

is not guaranteed to exist in $P_{r\theta}$. That is,

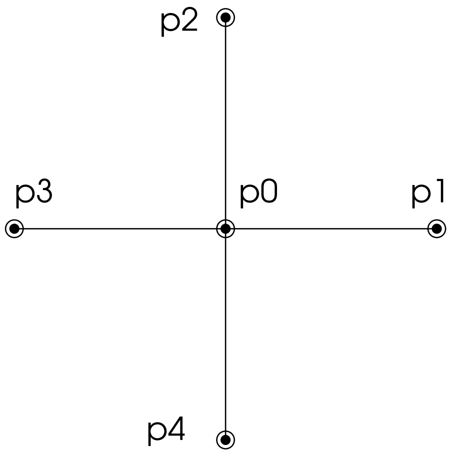
$$P_{r\theta}(i, j) \neq P_{r\theta}(j, i)$$

in general, since neighbourhood points (other than p_0) can exist anywhere within the boundaries of the image.

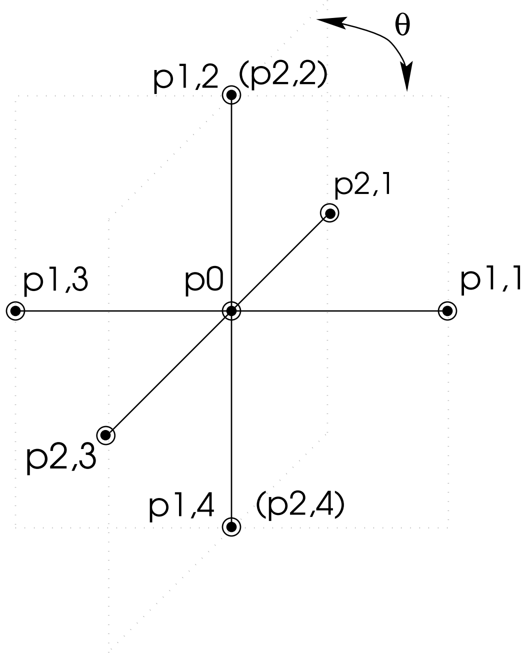
Figure 4.2 shows some different neighbourhoods using both the pixel distance and the radial distance. Note the difference between Figure 4.2(a) using the pixel distance and Figure 4.2(b) using the radial distance. The neighbourhood using the radial distance requires interpolation while the other does not.

Based on the formulation of the cooccurrence matrix in Equation (4.4), calculating the 3D cooccurrence matrix is straightforward. Given an $M \times N \times L$ volume, V, with pixel locations (m, n, l) the definition of the 3D cooccurrence matrix is as follows:

$$P_{d\theta}(i, j) = 1/R \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{l=0}^{L-1} \sum_{b=1}^{360/\theta} \sum_{k=1}^{360/\theta} \delta(f(p_0) = i \text{ and } f(p_k, b) = j \text{ and } p_k, b \in I), \tag{4.5}$$



(a) 2D



(b) 3D

Figure 4.3: Neighbourhoods for p_0 with $\theta = 90°$ and $d = 1$ (pixel distance).

where

$$R = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{l=0}^{L-1} \sum_{b=1}^{360/\theta} \sum_{k=1}^{360/\theta} \delta(p_k, b \in I),$$

and R is the normalising constant and p_k, b is the kth neighbourhood point on plane b. Figure 4.3 illustrates the geometry for this type of neighbourhood. In the 2D case, as shown in figure 4.3(a), all neighbourhood points lie on the XY plane, $b1$. In order to determine the remaining neighbourhood points, the plane is rotated by an angle θ about the axis intersecting point p_0 and parallel to the Y axis. The new plane, $b2$, will have the same number of neighbourhood points as $b1$ in the same orientation within the plane, but the locations of the points in 3D will be different. The exception is the points which lie on the axis of rotation which always remain at the same 3D location.

Figure 4.3(b) illustrates the neighbourhood of p_0 for $\theta = 90°$ and $d = 1$ (pixel distance). There are a total of 6 unique points in the neighbourhood ($p_{2,2}$ and $p_{2,4}$ are duplicate on-axis points). For $\theta = 45°$, the neighbourhood consists of 4 planes with 8 points on each. Accounting for the duplicate on-rotational axis points, the number of neighbourhood points is $8+3*(8-2) = 26$. As with the 2D case, Equation (4.5) can be modified to use the radial distance rather than the pixel distance

$$P_{r\theta}(i,j) = 1/R \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{l=0}^{L-1} \sum_{b=1}^{360/\theta} \sum_{k=1}^{360/\theta} \delta(f(p_0) = i \; and \; f(p_k, b) = j \; and \; p_k, b \in I), \tag{4.6}$$

where

$$R = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \sum_{l=0}^{L-1} \sum_{b=1}^{360/\theta} \sum_{k=1}^{360/\theta} \delta(p_k, b \in I)$$

and p_k, b is at a radial distance r from p_0.

4.1.2 Cooccurrence Features

Once the cooccurrence matrix has been calculated from 3D data, it remains to determine feature measures from the matrix. Since the matrix is of the same form as the classical cooccurrence matrix (except containing more information), it is possible to use the usual cooccurrence features proposed in the literature [39, 92, 37, 24].

Ohanian and Dubes [86] suggest that a small subset of Haralick's cooccurrence features is sufficient for characterising many types of textures. The suggested features are the angular second moment (ASM) which measures the homogeneity of an image, the contrast feature (CON) which measures the contrast in an image, the correlation feature (COR) which measures the linear dependency in an image, and the entropy feature (ENT) which measures the complexity of an image. The definition of these measures is given as follows:

$$ASM = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i,j)^2, \tag{4.7}$$

$$CON = \sum_{n=0}^{G-1} n^2 \sum_{|i-j|=n} P(i,j), \tag{4.8}$$

$$COR = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{ijP(i,j) - \mu_x \mu_y}{\sigma_x \sigma_y}, \tag{4.9}$$

$$ENT = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} -P(i,j) ln(P(i,j)), \tag{4.10}$$

where

$$\mu_x = \sum_{i=0}^{G-1} i \sum_{j=0}^{G-1} P(i,j), \quad \mu_y = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} jP(i,j),$$

$$\sigma_x = \sum_{i=0}^{G-1} (i - \mu_x)^2 \sum_{j=0}^{G-1} P(i,j), \quad \sigma_y = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (j - \mu_y)^2 P(i,j).$$

Peckinpaugh [92] suggests additional features which are commonly calculated from the cooccurrence matrix. These features are cluster shade (CS), cluster prominence (CP), Inertia (IN) and local homogeneity (LH). The definition of these measures is given as follows:

$$CS = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i + j - \mu_x - \mu_y)^3 P(i,j) \tag{4.11}$$

$$CP = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i + j - \mu_x - \mu_y)^4 P(i,j) \tag{4.12}$$

$$IN = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i - j)^2 P(i,j) \tag{4.13}$$

$$LH = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{1}{1 + (i - j)^2} P(i,j) \tag{4.14}$$

4.2 Statistical Features

Statistical features are simple measures of texture that provide an indication of the actual gray levels and their variation. Unlike the cooccurrence based features, statistical features do not consider the spatial relationship among the gray levels.

The usual statistical features are the mean and standard deviation. Since the spatial relationship of the grey levels is not an issue, calculating the mean and standard deviation in 3D is not more difficult than calculating them in 2D. The complexity does increase, however, as the number of operations required increases with the dimension.

The three statistical features used are the mean (MEAN) which measures the average pixel intensity in the image, the standard deviation (SDEV) and the average deviation (ADEV). The last two features measure the variation of the pixel intensities, $p_{i,j}$, with respect to the average pixel intensity. The ADEV feature is defined as

$$ADEV = \frac{1}{N^2} \sum_i \sum_j |p_{i,j} - MEAN|, \; 0 \leq i, j < N$$

which is basically an approximation of the usual standard deviation.

4.3 Experiments

It has been found in the experiments on ultrasound volume data in Chapter 3 that the measures based on the 3D cooccurrence matrix resulted in feature spaces with higher probability of correct classification than those based on the 2D cooccurrence matrix. Rather than reiterate these results, it is the purpose of these experiments to analyse the form of the cooccurrence matrix itself to understand the effect of the cooccurrence parameters. In doing so, it is possible to determine if the measurement of the neighbourhood frequencies using the radial distance has any advantage over the measurement using the pixel distance.

Both 2D and 3D regions from the bovine ultrasound data, as described in Section 3.6, were used in the experiments. The 2D regions were 16×16 pixels and the 3D regions were $16 \times 16 \times 16$ pixels. From these regions the cooccurrence matrices were constructed and the cooccurrence measures calculated. Timing information was obtained by calculating 2436 cooccurrence matrices (and the corresponding measurements) from randomly selected regions. The total CPU time was then divided

(a) $d = 1$

(b) $r = 1$

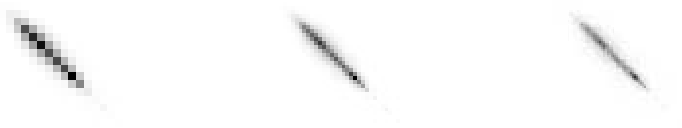
Figure 4.4: 3D Cooccurrence matrix obtained with G = 128, θ = 90 and given distance pixel distance d, and radial distance r.

by this number to give the average number of CPU seconds required to compute the matrix along with its measures. All matrices are normalised to the range [0,255] and size 128×128 for display.

4.3.1 Results

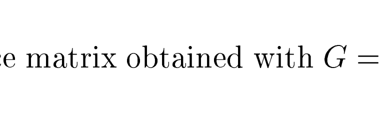
Figures 4.4 - 4.6 show the resulting 3D cooccurrence matrices for various combinations of the parameters G, θ, and r (d). All matrices were constructed from the same 3D region in a corpus luteum (C1) sample volume. Figures 4.4(a) and 4.4(b) show the matrices for $G = 128$, $\theta = 45$ with $d = 1$ and $r = 1$, respectively. Figures 4.5(a), 4.5(b) and 4.5(c) show the matrices for $\theta = 90$, $r = 1$, with $G = 32$, $G = 64$ and $G = 128$, respectively. Figures 4.6(a) and 4.6(b) show the matrices for $G = 128$, $r = 1$, with $\theta = 45$ and $\theta = 90$, respectively. From these figures it can be seen that there is not much difference among the matrices in their normalised form. The exception is Figure 4.5 in which G is varied. Since the matrices are normalised, the matrices calculated with $G = 32$ and $G = 64$ appear blocky. This indicates that as G increases, more detailed information is available about the frequency distribution of intensities. Note, however, that the computational expense also increases.

Figure 4.7 shows the average number of CPU seconds required to compute the 3D cooccurrence matrix and the 8 feature measures for the given combination of θ and



(a) $G = 32$ (b) $G = 64$ (c) $G = 128$

Figure 4.5: 3D Cooccurrence matrix obtained with $r = 1$, $\theta = 90$ and given values of G.



(a) $\theta = 45$ (b) $\theta = 90$

Figure 4.6: 3D Cooccurrence matrix obtained with $G = 128$, $r = 1$ and given values of θ.

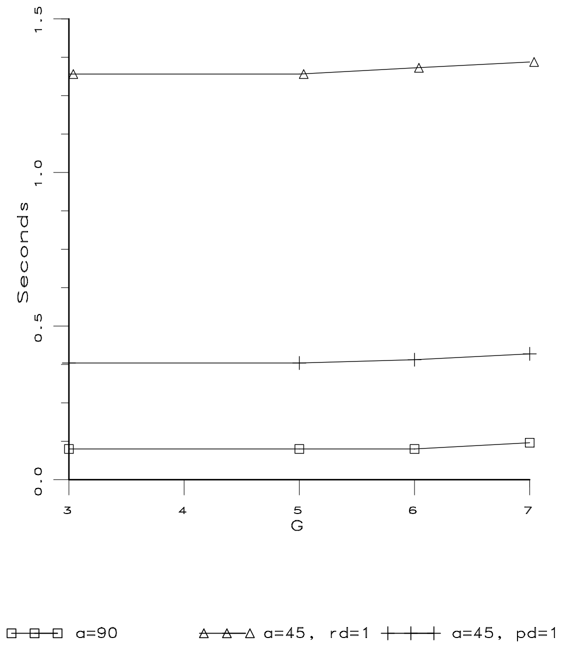


Figure 4.7: Time to compute cooccurrence matrix for given θ and type of distance measurement.

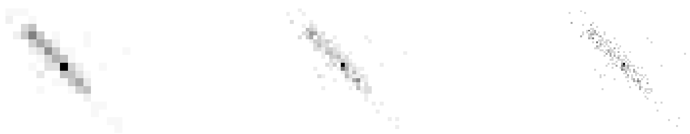
distance measurement. Although there was little difference among the matrices in the above figures, it is apparent from Figure 4.7 that the computational requirements to construct the matrices and their associated measurements are quite different.

The 2D cooccurrence matrices constructed from various combinations of the G, θ and d parameters are shown in Figures 4.8 and 4.9. It is apparent that there is much less information available in these matrices as compared to the 3D matrices.

Figures 4.10 and 4.11 show the resulting 2D and 3D matrices, respectively, obtained from the corpus luteum (C1), stroma (C2), and fluid (C3) volume samples. For the 3D cooccurrence matrices, there are obvious differences among the classes. The matrix in Figure 4.11(c) is significantly different from those in Figures 4.11(a) and 4.11(b). For the 2D cooccurrence matrices, there are also differences among the classes. It seems from these figures that either the 2D or 3D cooccurrence features could be used to effectively represent each of the three classes. However, experi-

(a) $\theta = 0$ (b) $\theta = 45$ (c) $\theta = 90$ (d) $\theta = 135$

Figure 4.8: 2D Cooccurrence matrix obtained with $G = 128$, $d = 1$ and given values of θ.



(a) $G = 32$ (b) $G = 64$ (c) $G = 128$

Figure 4.9: 2D Cooccurrence matrix obtained with $d = 1$, $\theta = 0$ and given values of G.

(a) C1 (b) C2 (c) C3

Figure 4.10: Cooccurrence matrix obtained from the given class with $d = 1$, $\theta = 90$ and $G = 128$.

(a) C1 (b) C2 (c) C3

Figure 4.11: 2D Cooccurrence matrix obtained from the given class with $d = 1$, $\theta = 0$ and $G = 64$.

mental results in Section 3.6 revealed that the 3D cooccurrence measures provided better representations of the classes than the 2D cooccurrence measures.

4.4 Summary

In this chapter two classes of 3D texture features have been described: cooccurrence matrix based features and statistical features. Since it is desired to calculate the cooccurrence matrix from volume data it is necessary to redefine the matrix. Section 4.1.1 derived the new definition in 3D. This definition provides the ability to represent the texture in the volume in 3D, rather than in a collection of 2D matrices as has been done in the past. The definition also makes it possible to calculate measures from the cooccurrence matrix that are independent of the orientation of the texture in the volume. Traditionally, the cooccurrence matrix has always been dependent on the texture orientation. Experiments were performed to analyse various characteristics of the 3D and 2D cooccurrence matrices. The results showed that there was not a significant difference in the information content of 3D matrices constructed with different parameter combinations of θ and r (d). There is an increase in information when G is increased. The results also showed that there is significantly more information present in the 3D cooccurrence matrix as compared to the 2D cooccurrence matrix. Finally, the results showed that there is a difference in the amount of time required to construct the 3D cooccurrence matrix as θ decreases or the radial distance, r, is used instead of the pixel distance, d.

It was found in the experiments on bovine ultrasound volume data in Chapter 3 that the measures based on the 3D cooccurrence matrix resulted in feature spaces with a higher probability of correct classification than those based on the 2D cooccurrence matrix. In Chapter 5, it is described how these 3D cooccurrence based feature spaces are used by the Segmentation module to segment the bovine data. The use of these feature spaces provides the ability to successfully segment the corpus luteum and stroma; two visually very similar tissues. This is significant as differentiation of these tissues was not possible using 2D cooccurrence measures of texture in previous work [80, 79, 77].

Chapter 5

3D Segmentation

Due to the increasing popularity of data visualisation, segmentation of 3D datasets is becoming more important. Many approaches rely heavily on the selection of a threshold (*isosurface rendering*) or mapping function (*volume rendering*) in determining which points in the volume should contribute to the visualised surface(s) of interest. The amount of noise in the data has a significant effect on the quality of the rendered image.

Using 2D segmentation techniques on 3D data, the potential increase in information that is available in the third dimension is not used because typical processing involves segmenting each 2D image and then combining the resulting segmented images in some manner. This also implies that the third dimension is not treated in the same manner in which the 2D images are. Another problem with 2D segmentation is the orientation in which the planes of the volume are examined. Figure 5.1 illustrates two perspectives of a simple volume composed of alternating black and white planes. In the left image the volume seems to be composed of planes having a "striped" texture. In the right image the volume appears to be composed of uniform

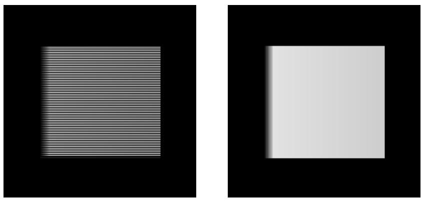


Figure 5.1: Two different perspectives of the same volume. The volume is composed of alternating black and white planes along the Z axis.

planes. However, by examining the entire volume, rather than a single plane at a time, this ambiguity would not arise, thus, the motivation for 3D segmentation.

5.1 3D MTS

The 3D Segmentation module of the proposed system must have the ability to segment isometric volume data. It must also be able to achieve good segmentation results when noise is present in the volume data. Finally, the 3D Segmentation module should use appropriate measures of homogeneity (such as texture measures) in performing segmentation. It is hypothesised that the 3D Segmentation module must possess these features in order to achieve accurate segmentation of the ultrasound volume data. One approach to segmentation which possesses these features is the Multiresolution Texture Segmentation (MTS) algorithm [77, 81, 82, 78]. Previous research demonstrated that a series of 2D ultrasound images could be segmented with good results. The following are a number of desirable characteristics of the MTS algorithm:

- examines texture at multiple resolutions,
- robust in the presence of noise, and
- similarity measure configurable to application.

It is important that these characteristics be maintained in the 3D Segmentation module of the proposed system.

Although the MTS algorithm has demonstrated success in segmenting noisy 2D images, it has some weaknesses that limit its use in the proposed 3D application:

- makes no use of third dimension in evaluating texture,
- uses an *ad hoc* similarity measure.

The 3D MTS algorithm is proposed to overcome the weaknesses of the original MTS algorithm.

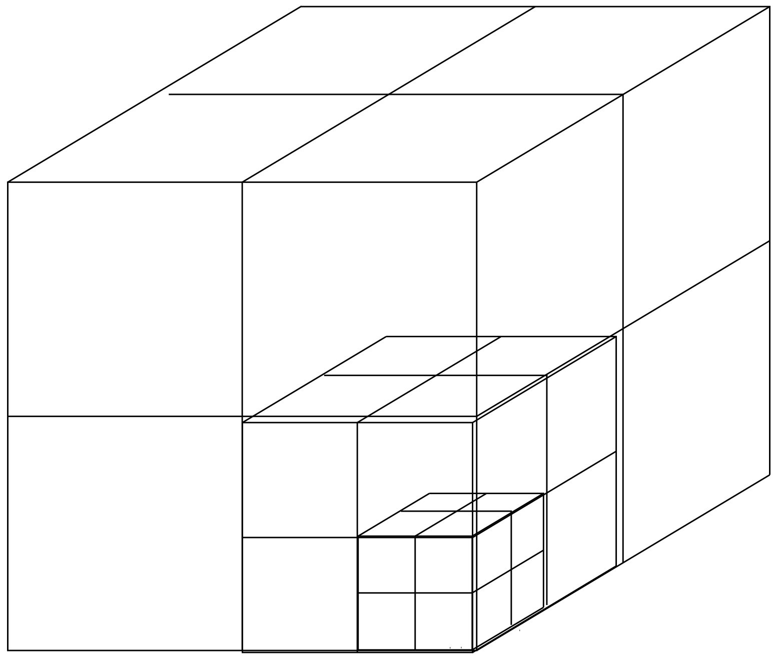


Figure 5.2: Octree structure.

The MTS algorithm is based on the quadtree structure (see Figure 2.15) which is inherently 2D. The use of an octree structure makes it possible to represent the 3D data. Using an octree, a block of $N \times N \times N$, rather than an $N \times N$, pixels is the unit of measure. This provides the ability to treat all three dimensions in an equal manner. Of course, this implies that the ability of the 3D MTS algorithm to segment volume data is dependent on how well the 3D pixel blocks can be characterised.

Figure 5.2 depicts the structure of an octree. An octree is a tree data structure in which each parent node has 8 child nodes. Each leaf node in the tree represents a block within the volume. For example, a single node octree represents the entire volume. As the tree is expanded down from the root, smaller blocks in the volume are represented by the leaf nodes. For an M^3 volume, the size of a block for a corresponding leaf node at level l is $(2^{(n-l)})^3$, where $0 \leq l \leq n$ and $M = 2^n$.

The basic mechanism of splitting and merging blocks in the MTS algorithm is inherited in the 3D MTS. A simulated annealing framework controls the manner in which this occurs. A block is split if its 8 children are *not similar* (see Section 5.1.1

for a discussion of the similarity measure). A neighbourhood of 8 *similar* blocks is merged into the parent block. The parameter α determines the cooling schedule. At an initial temperature, T, random splitting and merging of blocks are allowed to occur. T is decreased according to the prescribed cooling schedule and the number of random configurations decreases. The steady state is reached when T drops below a threshold at which no more random configurations occur. Since T is a monotonically decreasing function of α, smaller values of α result in a faster cooling schedule, thus decreasing the number of random splits and merges. An instant cooling schedule is achieved by not allowing any random configurations ($\alpha = 0$).

At every temperature interval, a total of Ω splits and merges are allowed. For high temperatures, the percentage of random splits and merges will be higher than the percentage of random splits and merges at low temperatures. When T is low enough, only those splits and merges which satisfy the similarity measure occur. The resulting state of the octree after iteration completes represents the final segmented volume.

The above describes the mechanics of how blocks are examined within a volume; however, it is yet to be described how blocks are determined to be similar or not similar. As well, it has not been mentioned how the identity of each block is determined. The next section discusses these issues.

5.1.1 Similarity Measure

The biggest weakness of the MTS algorithm lies in the similarity measure used to determine the homogeneity of regions. Recall from Equation (2.1) that $\Phi_x(C)$ measures how well the current block fits the texture region in the image. The main contributing factor to $\Phi_x(C)$ is the similarity measure, $d_C(P)$:

$$d_C(P) =_{1 \leq k < i}^{MIN} \left(\sum_{j=k+1}^{i} \delta \left(T(C_k) \neq T(C_j) \right) \right) \tag{5.1}$$

where

- P is either the parent of the leaf node C (merging) or is C itself (splitting),

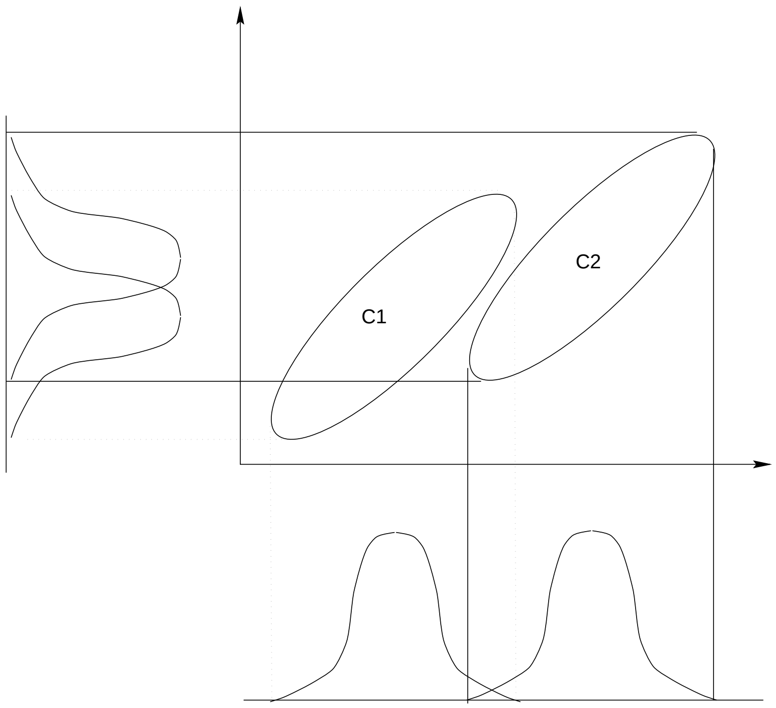


Figure 5.3: Distinct classes with overlap along the projected axis.

- i is the number of child nodes of P, since the quadtree structure is used, i=4, and
- $T(C_j)$ indicates the class of child block C_j.

Equation (5.1) defines the number of children with a class different from the class of the majority of the children. For example, if the classes for the four children of P are represented as $\{1,1,2,3\}$ then the class of the majority is 1 and the number of children with a different class is two, $d_C(P) = 2$.

The problem lies in the calculation of a block's class using $T(C_j)$. In the MTS algorithm, *ad hoc* rules are used in determining the similarity between block texture measures and the class parameters (determined *a priori*). The most limiting factor is that the similarity is based on one-dimensional projections of the measures (feature space). Figure 5.3 shows two classes which are distinct in a 2D feature space yet their projections result in overlap. In general, classes which are linearly separable (either by a line, plane, hyperplane, etc.) are not always separable along axial projections.

To overcome this deficiency in the 3D MTS algorithm, the *ad hoc* determination of a block's class using $T(C_j)$ is replaced by the Inck decision rule described in Section 3.2. The main advantage in using the modified decision rule is that it is based on well established theory in statistical pattern recognition. With this approach there is now a direct relationship between classification and segmentation which have traditionally been treated as separate topics.

Identifying the class membership of blocks within a volume using the 3D MTS algorithm is like performing "classification on the fly". The use of a multiresolution classifier, ϕ, as described in Section 3.5, is used to classify blocks at different resolutions. By replacing $T(C_j)$ in Equation (5.1) with ϕ, the similarity measure used in the 3D MTS algorithm becomes

$$d_C(P) = \overset{MIN}{\underset{1\leq k<i}{}} \left(\sum_{j=k+1}^{i} \delta\left(\phi(C_k) \neq \phi(C_j)\right) \right). \tag{5.2}$$

Note that both the classifier, ϕ, and the block (or pattern), C_j, are resolution dependent. Also note that use of the octree structure requires $i = 8$.

Using a multiresolution classifier based on the modified decision rule requires the *a priori* class probabilities. These probabilities can be determined in a number of ways:

- assume probabilities are equal for each class,
- use probabilities obtained while determining class distribution parameters (training),
- allow probabilities to be defined based on knowledge of application.

The easiest way is to assume the probabilities are equal. However, this prevents making use of any biases that may exist due to classes that are more or less likely than other classes. The second way is equally as easy but there is no basis to support the assumption that the *a priori* class probabilities in training are the same as that during segmentation. The final way is the most desirable, as additional knowledge about the data being segmented is made available to the classifier. However, if the knowledge is inaccurate then the performance of the classifier will be reduced.

5.1.2 Outputs

There are three types of output available from the 3D MTS. These outputs represent different characteristics of resulting segmentation. The first output type is the class probability distributions. This output is similar to the type of output produced by Momenan *et al.* [75] in which 2D ultrasound scans are “stained” with different colours indicating the probability of which a region belongs to a given reference class.

Figure 5.10(b) shows (2D) examples of the class probability output type. Each point in the volume represents a probability of belonging to a given class. A vector, P, at each point contains these *a posteriori* probabilities such that

$$\sum_{i=1}^{C} P[i] = 1$$

where C is the number of classes determined prior to segmentation. The vector P contains two additional elements, $P[C+1]$, representing membership in the distance reject classes, and $P[C+2]$, representing membership in the ambiguity reject class. These classes result from the use of the modified decision rule (see Section 3.3). When $P[C+1] > 0$, the *a posteriori* probabilities are meaningless as the point is distant from all C classes. When $P[C+2] > 0$, the membership of the point is ambiguous among two or more of the C classes.

This output type contains the most information about the class probabilities and is most readily visualised using a volume rendering technique.

The second type of output is the class labels. Figure 5.4 shows an example of the this type of output. For the class labels a decision is made as to the most probable class to which the point belongs. The decision is made by selecting the largest $P[i]$ at each point and assigning the corresponding class number as the output value. The class labels output type is a simpler form of the class probability distributions

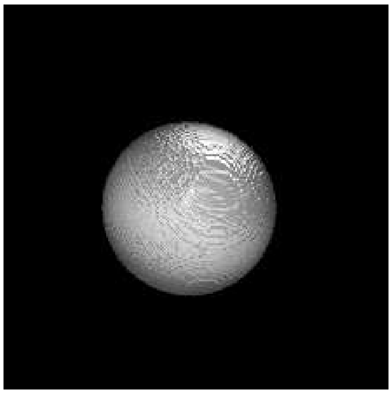


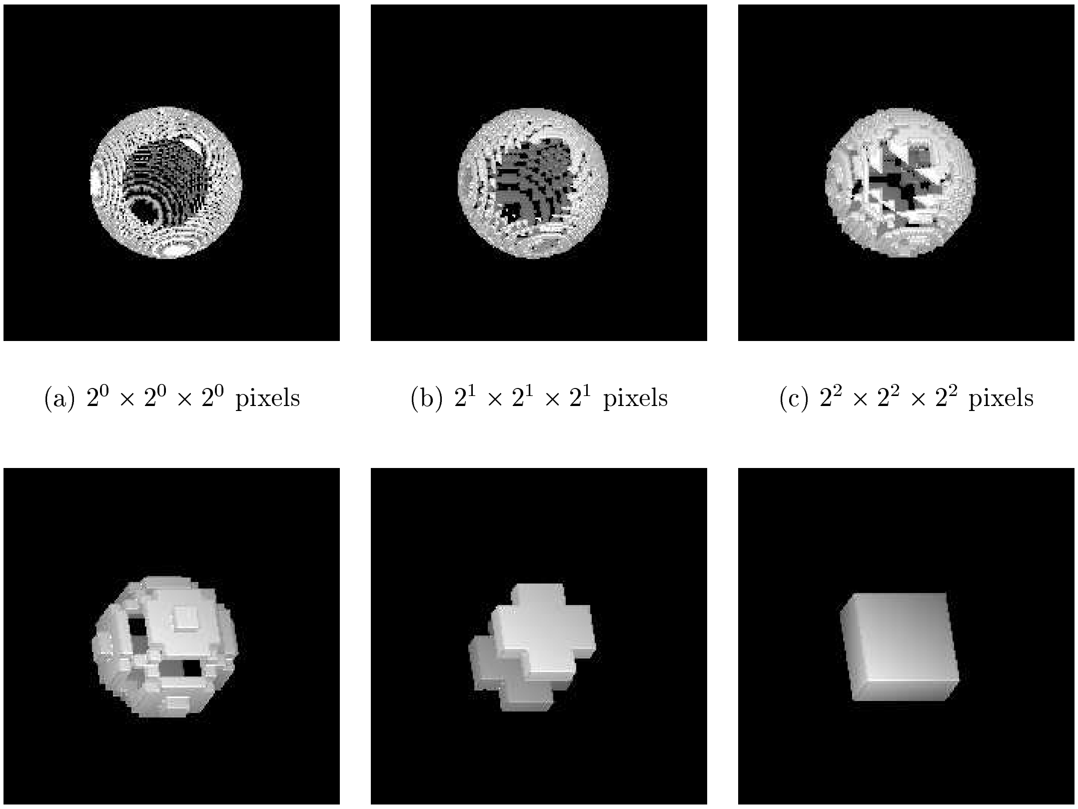
Figure 5.4: Output type 2: class labels.

as each point contains a single value indicating its class identity. This output type is most readily visualised using an isosurface rendering technique.

The final type of output is the block size distributions. Figure 5.5 shows an example of this type of output. Each point in the volume is assigned a value indicating its size (c.f. the point's level in the octree). This output type is most useful for analysing the performance of the 3D MTS. The distribution of the various block sizes throughout the volume indicates where boundaries and detailed regions occur. There are two interpretations of detailed regions. One is that the volume has small details which the algorithm is trying to determine. The second interpretation is that the algorithm could not determine the identity of certain regions in the volume. Successive splitting of the region resulted in small blocks whose identity are not known. This results in longer processing time and a decrease in segmentation accuracy. This output type is most readily visualised using an isosurface rendering technique.

5.2 Experiments

Experiments were performed to evaluate the effectiveness of the proposed 3D MTS algorithm in segmenting volume data. The following sections describe the experiments performed using synthesised (Section 5.2.1) and real (Section 5.2.2) data.



(a) $2^0 \times 2^0 \times 2^0$ pixels
(b) $2^1 \times 2^1 \times 2^1$ pixels
(c) $2^2 \times 2^2 \times 2^2$ pixels
(d) $2^3 \times 2^3 \times 2^3$ pixels
(e) $2^4 \times 2^4 \times 2^4$ pixels
(f) $2^5 \times 2^5 \times 2^5$ pixels

Figure 5.5: Output type 3: block size distribution. Each image depicts all blocks of a given size in the volume. A cut plane is used to allow visualisation of the interior of the object.

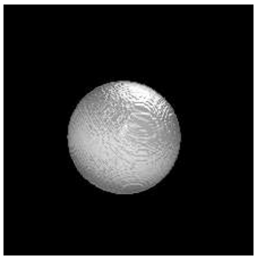
5.2.1 Synthesised Data

The goal of these experiments was to determine the accuracy in which segmentation of synthesised volume data could be achieved using the 3D MTS algorithm. Emphasis was placed on studying the effects of the $N \times N \times N$ block size so a simple homogeneity criteria (mean gray level) was used. Hundreds of experiments were performed on data sets which were synthesised by sampling analytic functions (sphere, ellipse, cube, and toroids) at discrete, isometric points in the Cartesian coordinate system. Each sample point within the boundaries of the function was assigned the value 200 while the other points were assigned the value 100. Gaussian noise (with varying σ) was added to the images to determine how robust the 3D MTS algorithm would be to a degradation in image quality. Volumes of size 64^3 pixels and 128^3 pixels were segmented. All the experiments were performed on a Sun Microsystems SPARCstation 10 with 64M of memory running SunOS 4.1.3. All timings are given in real time.

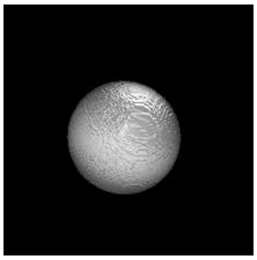
Two other segmentation algorithms were evaluated to provide a basis for comparison with the 3D MTS algorithm. A simple thresholding algorithm was used because it is currently a popular method for rendering an isosurface within a volume. The 2D MTS algorithm was also used because of its similarity to the 3D MTS algorithm. The 2D MTS algorithm was applied to each plane along the Z axis of the volume and the resulting segmented planes were combined into a single volume. The accuracy of each segmentation was determined by calculating the percentage of mislabelled pixels.

Results

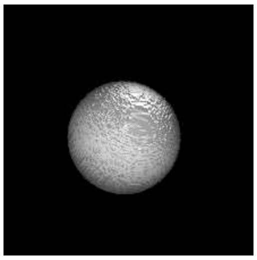
Figure 5.6 shows some sample segmentation results obtained by thresholding, the 2D MTS, and the 3D MTS algorithms. Figures 5.6(a), 5.6(e) and 5.6(i) show the original test volume (128^3 pixels) containing additive Gaussian noise with $\sigma = 0$, $\sigma = 40$ and $\sigma = 60$, respectively. The segmentation results are shown in Figures 5.6(b), 5.6(f),5.6(j) (thresholding),5.6(c), 5.6(g),5.6(k) (2D MTS), 5.6(d), 5.6(h) and



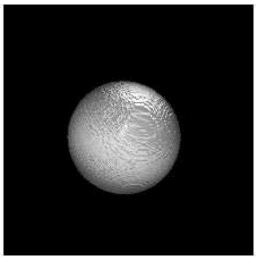
(a) original



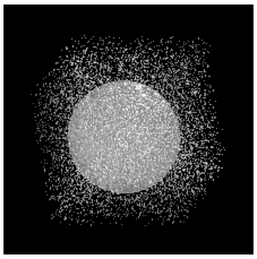
(b) Thresholding: $\sigma = 0$, accuracy: 100%



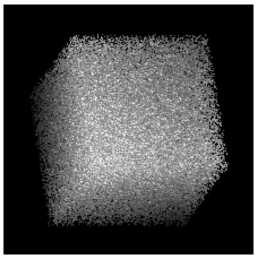
(c) 2D MTS, $\sigma = 0, \alpha = 0$, accuracy: 99.5%



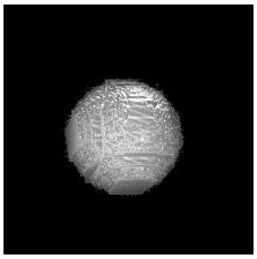
(d) 3D MTS, $\sigma = 0, \alpha = 0$, accuracy: 100%



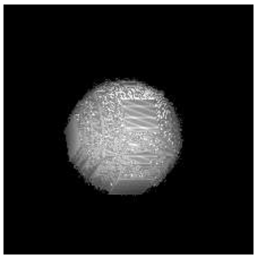
(e) original



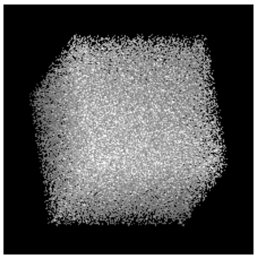
(f) Thresholding, $\sigma = 40$, accuracy: 89.3%



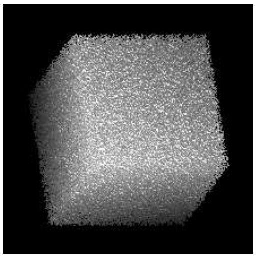
(g) 2D MTS, $\sigma = 40, \alpha = 0$, accuracy: 98.8%



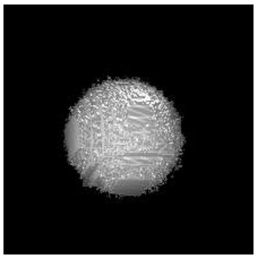
(h) 3D MTS, $\sigma = 40, \alpha = 0$, accuracy: 99%



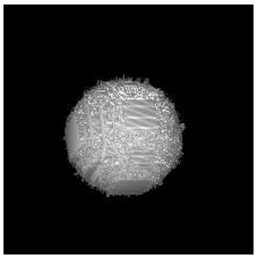
(i) original



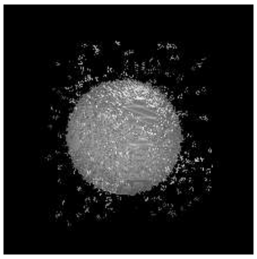
(j) Thresholding, $\sigma = 60$, accuracy: 78.2%



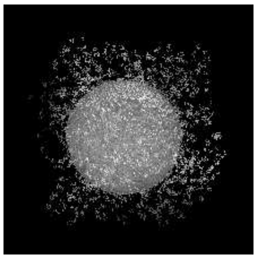
(k) 2D MTS, $\sigma = 60, \alpha = 0$, accuracy: 98.5%



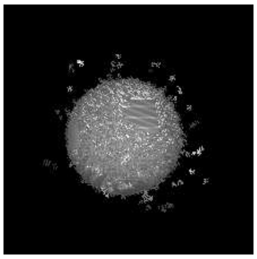
(l) 3D MTS, $\sigma = 60, \alpha = 0$, accuracy: 98.5%



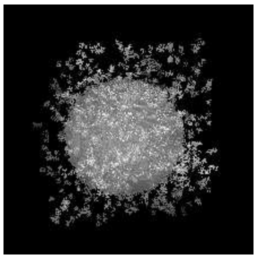
(m) 3D MTS, $\sigma = 40, \alpha = 0.3$, accuracy: 99.1%



(n) 3D MTS, $\sigma = 40, \alpha = 0.5$, accuracy: 98.8%



(o) 3D MTS, $\sigma = 60, \alpha = 0.1$, accuracy: 98.5%



(p) 3D MTS, $\sigma = 60, \alpha = 0.3$, accuracy: 97.9%

Figure 5.6: Segmentation results for sphere with given values of σ and α.

5.6(l) (3D MTS). When there is no noise present ($\sigma = 0$), each algorithm segments the volume almost perfectly. However, as the level of noise increases, the accuracy of the segmentations using thresholding decreases significantly while the accuracy of the segmentations obtained using the 2D and 3D MTS algorithms remains high.

Figures 5.6(m)-5.6(p) show some results obtained by the 3D MTS algorithm with different cooling schedules. The results seem to suggest that there is no advantage in slowing the cooling schedule (increasing the coefficient α) as there is no improvement in the accuracy of the segmentation. This is in agreement with the findings of Liu and Yang [61].

The 3D MTS algorithm required, on average, approximately twice the amount of time to segment the test images as compared to the 2D MTS algorithm (147 sec. vs. 70 sec.). As well, the segmentation times for both the 2D and 3D algorithms increased as the cooling schedule coefficient α was increased.

5.2.2 Real Data

The goal of these experiments was to determine the accuracy (qualitatively) in which segmentation of real volume data could be achieved using the 3D MTS algorithm. Data sets from two different applications were evaluated. The first data set consists of a linear scan of a fetus while the data sets from the second application consist of linear scans of bovine ovaries.

Fetal Volume Data

As with the synthesised data, a simple homogeneity criteria (mean gray level) was used in segmenting the fetal data. The volume data consists of 232 parallel slices from an *in vitro* linear scan of a fetus obtained using a 5-9 MHz ultrasound transducer. The inter-slice distance is 0.25 mm and the resolution of each image is 640×480 pixels (0.25 mm/pixel). A volume of size 256×256×128 was extracted from the original data and used for all the experiments. To facilitate faster processing and lower resource requirements, the volume was segmented as four, 128^3 pixel

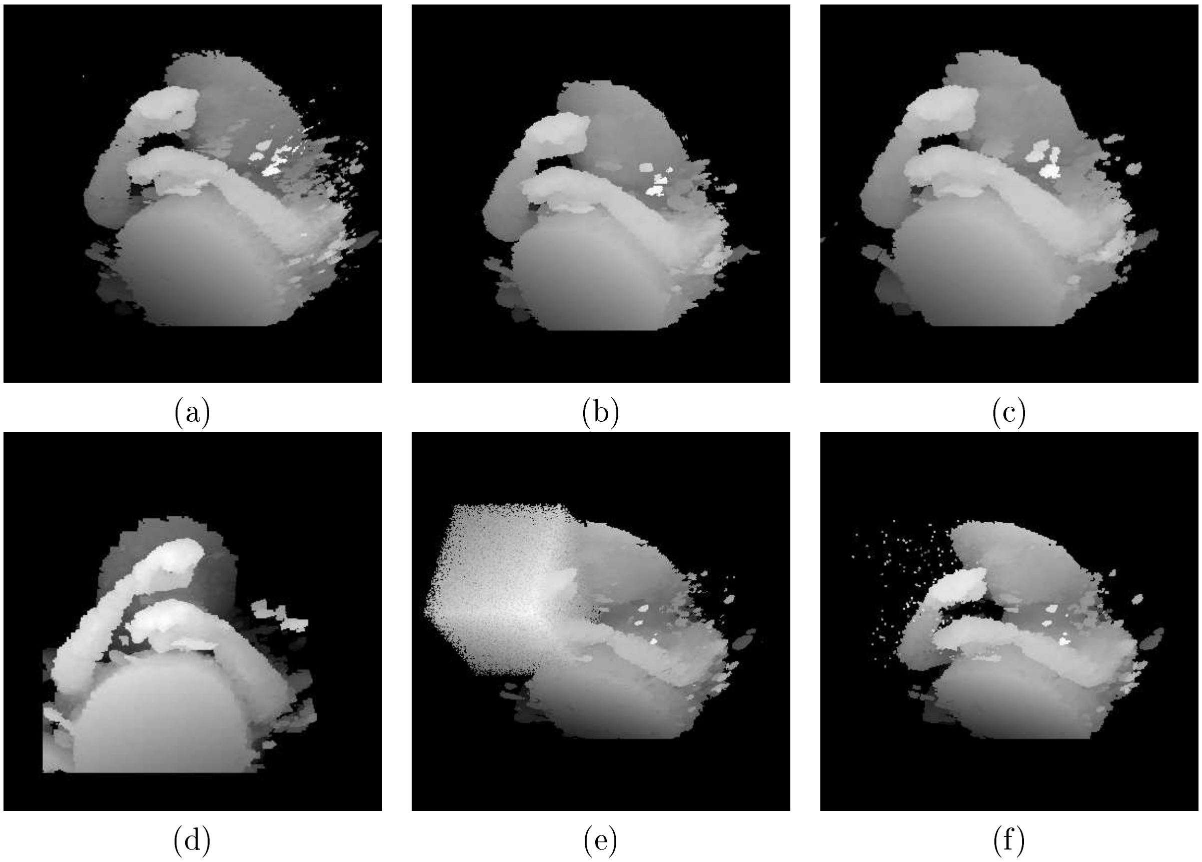


Figure 5.7: Segmentation results for original ultrasound data of a human fetus: (a) Thresholding, (b) 2D MTS, (c) and (d) 3D MTS (different perspectives). Results with additive noise, $\sigma = 40$: (e) Thresholding and (f) 3D MTS.

volumes independently. The resulting segmentations were combined to achieve the final segmentation results for the entire volume.

Results

Figure 5.7 shows some sample segmentation results obtained by thresholding, the 2D MTS, and the 3D MTS algorithms. Figures 5.7(a)-(d) show the results obtained for the different segmentation algorithms on the original volume data. Figure 5.7(e) shows the resulting segmentation produced by thresholding with noise ($\sigma = 40$) added to a 128^3 pixel volume within the original data. Figure 5.7(f) shows the resulting segmentation produced by the 3D MTS algorithm for the same noisy data. The results shown for both the 2D and 3D MTS algorithms were obtained using an instant cooling schedule ($\alpha = 0$).

It can be seen from these results that each algorithm segmented the original volume data with roughly the same quality. However, when noise is added to the data, thresholding produces poor results while the 3D MTS algorithm maintains its quality. It can be argued that the addition of noise to the original data produces a contrived example which only serves to exaggerate the capabilities of the 3D MTS algorithm. However, in an *in vivo* environment, there is a significant increase in the amount of noise present in the data. These results demonstrate that the 3D MTS algorithm should produce better segmentations than thresholding for noisy data.

The 3D MTS algorithm segmented the original data in approximately 348 seconds while the 2D MTS algorithm required approximately 709 seconds. Similar times (relative) resulted for slower cooling schedules (i.e. α increasing). This is a direct contrast with the segmentation times obtained for the synthesised data in which the 3D MTS algorithm required twice the amount of time as compared to the 2D MTS algorithm. This can be attributed to the quadtree (octree) representation of the data. For the 2D MTS algorithm, there are more leaves in total in the 128 quadtrees than the 1 octree representation used by the 3D MTS algorithm. The number of leaves depends on the homogeneity of the data. Thus, the time required to produce a segmentation using either algorithm is fairly data dependent.

Bovine Volume Data

The bovine data, as describe in Section 3.6, was used to determine the accuracy (qualitatively) in which segmentation could be achieved using the 3D MTS algorithm. A volume of size $400 \times 300 \times 400$ points was extracted from the interpolated data and used for all the experiments.

The multiresolution classification obtained for the bovine volume data using the Tissue Characterisation module in Section 3.6.4 was used as a basis for segmenting the bovine data. That is, the feature spaces (and associated class parameters) for resolutions $\beta_2 = 4$, $\beta_3 = 8$, $\beta_4 = 16$ and $\beta_5 = 32$ were used by the 3D MTS algorithm. To simplify processing, small (two dimensional) features spaces were chosen from the (up to) ten dimensional feature spaces, listed in Table 3.10, determined from the 3C dataset. The decision to use small feature spaces was based on the necessity for minimal computational requirements as well as a good probability of correct classification. To further simplify processing, these feature spaces were modified so that for each resolution, the feature space (31,95) was used. Note that both of these features are measures of local homogeneity based on the 3D cooccurrence matrix. As a precursor to segmentation, classification was performed with this feature space to confirm that the probability of correct classification remained high. For all segmentations, an instant cooling schedule ($\alpha = 0$) was employed.

For comparison purposes, a convolution method [120, 16, 72, 121, 119] was used. This method was chosen because it is commonly used to segment images. A window of size $16 \times 16 \times 16$ was convolved with the ultrasound volume at intervals of 8 pixels. At each location, a feature vector containing the 3D cooccurrence based features was calculated and evaluated using the Inck decision rule. The resulting output at each location was a vector containing the *a posteriori* class probabilities for each of the three classes present in the ultrasound data. The class associated with the highest probability in the vector was determined to be the class label assigned at the location. for both the 3D MTS and convolution methods, the class *a prior* probabilities were assumed to be equal (0.333).

Results

Figure 5.8 shows the results obtained when the 3D MTS algorithm is applied to the class sample regions, of size 32^3 points, used in determining the feature measures for the 3C dataset. Recall that the 3C dataset contains 3D samples of each of the three classes present in the bovine ultrasound volume data. Since it was assumed that each of the regions represented only one class, segmentation of each of the regions should be homogeneous. This is the case for class C3 (light gray) in Figure 5.8(c). In Figures 5.8(a) and 5.8(b), however, the segmented output for C1 (gray) and C2 (black) is not homogeneous. This suggests that there is some sampling error in the 3C dataset as the sample regions do not represent one class exclusively. Note, also, that the erroneously identified regions are based on measures

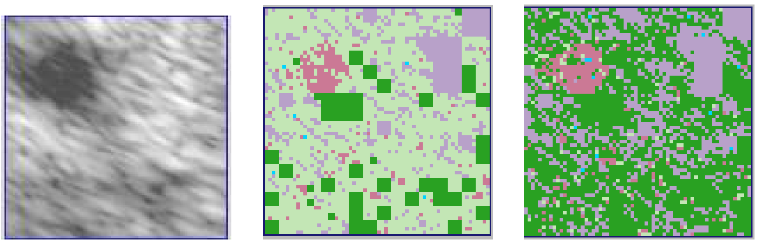
(a) C1 (b) C2 (c) C3

Figure 5.8: Segmentation error in class samples.

at a small resolution $\beta_2 = 4$. This confirms the hypothesis, described in Section 3.7, that segmentation based on the classification results at smaller resolutions results in higher probabilities of error.

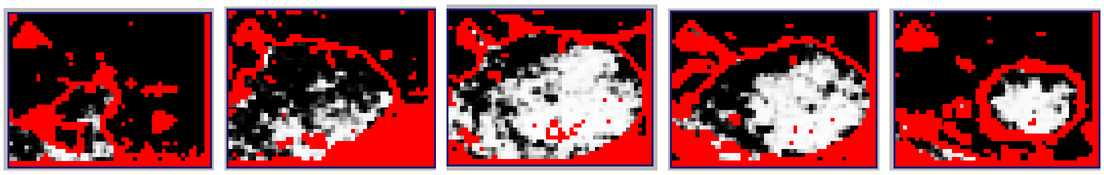
Figure 5.9 shows corresponding 2D slices selected from the original (Figure 5.9(a)) and segmented volumes based on values of $\epsilon = 0.05$ (Figure 5.9(b)) and $\epsilon = 1.0$ (Figure 5.9(c)). Since the probability of ambiguity reject increases as ϵ decreases, the resulting segmentation with $\epsilon = 0.05$ should contain more ambiguity rejected points than the segmentation with $\epsilon = 1.0$. It can be seen that this is the case. In Figure 5.9(b), the majority of points are identified as ambiguous (mostly between classes C1 and C2) while in Figure 5.9(c), many of the same points are identified as either C1 or C2, with a much higher probability of error.

Figure 5.10 shows the class probability distributions resulting from the convolution method with $\epsilon = 1.0$ (no ambiguity reject region). Figure 5.10(a) illustrates the *a posteriori* probability of points belonging to class C1 (corpus luteum), Figure 5.10(b), the *a posteriori* probability of class C2 (stroma) and Figure 5.10(c), the *a posteriori* probability of class C3 (fluid). Points displayed as white have a high probability of class membership while points displayed as black have a low probability of class membership. Red indicates those points which have been distance rejected. It can be seen from these images that fluid is quite prominently distinguished from

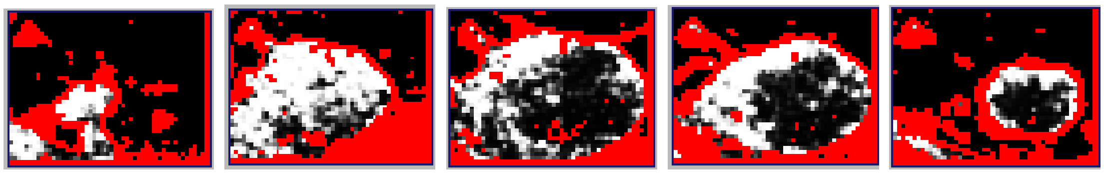


(a) original (b) $\epsilon = 0.05$ (c) $\epsilon = 1.0^*$

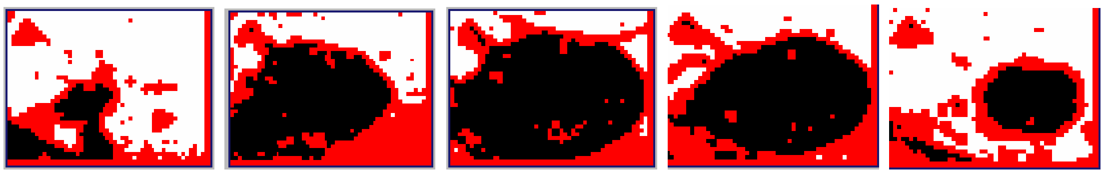
Figure 5.9: Segmentation results for different values of ϵ ($^*\epsilon = 0.5$ for resolution $\beta^1 = 2$); light green - ambiguity rejected, blue - distance rejected, green - C1, purple - C2, pink - C3.



(a) Probability of C1, slices 0, 34, 67, 83, 101



(b) Probability of C2, slices 0, 34, 67, 83, 101



(c) Probability of C3, slices 0, 34, 67, 83, 101

Figure 5.10: Probability of given class for selected slices from volume; red indicates point is distance rejected, black to white is probability 0.0 to 1.0.

the corpus luteum and stroma. As well, the corpus luteum is discriminated from the stroma with a high probability of correct classification. As there is a transition from one homogeneous region to the next, a "band" of distance rejected points is identified. This is intuitive as the 16^3 window overlaps more than one region, thus producing a feature vector which is statistically different from either of the class parameters.

Figure 5.11 shows different perspectives of the corpus luteum in the segmented volume using the convolution method. Identification of the corpus luteum was determined from the points with a class probability ≥ 0.95 for C1. Figure 5.12 shows different perspectives of the stroma in the segmented volume using the convolution method. Identification of the stroma was determined from the points with a class probability ≥ 0.95 for C2. Finally, Figure 5.13 shows different perspectives of the fluid in the segmented volume. Note the clear distinction among the corpus luteum, stroma and surrounding fluid. Although it is not totally apparent in the depictions of these structures, the segmentation is subjectively determined to be quite good.

Figures 5.14 - 5.16 show different perspectives of the regions identified as corpus luteum, stroma and fluid, respectively, using the 3D MTS algorithm with $\epsilon = 1.0$. Since the 3D MTS algorithm requires the volume data to have dimensions which are an even power of 2, the $384 \times 256 \times 384$ sub-volume originating at the origin of the original volume was segmented.

Comparison of the resulting segmentations obtained by the 3D MTS algorithm in Figures 5.14 and 5.15 to those obtained by the convolution method reveal that the corpus luteum and stroma are poorly distinguished. It is apparent that a number of blocks have been erroneously segmented. Since there is no ambiguity reject region, there is more uncertainty when a decision is made in classifying a block. Further, many of the blocks are classified at a smaller resolution, $\beta_2 = 4$. As stated above, at this resolution there is a much less chance of correctly classifying a block. Examination of the block distributions for the 3D MTS segmentations revealed that over 90% of the blocks in the volume were being classified at this resolution. Since a high probability of correct classification is not possible at this resolution, it was

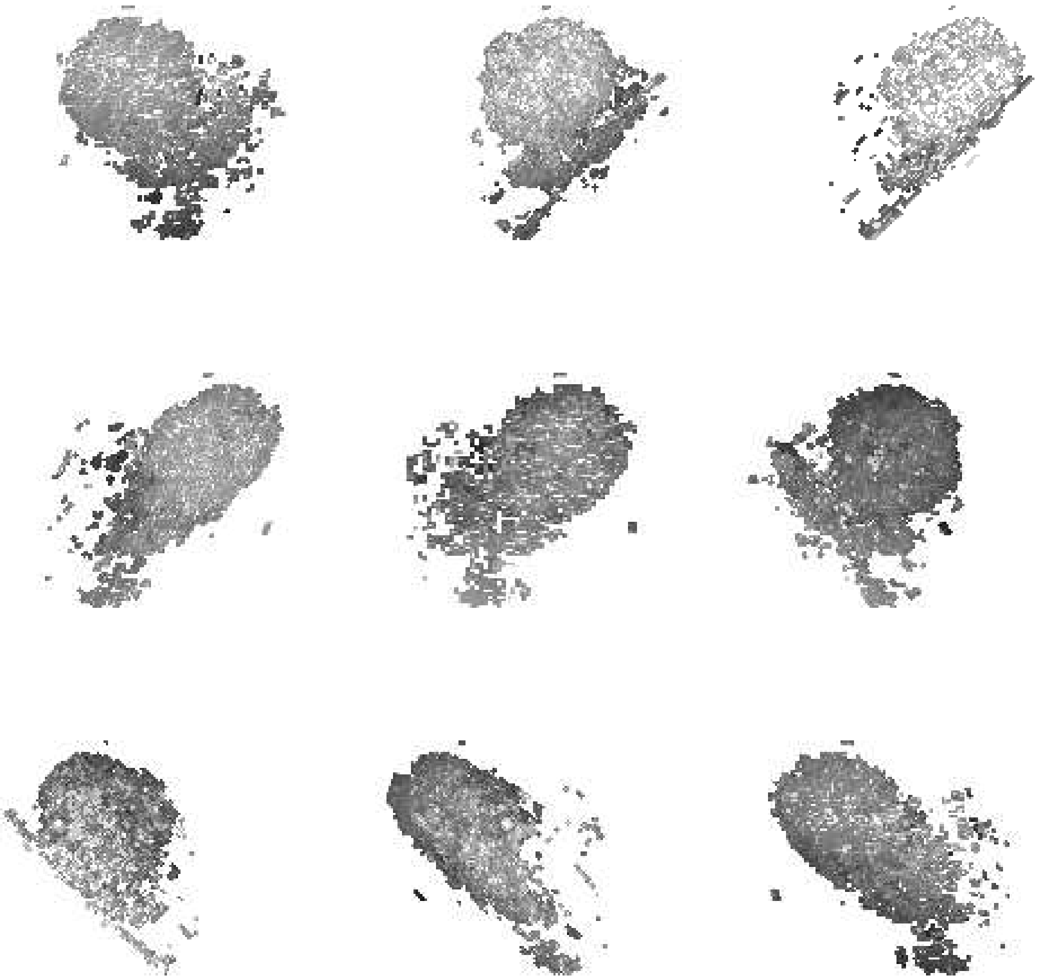


Figure 5.11: Bovine ovary animation of corpus luteum (C1) using convolution. Perspectives shown for 40° rotational increments about the y axis.

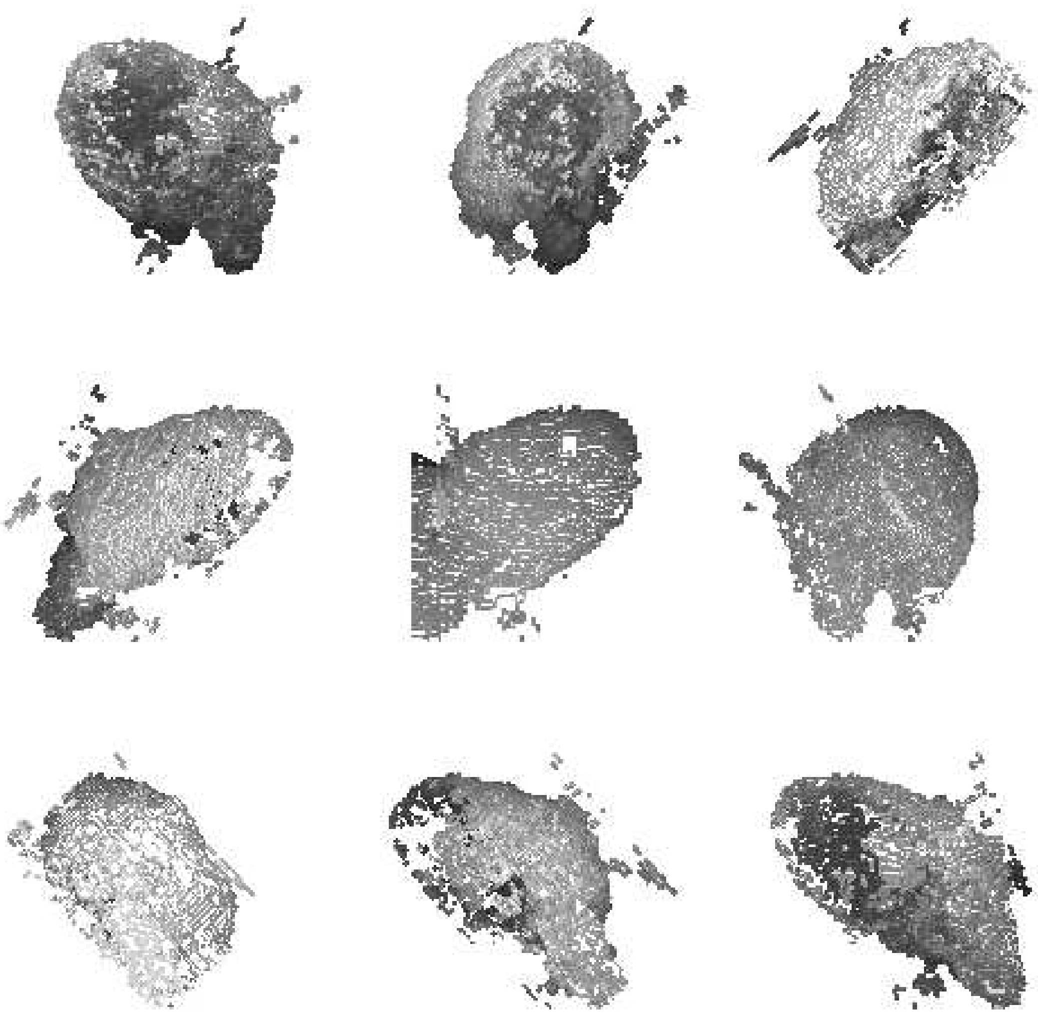


Figure 5.12: Bovine ovary animation of stroma (C2) using convolution. Perspectives shown for 40° rotational increments about the y axis.

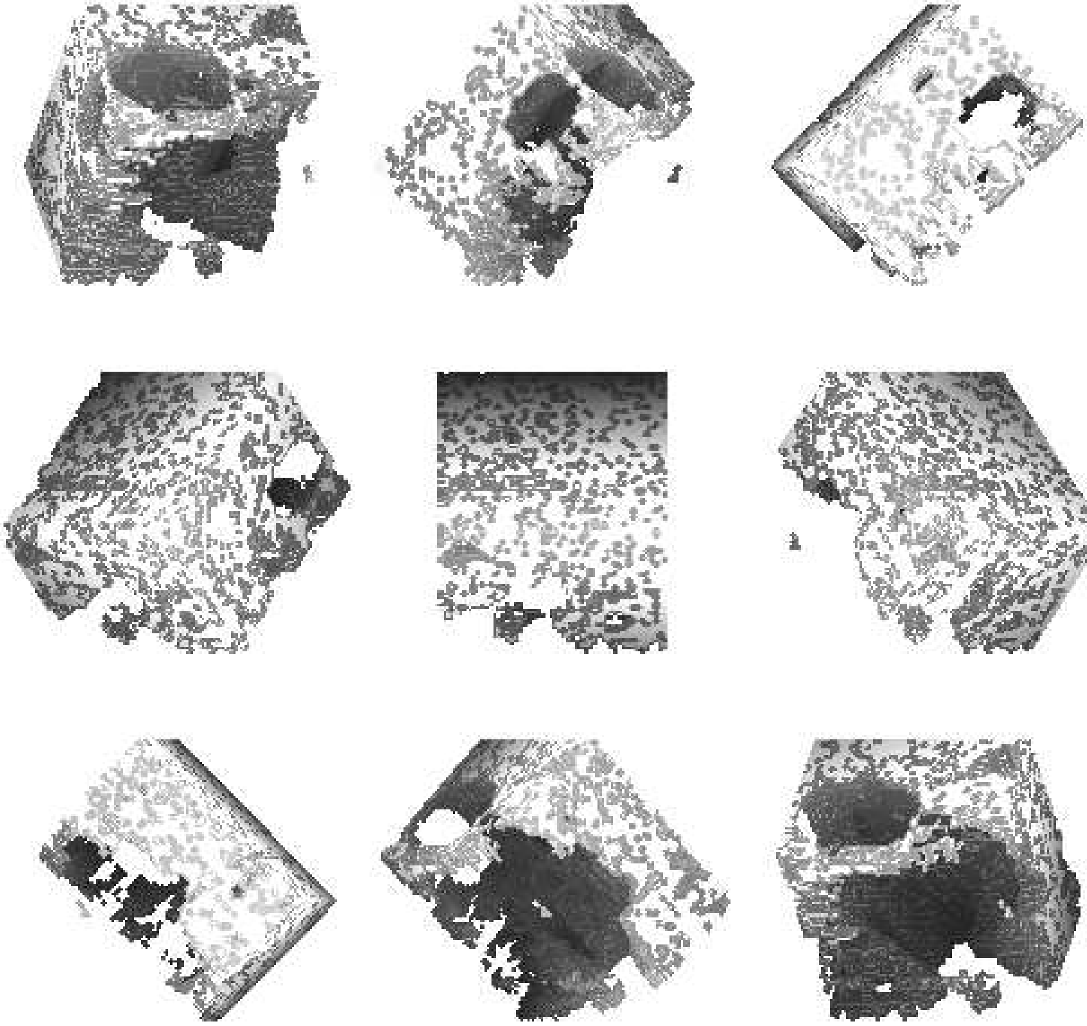


Figure 5.13: Bovine ovary animation of fluid (C3) using convolution. Perspectives shown for 40° rotational increments about the y axis.

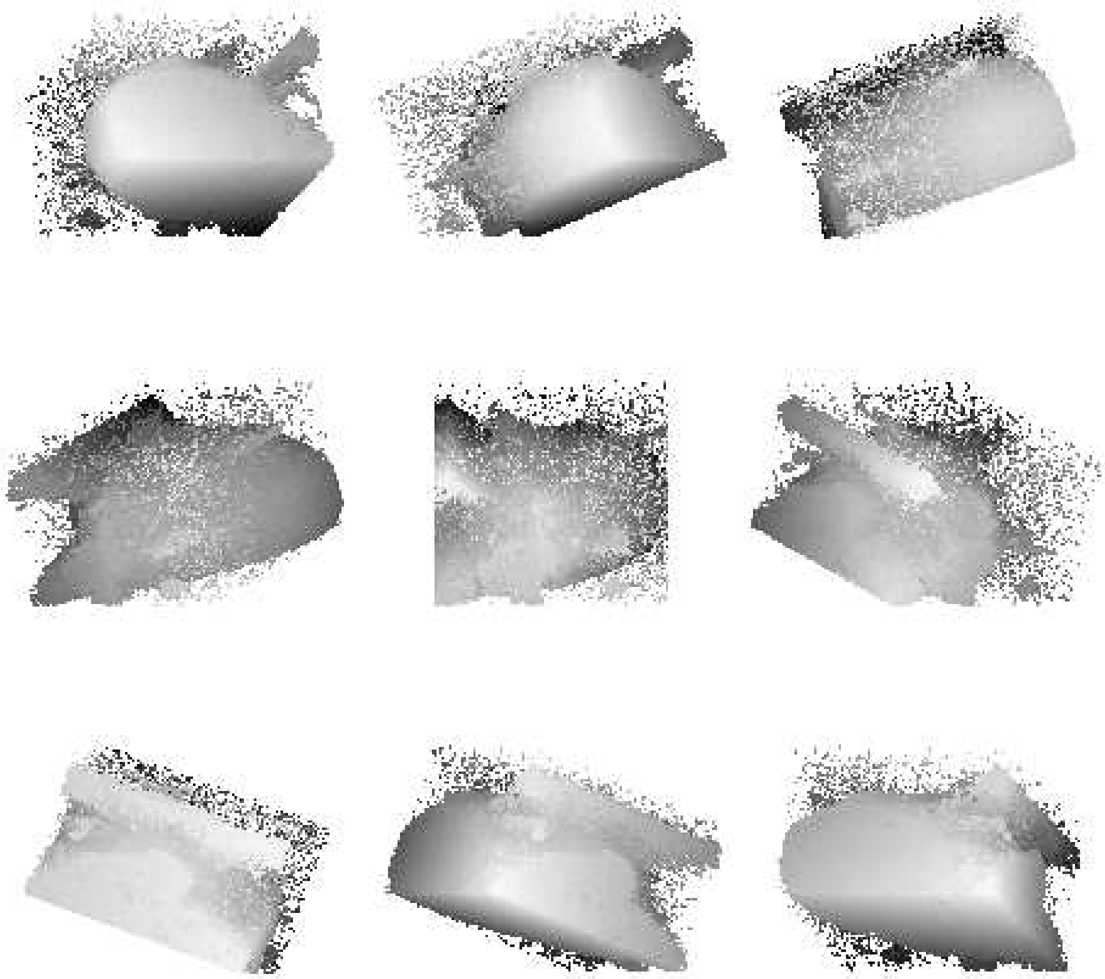


Figure 5.14: Bovine ovary animation of corpus luteum (C1) using the 3D MTS algorithm. Perspectives shown for 40° rotational increments about the y axis.

necessary to modify how features were calculated for these small blocks.

Figures 5.17 - 5.19 show different perspectives of the corpus luteum, stroma and fluid, respectively, identified using the 3D MTS algorithm with $\epsilon = 0.05$. By decreasing ϵ, a safeguard against classification error was achieved. However, many blocks at resolution $\beta_2 = 4$ and $\beta_3 = 8$ were identified as ambiguous. Instead of leaving these blocks labelled as ambiguous, a one step refinement was made to the 3D MTS algorithm. All ambiguous blocks at resolution $\beta_3 = 8$ (the smallest resolution

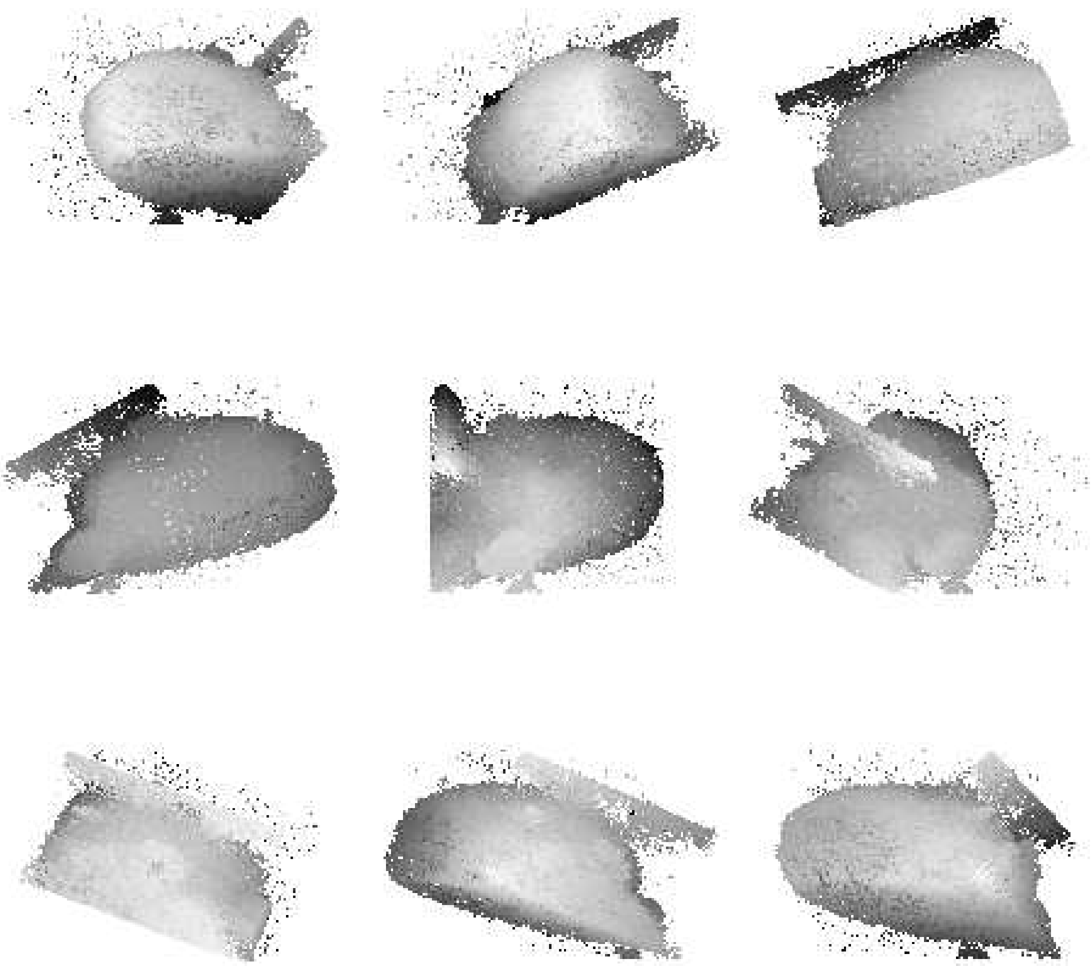


Figure 5.15: Bovine ovary animation of stroma (C2) using the 3D MTS algorithm. Perspectives shown for 40° rotational increments about the y axis.

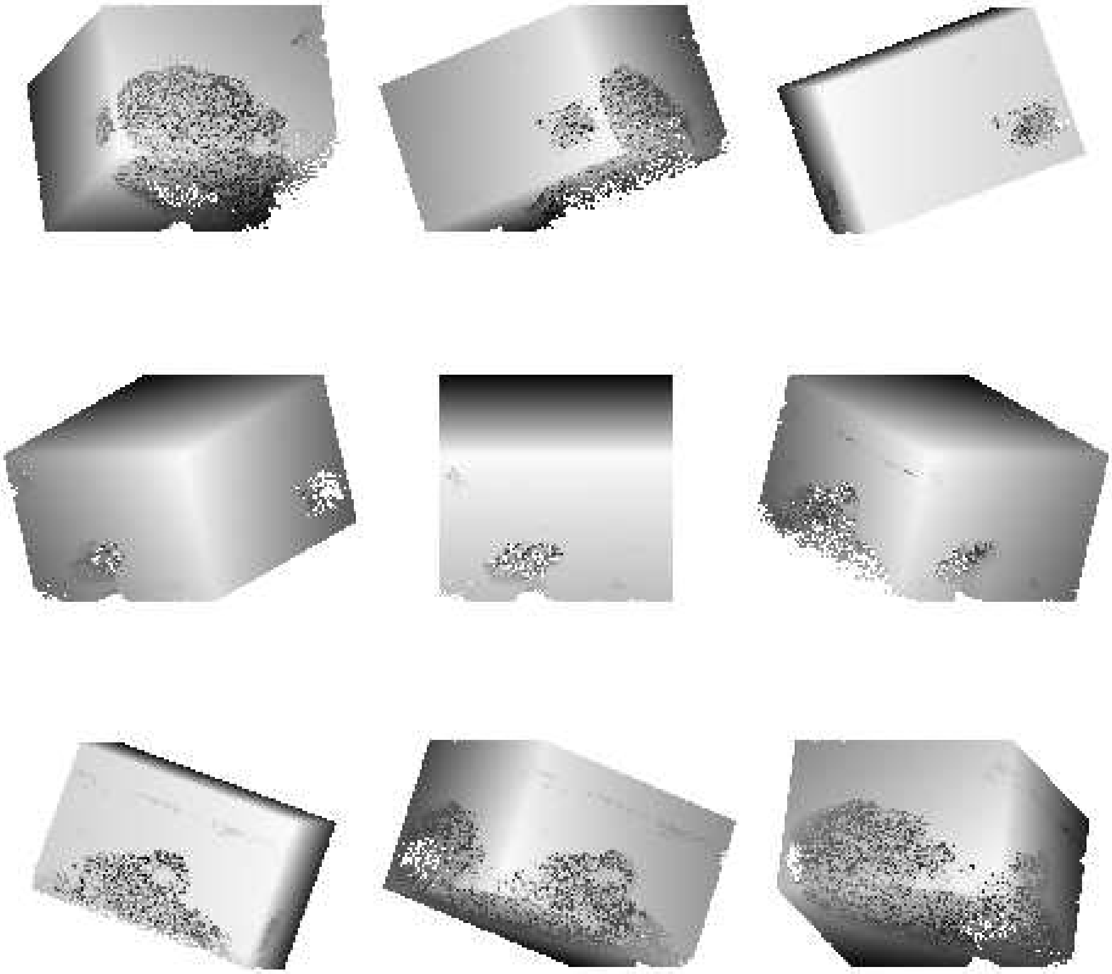


Figure 5.16: Bovine ovary animation of fluid (C3) using the 3D MTS algorithm. Perspectives shown for 40° rotational increments about the y axis.

was limited to $\beta_3 = 8$) had their feature measures calculated at a larger resolution, $\beta_4 = 16$, in a "one-time only" fashion (just as in the convolution method). The resulting probabilities were then used to determine the identity of the block. Figure 5.20 shows the distance rejected blocks remaining in the final segmentation. It can be seen from Figures 5.17 - 5.19 that this small refinement to the 3D MTS algorithm resulted in identification of the major structures in the bovine ovary with as much accuracy as those shown in Figures 5.14 - 5.16, obtained with the convolution method. The major advantage in using the modified version of the 3D MTS algorithm is that much less computational effort is required. The multiresolution capability allows the 3D MTS algorithm to adjust to varying levels of detail in the volume. That is, for large, homogeneous regions, features can be calculated from blocks at large resolutions. For the convolution method, the same resolution is used independent of the size of the homogeneous regions in the volume. Thus, for large homogeneous regions, features from a number of (smaller) blocks must be calculated.

The dominating operation in both methods is the calculation of the feature vectors from 16^3 blocks in the volume. For a 128^3 volume, the convolution method required 4096 operations. The real-time required to calculate the features was approximately 8000 seconds. For the refinement step of the 3D MTS algorithm, the number of operations averaged 71% of the number of operations required by the convolution method. The average time to complete the operations was approximately 4595 seconds. Additional overhead was required by the 3D MTS algorithm to produce the initial segmentation. On average, approximately 443 seconds were required to produce the initial segmentation.

5.3 Summary

This chapter described the details of the 3D Segmentation module. It was determined that the segmentation approach used must have the ability to segment isometric volume data, be able to achieve good segmentation results when noise is present in the volume data and should use appropriate measures of homogeneity

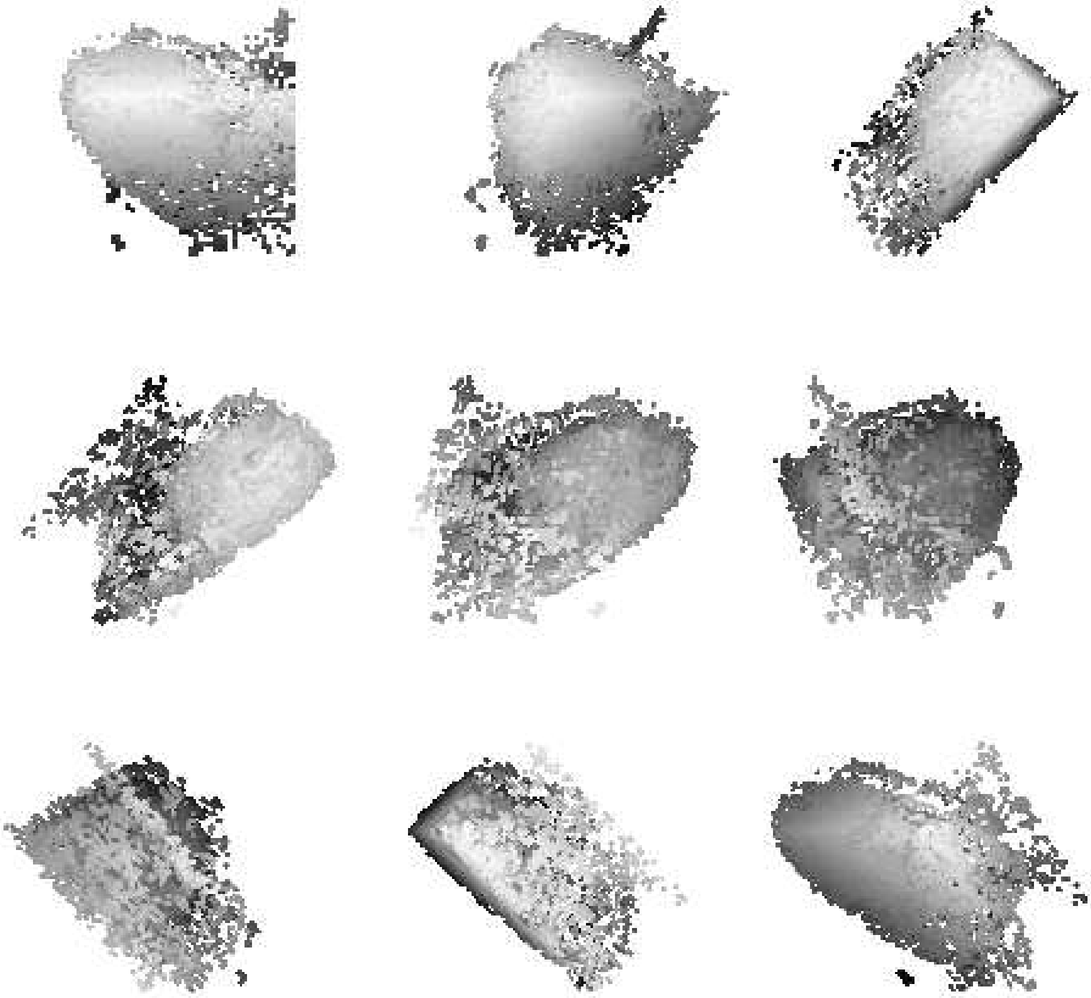


Figure 5.17: Bovine ovary animation of corpus luteum (C1) using the modified 3D MTS algorithm. Perspectives shown for 40° rotational increments about the y axis.

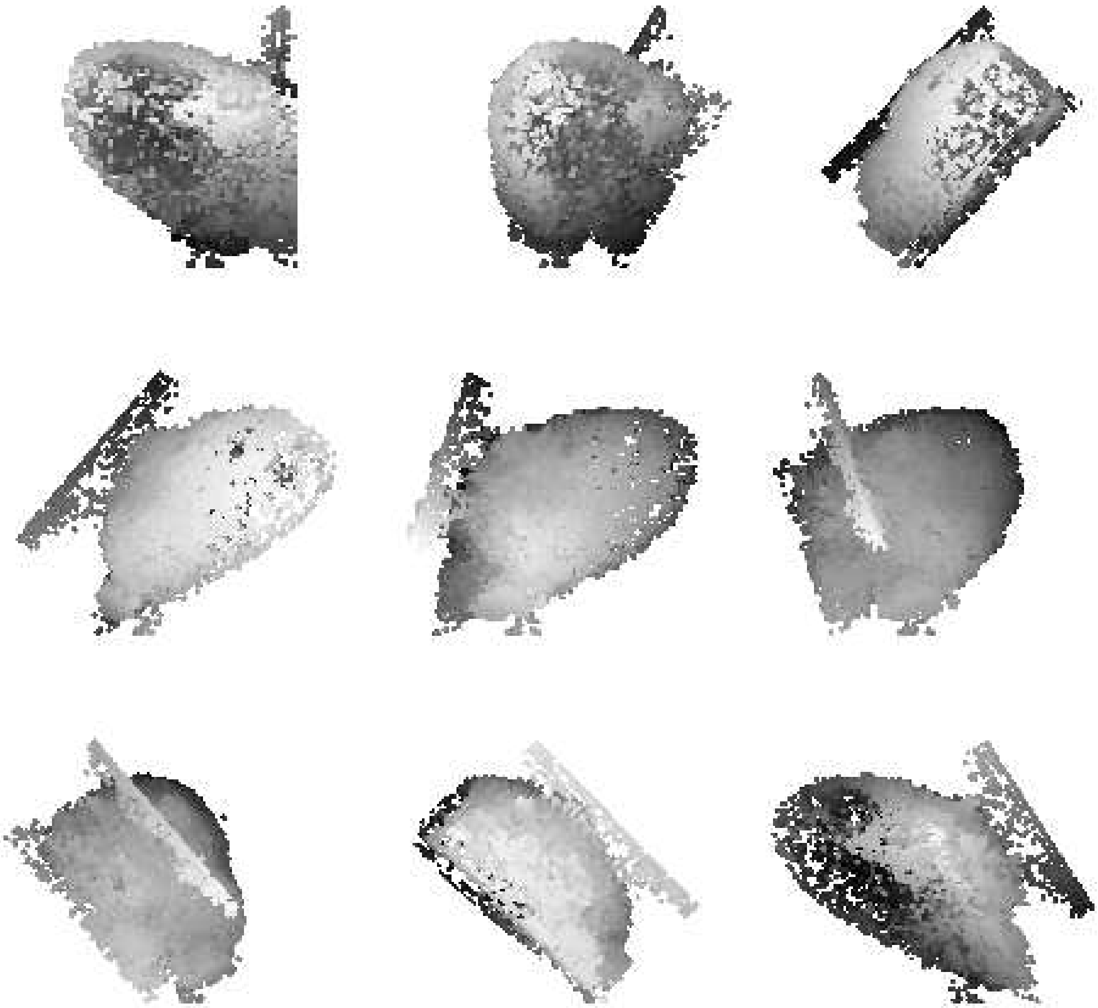


Figure 5.18: Bovine ovary animation of stroma (C2) using the modified 3D MTS algorithm. Perspectives shown for 40° rotational increments about the y axis.

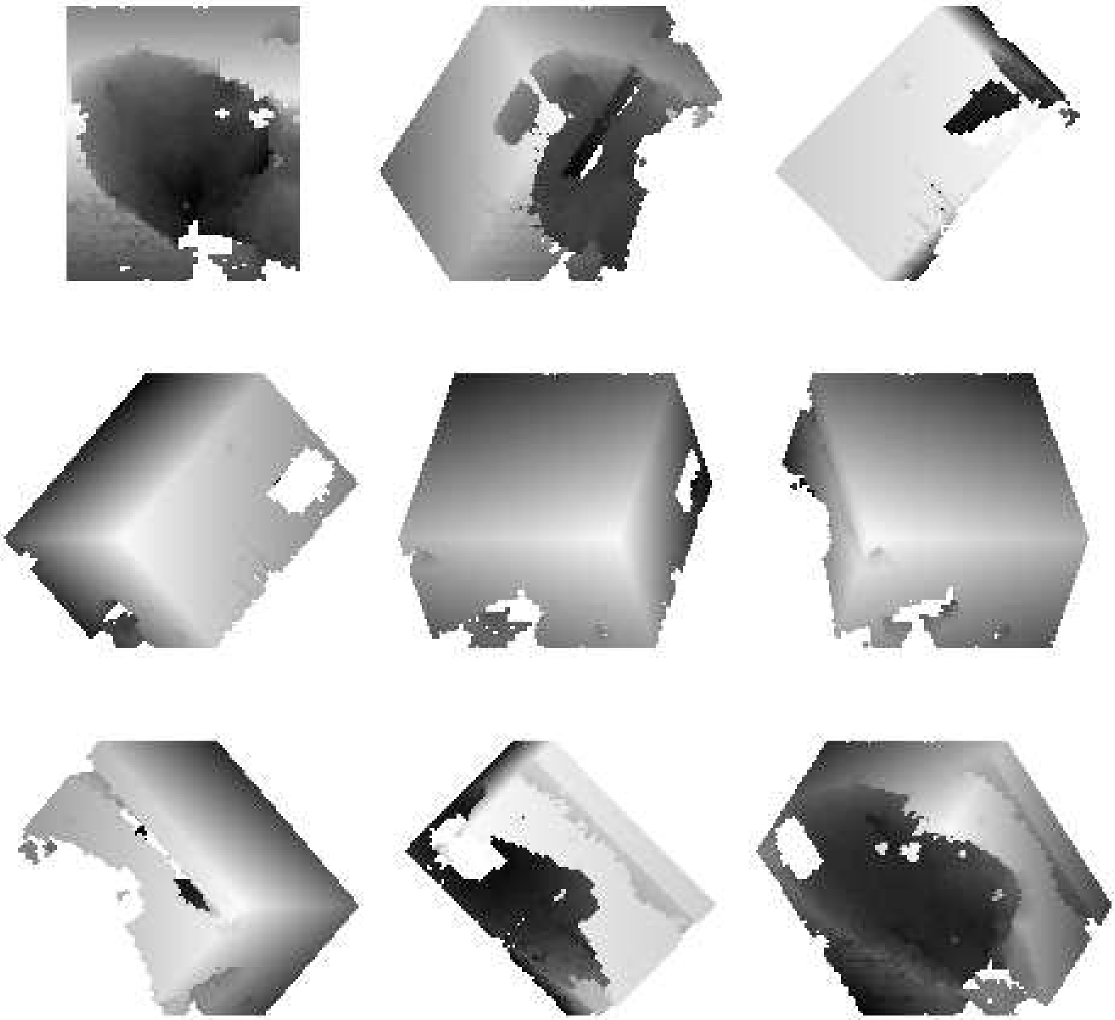


Figure 5.19: Bovine ovary animation of fluid (C3) using the modified 3D MTS algorithm. Perspectives shown for $40°$ rotational increments about the y axis.

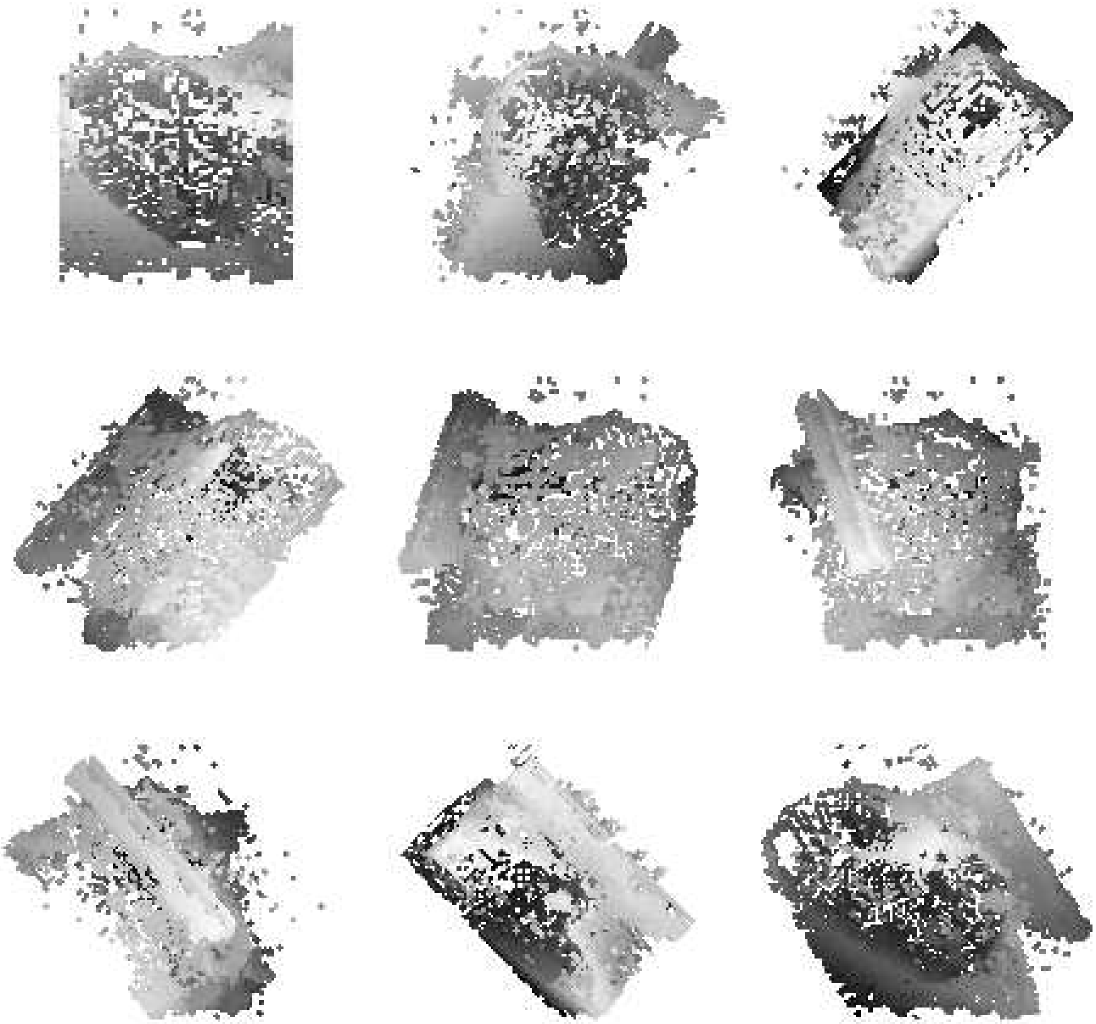


Figure 5.20: Bovine ovary animation of distance rejected points using the modified 3D MTS algorithm. Perspectives shown for 40° rotational increments about the y axis.

(such as texture measures) in performing the segmentation. One previous approach that possesses these features is the Multiresolution Texture Segmentation (MTS) algorithm [77, 81, 82, 78]. However, its inability to makes use of the third dimension in evaluating texture as well as its *ad hoc* similarity measure require that a new approach be determined. The 3D MTS algorithm was proposed as a new approach to be used in the 3D Segmentation module.

The 3D MTS algorithm is based on an octree structure which provides the ability to treat all three dimensions in the volume in an equal manner. The basic mechanism of splitting and merging blocks is controlled by a simulated annealing framework. The identity of blocks in the volume are determined using the Inck decision rule. An advantage in using this modified decision rule is that it is based on well established theory in statistical pattern recognition. Another advantage in using the Inck decision rule is that a direct relationship between classification and segmentation has been established. This provides the ability use a multiresolution classifier to determine the class parameters at each resolution. These parameters can then be used by the 3D MTS algorithm to determine the identity of unknown blocks in the volume using the Inck decision rule.

Numerous experiments were performed to evaluate the 3D MTS algorithm. For comparison purposes, the 2D MTS algorithm and a thresholding algorithm were also evaluated. The results obtained from the synthesised data suggest that both the original and 3D MTS algorithms are superior to the thresholding algorithm for segmenting noisy volume data. The results obtained from the ultrasound fetal data also support this conclusion.

The 3D MTS algorithm and a convolution method were used to segment bovine ultrasound data. For both methods, the texture was analysed using features based on the 3D cooccurrence matrix. The segmentation results from the convolution method showed a clear distinction among the corpus luteum, stroma and surrounding fluid. This is a significant result as the texture of the corpus luteum and stroma are visually very similar. When the 3D MTS algorithm was applied to the bovine data with $\epsilon = 1.0$ (no ambiguity reject) there was no clear distinction between the corpus luteum and the stroma. By setting $\epsilon = 0.05$ (thus, providing ambiguity reject) and refining the way small blocks are identified in the 3D MTS algorithm, the corpus luteum and stroma were clearly distinguished. The advantage of the modified version of the 3D MTS algorithm over the convolution method is that it required less time and computational effort to segment the bovine ultrasound volume data than the convolution method.

Chapter 6

3D Ultrasound Application

In order to demonstrate the functionality of the proposed system it is necessary to apply the system to a real problem. Throughout this thesis the system modules have been described and evaluated using 3D ultrasound data, yet, the entire process from start to end has not been demonstrated. It is the purpose of this chapter to demonstrate that the application of the proposed system to ultrasound volume data obtained from a 3D probe is possible.

6.1 Experiments

Two sets of ultrasound volume data are used for experimental evaluation. Both data sets, one a linear scan, the other a rotational scan, represent approximately the same "real life" volume. The objective of the experiments is to demonstrate that the proposed system can identify and model objects in ultrasound volume data. In determining the level of success/failure the system obtains, comparison of results from the linear and rotational data sets are continually made to determine if the 3D rotational ultrasound probe is a feasible method of image acquisition.

6.1.1 Data Acquisition

The state of the 3D ultrasound probe prototype placed a severe restriction on the availability of data for experimental evaluation. As a result only one data set from the probe was available. A linear scan of approximately the same volume was performed to provide a reference data set. All images were obtained with an ATL Mark 10 ultrasound device.

A 12 week old human embryo was immobilised in a tank filled with gel. The tank was placed on a translational stage so that a linear scan could be performed. The 3D rotational probe was fixed perpendicular to the fetus with the tip of the probe immersed in the gel. Acquisition of the linear scan was obtained by translating the tank (and fetus) underneath the probe from one end of the fetus to the other at 0.25mm increments. At each increment, a 2D slice was obtained resulting in a sequence of 163 images.

Acquisition of the rotational scan was obtained with the probe aligned over the middle of the fetus. The beam angle was chosen such that the entire fetus would be included in the sampled volume. The probe was rotated at 2° intervals through a 360° sampling range, thus sampling the entire volume twice. Only the first 91 (0° to 180°) images were used since, in the clinical environment, this is the maximum number of images that will be available.

Each 640×480, 8-bit ultrasound image was output to a frame grabber in a PC via an NTSC signal. It was then digitized and stored on disk for off-line processing. Note that in this configuration the ultrasound image undergoes a D/A (NTSC) and A/D (digitize) conversion. It is assumed that this does not introduce a significant amount of error.

In order to process the rotational volume data, it was first necessary to transform it from its cylindrical coordinate system representation to an isometric representation in the Cartesian coordinate system. A 350×185 pixel region surrounding the embryo was cropped from each image (same [x,y] offset for all images). Since no calibration data was available, the volume data was transformed using the mapping function $G(p')$ without any additional processing. The transformation resulted in a $350 \times 350 \times 185$ isometric volume. A 256×256 region surrounding the fetus was also cropped from each image in the linear scan volume data resulting in an $256 \times 256 \times 163$ isometric volume.

6.1.2 Results

Figure 6.1 shows the resulting segmented fetal volume obtained from the linear scan. A thresholding algorithm was used to distinguish fetal, from non-fetal tissue. Different perspectives of the segmented volume are shown as the volume is rotated through 360° at 20° intervals about the y axis. A fair amount of detail is present in the segmentation. There is also noise present in the segmented volume which is to due to echos received from the bottom of the tank.

Figure 6.2 shows the resulting segmented fetal volume obtained from the rotational scan. The same thresholding algorithm was used to distinguish fetal, from non-fetal tissue. Different perspectives of the segmented volume are shown as the volume is rotated about the x and y axes. It is obvious from these perspectives that a greater amount of noise is present in the rotational scan. The area around the feet of the fetus is especially corrupted with noise. Also note the circular pattern of the noise in this area. This indicates that the noise is present in the periphery of each of the images as the probe is rotated. Much of the noise can be attributed to echos from the tank floor. As well, some error is introduced in the sampling process.

Figure 6.1.2 illustrates the presence of sampling error in the resulting volume. There is a discontinuity across the middle horizon of the image. This discontinuity occurs precisely where the rotational scan starts and stops, indicating that the probe has a wobble about its rotational axis. The slice, selected perpendicular to the rotational scanning axis, shown in Figure 6.4 confirms the presence of the probe wobble.

Figure 6.5 shows the resulting segmented fetal volume obtained from the rotational scan using the 3D MTS algorithm. A multiresolution classification of the fetal data was performed by collecting 3D, representative samples of fetal and non-fetal tissue at resolutions $\beta_i = (2, 4, 8, 16, 32)$. Using the SFS algorithm and the Inck criterion function, it was found that a 1D feature space using only the mean (MEAN) could achieve a high probability of correct classification across all resolutions. With $\epsilon = 0.05$, the probability of correct classification averaged approximately 90% across

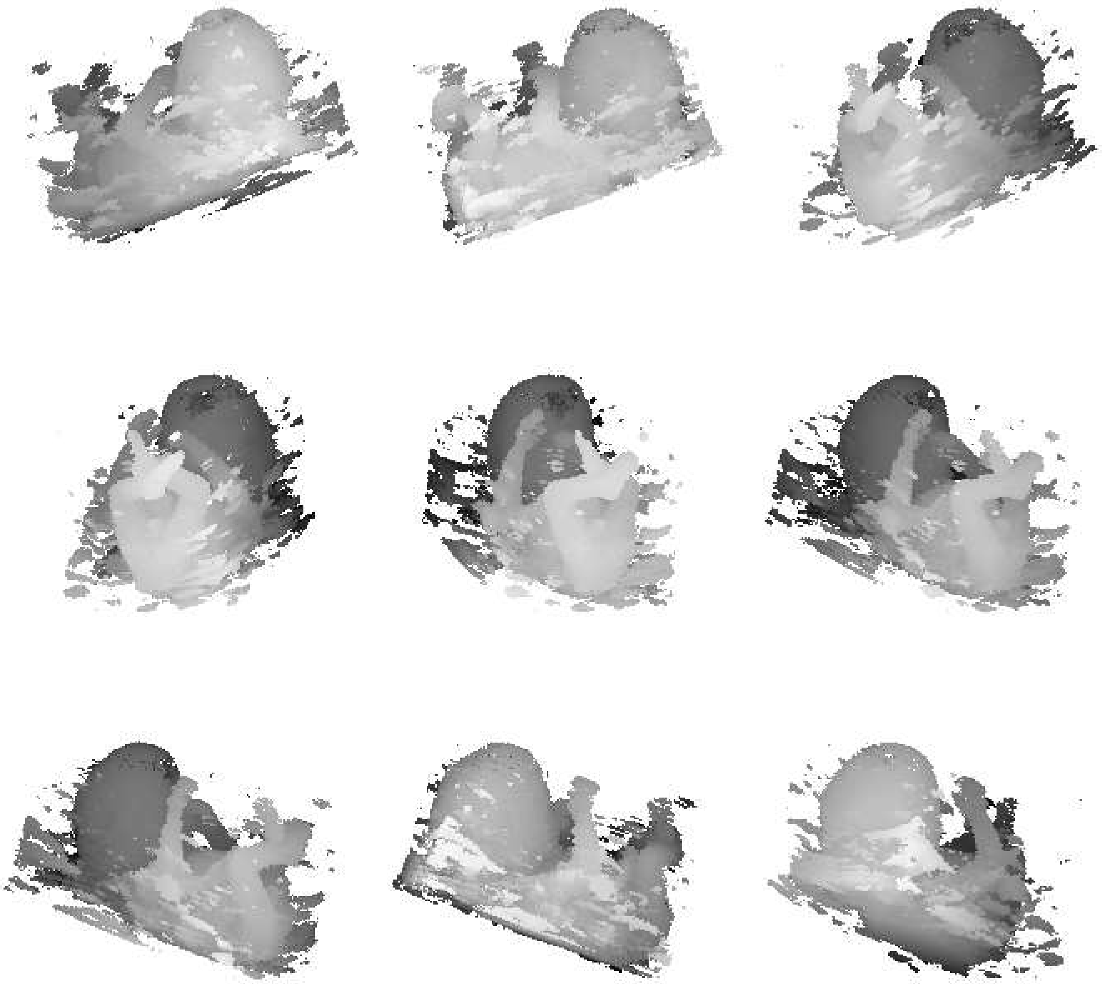


Figure 6.1: Animation of linear scanned fetal volume using thresholding. Perspectives shown for rotations about the y axis.

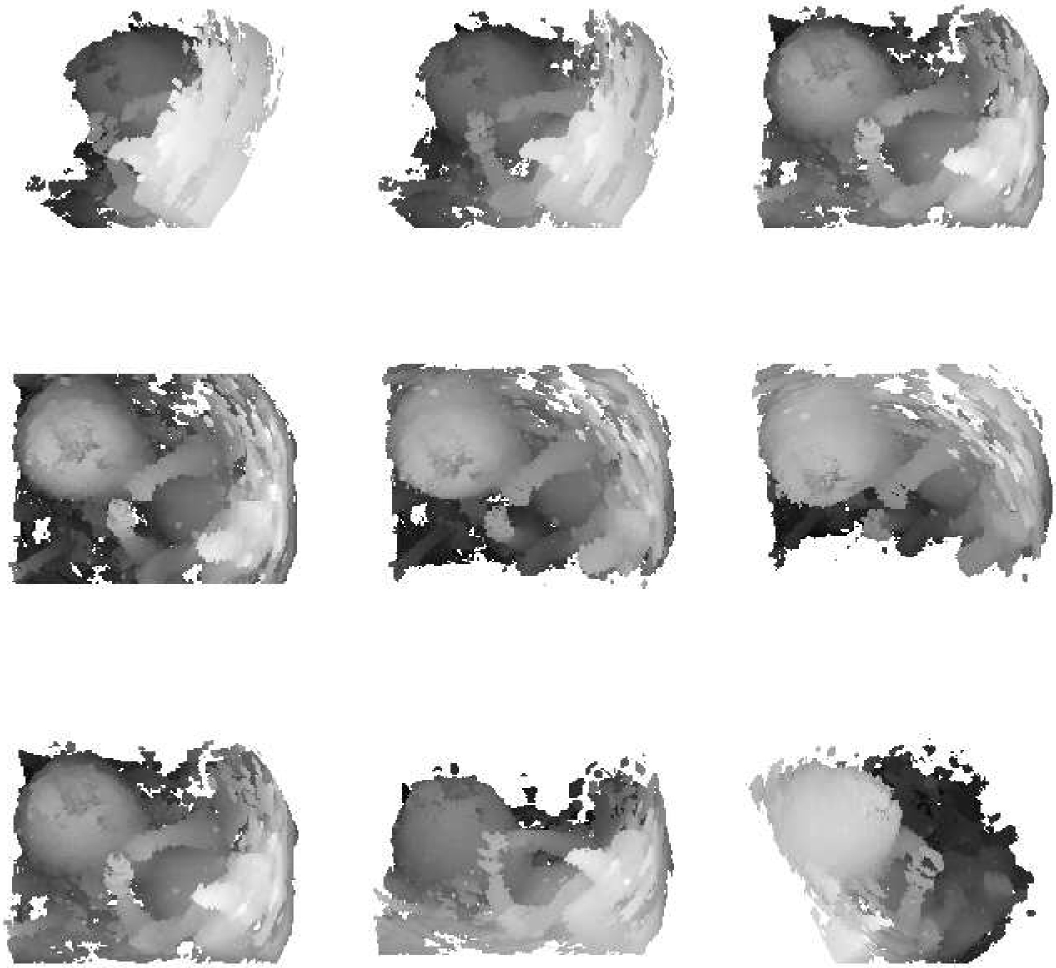


Figure 6.2: Animation of rotationally scanned fetal volume using thresholding. Perspectives shown for rotations about the x and y axis.

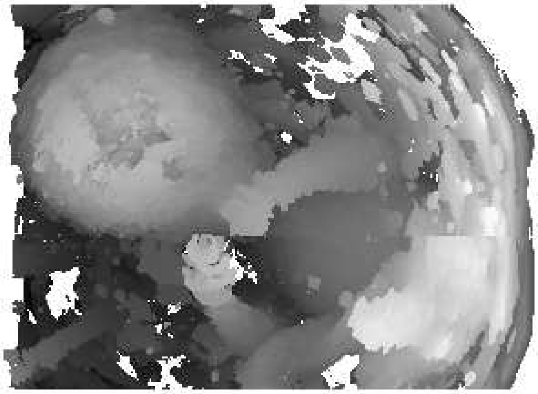


Figure 6.3: Rendering showing distortion of resulting segmentation due to probe wobble.

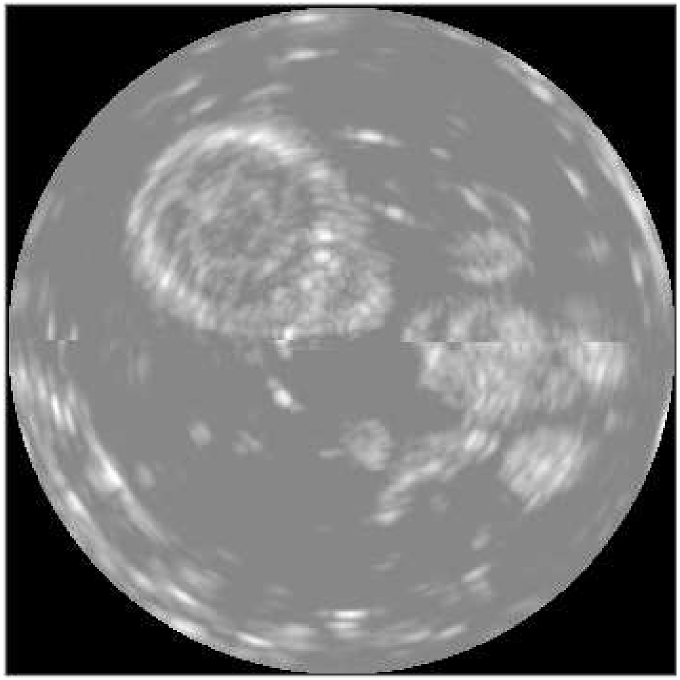


Figure 6.4: Perpendicular slice illustrating presence of probe wobble.

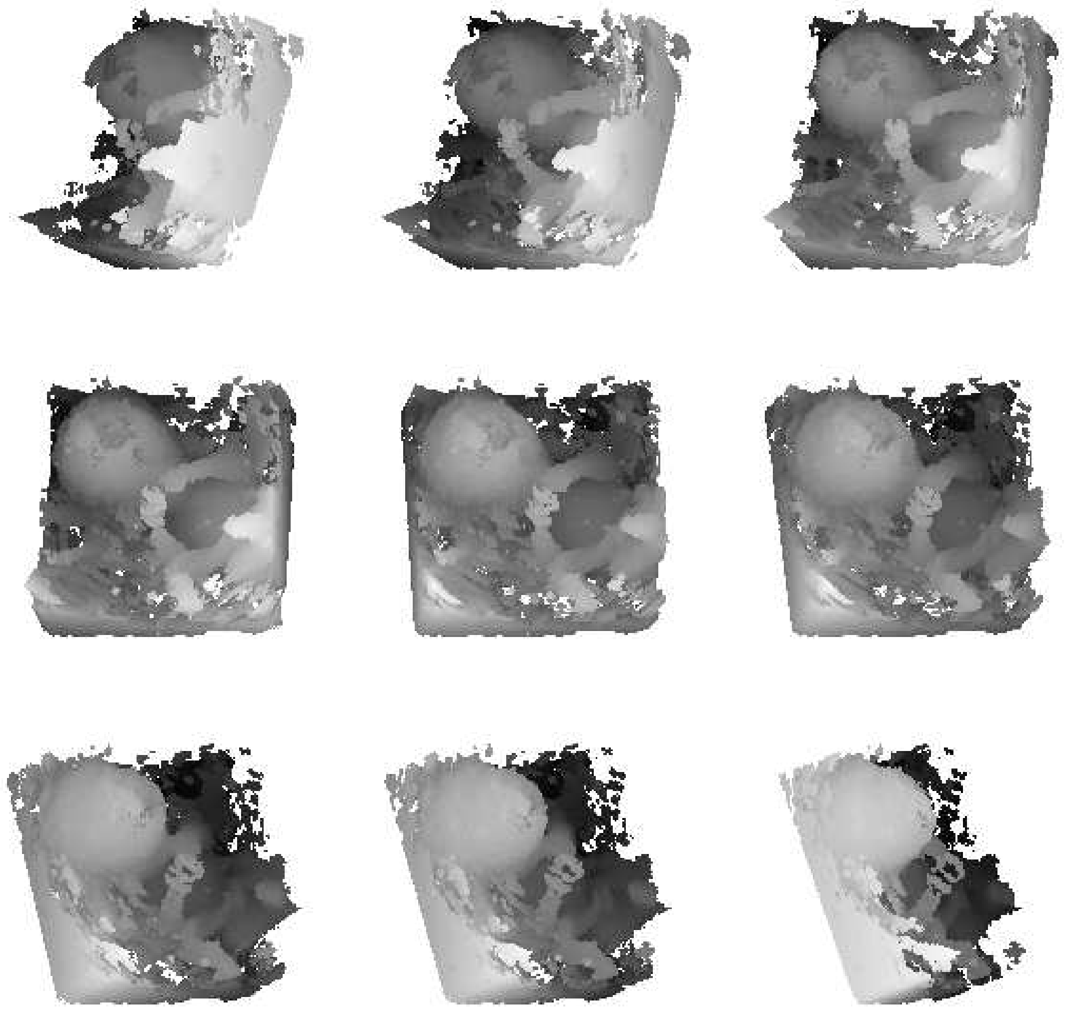


Figure 6.5: Animation of rotationally scanned fetal volume using the 3D MTS algorithm. Perspectives shown for rotations about the y axis.

all resolutions. The results from the multiresolution classification were used by the 3D MTS algorithm to segment the rotational volume. A volume of $256 \times 256 \times 128$ was cropped from the original $350 \times 350 \times 185$ volume and segmented. It can be seen from 6.5 that the results obtained by the 3D MTS algorithm are as good as those obtained using thresholding, as shown in Figure 6.2. A detailed analysis and classification of the noise in the fetal volume data would produce even better segmentation results, using the 3D MTS algorithm.

6.2 Summary

The purpose of this chapter was to demonstrate that the application of the proposed system to ultrasound volume data obtained from a 3D probe is possible. Since the limitations of the 3D ultrasound probe prototype placed severe restrictions on the availability of data, only one data set was available for experimental evaluation. Acquisition of the rotational scan was obtained with the 3D probe aligned over the middle of a 12 week fetus immobilised in a tank filled with gel. The volume data was transformed into the Cartesian coordinate system from which a $256 \times 256 \times 185$ isometric volume surrounding the fetus was subsequently processed. For comparison purposes, the same fetus was scanned linearly producing a $256 \times 256 \times 163$ point volume.

A thresholding algorithm was used to segment both linear and rotationally scanned data into fetal and non-fetal tissue. Different perspectives of the resulting segmentations were rendered and used as a basis for subjective evaluation of the quality of the segmentations. For both datasets, an adequate level of detail of the fetus were visible. However, a substantial amount of noise was produced from the bottom of the tank and interfered with visualisation. For the data from the 3D ultrasound probe, the noise prevented clear visualisation of the region around the feet of the fetus. A misalignment error was also present in the segmented volume suggesting that the probe had a wobble when the data was acquired.

The ultrasound volume data obtained from the 3D probe was also processed using the proposed system. First, the Tissue Characterisation module was used to build a multiresolution classification of the fetal data based on 3D, representative samples of fetal and non-fetal tissue. It was found that a 1D feature space based on the mean (MEAN) could used at all resolutions to accurately represent the fetal and non-fetal tissues present in the volume data. The results of the multiresolution classification in the Tissue Characterisation module where supplied to the 3D Segmentation module to be used in segmenting the fetal volume. Using the 3D MTS algorithm, the segmentation results were as good (subjectively) as those obtained using thresholding. A detailed analysis and classification of the noise in the fetal volume data would produce even better segmentation results using the 3D MTS algorithm.

Chapter 7

Conclusions

This chapter provides a summary of the thesis. The original contributions of this thesis are also presented. Finally, future work resulting from this thesis is discussed.

7.1 Summary

In Chapter 1 the main goal of the thesis and the tasks necessary to satisfy the goal were presented. An overview of the system was provided which is a realisation of the thesis goal. Five modules comprising the system were identified: the Tissue Characterisation module, the 3D Segmentation module, the Visualisation module, the Resampling module and the Ultrasound Imaging module. It was determined that standard techniques (see Section 2.4) could be employed in the Visualisation module to display the data. The publicly available software environment Khoros [147] provides enough functionality to meet the visualisation requirements of this research.

In Chapter 2 background material was provided. An overview of ultrasonic imaging was described in Section 2.1. This included a discussion of the main components of the ultrasound process as well as the characteristics of the resulting ultrasound images. A survey of the different models and techniques used to characterise ultrasound tissues was provided in Section 2.2. In Section 2.3, 2D and 3D segmentation methods were discussed as a way of identifying objects in 3D datasets. In Section 2.4 techniques for visualising both segmented and unprocessed data were described.

In Chapter 3 details of the Tissue Characterisation module were presented. The model is based on a statistical pattern recognition approach which provides a solid foundation on which tissue characterisation can be accomplished.

A new criterion function, the Inck (**Inc**omplete **k**nowledge) criterion function, was presented which approximates the error probability when M classes do not span the entire pattern space. This criterion is based on the probabilistic measures obtained from a modified version of Dubuisson and Masson's [26] statistical decision rule with reject. The error probability, or conversely the probability of correct classification, can be determined without having complete knowledge about the class distributions. The attractiveness of this criterion is that it is highly correlated with the statistical decision rule (with reject) which can be used by the classifier. Thus, the criterion provides a good indication of the classifier performance that can be expected with the reduced feature space.

It was also described how the robust statistical minimum volume ellipsoid estimator (MVE) proposed by Rousseeuw and Leroy [106] could be used to increase the classifier's tolerance to outlier data. The MVE estimator can theoretically provide an unbiased parameter estimation (in the case of a multivariate normal distribution, the mean and associated covariance matrix) describing a class distribution in the presence of up to 50% outlier data points.

A classifier design was presented which incorporates the new Inck criterion function. As well, it was described how the rejection classes of the modified statistical decision rule with reject could be used in a multiresolution classifier design to determine the class identity of patterns which can not be determined at a given resolution.

Numerous experiments were performed using both synthetic and real ultrasound volume data. The results obtained from the experiments showed that feature spaces based on the Inck criterion function resulted in higher probabilities of correct classification as compared with those obtained using probabilistic distance measures. Also, the decision to use MVE parameter estimation should be weighed against the expected presence of outliers. Finally, for smaller resolution classification, the probability of classification error, distance and ambiguity reject was high suggesting that segmentation of smaller details in the bovine ultrasound volume data, based on the classification data, will not be exceptional.

In Chapter 4 the 3D texture features used in the Tissue Characterisation module were described. Two classes of 3D texture features were described: cooccurrence matrix based features and statistical features. Since it is desired to calculate the cooccurrence matrix from volume data it is necessary to redefine the matrix. A new 3D definition of the matrix was presented which provides the ability to represent the texture in the volume in 3D, rather than in a collection of 2D matrices as has been done in the past. The definition also makes it possible to calculate measures from the cooccurrence matrix which are independent of the orientation of the texture in the volume. Traditionally, the cooccurrence matrix has always been dependent on the texture orientation.

Experiments were performed to analyse various characteristics of the 3D cooccurrence matrices. The results showed that there was not a significant difference in the information content of 3D matrices constructed with different parameter combinations of θ and r (d). There is an increase in information when G is increased. The results also showed that there is significantly more information present in the 3D cooccurrence matrix as compared to the 2D cooccurrence matrix. Finally, the results showed that there is a difference in the amount of time required to construct the 3D cooccurrence matrix as θ decreases or the radial distance, r, is used instead of the pixel distance, d.

In Chapter 5 details of the 3D Segmentation module were presented. It was determined that the segmentation approach used must have the ability to segment isometric volume data, be able to achieve good segmentation results when noise is present in the volume data and should use appropriate measures of homogeneity (such as texture measures) in performing the segmentation. The 3D MTS algorithm was proposed as a new approach to be used in the 3D Segmentation module.

The 3D MTS algorithm is based on an octree structure which provides the ability to treat all three dimensions in the volume in an equal manner. The basic mechanism of splitting and merging blocks is controlled by a simulated annealing framework.

The identity of blocks in the volume are determined using the Inck decision rule. An advantage in using this modified decision rule is that it is based on well established theory in statistical pattern recognition. Another advantage in using the Inck decision rule is that a direct relationship between classification and segmentation has been established. This provides the ability to use a multiresolution classifier to determine the class parameters at each resolution. These parameters can then be used by the 3D MTS algorithm to determine the identity of unknown blocks in the volume using the Inck decision rule. Numerous experiments were performed to evaluate the 3D MTS algorithm. For comparison purposes, the 2D MTS algorithm and a thresholding algorithm were also evaluated. The results obtained from the synthesised data suggest that both the original and 3D MTS algorithms are superior to the thresholding algorithm for segmenting noisy volume data. The results obtained from the ultrasound fetal data also support this conclusion.

The 3D MTS algorithm and a convolution method were used to segment bovine ultrasound volume data. For both methods, the texture was analysed using features based on the 3D cooccurrence matrix. The segmentation results from the convolution method showed a clear distinction among the corpus luteum, stroma and surrounding fluid. This is a significant result as the texture of the corpus luteum and stroma are visually very similar. When the 3D MTS algorithm was applied to the bovine data with $\epsilon = 1.0$ (no ambiguity reject), there was no clear distinction between the corpus luteum and the stroma. By setting $\epsilon = 0.05$ (thus, providing ambiguity reject) and refining the way small blocks are identified in the 3D MTS algorithm, the corpus luteum and stroma were clearly distinguished. The advantage of the modified version of the 3D MTS algorithm over the convolution method is that it required less time and computational effort to segment the bovine ultrasound volume data than the convolution method.

In Chapter 6 the application of the proposed system to ultrasound volume data obtained from a 3D probe was demonstrated. Since the limitations of the 3D ultrasound probe prototype placed severe restrictions on the availability of data, only one data set was available for experimental evaluation. Acquisition of the rotational scan was obtained with the 3D probe aligned over the middle of a 12 week fetus immobilised in a tank filled with gel. The volume data was transformed into the Cartesian coordinate system from which a $256 \times 256 \times 185$ isometric volume surrounding the fetus was subsequently processed. For comparison purposes, the same fetus was scanned linearly producing a $256 \times 256 \times 163$ point volume. A thresholding algorithm was used to segment both linear and rotationally scanned data into fetal and non-fetal tissue. Different perspectives of the resulting segmentations were rendered and used as a basis for subjective evaluation of the quality of the segmentations. For both datasets, an adequate level of detail of the fetus were visible. However, a substantial amount of noise was produced from the bottom of the tank and interfered with visualisation. For the data from the 3D ultrasound probe, the noise prevented clear visualisation of the region around the feet of the fetus. A misalignment error was also present in the segmented volume suggesting that the probe had a wobble when the data was acquired.

The ultrasound volume data obtained from the 3D probe was also processed using the proposed system. First, the Tissue Characterisation module was used to build a multiresolution classification of the fetal data based on 3D, representative samples of fetal and non-fetal tissue. It was found that a 1D feature space based on the mean (MEAN) could used at all resolutions to accurately represent the fetal and non-fetal tissues present in the volume data. The results of the multiresolution classification in the Tissue Characterisation module where supplied to the 3D Segmentation module to be used in segmenting the fetal volume. Using the 3D MTS algorithm, the segmentation results were as good (subjectively) as those obtained using thresholding. A detailed analysis and classification of the noise in the fetal volume data would produce even better segmentation results using the 3D MTS algorithm.

Details of the sampling process used in the Resampling module are presented in Appendix A. A simulation of the 3D probe was performed to quantise some of the errors that can result when performing a rotational volume scan and transforming it into an isometric 3D volume in the Cartesian coordinate system. A number of synthesised datasets were used in the simulations. The results of the simulations suggest that a probe wobble of up to $\phi, \gamma = 0.1°$ can be tolerated without introducing a significant amount of alignment error in the resulting volume.

7.2 Contributions

The first contribution of this thesis is the design, implementation, and testing of a software system for characterising, classifying, and segmenting volumetric ultrasound data. The system produces various outputs which can be visualised using standard rendering software. In the system, segmentation is viewed as way of "classifying on the fly". That is, the 3D Segmentation module uses the class information obtained by the Tissue Characterisation module to determine the identity of different regions within the volume. The emphasis of the segmentation algorithm is on determining the appropriate regions (location and size) within the volume that are to be "classified" using the *a priori* class knowledge.

The second contribution of this thesis is a unified multiresolution framework for segmentation and classification of (ultrasound) data. This contribution is significant as it ties together the topics of segmentation and classification which are traditionally treated independently. To the author's knowledge, this is one of the first attempts at unifying these topics in a common framework.

The third contribution is new statistical decision rule, Inck, for use as a criterion function in feature selection and a decision rule in classification of unknown patterns. Along with the Inck decision rule is contributed an algorithm for automating the selection of the parameters which determine how the rule is applied.

The fourth contribution of the thesis is the addition of a robust statistical parameter estimator, MVE, in the design of a classifier. Although the MVE algorithm has been used in prior applications [106], it has not been used in the context of "filtering out" outlier samples in training a classifier.

The fifth contribution of the thesis is the multiresolution design of a classifier. It is typical in the literature to neglect the importance of the dependence between the features used to classify patterns and the resolution at which the features are calculated. In previous research [77, 81, 82, 78], this dependence was demonstrated in segmenting ultrasound data of ovarian follicles. In this thesis, this concept was applied to the design of a classifier. This contribution is significant as it provides the ability to unify classifier design with a multiresolution segmentation algorithm, the 3D MTS algorithm.

The sixth contribution is a new 3D segmentation algorithm, the 3D MTS algorithm, which provides the ability to segment noisy, isometric, ultrasound volume data. The algorithm uses *a priori* knowledge, collected from a statistical pattern recognition classifier, in identifying homogeneous regions within the volume. This knowledge increases the likelihood of accurately segmenting noisy ultrasound data as compared with traditionally used segmentation approaches such as thresholding.

The seventh contribution of the thesis is the development of the 3D cooccurrence matrix. The matrix has two main advantages over the usual cooccurrence matrix. First, it provides the ability to represent the 3D texture in isometric volume data. Second, the definition of the 3D cooccurrence matrix makes it possible to calculate measures from the cooccurrence matrix which are independent of the orientation of the texture in the volume. Traditionally, the cooccurrence matrix has always been dependent on the texture orientation.

Finally, the eighth contribution of the thesis is that it makes possible the visualisation of ultrasound volume data obtained from a 3D ultrasound probe. As well, experimental evaluation and simulation of a 3D ultrasound probe are performed.

7.3 Future Work

The research presented in this thesis provides ample opportunity for future work in different areas. First, it would be desirable to obtain substantially more test data from the 3D ultrasound probe. This would provide the ability for extensive evaluation of the proposed system on the intended application of volumetric ultrasound data. Volumetric data from other imaging modalities (e.g. MRI, PET) should also be evaluated to verify the general applicability of the proposed system.

Further examination of 3D texture features would certainly benefit the system's ability to represent and identify tissues in ultrasound volume data. Most notably, the use of model-based features seem promising. However, their use depends on the availability of the ultrasound RF data. Given the reluctance of today's vendors to disclose RF data, the use of model-based features is likely not possible in an academic environment.

One way to improve the segmentation results achieved by the 3D MTS algorithm is to increase the *a priori* knowledge used when segmenting ultrasound volume data. In addition to the tissue characterisation of each class, information based on the anatomy of the object being segmented can be used. Sonka *et al.* [117] use various heuristics based on the biology of coronary arteries to help guide a border detection algorithm. The intravascular ultrasound data of coronary arteries exhibits characteristic artifacts for biologically significant structures within the arterial wall. By studying the these artifacts and knowing the anatomical relationships of the various structures in the coronary artery heuristics are formulated to guide the identification of the structures in the artery. Parvin *et al.* [91] use a model-guided approach in segmenting volumetric structures from MR images of the brain. The model is represented by a set of constraints which are based on the anatomical features of the brain. These constraints help guide the segmentation by ruling out various region configurations which are biologically implausible.

Another area that requires future work is in selecting representative, 3D samples of classes for use in training by the Tissue Characterisation module. It is necessary to have expert knowledge of the classes in order to obtain the class samples. Further, since 3D samples are desired, it is necessary to have some means of visualising the raw ultrasound data in 3D so that samples can be selected. Perhaps the best way to achieve this is to control the imaging environment so that samples of only one tissue type at a time are obtained. This would alleviate the need for an expert to evaluate the resulting volume data to select appropriate samples.

The direct calculation of the 3D cooccurrence measures as presented in this thesis are computationally expensive. Peckinpaugh [92] suggests some optimisations which provide a substantial decrease in the computational complexity of the measures. These optimisations could be implemented to provide a significant decrease in the time required to segment volume data.

Normalisation of the volumetric ultrasound data is essential in achieving consistent results among datasets. Such parameters as the zoom factor, time gain compensation, and other machine dependent factors must be set in such a way that consistency is obtained. In this thesis, normalisation of the ultrasound data was difficult to obtain as this fell beyond the scope of the research. Future versions of the Ultrasound Imaging module of the proposed system must provide the means for normalising the data which is subsequently processed by the remaining modules in the system.

References

[1] D. Abramson. A very high speed architecture for simulated annealing. *Computer*, pages 27–36, May 1992.

[2] S. Aeberhard, D. Coomans, and L. De Vel. Comparative analysis of statistical pattern recognition methods in high dimensional settings. *Pattern Recognition*, 27(8):1065–1077, 1994.

[3] H.J. Antonisse. Image segmentation in pyramids. *Computer Graphics, and Image Processing*, 19:367–383, 1982.

[4] S. Arya, D. DeMenthon, P. Meer, and L. Davis. Textural analysis of range images. *Pattern Recognition Letters*, 12:619–626, October 1991.

[5] M. Bajura, H. Fuchs, and R. Ohbuchi. Merging virtual objects with the real world: Seeing ultrasound imagery within the patient. In *Proceedings of SIGGRAPH '92*, volume 26(2), pages 203–210, Chicago, Illinois, July 26-31 1992.

[6] S. Baker and S. Nayar. A theory of pattern rejection. Technical Report CUCS-013-95, Department of Computer Science, Columbia University, New York, NY 10027, June 1995.

[7] F. Blayo, Y. Cheneval, A. Guerin-Dugue, R. Chentouf, C. Aviles-Cruz, J. Madrenas, M. Moreno, and J. Voz. Deliverable r3-b4-p task b4: Benchmarks. Technical Report ESPRIT Basic Research Project Number 6891, Enhanced Learning for Evolutive Neural Architecture, June 1995.

[8] C. Bouman and B. Liu. Multiple resolution segmentation of textured images. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 13(2):99–113, 1991.

[9] G. Braccini and O. Salvetti. Morphometric analysis of sonographic images by spatial geometric modeling. *Computerized Medical Imaging and Graphics*, 16(2):93–108, 1992.

[10] J. D. Bronzino. *Biomedical Engineering and Instrumentation Basic Concepts and Applications.* PWS Publishers, Bonston, Massachusetts, 1986.

[11] C. Burckhardt. Speckle in b-mode scans. *IEEE Transactions on Sonics and Ultrasonics*, SU-25(1):1–6, January 1978.

[12] V. Cantoni and S. Levialdi. *Pyramidal Systems for Computer Vision*, pages 215–288. Springer-Verlag, 1986.

[13] D.A. Carpenter, G. Kossoff, and K.A. Griffiths. Correction of distortion in us images caused by subcutaneous tissues: Results in tissue phantoms and human subjects. *Radiology*, 195:563–567, 1995.

[14] P.L. Carson, C.R. Meyer, E.H. Chiang, R.G. Faix, and T.I. Marks. Ultrasound attenuation coefficient in the fetal liver as a function of gestational age. *Ultrasound in Medicine and Biology*, 16(4):399–407, 1990.

[15] C. Chang and S. Chatterjee. A hybrid approach toward model-based texture segmentation. *Pattern Recognition*, 25(5):519–531, 1992.

[16] C. Chen, J.S. Daponte, and M.D. Fox. Fractal feature analysis and classification in medical imaging. *IEEE Transactions on Medical Imaging*, 8(2):133–142, June 1989.

[17] D.S. Chen and B.G. Schunck. *Robust statistical methods for building classification procedures*. Artificial Intelligence Laboratory, Department of EE and CS, University of Michigan - Ann Arbor, April 1990.

[18] L. Chen, G.T. Herman, R.A. Reynolds, and J.K. Udupa. Surface shading in the cuberville environment. *IEEE Computer Graphics and Applications*, pages 33–43, December 1985.

[19] Y.Q. Chen, M.S. Nixon, and D.W. Thomas. Statistical geometrical features for texture classification. *Pattern Recognition*, 28(4):537–552, 1995.

[20] C.K. Chow. An optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, IT-16:41–46, 1970.

[21] D. A. Christensen. *Ultrasonic Bioinstrumentation*. John Wiley & Sons, New York, New York, 1988.

[22] F.S. Cohen. Maximum likelihood unsupervised textured image segmentation. *CVGIP: Graphical Models and Image Processing*, 54(3):239–251, 1992.

[23] F.S. Cohen and D.B. Cooper. Simple parallel hierarchical and relaxation algorithms for segmenting noncausal markovian random fields. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 9(2):195–219, March 1987.

[24] G. Corneloup, J. Moysan, and I. Magnin. Bscan image segmentation by thresholding using cooccurrence matrix analysis. *Pattern Recognition*, 29(2):281–296, 1996.

[25] R.A. Devijver and J. Kittler. *Pattern Recognition: A Statistical Approach.* Prentice-Hall International Inc., 1982.

[26] B. Dubuisson and M. Masson. A statistical decision rule with incomplete knowledge about classes. *Pattern Recognition*, 26(1):155–165, 1993.

[27] I.M. Elfadel and R.W. Picard. Gibbs random fields, cooccurrences, and texture modeling. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 16(1):24–37, 1994.

[28] E.J. Feleppa, F.L. Lizzi, D.J. Coleman, and M.M. Yaremko. Diagnostic spectrum analysis in opthalmology: a physical perspective. *Ultrasound in Medicine and Biology*, 12(8):623–631, 1986.

[29] E.J. Feleppa and M.M. Yaremko. Ultrasonic tissue characterization for diagnosis and monitoring. *IEEE Engineering in Medicine and Biology Magazine*, 6:18–26, December 1987.

[30] J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes. *Computer Graphics Principles and Practice.* Addison-Wesley, 2nd edition, 1990.

[31] N. Friedland and D. Adam. Automatic ventricular cavity boundary detection from sequential ultrasound images using simulated annealing. *IEEE Transactions on Medical Imaging*, 8(4):344–353, December 1989.

[32] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press Inc., 1990.

[33] K. Fukunaga and R. Hayes. Effects of sample size in classifier design. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 11(8):873–885, August 1989.

[34] D. Geman, S. Geman, C. Graffigne, and P. Dong. Boundary detection by constrained optimization. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 12(7):609–628, July 1990.

[35] R. Gilchrist. A multi-mode 3D renderer for clinical and scientific visualization. Technical report, ISG Technologies, 3030 Orlando Drive, Missisauga, Ontario, Canada, L4V 1S8, 1993.

[36] O.H. Gilja, N. Thune, K. Matre, T. Hausken, S. Odegaard, and A. Berstad. In vitro evaluation of three-dimensional ultrasonography in volume estimation of abdominal organs. *Ultrasound in Medicine and Biology*, 20(2):157–165, 1994.

[37] C.C. Gotlieb and H.E. Kreyszig. Texture descriptors based on co-occurrence matrices. *Computer Vision, Graphics, and Image Processing*, 51(1):70–86, July 1990.

[38] A.D. Gross and A. Rosenfeld. Multiresolution object detection and delineation. *Computer Vision, Graphics, and Image Processing*, 39(1):102–115, 1987.

[39] R. Haralick, K. Shanmugam, and I. Dinstein. Textural features for images classification. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-3(6):610–621, 1973.

[40] A. Herment, J.P. Guglielmi, P. Dumee, P. Peronneau, and P. Delouche. Limitations of ultrasound imaging and image restoration. *Ultrasonics*, 25:267–273, 1987.

[41] T.H. Hong and A. Rosenfeld. Compact region extraction using weighted pixel linking in a pyramid. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 6(2):222–229, March 1984.

[42] M.F. Insana, R.F. Wagner, B.S. Garra, R. Momenan, and T.H. Shawker. Pattern recognition methods for optimizing multivariate tissue signatures in diagnostic ultrasound. *Ultrasonic Imaging*, 8:165–180, 1986.

[43] T.R. Jackson and M.B. Merickel. Applications of hierarchical image segmentation techniques: Aorta segmentation. *Computerized Medical Imaging and Graphics*, 16(5):333–343, 1992.

[44] M. James. *Classification Algorithms.* William Collins Sons & Co. Ltd., 1985.

[45] M.E. Jernigan and F. D'Astous. Entropy-based texture analysis in the spatial frequency domain. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 6(2):237–243, March 1984.

[46] M. Joliot and B.M. Mazoyer. Three-dimensional segmentation and interpolation of magnetic resonance brain images. *IEEE Transactions on Medical Imaging*, 12(2):269–277, June 1993.

[47] J.W. Klinger Jr., C.L. Vaughan, T.D. Fraker Jr., and L.T. Andrews. Segmentation of echocardiographic images using mathematical morphology. *IEEE Transactions on Biomedical Engineering*, 35(11):925–934, November 1988.

[48] M. Kallergi, K. Woods, L.P. Clarke, W. Qian, and R.A. Clark. Image segmentation in digital mammography: Comparison of local thresholding and region growing algorithms. *Computerized Medical Imaging and Graphics*, 16(5):323–331, 1992.

[49] M. Karaman, A. Kutay, and H. Bozdagi. An adaptive speckle suppression filter for medical ultrasonic imaging. *IEEE Transactions on Medical Imaging*, 14(2):283–292, June 1995.

[50] C. Kervrann and F. Heitz. A markov random field model-based approach to unsupervised texture segmentation using local and global spatial statistics. *IEEE Transactions On Image Processing*, 4(6):856–862, June 1995.

[51] T. Kling, K. Shung, and G Thieme. Reverberation reduction in ultrasonic b-mode images via dual frequency image subtraction. *IEEE Transactions on Medical Imaging*, 12(4):792–802, 1993.

[52] M. Kreutz, B. Volpel, and H. Janben. Scale-invariant image recognition based on higher-order autocorrelation features. *Pattern Recognition*, 29(1):19–26, 1996.

[53] L. Landini and L. Verrazzani. Spectral characterization of tissues microstructure by ultrasounds: a stochastic approach. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, 37(5):448–456, September 1990.

[54] T. Lei and W. Sewchand. Statistical approach to x-ray ct imaging and its applications in image analysis-part 2: A new stochastic model-based image segmentation technique for x-ray ct image. *IEEE Transactions on Medical Imaging*, 11(1):62–69, March 1992.

[55] M. Levoy. Display of surfaces from volume data. *IEEE Computer Graphics and Applications*, 8(3):29–37, May 1988.

[56] M. Levoy. Volume rendering by adaptive refinement. *The Visual Computer*, 6:2–7, 1990.

[57] S.Z. Li. Robustizing robust m-estimation using deterministic annealing. *Pattern Recognition*, 29(1):159–166, 1996.

[58] L.M. Lifshitz and S.M. Pizer. A multiresolution hierarchical approach to image segmentation based on intensity extrema. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 12(6):529–540, June 1990.

[59] S. Liou and R.C. Jain. An approach to three-dimensional image segmentation. *CVGIP: Image Understanding*, 53(3):237–252, May 1991.

[60] D. Liu and M. Saito. A new method for estimating the acoustic attenuation coefficient of tissue from reflected ultrasonic signals. *IEEE Transactions on Medical Imaging*, 8(1):107–110, March 1989.

[61] J. Liu and Y.H. Yang. Multiresolution color image segmentation. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 16(7):689–700, 1994.

[62] S. Liu and M.E. Jernigan. Texture analysis and discrimination in additive noise. *Computer Vision, Graphics, and Image Processing*, 49(1):52–67, January 1990.

[63] F. Lizzi and M. Laviola. Tissue signature characterization utilizing frequency domain analysis. In *IEEE Ultrasonics Symposium Proceedings*, pages 714–719, New York, November 1976.

[64] F.L. Lizzi, M. Greenebaum, E.J. Feleppa, and M. Elbaum. Theoretical framework for spectrum analysis in ultrasonic tissue characterization. *Journal of the Acoustical Society of America*, 73(4):1366–1373, April 1983.

[65] F.L. Lizzi, D.L. King, M.C. Rorke, J. Hui, M. Ostromogilsky, M.M. Yaremko, E.J. Feleppa, and P. Wai. Comparison of theoretical scattering results and ultrasonic data from clinical liver examinations. *Ultrasound in Medicine and Biology*, 14(5):377–385, 1988.

[66] F.L. Lizzi, M. Ostromogilsky, E.J. Feleppa, M.C. Rorke, and M.M. Yaremko. Relationship of ultrasonic spectral parameters to features of tissue microstructure. *IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control*, UFFC-33(3):319–329, May 1986.

[67] B.C. Lovell and A.P. Bradley. The multiscale classifier. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 18(2):124–137, February 1996.

[68] S.W. Lu and H. Xu. Textured image segmentation using autoregressive model and artificial neural network. *Pattern Recognition*, 28(12):1807–1817, 1995.

[69] T. Malzbender. Fourier volume analysis. *ACM Transactions on Graphics*, 12(3):233–250, July 1993.

[70] J. Mao and A.K. Jain. Texture classification and segmentation using multiresolution simultaneous autoregressive models. *Pattern Recognition*, 25(2):173–188, 1992.

[71] G.F. McLean. Vector quantization for texture classification. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(3):637–649, 1993.

[72] M. McNitt-Gray, H. Huang, and J. Sayre. Feature selection in the pattern classification problem digital chest radiograph segmentation. *IEEE Transactions on Medical Imaging*, 14(3):537–547, September 1995.

[73] P. Meer. Stochastic image pyramids. *Computer Vision, Graphics, and Image Processing*, 45:269–294, 1989.

[74] R. Momenan, M.H. Loew, M.F. Insana, R.F. Wagner, and B.S. Garra. Application of pattern recognition techniques in ultrasound tissue characterization. *Proceedings - International Conference on Pattern Recognition*, 1:608–612, 1990.

[75] R. Momenan, R Wagner, B. Garra, M. Loew, and M. Insana. Image staining and differential diagnosis of ultrasound scans based on the mahalanobis distance. *IEEE Transactions on Medical Imaging*, 13(1):37–47, 1994.

[76] P. Morse and K. Ingard. *Theoretical Acoustics.* Mcgraw Hill, New York, New York, 1968.

[77] R.E. Muzzolini. Segmentation of ultrasound images using texture discrimination. Master's thesis, Department of Computer Science, University of Saskatchewan, Saskatoon, Saskatchewan, Canada, S7N 0W0, October 1991.

[78] R.E. Muzzolini, Y.H. Yang, and R.A. Pierson. A framework for the evaluation of feature detectors with application to diagnostic ultrasound images (an abridged version of [82]). In *Conference Record of the 1991 IEEE Nuclear Science Symposium and Medical Imaging Conference*, pages 2201–2204, Santa Fe, NM, November 1991.

[79] R.E. Muzzolini, Y.H. Yang, and R.A. Pierson. A multiresolution texture segmentation approach with application to diagnostic ultrasound images (an abridged version of [80]). In *IEEE Medical Imaging Conference*, pages 2205–2209, Santa Fe, NM, November 1991.

[80] R.E. Muzzolini, Y.H. Yang, and R.A. Pierson. Multiresolution texture segmentation with application to diagnostic ultrasound images. *IEEE Transactions on Medical Imaging*, 12(1):108–123, March 1993.

[81] R.E. Muzzolini, Y.H. Yang, and R.A. Pierson. Multidimensional texture characterization. Technical report, University of Saskatchewan, 1994.

[82] R.E. Muzzolini, Y.H. Yang, and R.A. Pierson. Texture characterization using robust statistics. *Pattern Recognition*, 27(1):119–134, 1994.

[83] T.R. Nelson and T.T. Elvins. Visualization of 3D ultrasound data. *IEEE Computer Graphics and Applications*, pages 50–57, November 1993.

[84] T.R. Nelson and D.H. Pretorius. Three-dimensional ultrasound of fetal surface features. *Ultrasound in Obstetrics and Gynecology*, 2:166–174, 1992.

[85] D.R. Ney, E.K. Fishman, D. Magid, and R.A. Drebin. Volumetric rendering. *IEEE Computer Graphics and Applications*, pages 24–32, March 1990.

[86] P.P. Ohanian and R.C. Dubes. Performance evaluation for four classes of textural features. *Pattern Recognition*, 25(8):819–833, 1992.

[87] R. Ohbuchi, D. Chen, and H. Fuchs. Incremental volume reconstruction and rendering for 3D ultrasound imaging. *SPIE*, 1808:312–323, 1992.

[88] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1):51–59, 1996.

[89] J. Ophir and N.F. Macklad. Digital scan converters in diagnostic ultrasound imaging. *Proc IEEE*, 67:654–664, 1979.

[90] N.R. Pal and S.K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277–1294, 1993.

[91] B. Parvin, W. Johnston, and D. Robertson. Visualization of volumetric structures from mr images of the brain. In *AAAI Conference on Application of Computer Vision to Medical Imaging*, Palo Alto, CA, March 21-23 1994.

[92] S.H. Peckinpaugh. An improved method for computing gray-level cooccurrence based texture measures. *CVGIP: Graphical Models and Image Processing*, 53(6):574–580, 1991.

[93] A. Pierce. *Acoustics: An introduction to its physical principles and applications*. McGraw Hill, New York, New York, 1981.

[94] M. Pietikainen and A. Rosenfeld. Image segmentation by texture using pyramid node linking. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(12):822–825, December 1981.

[95] R. Powis and W. Powis. *A Thinker's Guide to Ultrasonic Imaging*. Urban & Schwartzenberg Inc., Baltimore, Maryland, 1984.

[96] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C*, pages 343–352. Cambridge University Press, 1988.

[97] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, second edition, 1992.

[98] T. Rauber, M. Barata, and A. Steiger-Garcao. A toolbox for analysis and visualization of sensor data in supervision. Technical report, Universidade Nova de Lisboa - Faculdade de Ciencias e Tecnologia, Deparamento de Informatica - Intelligent Robotics Group, 2825 Monte de Caparica, Portugal.

[99] T.W. Rauber. Tooldiag - a pattern recognition toolbox. Technical report, Universidade Nova de Lisboa - Department of Electrical Engineering & UNINOVA - Center for Intelligent Robotics, Quinta da Torre, 2825 Monte da Caparica, Portugal, Copyright (C) 1992,1993,1994.

[100] T.R. Reed and J.M. Du Buf. A review of recent texture segmentation and feature extraction techniques. *CVGIP: Image Understanding*, 57(3):359–372, May 1993.

[101] J. M. Reid. Doppler ultrasound. *IEEE Engineering in Medicine and Biology Magazine*, 6(4):14–17, December 1987.

[102] R.A. Robb and C. Barillot. Interactive display and analysis of 3-d medical images. *IEEE Transactions on Medical Imaging*, 8(3):217–226, September 1989.

[103] R. Romijn, J. Thijssen, and G. Van Beuningen. Estimation of scatterer size form backscattered ultrasound: a simulation study. *IEEE Transactions on*

Ultrasonics, Ferroelectrics and Frequency Control, 36(6):593–606, November 1989.

[104] R. Romijn, J. Thijssen, B. Oosterveld, and A. Verbeek. Ultrasonic differentiation of intraocular melanomas: parameters and estimation methods. *Ultrasonic Imaging*, 13:27–55, 1991.

[105] K. Rose, E. Gurewitz, and G. Fox. Constrained clustering as an optimization method. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 15(8):785–794, August 1993.

[106] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection.* John Wiley and Sons, 1987.

[107] G. Sakas, L. Schreyer, and M. Grimm. Preprocessing and volume rendering of 3d ultrasonic data. *IEEE Computer Graphics and Applications, Special Issue*, 15(4):47–54, July 1995.

[108] G. Sakas and S. Walter. Extracting surfaces from fuzzy 3d-ultrasound data. In *Computer Graphics Proceedings*, pages 465–474, Los Angeles, California, August 6-11 1995. SIGGRAPH '95, ACM SIGGRAPH.

[109] E. Salari and Z. Ling. Texture segmentation using hierarchical wavelet decomposition. *Pattern Recognition*, 28(12):1819–1824, 1995.

[110] H. Samet. Neighbor finding techniques for images represented by quadtrees. *Computer Graphics, and Image Processing*, 18:37–57, 1982.

[111] H. Samet and R.E. Webber. Hierarchical data structures and algorithms for computer graphics (part 1). *IEEE Transactions On Computer Graphics and Applications*, pages 48–68, May 1988.

[112] H. Samet and R.E. Webber. Hierarchical data structures and algorithms for computer graphics (part 2). *IEEE Transactions On Computer Graphics and Applications*, pages 59–75, July 1988.

[113] R.F. Sarraga. Computer modeling of surfaces with arbitrary shapes. *IEEE Computer Graphics and Applications*, pages 67–77, March 1990.

[114] P.M. Shankar, J.M. Reid, H. Ortega, C.W. Piccoli, and B.B. Goldberg. Use of non-rayleigh statistics for the identification of tumors in ultrasonic b-scans of the breast. *IEEE Transactions on Medical Imaging*, 12(4):687–692, 1993.

[115] H.C. Shen, C.Y. Bie, and D.K. Chiu. A texture-based distance measure for classification. *Pattern Recognition*, 26(9):1429–1437, 1993.

[116] A. Shmulewitz. Ultrasonic multifeature maps of liver based on an amplitude loss technique and a conventional b-scan. *IEEE Transactions on Biomedical Engineering*, 39(5):445–449, May 1992.

[117] M. Sonka, X. Zhang, M. Siebes, M.S. Bissing, S.C. DeJong, S.M. Collins, and C.R. McKay. Segmentation of intravascular ultrasound images: A knowledge-based approach. *IEEE Transactions on Medical Imaging*, 14(4):719–732, December 1995.

[118] M. Spann and A.E. Grace. Adaptive segmentatin of noisy and textured images. *Pattern Recognition*, 27(12):1717–1733, 1994.

[119] E. Steen and B. Olstad. Volume rendering of 3d medical ultrasound data using direct feature mapping. *IEEE Transactions on Medical Imaging*, 13(3):517–525, September 1994.

[120] C.V. Stewart. Minpran: A new robust estimator for computer vision. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 17(10), October 1995.

[121] J. Strand and T. Taxt. Local frequency features for texture classification. *Pattern Recognition*, 27(10):1397–1406, 1994.

[122] K.C. Strasters and J.J. Gerbrands. Three-dimensional image segmentation using a split, merge and group approach. *Pattern Recognition Letters*, 12:307–325, May 1991.

[123] A. Teuner, O. Pichler, and B.J. Hosticka. Unsupervised texture segmentation of images using tuned matched gabor filters. *IEEE Transactions On Image Processing*, 4(6):863–870, June 1995.

[124] J.G. Thomas, R.A. Peters II, and P. Jeanty. Automatic segmentation of ultrasound images using morphological operators. *IEEE Transactions on Medical Imaging*, 10(2):180–186, 1991.

[125] U. Tiede, K.H. Hoehne, M. Bomans, A. Pommert, M. Riemer, and G. Wiebecke. Surface rendering. *IEEE Computer Graphics and Applications*, pages 41–53, March 1990.

[126] C. Tsai, Y. Sun, P. Chung, and J. Lee. Endocardial boundary detection using a neural network. *Pattern Recognition*, 26(7):1057–1068, 1993.

[127] J.K. Udupa and D. Odhner. Fast visualization, manipulation, and analysis of binary volumetric objects. *IEEE Computer Graphics and Applications*, 11(6):53–62, November 1991.

[128] J.K. Udupa and D. Odhner. Shell rendering. *IEEE Computer Graphics and Applications*, pages 58–67, November 1993.

[129] M. Unser. Sum and difference histograms for texture classification. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, PAMI-8(1):118–125, 1986.

[130] M. Unser and M. Eden. Multiresolution feature extraction and selection for texture segmentation. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 11(7):717–728, 1989.

[131] P.J.M. van Laarhoven and E.H.L. Aarts. *Simulated annealing: theory and applications*. Kluwer Academic Publishers, 1988.

[132] N. Venkateswarlu, S. Balaji, P. Raju, and R. Boyle. Some further results of three stage ml classification applied to remotely sensed images. *Pattern Recognition*, 27(10):1379–1396, 1994.

[133] R. Waag, J. Astheimer, and J. Smith III. Analysis and computations of measurement system effects in ultrasonic scattering experiments. *Journal of the Acoustical Society of America*, 91(3):1284–1297, March 1992.

[134] R.C. Waag. A review of tissue characterization from ultrasonic scattering. *IEEE Transactions on Biomedical Engineering*, BME-31(12):884–893, December 1984.

[135] A. Wacker and T. El-Sheikh. Average classification accuracy over collections of gaussian problems - common covariance matrix case. *Pattern Recognition*, 17(2):259–273, 1984.

[136] R. Wagner, S. Smith, J. Sandrik, and H. Lopez. Statistics of speckle in ultrasound b-scans. *IEEE Transactions on Sonics and Ultrasonics*, 30(3):156–163, May 1983.

[137] E. Walach and A. Shmulewitz. Local tissue attenuation images based on pulsed-echo ultrasound scans. *IEEE Transactions on Biomedical Engineering*, 36(2):211–221, February 1989.

[138] A. Wallin. Constructing isosurfaces from ct data. *IEEE Computer Graphics and Applications*, 11(6):28–33, November 1991.

[139] A. Watt and M. Watt. *Advanced Animation and Rendering Techniques Theory and Practice*. Addison-Wesley, 1992.

[140] P. N. T. Wells. Medical ultrasonics. *IEEE Spectrum*, pages 44–51, December 1984.

[141] C.S. Won and J. Derin. Unsupervised segmentation of noisy and textured images using markov random fields. *CVGIP: Graphical Models and Image Processing*, 54(4):308–328, 1992.

[142] C. Wu and Y. Chen. Statistical feature matrix for texture analysis. *CVGIP: Graphical Models and Image Processing*, 54(5):407–419, September 1992.

[143] C. Wu and Y. Chen. Multi-threshold dimension vector for texture analysis and its application to liver tissue classification. *Pattern Recognition*, 26(1):137–144, 1993.

[144] C. Wu, Y. Chen, and K. Hsieh. Texture features for classification of ultrasonic liver images. *IEEE Transactions on Medical Imaging*, 11(2):141–152, June 1992.

[145] L.X. Yao, J.A. Zagzebski, and E.L. Madsen. Backscatter coefficient measurements using a reference phantom to extract depth-dependent instrumentation factors. *Ultrasonic Imaging*, 12:58–70, 1990.

[146] S. R. Yhann and T.Y. Young. Boundary localization in texture segmentation. *IEEE Transactions On Image Processing*, 4(6):849–856, June 1995.

[147] M. Young, D. Argiro, and S. Kubica. Cantata: Visual programming environment for the khoros system. *Computer Graphics*, 29(2):22–24, May 1995.

[148] T.Y. Young and K. Fu. *Handbook of Pattern Recognition and Image Processing*. Academic Press Inc., 1986.

[149] C. Zhuang and S. Dunn. The amplitude varying rate statistical approach for texture classification. *Pattern Recognition Letters*, 11:143–149, February 1990.

Appendix A

3D Ultrasound Probe Simulation

Since the development of a 3D ultrasound probe is beyond the scope of this thesis, it is somewhat difficult to determine the accuracy of the data obtained from the probe. Further, little experimental data is currently available from the probe, so it is necessary to make a number of assumptions about the acquisition process. The weakest of these assumptions is that there is little or no alignment error introduced by the mechanical rotation of the probe. If this assumption is violated the resulting reconstruction will be distorted. It is also assumed that the motor does not induce any movement (chatter) of the transducer when a single image is being obtained. If this assumption is violated the image will contain motions artifacts which will be impossible to remove. Preliminary results obtained from the 3D ultrasound probe indicate that this assumption is valid (at least in the experimental environment). However, it is not apparent that perfect alignment of the slices can be assumed. Since calibration data is currently not available, it is necessary to determine how much of an effect misalignment of the slices has on the reconstruction of the volume data. A computer simulation of the 3D ultrasound probe provides the ability to study alignment errors to determine the effect on resulting volume data.

Figure A.1 illustrates the layout for a typical 3D ultrasound scan. The probe is oriented perpendicularly to the volume at the centre of the XY plane. A scanning plane perpendicular to the XY plane defines a slice within the volume which is acquired by the probe. As the probe is rotated successive slices are sampled from the volume. Due to mechanical misalignment within the probe, the scanning plane does not remain perpendicular to the XY plane as it is rotated.

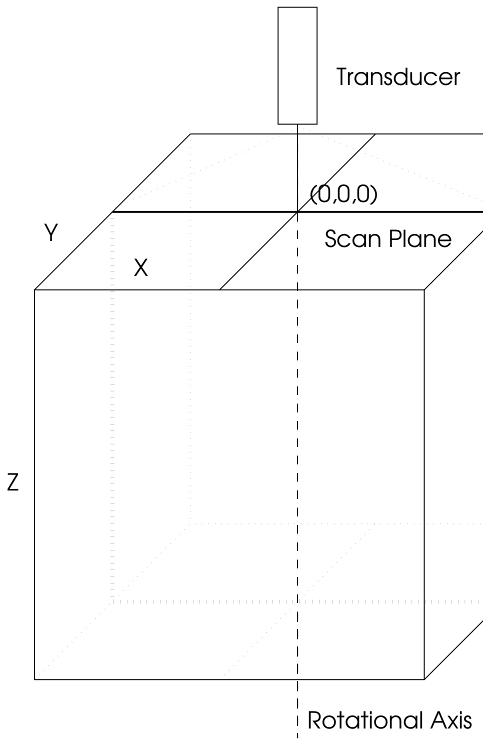


Figure A.1: Layout for a typical 3D ultrasound scan.

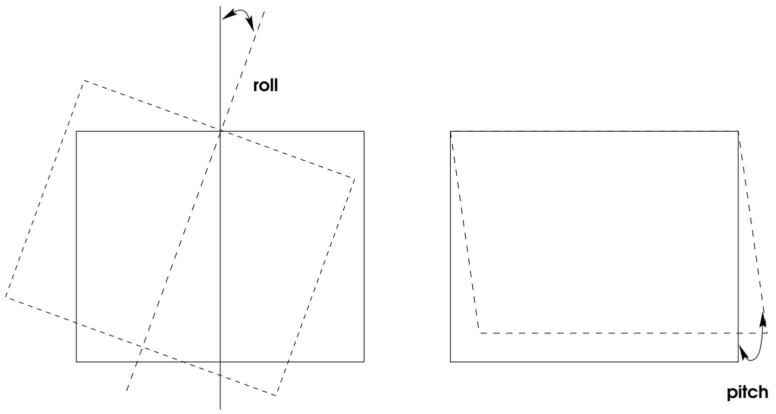


Figure A.2: Model simulating probe wobble during scan.

Figure A.2 depicts the type of alignment errors modelled in the simulation. The simulation model assumes three degrees of freedom in the scanning process:

- θ - angular increment between successive scans,
- ϕ - pitch, angular displacement from vertical,
- γ - roll, angular displacement from horizontal.

Ideally, ϕ and γ are small and do not introduce a significant error, relative to the resolution of the ultrasound probe. These two angles characterise a "wobble" as the transducer is rotated. In the model, it is assumed that these angles are constant during the entire scan. As well, translational errors are not modelled.

The wobble of the transducer is characterised by a matrix W that displaces the scanning plane from true vertical and horizontal as follows:

$$W = \begin{bmatrix} cos(\gamma) & 0 & -sin(\gamma) \\ sin(\phi)sin(\gamma) & cos(\phi) & sin(\phi)cos(\gamma) \\ cos(\phi)sin(\gamma) & -sin(\phi) & \cos(\phi)\cos(\gamma) \end{bmatrix} \tag{A.1}$$

The scanning plane is incrementally rotated through the volume at increments of $\theta°$. This rotation is achieved with the following matrix S:

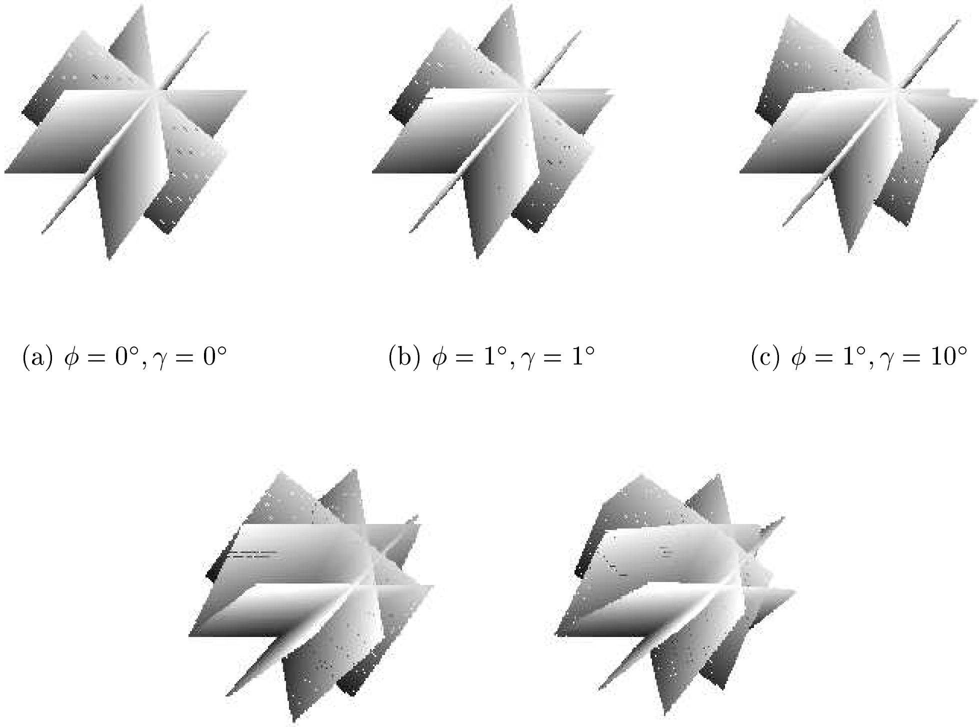
$$S = \begin{bmatrix} cos(a) & sin(a) & 0 \\ -sin(a) & cos(a) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.2}$$

where $a = s\theta$, $0 \leq s \leq 180/\theta$, and s is the number of slices sampled in the volume. The product of W and S yields the final rotational matrix R

$$R = WS. \tag{A.3}$$

Since s is incremented throughout the simulation S and R must be recalculated at every sampling interval.

Figure A.3(a) shows the orientation of the scanning planes through 180° for $\theta = 45°$ which results in $s = 5$ slices sampled within the volume. Figures A.3(b)



(a) $\phi = 0°, \gamma = 0°$ (b) $\phi = 1°, \gamma = 1°$ (c) $\phi = 1°, \gamma = 10°$

(d) $\phi = 10°, \gamma = 1°$ (e) $\phi = 10°, \gamma = 10°$

Figure A.3: Scan plane orientation with $\theta = 45°$.

- A.3(e) show the orientation of the scanning planes for various values of ϕ and γ. Note that as the wobble intensifies the first and last slices are no longer coplanar.

In the real environment it assumed that the wobble about the rotational axis is negligible, thus, $W \approx I$ (the identity matrix). The sampled points can be mapped directly into a cylindrical coordinate system. To model this in the simulation, all sampled points are also mapped into a cylindrical coordinate system, ignoring the effect W has on the location of the scanning planes. This assumption is represented by relaxing the equality sign in Equation (A.3)

$$R = WS \approx IS = S. \tag{A.4}$$

The extent to which W varies from I will determine how much error is introduced.

A point $p, p = [x, y, z]$, on a scan plane in the original volume is mapped to the point $p', p' = [\theta', r, z]$, in the cylindrical coordinate system using the mapping function F(p)

$$p' = F(p) \tag{A.5}$$

where $r = xcos(\theta') + ysin(\theta')$, $\theta' = s\theta$, and x and y are offsets from the centre of the XY plane at which the axis of rotation intersects. Points are mapped back to the Cartesian coordinate system using the inverse mapping function, $G(p')$

$$p = G(p') \tag{A.6}$$

where $x = rcos(\theta')$ and $y = rsin(\theta')$. Due to the discrete sampling of the volume (controlled by θ) and the approximation in Equation (A.4) the forward and inverse mappings do not necessarily result in points p having the same value, i.e.,

$$G(F(p)) \neq p. \tag{A.7}$$

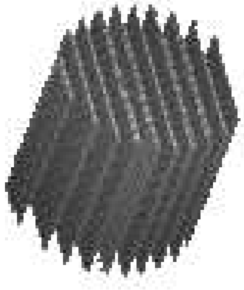
There are three main objectives of the simulation:

- determine how the sampling interval θ affects the accuracy of the volume data,
- determine methods for reducing aliasing in the volume data,
- determine the extent to which transducer wobble introduces artifacts in the volume data.

The following sections describe each of these objectives in detail as well as the volume data used in the simulations.

A.1 Data Sets

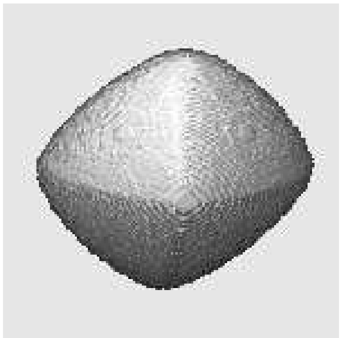
A number of data sets were synthesised and used in the simulation of the 3D ultrasound probe. Figure A.4 depicts renderings of the volume data used in evaluating



(a) sine wave - 64^3

(b) sphere - 64^3

(c) cube - 64^3



(d) superellipsoid - 128^3

(e) supertoroids - 128^3

Figure A.4: Data sets used in simulating the 3D ultrasound probe.

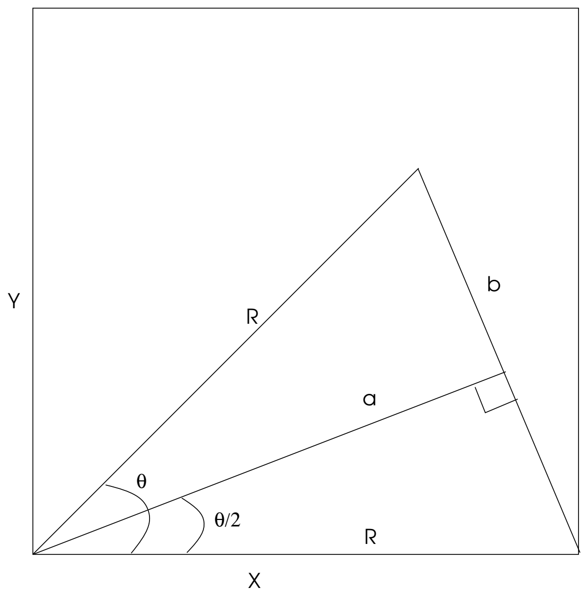


Figure A.5: Geometry of sampling process.

the probe wobble as well as the sampling interval. Figures A.4(a) - A.4(c) consist of a volume of 64^3 pixel while Figures A.4(d) and A.4(e) are 128^3 pixels. The sine wave data set was used because it has pixel variations throughout the entire volume. As well, any distortions incurred in the sampling process are visually apparent as edges are either curved or contain aliasing effects. The dataset was generated using the function

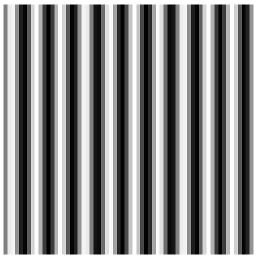
$$I(x, y, z) = sin(x\pi/180),$$

where $I(x, y, z)$ represents the intensity of the point at (x, y, z).

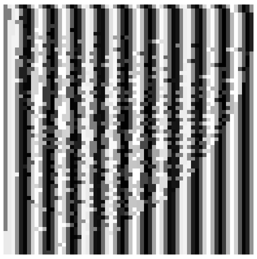
The remainder of the data sets were chosen because of their various shapes. As well, larger volumes (128^3) were used because sampling error increases as the distance from the transducer increases.

A.2 Sampling Interval

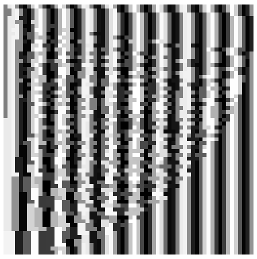
The first objective of the simulation is to determine the effect that the sampling interval has on the resulting volume data. Without loss of generality, the 2D case can be considered where a plane in the positive quadrant is sampled at radial increments



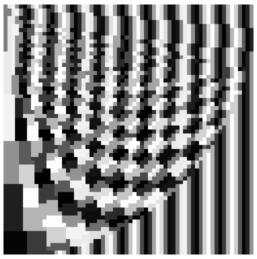
(a) original



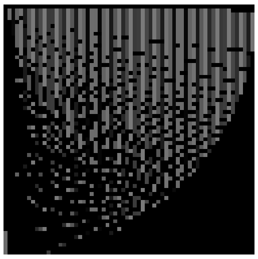
(b) $\theta = 1$



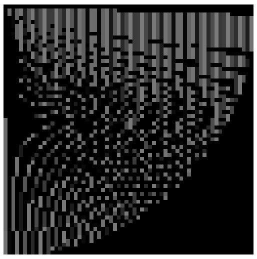
(c) $\theta = 2$



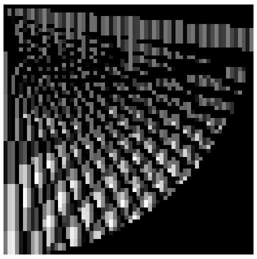
(d) $\theta = 5$



(e) $\theta = 1$



(f) $\theta = 2$



(g) $\theta = 5$

Figure A.6: Original, sampled and difference images at varying sampling intervals.

from the origin (c.f. Equation (A.5) and Equation (A.6) where the sampling process is independent of the z axis). Figure A.5 illustrates the geometry of the sampling process (sampling in the other quadrants is obtained by symmetry). The sampling interval θ determines the number of radials that are sampled along in the quadrant. Note that the region close to the origin is sampled more densely while sampling is sparse as the distance from the origin approaches R. From Figure A.5 the following relations can be derived:

$$b = 2Rsin(\theta/2), \tag{A.8}$$

$$r = a(2sin(\theta/2))^{-1}. \tag{A.9}$$

In order to minimise the sampling error in the periphery of the region it is necessary to have b small. This requires θ to be small as well. However, as θ decreases, the time required to sample the region increases as more samples are collected. To minimise patient/transducer movement sampling must be accomplished in real time. It has been determined that up to 3 seconds can be tolerated without introducing significant error. Acquisition of a single ultrasound image with one focal zone requires approximately $1/30^{th}$ of a second. Given the 3 second maximum sampling time and a $180°$ sampling arc, the minimum angle for θ is $2°$. When the ultrasound image is acquired using three focal zones the rate drops to approximately 17 images per second. This requires $\theta \geq 3.5°$ in order to sample the region in under 3 seconds. Assuming a single focal zone and using Equation (A.8) the distance between two sampled points at a radial distance R is $b = 0.035R$. For a typical ultrasound scan, $R = 320$ and the distance between the points is $b = 11.2$. This distance is substantial as small details are lost in the sampling process. The point along the radial sampling axis at which a unit distance exists between equidistant radial samples occurs at $a = 1$. Using Equation (A.9), $r = 28.6$. Thus, points only within 29 pixels of the origin do not incur any error due to undersampling of the region.

Simulations were conducted using a 64×64 pixel image of a sine wave shown in Figure A.6(a). The sampling origin is at the top left corner of the image with $R = 64$. The sampling interval was varied from $\theta = 1$ to $\theta = 45$. Points were sampled from the original image into the cylindrical coordinate system using Equation (A.5) and then back to the Cartesian coordinate system using Equation (A.6). Points whose coordinates do not fall at discrete locations are approximated to the nearest discrete locations (Section A.3 describes alternatives to this method).

The resulting images for $\theta = 1, 2$ and 5 are shown in in Figures A.6(b) - A.6(d). By comparing these images to the original image in Figure A.6(a), it can be observed that for even a small sampling interval, error is introduced. The amount of error that is introduced in the sampling process can be visualised by taking the pointwise difference between the original image and a sampled image. The magnitude of the difference at each pixel forms a difference image whose intensities are proportional to the magnitude. In Figures A.6(e) - A.6(g) the difference images are shown for the corresponding sampled images with $\theta = 1, 2$ and 5. It can be concluded that a large number of pixels have their original value modified due to sampling. For

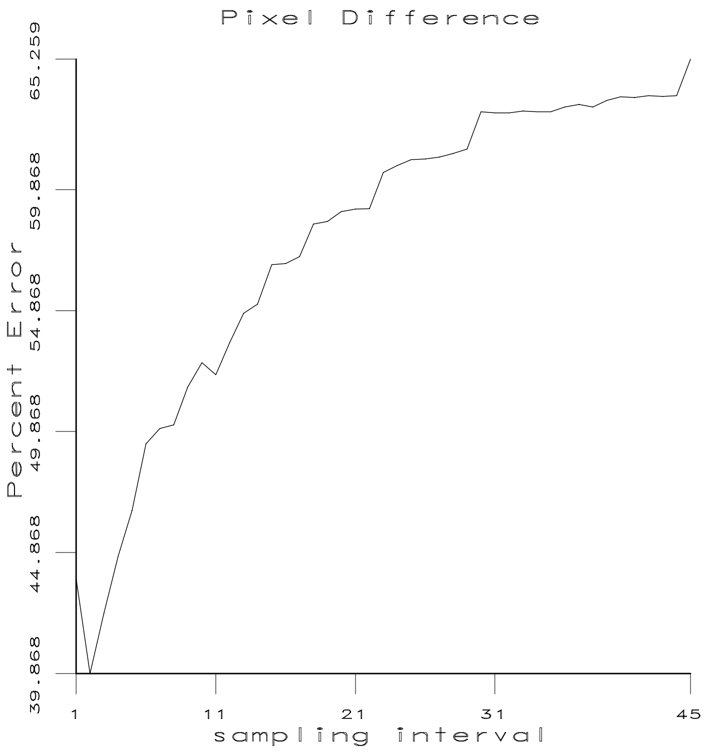
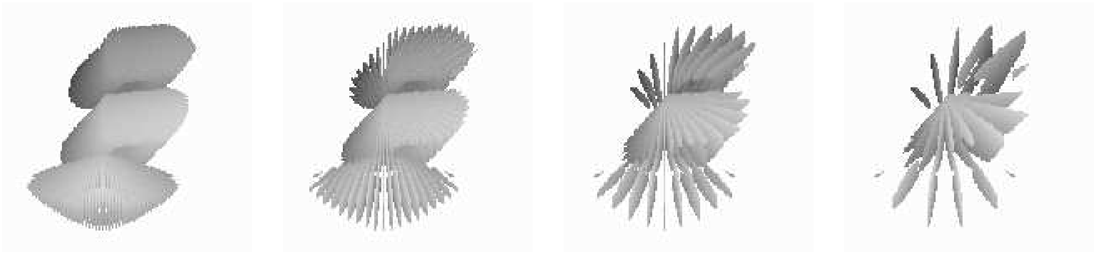


Figure A.7: Percentage of the total number of pixels in which there is a difference between the original and sampled images.



(a) $\theta = 2$ (b) $\theta = 5$ (c) $\theta = 10$ (d) $\theta = 20$

Figure A.8: Supertoroids volume data at varying sampling intervals.

$\theta = 1$, $b \approx 1$ so no points are undersampled. The error is due to the discretisation of point locations during the sampling process. For $\theta \leq 2$, points in the periphery of the sampling range begin to be undersampled so additional error is incurred. Figure A.7 shows the percentage of the total number of pixels in which there is a difference between the original and sampled images as a function of the sampling interval. As expected, the error increases as θ increases (except for the minimum at θ). The results in the graph are biased as a number of points in the original image are beyond the radial sampling range (points in the bottom right corner). This bias only has an effect when making absolute comparisons of pixel differences for a given θ. It can be removed by scaling each value by the percentage of points within the radial sampling range. For the sine wave example, 79% of the points are within the sampling range (since sampling range is over $90° = \pi/4$). This results in a difference range of $[50.8°, 83.1°]$.

An example of sampling volume data is shown in Figure A.8. Depicted in each image is the portion of the supertoroids falling on the sampling planes in calculating $F(p)$. It is apparent that for $\theta \geq 5$ the shape information lost in sampling to the cylindrical coordinate system is significant. For example, the bottom supertoroid in Figure A.8(b) is sampled much less than the middle supertoroid. The result is that the bottom supertoroid will be deformed when sampled back to the Cartesian coordinate system.

Based on the limited sampling time and the results obtained from the above simulation, a sampling interval of $\theta = 2$ is used for the remainder of the simulations.

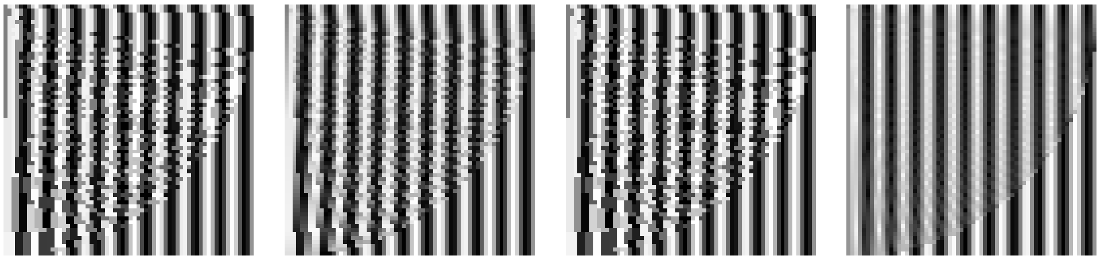
A.3 Anti-Aliasing

The sampled images of the sine wave in Figure A.6 all contain jagged edges (aliasing) due to the discrete nature of the sampling process. It is possible to reduce this effect by applying an anti-aliasing technique during sampling. One basic way is to use interpolation. In the 2D case, bilinear interpolation is used to determine a point's value by taking a weighted average of its four nearest neighbours lying at discrete locations. In the 3D case, trilinear interpolation takes the information from the 8 neighbouring points.

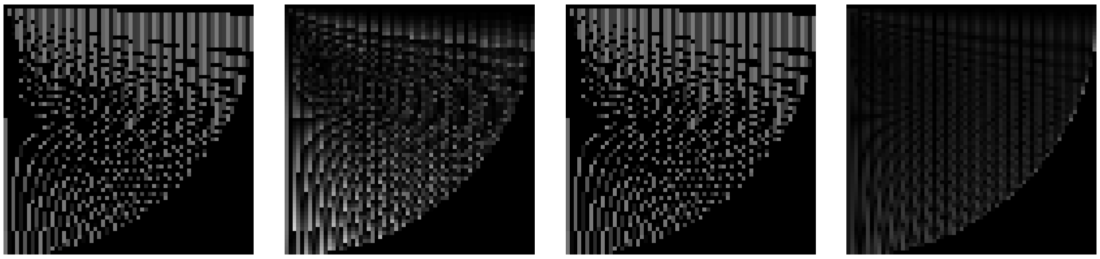
Since the purpose of $F(p)$ is to model the way in which the ultrasound probe acquires data it is justifiable to use interpolation during sampling. Consideration of the characteristics of a single ultrasound image indicates that a pixel's intensity is representative of an echo caused by a reflector in a given region. Since the ultrasound beam has a limited resolution (0.25mm), the echo, and correspondingly the pixel, is a sum of the reflected echos from multiple reflectors in the region. Admittedly, this process is complex and most likely not modelled with simple interpolation but is adequate for the purposes of the simulation. Use of interpolation in calculating $G(p')$ is not as readily justifiable. However, if the aliasing effect can be reduced while minimising any increases in error then interpolation is deemed justifiable.

Figure A.9 shows the resulting sampled and difference images for the sine wave (Figure A.6(a)). Interpolation was used in both $F(p)$ and $G(p')$ in Figure A.9(d). Qualitatively, this image has the least amount of aliasing. The corresponding difference image in Figure A.9(h) also seems to indicate that the use of interpolation in both $F(p)$ and $G(p')$ results in the least amount of difference between the sampled and original images (the darker the difference image, the smaller the pointwise differences).

Figure A.10(a) shows the relationship between the size of the sampling interval,

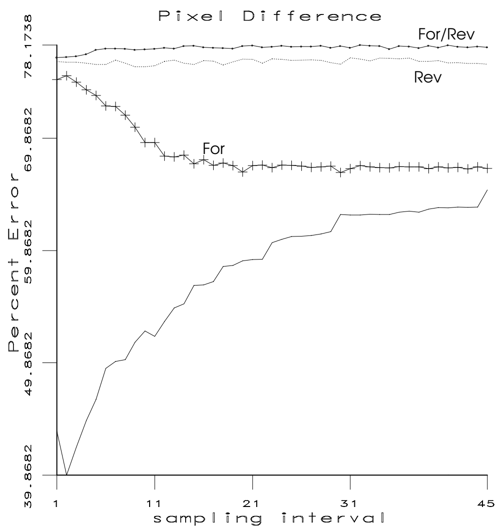


(a) none (b) F(p) (c) G(p') (d) F(p)/G(p')

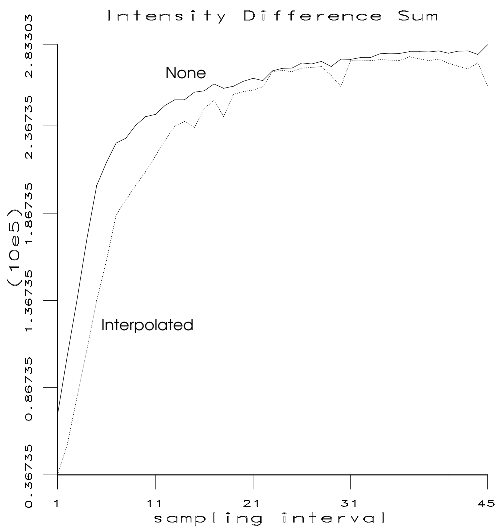


(e) none (f) F(p) (g) G(p') (h) F(p)/G(p')

Figure A.9: Interpolation used while calculating F(p) and G(p'), $\theta = 2$ for sampled image and resulting difference image.



(a) Percentage of pixels in sampled image different from original image (unlabelled used no interpolation)



(b) Sum of pointwise difference between sampled and original images with and without interpolation.

Figure A.10: Results obtained with interpolation ($F(p)$ is "For", $G(p')$ is Rev) and without interpolation.

θ, and the percentage of pixels different between the original and sampled images for differing use of interpolation. Two pixels are different if their intensity values are not equal and they are located at the same corresponding locations in two different images. When interpolation is used the percentage of different pixels is approximately constant, independent of θ. Removing the bias in calculating the percentage of different pixels (as discussed in Section A.2) reveals that almost every original pixel within the radial sampling range has a different corresponding value in the sampled image. When compared to the results without interpolation for $F(p)$ it seems that using interpolation significantly increases the error in the sampled image. However, counting the number of different pixels is certainly not favourable to interpolation as every pixel is a function of its neighbours. Only those points which lie at discrete locations in either coordinate system will maintain their original value.

In order to get a better measure of the error in the sampled images the sum of the pointwise differences is used. Figure A.10(b) shows that for smaller sampling intervals, the use of interpolation (for both $F(p)$ and $G(p')$) results in a smaller pointwise difference sum than for sampling without interpolation. Based on this and the qualitative analysis above, it is concluded that the use of interpolation in the sampling process is justifiable.

A.4 Probe Wobble

The simulations described in the previous sections have assumed perfect alignment about the axis of rotation. The objective of these simulations is to determine the amount of sampling error introduced when there is not perfect alignment. This misalignment is characterised by a wobble about the rotational axis which is parameterised by the matrix W (see Section A). By varying W from I, the identity matrix, it is possible to simulate varying degrees of transducer wobble. Since the underlying assumption for all simulations is that $W \approx I$, the use of Equation (A.3) in sampling with $F(p)$ will result in an increase in error when $G(p')$ is applied to map the points back to Cartesian coordinate system.

In order to simulate a realistic level of transducer wobble, a wide range of values for ϕ and γ, the parameters of W, were evaluated. Both ϕ and γ took on values in the range [0.00001,10]. It is assumed that an angle less than 0.00001° will produce a negligible amount of wobble. It is also assumed that the probe can be engineered with enough accuracy to keep the wobble under 10° for both ϕ and γ.

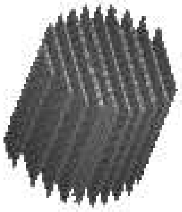
Figure 5.6 shows results for some of the simulations using the sine wave, sphere, cube, supertoroids, and superellipsoid volume data sets. All simulations were performed with a sampling interval of $\theta = 2°$. As ϕ and γ increase, the distortion of the object in the sampled image increases. There is a noticeable distortion in the supertoroids in Figure A.11(e) for $\phi = 1$ and $\gamma = 1$. This gets much worse for larger values as shown in Figures A.11(f) - A.11(h) and A.11(j).

The graphs in Figure A.12 display the relationship between the amount of transducer wobble and the percentage of difference pixels. In this context, the percentage of difference pixels is calculated between a sampled volume and an established baseline volume, rather than the original volume. The baseline is established by sampling the original volume *without* any transducer wobble ($i.e. \phi = 0,\ \gamma = 0$). From the graphs, it is concluded that a transducer wobble of up to 0.001° for ϕ and γ can be tolerated without affecting the sampling process as the amount of sampling error introduced is insignificant.

Further analysis of the simulation results is possible by examining the graphs in Figure A.13. These graphs display the relationship between the amount of transducer wobble and the sum of the pointwise differences. The pointwise differences are calculated from a difference volume obtained from the established baseline with no wobble and the sampled volumes with varying degrees of wobble. The pointwise difference is calculated as follows:

$$PWD = \sum_x \sum_y \sum_z |B(x,y,z) - V(x,y,z)| \tag{A.10}$$

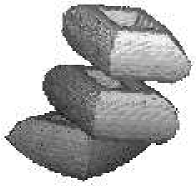
where B is the baseline volume, V is the sampled volume with wobble, and (x, y, z) range over the size of the volume (both B and V are the same size). These graphs



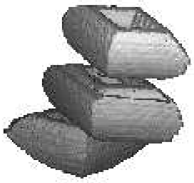
(a) Sine wave - $\phi = 0,\ \gamma = 0$

(b) Sphere - $\phi = 0,\ \gamma = 0$

(c) Cube - $\phi = 0,\ \gamma = 0$

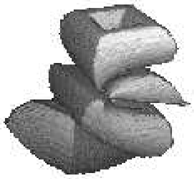


(d) Supertoroids - $\phi = 0,\ \gamma = 0$



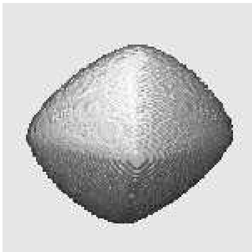
(e) Supertoroids - $\phi = 1,\ \gamma = 1$

(f) Supertoroids - $\phi = 1,\ \gamma = 10$

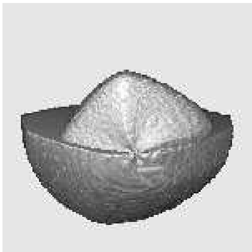


(g) Supertoroids - $\phi = 10,\ \gamma = 1$

(h) Supertoroids - $\phi = 10,\ \gamma = 10$

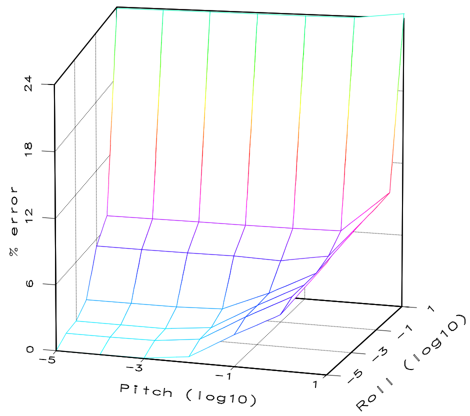


(i) Superellipsoid - $\phi = 0,\ \gamma = 0$

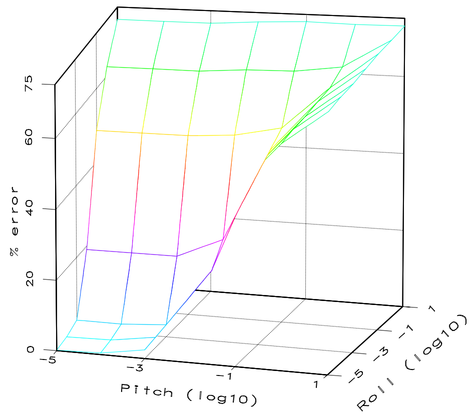


(j) Superellipsoid - $\phi = 10,\ \gamma = 10$

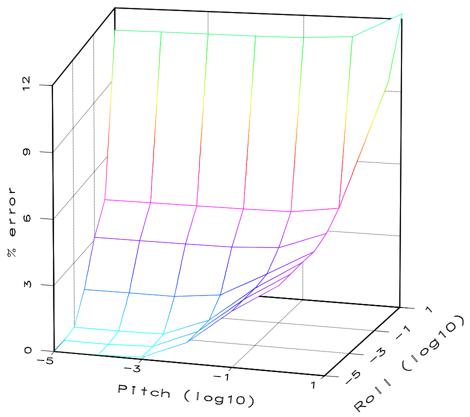
Figure A.11: Sampled volumes with $\theta = 2$.



(a) Sphere

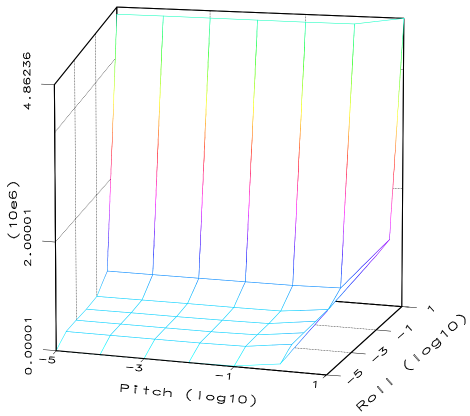


(b) Sine wave

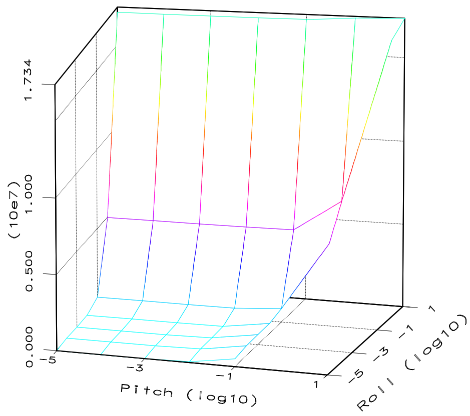


(c) Supertoroids

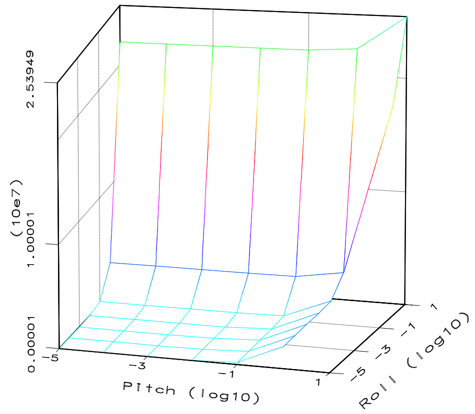
Figure A.12: Percentage of pixels different for varying degree of wobble, $\theta = 2$.



(a) Sphere



(b) Sine wave



(c) Supertoroids

Figure A.13: Sum of pointwise differences for varying degree of wobble, $\theta = 2$.

are typical of the simulation results obtained for other data sets. It can be seen from the graphs that PWD remains constant, near 0, for $\phi, \gamma \leq 0.1$. It can be concluded that although points in the sampled volume are different from the baseline for $0.001 < \phi, \gamma \leq 0.1$ the amount they differ by is minimal. This suggests that a larger amount of transducer wobble can be tolerated without introducing significant sampling error.

The remaining figures in this section illustrate the effect a transducer wobble can have on sampled data if the transducer is not calibrated to account for the wobble. Figure A.14 shows the supertoroids volume for $\phi = 1°$ and $\gamma = 1°$, Figure A.15 shows the supertoroids volume for $\phi = 10°$ and $\gamma = 10°$ and Figure A.16 show the superellipsoid volume for $\phi = 10°$ and $\gamma = 10°$.

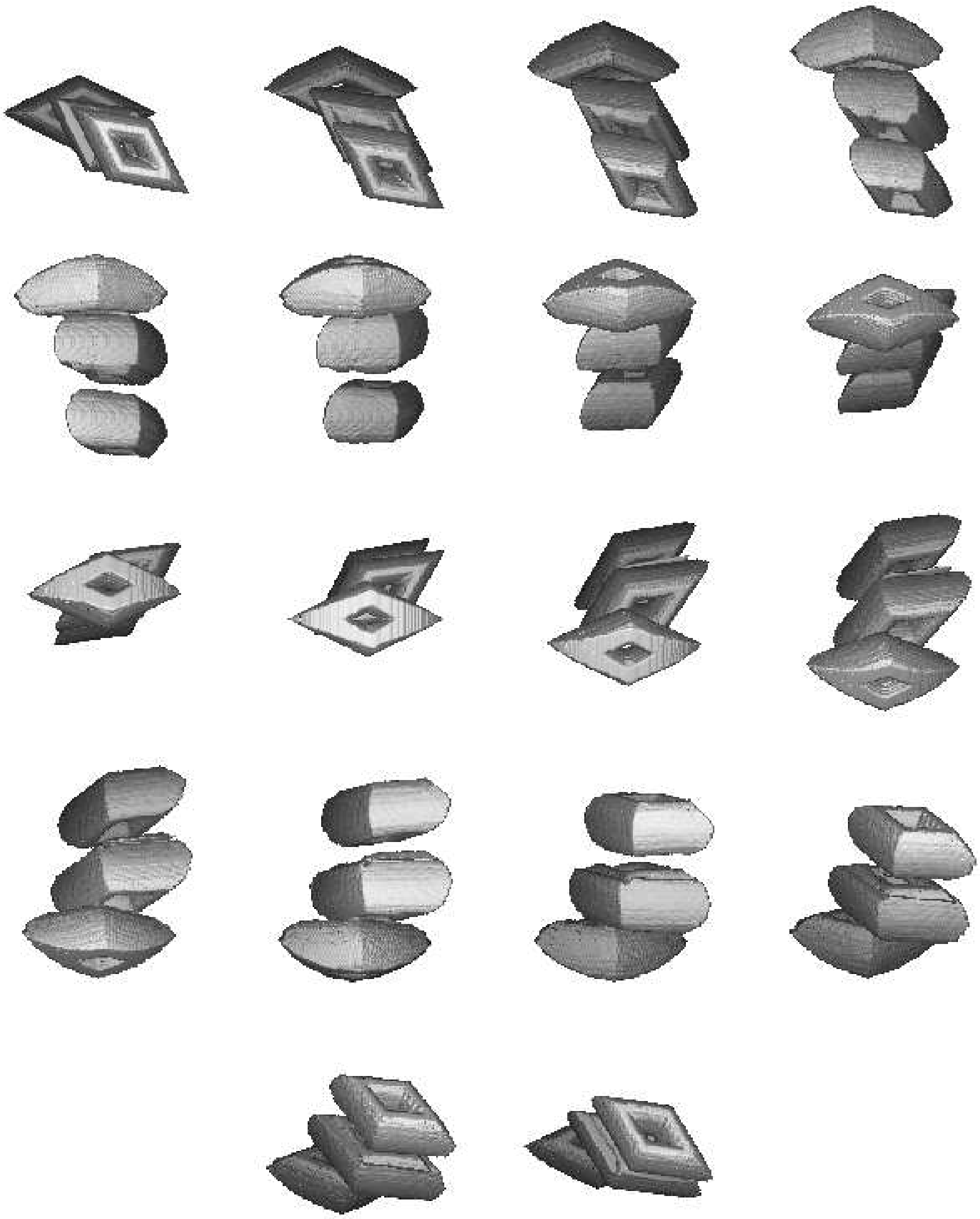


Figure A.14: Supertoroids animation, $\phi = 1^{\circ}, \gamma = 1^{\circ}$. Perspectives shown for 20° rotational increments about the x axis.

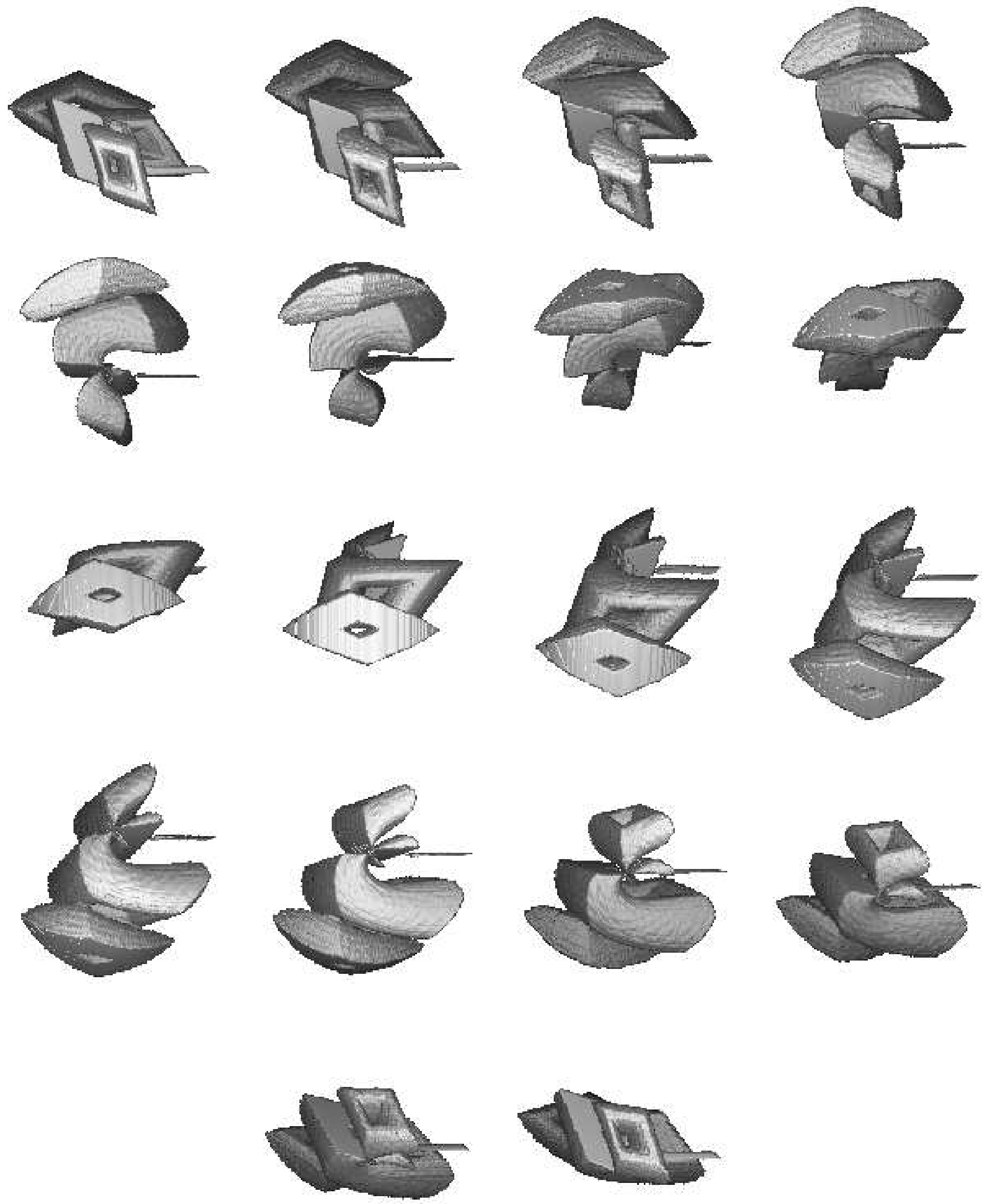


Figure A.15: Supertoroids animation, $\phi = 10^{\circ}, \gamma = 10^{\circ}$. Perspectives shown for 20° rotational increments about the x axis.

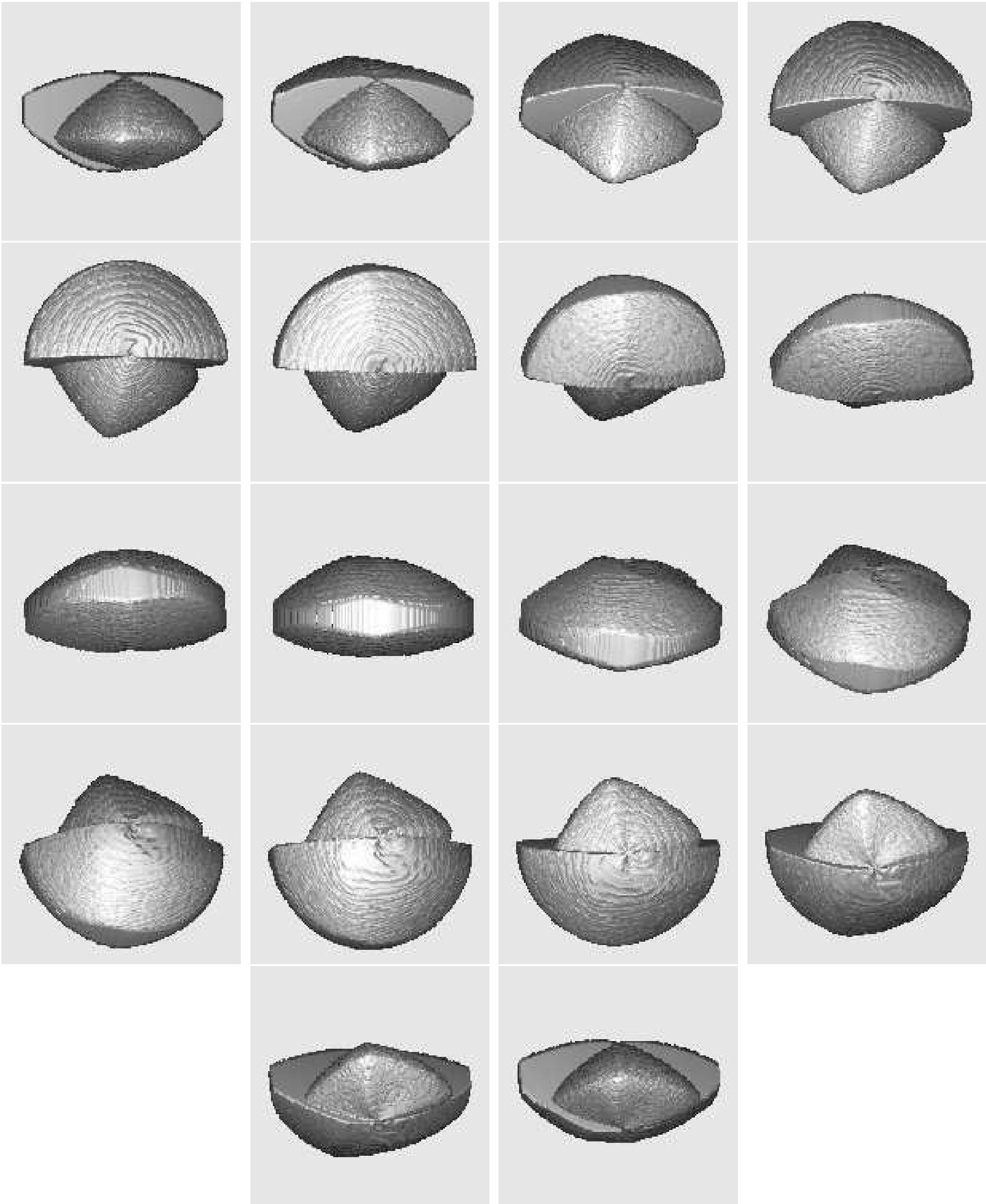


Figure A.16: Superellipsoid animation, $\phi = 10°, \gamma = 10°$. Perspectives shown for 20° rotational increments about the x axis.

Appendix B

Probabilistic Distance Measures

For M normally distributed classes with multivariate mean, $\mu_{\mathbf{i}}$, and covariance, $\mathbf{\Sigma_i}$, $1 < i, j \leq M$, the feature selection criterion function [25] is defined as

$$J = \sum_{i=1}^{M} \sum_{j=1}^{M} P(\omega_i) P(\omega_j) J(i,j), \; i \neq j$$

where $P(\omega_i)$ is the *a priori* probability of class i and $J(i,j)$ is the probabilistic distance measure between classes i and j using one of the probabilistic measures given below:
Chernoff

$$J_C(i,j) = \frac{1}{2} \; s(1-s)(\mu_{\mathbf{j}} - \mu_{\mathbf{i}})^t((1-s)\mathbf{\Sigma_i} + s\mathbf{\Sigma_j})^{-1}(\mu_{\mathbf{j}} - \mu_{\mathbf{i}}),$$

$$+\frac{1}{2} \; \ln \frac{|(1-s)\mathbf{\Sigma_i} + s\mathbf{\Sigma_j}|}{|\mathbf{\Sigma_i}|^{1-s}|\mathbf{\Sigma_j}|^s}, \quad s \in [0.0, 1.0],$$

Bhattacharyya

$$J_B(i,j) = \frac{1}{8} \; (\mu_{\mathbf{j}} - \mu_{\mathbf{i}})^t \left(\frac{\mathbf{\Sigma_i} + \mathbf{\Sigma_j}}{2} \right)^{-1} (\mu_{\mathbf{j}} - \mu_{\mathbf{i}}) + \frac{1}{2} \; \ln \frac{|\frac{1}{2}(\mathbf{\Sigma_i} + \mathbf{\Sigma_j})|}{(|\mathbf{\Sigma_i}||\mathbf{\Sigma_j}|)^{\frac{1}{2}}},$$

Matusita

$$J_T(i,j) = \sqrt{2(1 - \exp(-J_B(i,j)))},$$

Divergence

$$J_D(i,j) = \frac{1}{2}(\mu_{\mathbf{j}} - \mu_{\mathbf{i}})^t(\mathbf{\Sigma_i}^{-1} + \mathbf{\Sigma_j}^{-1})(\mu_{\mathbf{j}} - \mu_{\mathbf{i}}) + \frac{1}{2} \; \mathrm{tr}(\mathbf{\Sigma_i}^{-1}\mathbf{\Sigma_j} + \mathbf{\Sigma_j}^{-1}\mathbf{\Sigma_i} - 2\mathrm{I}),$$

I is the identity matrix,

Mahalanobis (when $\mathbf{\Sigma_j} = \mathbf{\Sigma_i} = \mathbf{\Sigma}$)

$$J_M(i,j) = (\mu_{\mathbf{j}} - \mu_{\mathbf{i}})^t \mathbf{\Sigma}^{-1} (\mu_{\mathbf{j}} - \mu_{\mathbf{i}}),$$

Patrick_Fisher

$$J_P(i,j) = ((2\pi)^d |2\mathbf{\Sigma_i}|)^{-\frac{1}{2}} + ((2\pi)^d |2\mathbf{\Sigma_j}|)^{-\frac{1}{2}}$$

$$-2((2\pi)^d |\mathbf{\Sigma_i} + \mathbf{\Sigma_j}|)^{-\frac{1}{2}} \ \exp(-\frac{1}{2}\ (\mu_\mathbf{j} - \mu_\mathbf{i})^t (\mathbf{\Sigma_i} + \mathbf{\Sigma_j})(\mu_\mathbf{j} - \mu_\mathbf{i})).$$

Appendix C

Glossary

affine equivariant estimate An approximation of the true value which commutes with a linear transformation of the space from which the true value exists [106].

attenuation The process of energy loss from a wave for any reason [95].

Cartesian coordinate system A parameterisation of the space in which each dimension is perpendicular to all other dimensions. A 3D Cartesian coordinate system is often denoted as (x, y, z).

characterisation The representation of all members of a class with a set of standard attributes (texture features).

classification The determination of the identity (class) of an unknown pattern.

Cylindrical coordinate system A parameterisation of the space in which a plane is rotated about an axis. A cylindrical coordinate system is often denoted as (r, θ, z), where θ describes angle of rotation of the plane.

feature space The space defined by the feature vectors [25].

Inck **Inc**omplete **K**nowledge; a new statistical decision rule, proposed in this thesis, for use as a criterion function in feature selection and as a decision rule in classification of unknown patterns.

isometric Equal scale in all dimensions.

lateral translation Perpendicular movement of the scanning plane along the scanning axis.

minimum volume ellipsoid estimator A robust statistical method in which the smallest ellipsoid containing at least h of N points is used to determine the parameters of the data.

mixture density A probability density function which is the sum of all class probability density functions.

MTS Multiresolution Texture Segmentation [80].

multiresolution Examination of a region at varying levels of granularity.

multivariate point cloud The existence of points in multi-dimensional space.

MVE Minimum Volume Ellipsoid [106].

noise The undesirable component of a signal which is added to the original signal due to a known or unknown process.

outliers Points in a dataset which are "far" from the majority of points.

probabilistic measures Values based on the probability distribution of a class.

reduced feature space A subspace of the original feature space.

reference phantom A composition of graphite material in oil which is used to emulate the acoustical properties of soft tissue.

Resampling Module The system module which transforms volume data from a non-isometric representation to an isometric representation.

RF signal A signal in the radio frequency range.

robust statistics Methods for describing the majority of points in a dataset in the presence of outliers.

scatter The dispersion of a sound wave by a tissue in a characteristic pattern.

sector scan A pie shaped scanning profile resulting from a curved array transducer.

segmentation The identification of homogeneous regions in an image.

speckle Undesirable phase sensitive interference patterns caused by echoes arriving at the transducer at the same time as the primary echoes from tissue interfaces and scattering bodies [95].

supervised training The use of known patterns to establish prototypical representations of a class(es).

texture The recurring pattern of the variations of feature values at a given resolution.

texture feature A number, calculated from region in an image, which is a compact representation of the variation in grey level intensities within the region.

unbiased The mean of the estimates is the same as the value of the parameter estimated.