Multi-Modal Registration for Image-Guided Therapy

by

Tamara S. Williams

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Science in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
OCT 2 3 2000
LIBRARIES

August 2000
[September 2000]

© Tamara S. Williams, MM. All rights reserved.

The author hereby grants to MIT permission to reproduce and distribute publicly paper and electronic copies of this thesis document in whole or in part.

BARKER

Author .....
Department of Electrical Engineering and Computer Science
August 15, 2000

Certified by .....
W.E.L. Grimson
Bernard Gordon Professor of Medical Engineering
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Multi-Modal Registration for Image-Guided Therapy

by

Tamara S. Williams

Submitted to the Department of Electrical Engineering and Computer Science
on August 15, 2000, in partial fulfillment of the
requirements for the degree of
Master of Science in Electrical Engineering and Computer Science

Abstract

The real-time monitoring of non-stationary targets for image guided therapy is critical for accuracy in determining the relative locations of organs. Currently, methods for obtaining real-time monitoring using image guidance involve expensive intra-operative equipment, ionizing radiation, or are limited to surface imaging of areas accessible through videoendoscopic tools. Of all imaging modalities, ultrasound is the most cost-effective, portable, subsurface-capable, and non-ionizing. However, low contrast and high speckle noise content have prohibited its widespread use in image-guided therapy.

We have sought solutions to two key elements involved in the registration of ultrasound with CT/MR: feature extraction in ultrasound and accurate multi-modal registration. A comparison is provided that illustrates substantial improvements in 2-D ultrasound edge detection using phase-based methods as opposed to traditional gradient-based methods. The elastic registration of 3-D CT/MR is accomplished by defining point correspondences used to compute a dense deformation field, which maps the atlas image (CT) from its coordinate system into the patient image (MR). We performed quantitative and qualitative error analysis to determine the accuracy of the registration. Also, a preliminary investigation into the Iterative Closest Point Algorithm (ICP) was begun and results are shown for rigid and non-rigid point patterns with the objective of extending the algorithm to register ultrasound edge points with CT/MR surface points.

Thesis Supervisor: W.E.L. Grimson
Title: Bernard Gordon Professor of Medical Engineering

Acknowledgments

I would like to thank my advisor, W.E.L. Grimson of the Artificial Intelligence Laboratory, for his guidance throughout this project and his wisdom in suggesting numerous papers that continue to assist me in my research goals. Much appreciation and thanks is extended to Carl-Fredrik Westin of Brigham and Women's Hospital and William Wells, III for their continual efforts in helping me with the computational results provided in this thesis. I also thank Homer Pien and Anthony Sacramone of Draper Laboratory and Steven Dawson of Massachusetts General Hospital for providing the ultrasound and CT data. Thanks to Julian Rathert of BWH for providing the point set correspondences in the CT/MR data. This work may not have been possible without the support of Lucent Technologies. Lucent's fellowship support gave me the freedom to seek an advisor and laboratory that would entirely suit my research interests.

A sincere thanks goes to my special friend, Robert Munroe, for his daily encouragement and faith in me that I could accomplish any goal set before me. Last, but definitely not least, I thank my family who offers great support, especially my mother, Brenda, and brother, Gregory. Their strength and courage have inspired me more than they may ever know.

Contents

Chapter 1

Introduction

1.1 Motivation of Multi-Modal Registration

The development and refinement of medical imaging modalities, which aid in non-invasively studying the internal structures of the human body, is essential for complementing the human visual system. The future of minimally invasive therapy is highly dependent on our ability to precisely target and deliver therapy. To succeed, such therapy must have target-relative information. Conventionally, image-guided procedures undergo two separate steps - a pre-operative imaging step in which the target site is identified, and an intra-operative step in which realtime or near-realtime imaging is used to guide the procedure [18]. Pre-operative imaging is sufficient for image guided procedures in which organs are relatively immobile. However, a realtime, intra-operative imaging technique is necessary for image guidance in which organs shift with respect to time such as the pulsating heart and abdominal organs, which are constantly in motion due to respiration.

Pre-operative imaging is typically performed via Computed Tomography (CT) or Magnetic Resonance Imaging (MR), although X-ray is sometimes used (e.g., for mammography). Intra-operative imaging can be CT, MR, X-ray, fluoroscopy, visible light or ultrasound. For some procedures involving only stationary targets, pre-operative imagery is sufficient. However, real-time monitoring of non-stationary targets for image-guided therapy is critical for the accurate determination of relative organ lo-

cations. Currently, methods for obtaining real-time monitoring using image guidance require expensive intra-operative equipment, expose ionizing radiation to the patient, or are limited to surface imaging of areas accessible through videoendoscopic tools. For example, the intra-operative MRI, which provides high quality imagery, will most likely remain available only to research centers and teaching hospitals due to high cost. In addition, fluoroscopy provides continuous X-ray imaging, which becomes prohibitive for long procedures due to risks of prolonged exposure to ionizing radiation. And videoendoscopy provides natural target-relative information, yet cannot be used in solid mass organs.

Of all imaging modalities, ultrasound is the most cost-effective, portable, subsurface-capable, and non-ionizing. However, its widespread use has been hampered due to its low contrast, high speckle noise content, and to a lesser degree limits on visibility due to bone structures or air pockets within the line of sight. Due to inhomogeneous tissue characteristics and its impact on ultrasound propagation, registration of CT/MR and ultrasound cannot be performed rigidly. We wish to improve the visualization of intra-operative ultrasound through its registration with high-resolution CT/MR imagery. CT provides good bone contrast of a medical image while MR provides good soft tissue contrast thus, a pre-operative registration of these two modalities will provide an optimal data set for registration with ultrasound. Surfaces of the CT/MR registration result can be segmented and points extracted to match with edge points of ultrasound.

We will demonstrate the ability to register pre-operative CT/MR in a non-realtime manner, so that in future work point-to-point correspondence to an intra-operative ultrasound can be obtained. This novel method of improving the visualization of ultrasound imagery is very inviting due to the overwhelming preference of users for high contrast CT/MR imagery and the low costs and high accessibility aspects of ultrasound. This approach enables an *intra*-operative CT/MR imagery from which image guidance can be performed, but without incurring the costs and risks associated with continuous CT/MR imaging.

1.2 Thesis Overview

The intent of this thesis was to present a method for the non-realtime registration of CT, MR, and ultrasound. We realized that the CT/MR and ultrasound registration problem would be attainable through many carefully planned steps thus, we pursued a divide-and-conquer approach to our solution. Our proposal for achieving the registration of CT/MR and ultrasound included the following stages:

- Extract edges from ultrasound to isolate boundaries
- Determine an initial alignment between MR/CT and ultrasound
- Select a set of surface points from MR/CT, and use ICP to match the points to edges in ultrasound. This result provides an initial correspondence between the two data sets
- Use polynomial warps to refine the matching, taking the correspondence returned by ICP as the set of matched points
- Use the final result to warp the entire CT/MR image to ultrasound

As a precursor to CT/MR registration with ultrasound, we decided to provide the reader with a platform of general ultrasound knowledge in Chapter 2, which would ideally lead to a greater understanding and appreciation of the benefits of using ultrasound in registration procedures. The main point that we wanted to relay was the difficulty in ultrasound feature extraction due to high amounts of speckle noise content and insufficiency of intensity-based techniques due to ultrasounds' invariant intensity property.

The first stage of this project was to successfully detect edges in ultrasound images. Implementing the work of Kovesi [23], who proposed phase-based methods versus traditional gradient-based methods for feature detection in "noisy" data, we were able to filter out most of the speckle noise and extract true edges in ultrasound liver data. Kovesi's work was explored because he constructed low level image measures that had a response independent of image illumination and/or contrast. In addition, he showed

how congruency of the local phase over many scales can be used as an illumination and contrast invariant measure of feature significance of points in images. Although Kovesi did not explore results on ultrasound data, Mulet-Parada and Noble [31] used the algorithm on echocardiography images, which inspired our interest to explore the algorithms' performance on ultrasound liver data in Chapter 3.

Upon detecting ultrasound edges to isolate organ boundaries, an initial alignment between CT/MR and ultrasound is necessary. After selecting a set of surface points from the segmented CT/MR data, the Iterative Closest Point Algorithm can be used to match these points to the ultrasound edges. We computed an initial transformation from moment information of an ultrasound image and simulated ultrasound image in Chapter 5. Unfortunately, we did not have corresponding ultrasound data to our CT/MR prostate data hence, the ultrasound and simulated ultrasound result.

ICP works as follows: it uses an initial transform to find the closest points between the transformed image and the original image; finds the closest points; computes a registration; and repeats this process until the mean-square error falls below a defined threshold value. We have produced point sets to test the ICP algorithm for translation and rotation computation. In using Rodrigues' formula to estimate rotation, we looked at the fact that non-linear optimization functions, such as Powell's method, would have to be used to estimate the rotation parameter for ICP. Instead, we used small angle approximations and performed vector analysis to produce a linear rotation and determined the rotation angle that produced the least squares error between the data-model pairings. ICP results are shown for these simple point patterns. In addition, we used Chui and Rangarajan's [5] demo, which was intended to compare ICP to MPM results on simulated and sulcal point sets, to obtain ICP results from various corresponding points in ultrasound and CT liver data.

The 3-D CT/MR registration of prostate data was achieved by using point correspondences and a polynomial warping algorithm to create a dense deformation field, which was used to transform the patients' MR data to an atlas CT image. When corresponding ultrasound data becomes available, we will select a group of surface points from segmented CT/MR, and use ICP to match these points to the ultrasound

data. In future work, the initial correspondence found from ICP will be refined by the elastic warping algorithm used in the CT/MR registration. In addition to the fact that automatic correspondence can be found by ICP, the interpretation tree search method was discussed as another option in Chapter 4. The final analysis will be to warp the entire MR/CT data set to the ultrasound data. In future work, real-time registration methods will be investigated and incorporated into our registration results.

We sought to determine and validate the accuracy of our CT/MR registration by performing numerous experiments in Chapter 6. We created an error analysis algorithm in which we placed a set of corresponding points on reserve and used a separate set of points to compute the registration. The interpolated dense deformation field was then applied to the reserve points and error analysis was performed. The parameters which we varied were polynomial order, quantity of point set data, and type of point distribution. Tabulated and illustrated error and registration results are provided in Chapter 6 along with error validation results.

Chapter 2

Imaging Modalities

2.1 Ultrasound Imagery

Most encounters with ultrasound in the non-medical community occur when expectant mothers receive a scan to inquire about the health and/or sex of their unborn child. Although it may take a medical professional to fully interpret the unintuitive image, ultrasound provides a preview into life formation that would otherwise not be possible. There are other ways of producing images of the inner body, but they are generally either more expensive than ultrasound, or they involve exposure to x-rays or other harmful radiation. Ultrasound machines are comparatively cheap and are considered safe, even for scanning the fetus in utero. They are quick and efficient to use, and not unpleasant for the person being scanned. Their main drawback is the speckled nature of the image that has prohibited its widespread use.

2.1.1 Ultrasound Image Generation

Anatomic imaging with ultrasound is accomplished with a pulse-echo technique. Pulses of ultrasound are generated by a transducer and sent into the patient, where they produce echoes at organ boundaries and within tissues. These echoes return to the transducer, where they are detected and imaged on an instrument. The transducer both generates the ultrasound pulses and detects the returning echoes. The ultra-

sound instrument processes the echo information and generates appropriate dots, which form the ultrasound image on the display. The brightness of each dot corresponds to the echo strength, producing what is known as a gray-scale image. The location of each dot corresponds to the anatomic location of the echo-generating structure. The positional information is determined by knowing the direction of the pulse when it enters the patient and measuring the time for its echo to return to the transducer. If an echo-generating structure is moving, the echo will have a different frequency than that of the pulse emitted from the transducer. This is the Doppler effect, which is put to use in blood flow detection and measurement [25].

There are four principle modes of operation used in ultrasonic procedures and they are A mode, T-M mode, B mode and Doppler. Those in most common use are the brightness mode (B mode, B scan, or gray scale) and motion mode (M mode). The display device used in each case is a cathode ray tube. This tube generates a sharply focused beam of electrons that produces a spot of light on the phosphor-coated inner front screen of the tube. The B scan is a brightness image that represents a cross section of the object through the scanning plane, as if the sound beam cut a section through the tissue. Each individual image is called a frame. Because several frames can be acquired and presented in each second of time, this is called a real-time display. The other common display mode, M mode, is used to show the motion of cardiac structures. It is a display form that illustrates depth versus time [26].

2.2 2-D vs. 3-D Ultrasound

One of the main drawbacks of ultrasound is the fact that only a two-dimensional slice is produced at any given time. The main clinical consequence of this two-dimensional limitation is that it makes it difficult to measure the volumes of structures in the body. This is a common requirement as volumes are used to assess the progression of disease or its response to treatment. They are also useful for calculating drug dosage. Current practice for volume measurement generally involves the use of heuristic formulas. A number of key dimensions are measured and used to provide a volume estimate

by combining them with scaling factors learned from previous experience. These techniques can be incorrect by as much as 50 percent [38].

Prager and Gee's research has focused on developing a more accurate approach to volume measurement, based on 3-D ultrasound. A commercial 2-D ultrasound machine can produce images representing a number of slices through the structure of interest. To find the volume, one should know the relative position and orientation of the slices in 3-D space. This information can be obtained by attaching a position sensing device to the ultrasound scanner probe. Prager, et al. used a magnetic tracking system. The trackers consist of a fixed transmitter with three orthogonal coils of wire (about a 2 inch cube), and a receiver (about half an inch in diameter), similarly containing three smaller coils of wire. The transmitter generates harmless magnetic fields, that are picked up by the receiver and used by the instrument to calculate their relative position and orientation. The transmitter is mounted in a fixed location close to the patient and the receiver is rigidly attached to the probe of the ultrasound machine as shown in Figure 2-1.

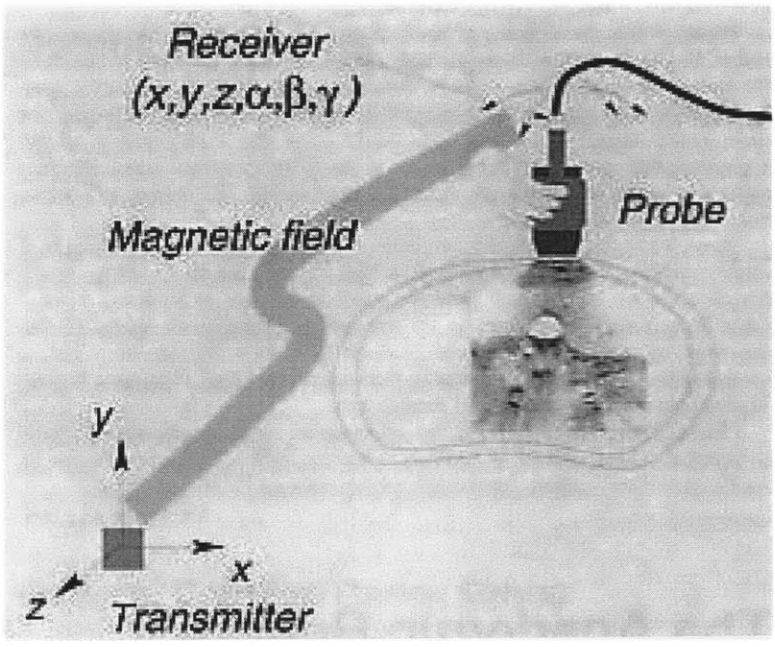


Figure 2-1: The general configuration of a freehand 3-D ultrasound system, showing the probe of the 2-D ultrasound machine and the magnetic position sensor; courtesy of [38]

The arrangement of a conventional 2-D ultrasound machine with a position sensor attached to the probe has other uses. If the images, and matching positions, are recorded fast so that a dense block of data is obtained, it is possible to calculate slices that cannot be obtained by direct scanning, for example, slices entirely beneath the skin. Non-planar slices can also be computed to 'flatten out' curved parts of the body and make certain aspects of the structure easier to understand.

2.2.1 Echocardiography

One of the greatest advantages of using ultrasound is that the image frame rate typically ranges between 10 and 50 images per second, depending on the type of probe and the scanning mode. This allows the imaging of moving tissue and acquisition of ultrasound to be a few seconds compared to several minutes needed in the case of CT or MR [17]. One of the most popular applications of 3-D ultrasound is in echocardiography. When the acquisition is performed with annular arrays, echocardiography provides thinner tomographic slices, which reduces the danger of merging adjacent structures in the direction perpendicular to the scan. While 3-D echography raises difficult issues, it has attracted much attention because 2-D echocardiography is currently used to assess heart functions and performances. 2-D images are successfully used to evaluate the cardiac function and show a wide variety of vascular disorders or myocardial disease but is far from ideal to understand the 3-D morphology of the beating heart and the estimation of the heart cavities, particularly the left ventricle, is far from accurate. Generally volume estimation is derived from the areas of two cross sections of the ventricle, modeling it by a crude shape such as an ellipsoid. The shape of a normal ventricle is far more complex, not to mention the diseased heart. Deriving the exact volume from 3-D echography will largely improve the accuracy of this important feature.

2.3 Biological Effects of Ultrasound

In the development of ultrasound imaging, there was a conscious, deliberate effort to utilize a minimum of ultrasonic energy, to observe clinical evidence of side effects, and to experimentally determine the parameters of safe exposure. Although knowledge of these parameters and of biological effects is incomplete, it has been impossible to substantiate any evidence of harmful effects of diagnostic ultrasound. A wide range of energy levels are used in many medical and non-medical applications of ultrasound. Those used in diagnostic ultrasound are at the low end of the spectrum while higher average intensity levels are used to produce localized deep heating of tissue [26]. The possibility of hazard from exposure to ultrasound depends upon the physical characteristics of the energy being utilized. To reasonably evaluate the probability of harmful effects, the effects at different physical parameters and dosage regimens of ultrasound irradiation must be understood [20].

According to Kremkau [25], it seems reasonable to assume that there is some small risk in diagnostic ultrasound because ultrasound is a form of energy and has the potential to produce a biological effect that could constitute risk. Knowledge of the effects of ultrasound come from several sources including experimental observations in cell suspensions and cultures, plants, and experimental animals; epidemiologic studies with humans; and an understanding of interaction mechanisms, such as heating and cavitation. Information from *in vitro* and *in vivo* experimental studies has yielded no known risks in the use of diagnostic ultrasound. Thermal and mechanical mechanisms have been considered, but do not appear to operate significantly at diagnostic intensities. Currently, there is no known risk associated with the use of diagnostic ultrasound. Experimental animal data have helped to define the intensity-exposure time region in which bioeffects can occur. However, physical and biological differences between the two situations make it difficult to apply results from one risk assessment to the other. In the absence of known risk, but recognizing the possibility that bioeffects could be occurring that are subtle, of low incidence, or delayed, a conservative approach to the medical use of ultrasound is recommended by Kremkau.

2.4 Speckle Noise

Acoustic speckle results from interference effects of scattered sound from the distribution of scatterers in the tissue. Because the ultrasound pulse encounters several scatterers at any point in its travel, several echoes are generated simultaneously. These may arrive at the transducer in such a way that they reinforce (constructive interference) or partially or totally cancel (destructive interference) each other [25]. The result is a dot pattern that represents an interference pattern of the scatterer distribution scanned as opposed to the direct representation of individual scatterers. This acoustic speckle is similar to the speckle phenomenon observed when a laser is shown on a wall.

2.4.1 Filtering

One of the most traditional methods of reducing speckle noise is to apply median filtering. In median filtering, the gray level of each pixel is replaced by the median of the gray levels in a neighborhood of that pixel, instead of by the average. In order to apply median filtering in a neighborhood of a pixel, the pixel values and its neighbors are sorted, the median is computed and assigned to the pixel value. Unfortunately, median filtering is not robust enough to satisfactorily reduce the speckle in ultrasound. Thus, we decided to explore more robust orientable odd and even log-Gabor wavelet filters for enhancement as introduced by Knutsson and Granlund [21]. Gabor filters are defined by harmonic functions modulated by a Gaussian distribution. Gabor filters bear some similarity to Fourier filters, but (by the Gaussian damping terms) are limited to certain frequency bands. With a judicious choice of frequencies, e.g. by octaves, a succession of Gabor filters can be assimilated into a wavelet transform, and do an excellent job in image or information compaction.

2.5 Magnetic Resonance

Magnetic Resonance Imaging (MRI) is an imaging technique used primarily in medical settings to produce high quality images of the inside of the human body. MRI is based on the principles of nuclear magnetic resonance (NMR), a spectroscopic technique used by scientists to obtain microscopic chemical and physical information about molecules. MRI started out as a tomographic imaging technique, that is, it produced an image of the NMR signal in a thin slice through the human body. MRI has advanced beyond a tomographic imaging technique to a volume imaging technique [16].

Magnetic Resonance can produce volumetric renderings of anatomical structures based on the response of their components' atomic nuclei to an external magnetic field. MR volumes are produced and processing is taken over tissues located in adjacent transverse slices of anatomy, which are stacked to form a volumetric image. The *in-slice* resolution is typically better that the resolution between slices. A range of dimension is available for various MR acquisitions. For example, high resolution MR is typically composed of slices containing 256×256 pixels ($0.9^2 mm^2$) of area, and the slices are spaced by 1 mm. Low resolution MR has the same number of pixels as high resolution MR, but their pixel areas are larger ($1.25^2 mm^2$), and their slices are separated by 4 mm [36]. Within slices, pixel intensities map the alignment intensity of hydrogen nuclei (protons) at those locations to an external, uniform magnetic field. Variations in the imaging technique emphasize different characteristic magnetic relaxation times (denoted T1 and T2) of aligned protons. These properties relate directly to the proton density of the nuclei, not the tissue density. Thus, MR renders soft tissue variations very well due to their high hydrogen concentrations and differing chemical composition. On the other hand, as hard bone is relatively anhydrous, it is given the same low intensity as air [7].

MR imaging is subject to a number of problems including intensity variations and geometric distortions. The intensity variations are the product of non-uniformities in the imaging apparatus and inhomogeneities in the magnetic fields and RF pulses

used in imaging [37]. Geometric distortions are most pronounced at boundaries between regions of significantly different magnetic susceptibility (e.g. soft tissue/bone, tissue/air).

2.6 Computed Tomography

In addition to MR, Computed Tomography (CT) can be used to produce volumetric images of anatomical structures by the stacking of slices produced from measurements taken over transverse slices of anatomy [44]. A range of dimension is available for various CT acquisitions. High resolution CT is typically composed of slices containing 256×256 or 512×512 pixels of area $0.6^2 - 0.9^2 mm^2$, with slices spaced by 1.5 mm. Low resolution CT has slices equal to the same number of pixels, with pixel areas being $0.65mm^2$, and their slices separated by 4 mm [36].

CT slices are produced by rotating an apparatus around the patient which casts a plane of x-rays through the patient in a number of different directions. Processing is then performed such that the CT slice gives a map of the x-ray attenuating properties of the tissue section at each position. Conventionally, each of these attenuation coefficients are rescaled by a linear function to produce an intensity called a CT *number*. CT intensities differ from those given by fluoroscopy, as CT is processed with x-rays of energy less than those used in fluoroscopic radiography. This has the effect that there is less contrast between the attenuation coefficients of bone and soft tissue. Modern CT imaging does not suffer from any significant geometric distortions, and is often used as the "gold standard" basis for trying to correct geometric distortion in MR [7].

2.7 Summary

We deemed a detailed analysis of ultrasound necessary because we hypothesize that it is ideally the intra-operative modality of choice for real-time registration. With the notable attributes of ultrasound in it's low visualization state, we recognize that a

remarkable enhancement of ultrasound could revolutionize the field of image-guided therapy. Applicable procedures include cardiac surgery as well as numerous surgeries that are performed in the abdominal area. We realize that this task is far from trivial and our aim is to use other imaging techniques such as CT and MR to enhance ultrasound so that it is suitable for various image-guided procedures.

Chapter 3

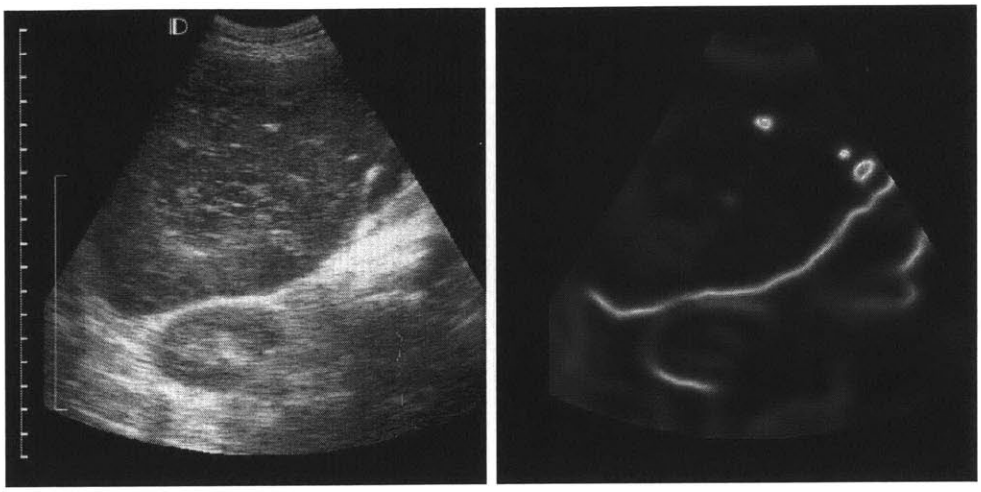
Image Features

3.1 Acoustic Feature Detection

Image enhancement results can be achieved through segmentation, which consists of a partitioning of an image into regions of interest. Regions can be extracted by re-grouping voxels which form homogeneous areas with respect to a given homogeneity criterion, or by detecting edges, which are points of transition between two homogeneous areas [32]. Because ultrasound boundaries are not clearly detectable due to high speckle content and low contrast, finding ways to detect edges seemed reasonable. Edges are described as the sharp intensity changes in a gray-level input image. Among commonly used edge detectors are the Marr-Hildreth Laplacian of Gaussian, Canny, Sobel, and Prewitt. Applying any of these operators to the image reduces the sensory input to an array of edge points, where a 1 in a pixel indicates an edge point, and all other points are 0 [13]. Figures 3-1 and 3-2 illustrates segmented ultrasound and CT data. Note that the ultrasound image in this example was enhanced to make edges easier to detect.

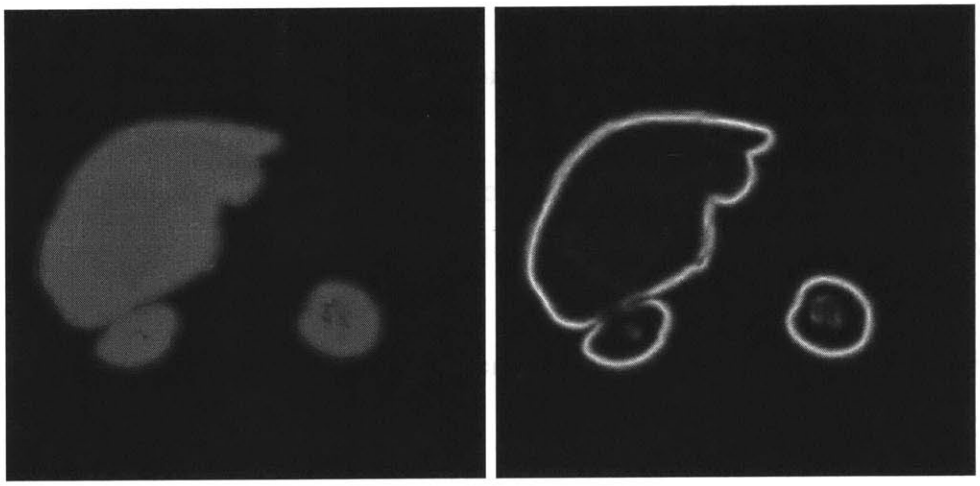
3.2 Invariant Measures in Images

Kovesi [23] defined invariant measures as features that remain constant over a wide range of viewing conditions. He noted that although some effort has been devoted to



(a) Ultrasound Image (b) Ultrasound Edges

Figure 3-1: Ultrasound image and its corresponding edge information



(a) CT image (b) CT Edges

Figure 3-2: Segmented surface of CT and it's corresponding edge information

investigating invariant measures of higher level structures in images, little attention has been paid to the invariant quantities that may exist in low-level or early vision tasks such as feature detection or feature matching. The two main properties of invariant measures are that they must be dimensionless and they must represent some meaningful and useful quality. Because objects can appear with arbitrary orientation and spatial magnification along with arbitrary brightness and contrast, the search for invariant quantities is very important for computer vision. Kovesi sought to construct low-level image measures that have a response independent of illumination and/or contrast.

As Kovesi noted, there are many algorithms that involve the minimization of some *energy*; often the *energy* is defined to be the addition of many components which have different units. For example, energy minimizing splines (snakes) are usually formulated in terms of the minimization of an energy that is made up of an intensity gradient term and a spline bending term [19]. These two quantities while meaningful, are not dimensionless. This means that for energy minimizing splines to be effective, their parameters have to be tuned carefully for each individual application. The parameters are used to balance relative importance of individual components of the overall energy. If say, the overall image contrast was halved one would need to double the weighting applied to the intensity gradient term to retain the same snake behavior. If one was to somehow replace the intensity gradient and spline bending terms with dimensionless quantities that represented, in some way, the closeness of the spline to the feature and the deformation of the spline, one would be able to use fixed parameters over wider classes of images [23].

3.3 Gradient-based Feature Detection

The majority of work in the detection of low-level image features has been concentrated on the identification of step discontinuities in images using gradient-based operators. One of the most widely used gradient-based detectors used is that of Canny [4] who developed infinite impulse response (IIR) filters for the detection of edges via

zero-crossings. Although Canny's edge detector works well on 1-D step edges in the presence of noise, difficulties arise at 2-D features such as corners and junctions where the intensity gradient becomes poorly defined. As shown in Figure 3-3, much of the speckle is still present in the ultrasound image after the Canny Detector is applied. Thresholding hysteresis was applied with multiple pairs of upper and lower threshold values.

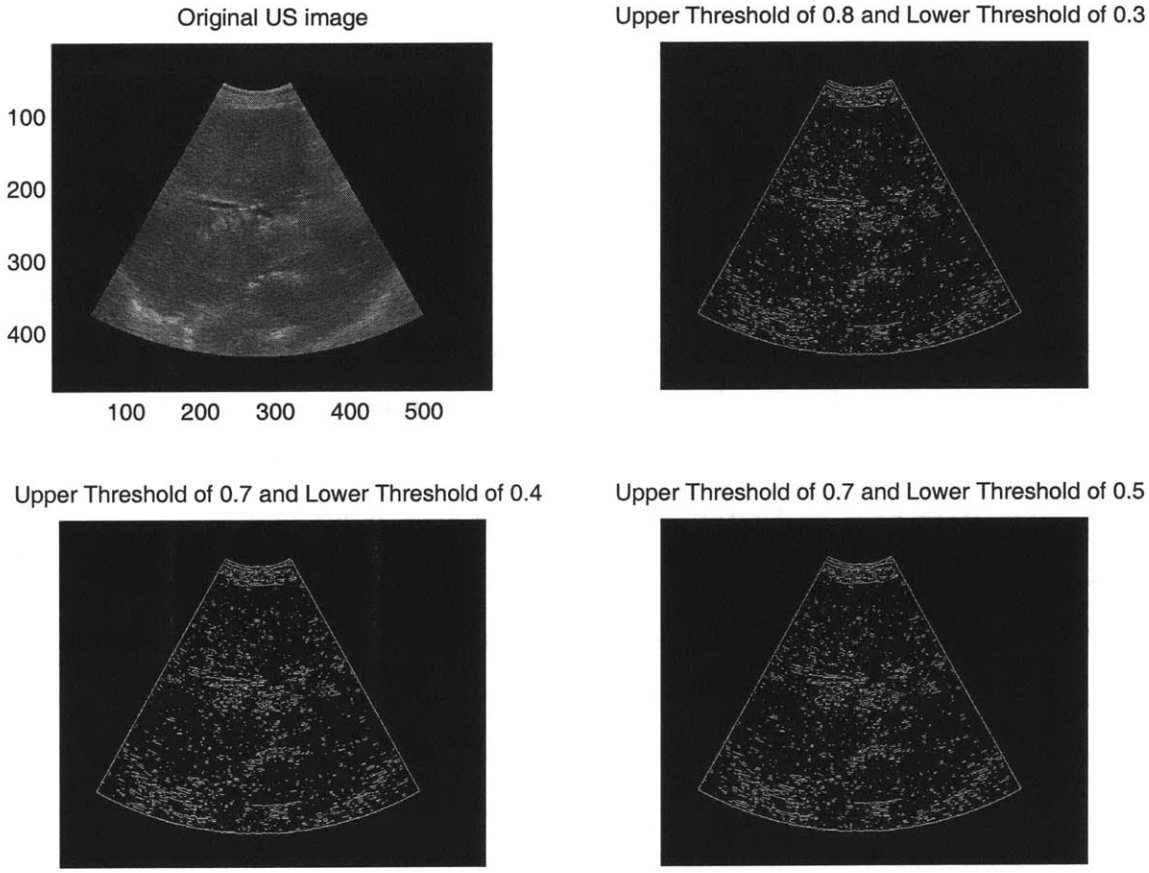


Figure 3-3: Canny edge results after hysteresis thresholding

Kovesi stressed that a major problem with gradient-based operators is that they assume edges are step discontinuities. However, as Perona and Malik [35] pointed out, many image features are represented by a combination of step, delta, roof and ramp profiles. A second problem with gradient-based edge detectors is that they typically characterize edge strength by the magnitude of the intensity gradient. Thus, the perceived strength or significance of an edge is sensitive to illumination and spa-

tial magnification variations. Any gradient-based edge detector will need to use an appropriately modified threshold. However, generally one does not know in advance the level of contrast present in an image or its magnification. The image gradient values that correspond to significant edges are usually determined empirically. Canny introduced the idea of thresholding hysteresis which has proved to be a useful heuristic for maintaining the continuity of threshold edges, though one then has the problem of determining two threshold levels.

3.4 Local Energy and Phase Congruency

The local energy model of feature perception is not based on the use of local intensity gradients for feature detection, but rather assumes that features are perceived at points in an image where the Fourier components are maximally in phase [1] [23]. Step, line and roof edges, and Mach bands are among the wide range of feature types that give rise to points of high phase congruency. Investigations into the phenomenon of Mach bands by Morrone, et al. [29] led to the development of the local energy model. Mach bands are illusory bright and dark bands that appear on the edges of trapezoidal intensity gradient ramps, for example, on the edges of shadows. Morrone, et al. showed that the Fourier components of a signal are maximally in phase, though not exactly in phase, at the points where we perceive Mach bands. These results led to their hypothesis that we perceive features in images at points of high phase congruency.

With phase data demonstrated as being important in the perception of images it is natural that one should pursue the development of a feature detector that operates on the basis of phase information. From their work on Mach bands Morrone and Owens [28] recognized that the local energy model had applications in feature detection for computer vision. To illustrate how this works, Figure 3-4 shows a simple test image that contains a variety of features at different contrasts. Figure 3-5(a) shows

[1] Phase refers to local phase in this document. We are concerned with the local phase of the signal at some position x.

the output of a simple gradient-based edge detector (here, the Sobel operator). Note that the output depends on the relative contrast of the edge, and that the output for line features is two edges, one on either side of the line. Figure 3-5(b) shows the output of the local energy (or phase congruency) detector. Here we note that the output is a uniform response, regardless of the type or contrasts of the feature involved.

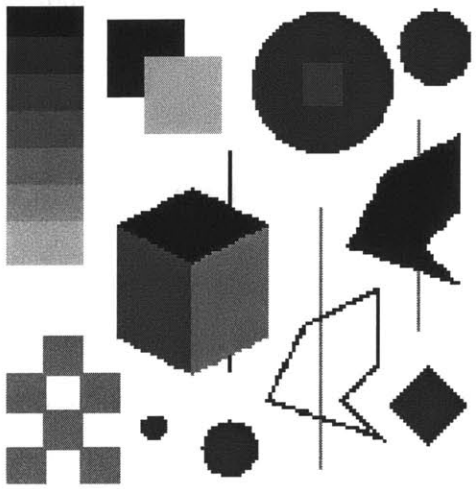
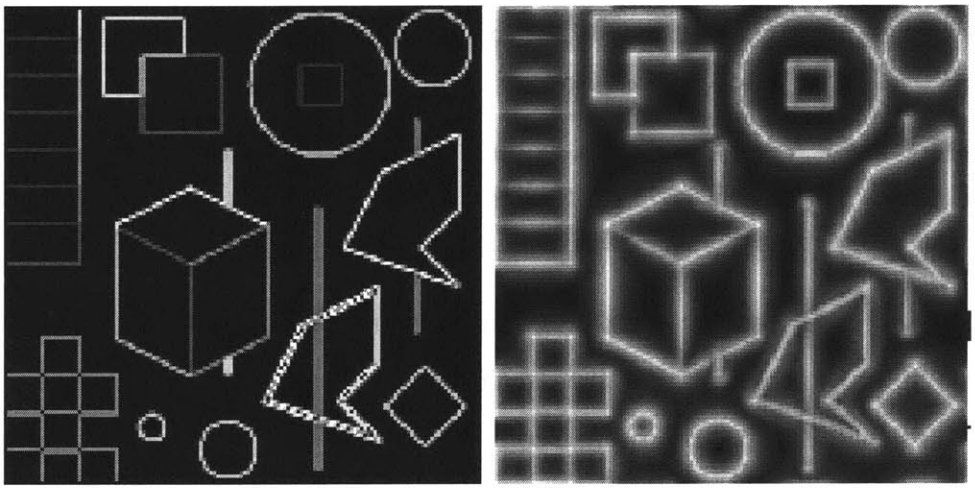


Figure 3-4: A Simple Test Image; courtesy of Kovesi [24]



(a) The Output of the Sobel Operator

(b) The Phase Congruency Map

Figure 3-5: Comparison of Sobel Operator and PC; courtesy of Kovesi [24]

3.4.1 Phase Congruency Definition

Kovesi developed the phase congruency function in 1-D from the Fourier series expansion of a signal, I at some location, x,

$$I(x) = \sum_n A_n \cos(n\omega x + \phi_{n_0}) \tag{3.1}$$

$$= \sum_n A_n \cos(\phi_n(x)), \tag{3.2}$$

where A_n represents the amplitude of the n^{th} cosine component, ω is a constant (usually 2π), and ϕ_{n_0} is the phase offset of the n^{th} component (the phase offset also allows sine terms in the series to be represented). The function $\phi_n(x)$ represents the *local* phase of the Fourier component at position x.

Morrone, et al. defines the phase congruency function as

$$PC(x) = max_{\bar{\phi}(x)\in[0,2\pi]} \frac{\sum_n A_n \cos(\phi_n(x) - \bar{\phi}(x))}{\sum_n A_n}. \tag{3.3}$$

The value of $\bar{\phi}(x)$ that maximizes Equation 3.3 is the amplitude weighted mean local phase angle of all the Fourier terms at the point being considered. Taking the cosine of the difference between the actual phase angle of a frequency component and this weighted mean, $\bar{\phi}(x)$, generates a quantity approximately equal to one minus half this difference squared (the Taylor expansion of $\cos(x) \approx 1 - x^2/2$ for small x). Thus finding where phase congruency is a maximum is approximately equivalent to finding where the weighted variance of local phase angles, relative to the weighted average local phase, is minimum.

3.4.2 Phase Congruency via Local Energy

Due to the fact that phase congruency is a non-trivial quantity to calculate, Venkatesh and Owens [43] set out to show that points of maximum phase congruency can be calculated equivalently by searching for peaks in the local energy function. The local energy function is defined for a 1-D luminance profile, $I(x)$, as the modulus of a

complex number,

$$E(x) = \sqrt{F^2(x) + H^2(x)}, \tag{3.4}$$

where $F(x)$ is $I(x)$ with the DC Component removed and $H(x)$ is the Hilbert transform of $F(x)$ (a 90 degree phase shift of $F(x)$). Venkatesh, et al. proved that energy is equal to phase congruency scaled by the sum of the Fourier amplitudes, that is

$$E(x) = PC(x) \sum_n A_n. \tag{3.5}$$

Thus the local energy function is directly proportional to the phase congruency function, so peaks in local energy will correspond to peaks in phase congruency.

Rather than compute local energy via the Hilbert transform of the original luminance profile one can calculate a measure of local energy by convolving the signal with a pair of filters in quadrature. This gives two signals, each being a bandpassed version of the original, and one being a 90 degree phase shift of the other. Odd and even-symmetric Gabor functions can be used for the quadrature pair of filters. Local energy is defined by

$$E(x) = \sqrt{(I(x) * M^e)^2 + (I(x) * M^o)^2}, \tag{3.6}$$

where M^e and M^o denote the even and odd symmetric filters in quadrature.

While the use of the local energy function to find peaks in phase congruency is computationally convenient it does not provide a dimensionless measure of feature significance as it is weighted by the sum of the Fourier component amplitudes, which have units *lux*. Thus, like derivative based feature detectors, local energy suffers from the problem that we are unable to specify in advance what level of response corresponds to a significant feature. Despite this, local energy remains a useful measure in that it responds to a wide range of feature types. Phase congruency, on the other hand, is a dimensionless quantity. We obtain it by normalizing the local energy function; dividing energy by the sum of the Fourier amplitudes. Values of phase con-

gruency vary from a maximum of 1, indicating a very significant feature, down to 0 indicating no significance. This property offers the promise of allowing one to specify universal feature thresholds, that is, we could set thresholds before an image is seen and have truly automated feature detection [23].

3.5 Calculating Phase Congruency

One issue we must also consider is that phase congruency as defined in Equation 3.3 does not take into account the spread of frequencies that are congruent at a point. Significant feature points are presumably ones with high information content; a point of phase congruency indicates a point of high information content only if we have a wide range of frequencies present. We do not gain much information from knowing the phase congruency of a signal which has only one frequency component.

Kovesi constructed a weighting function that devalued phase congruency at locations where the spread of filter responses is narrow. This spread is given by,

$$s(x) = \frac{1}{N}\left(\frac{\sum_n A_n(x)}{\varepsilon + A_{max}(x)}\right), \tag{3.7}$$

where N is the total number of scales considered, $A_{max}(x)$ is the amplitude of the filter pair having maximum response at x. In addition, a weighting function is defined by applying a sigmoid function to the filter response spread value,

$$W(x) = \frac{1}{1 + e^{\gamma(c - s(x))}}, \tag{3.8}$$

where c is the threshold value for which phase congruency values become penalized and γ is a gain that controls sharpness of the cutoff.

Because phase localization was poor for blurred images, Kovesi introduced a more sensitive phase deviation function on which to base the calculation of phase congruency,

$$\Delta\Phi(x) = \cos(\phi_n(x) - \bar{\phi(x)}) - |\sin(\phi_n(x) - \bar{\phi(x)})|. \tag{3.9}$$

3.5.1 Calculating Phase Congruency via Wavelets

Linear phase filters preserve phase information and thus can be used as non-orthogonal wavelets that are in symmetric/antisymmetric quadrature pairs. Following a similar approach to Morlet et al. [27], Kovesi used logarithmic Gabor functions as suggested by Field [9] partly because the filters have a Gaussian transfer function when viewed on the logarithmic frequency scale. In addition, Log Gabor filters allow arbitrarily large bandwidth filters to be constructed while still maintaining a zero DC component in the even-symmetric filter. On the linear frequency scale, the Log Gabor function has a transfer function of the form

$$\mathcal{G}(\omega) = e^{\frac{-(\log(\omega/\omega_0))^2}{2(\log(\kappa/\omega_0))^2}}, \tag{3.10}$$

where ω_0 is the filter's center frequency. To obtain constant shape ratio filters the term κ/ω_0 must also be held constant for varying ω_0. For example, a κ/ω_0 value of 0.75 will result in a filter bandwidth of approximately one octave (difference between two frequencies) and a value of 0.55 will result in a two-octave bandwidth.

Letting I denote the signal and M_n^e and M_n^o denote the even-symmetric (cosine) and odd-symmetric (sine) wavelets at a scale n, Kovesi showed that we can think of the responses of each quadrature pair of filters as forming a response vector,

$$[e_n(x), o_n(x)] = [I(x) * M_n^e, I(x) * M_n^o]. \tag{3.11}$$

The amplitude of the transform at a given wavelet scale is given by

$$A_n(x) = \sqrt{e_n(x)^2 + o_n(x)^2}, \tag{3.12}$$

and the phase is given by

$$\phi_n(x) = \arctan 2(e_n(x), o_n(x)). \tag{3.13}$$

At each point x in a signal, we will have an array of these response vectors, one

vector for each scale of filter. These response vectors form the basis of our localized representation of the signal, and they can be used in exactly the same way as Fourier components to calculate phase congruency.

The design of the wavelet filter bank needs to be such that the transfer function of each filter overlaps sufficiently with its neighbors so that the sum of all the transfer functions forms a relatively uniform coverage of the spectrum. This is necessary since phase congruency is of interest only if it occurs over a wide range of frequencies.

The fact that phase congruency becomes undefined if all the Fourier amplitudes are very small can be addressed by adding a small positive constant, ε, to the denominator of Equation 3.3. Thus,

$$PC(x) = \frac{E(x)}{\sum_n A_n(x) + \varepsilon}, \tag{3.14}$$

where $E(x)$ is defined in Equation 3.4. The appropriate value of ε depends on the precision with which we are able to perform convolutions and other operations on our signal; it does not depend on the signal itself. We chose to use 0.0001 as our ε value in the calculations at the end of this chapter.

3.5.2 Extension to 2-D

A logical way to construct 2-D filters in the frequency domain is to use polar-separable 2-D Gaussians. In the radial direction, along the frequency axis, the filters are designed in the same way as the 1-D filters (that is, log Gaussians with geometrically increasing centered frequencies and bandwidths). In the angular direction, the filters have Gaussian cross-sections, where the ratio between the standard deviation and the angular spacing of the filters is some constant. This ensures a fixed length-to-width ratio of the filters in the spatial domain. Thus, the cross-section of the transfer function in the angular direction is

$$G(\theta) = e^{-\frac{(\theta-\theta_0)^2}{2\sigma_\theta^2}}, \tag{3.15}$$

where θ_0 is the orientation angle of the filter, and σ_θ is the standard deviation of the Gaussian spreading function in the angular direction. This is set to be some fixed ratio of the orientation spacing between the filters to ensure even coverage of the spectrum in all orientations. Kovesi found that a filter orientation spacing of 30 degrees provides a good compromise between the need to achieve an even spectral coverage while minimizing the number of orientations. The use of more filter orientations does not change the quality of the results significantly. The final arrangement of filters results in a rosette of overlapping polar-separable 2-D Gaussians in the frequency plane.

To combine the data over all the orientations, Kovesi gives the following 2-D phase congruency equation,

$$PC_2(x) = \frac{\sum_o \sum_n W_o(x) \lfloor A_{no}(x) \Delta\Phi_{no}(x) - T_o \rfloor}{\sum_o \sum_n A_{no}(x) + \varepsilon}. \tag{3.16}$$

where o denotes the index over orientations. This equation results from calculating the energy, $E(x)$, in each orientation, compensating for the influence of noise T_o, applying the weighting for frequency spread, and summing over all orientations. This sum is then normalized by dividing by the sum over all orientations and scales of the amplitudes of the individual wavelet responses at that location in the image.

3.6 Scale via High-Pass Filtering

As far as phase congruency is concerned, the natural scale parameter to vary is the size of the window in the image over which we perform the local frequency analysis. In the context of the use of wavelets to calculate phase congruency, the scale of analysis is specified by the spatial extent of the largest filter in the wavelet bank. In this approach, high-pass filtering is used to specify the analysis scale. Low-frequency components (those having wavelengths larger than the window size) are cut out while high-frequency components are left intact.

Kovesi proposes that multiscale analysis be done by considering phase congruency of differing high-passed versions of an image.The high-pass images are constructed

from the sum of band-passed images, with the sum ranging from the highest frequency band down to some cutoff frequency. With this approach, no matter what scale we consider, all features are localized precisely and in a stable manner. There is no drift of features that occurs with low-pass filtering. All that changes with analysis at different scales is the relative significance of features. This provides an ideal environment for the use of coarse-to-fine strategies.

3.7 Phase Congruency and Edge Results in Ultrasound Images

A number of parameters could vary in the calculation of phase congruency for edge extraction in ultrasound liver data. Our choice of parameters for ultrasound organ boundary detection is provided below. We conducted numerous trials with filter wavelengths ranging from 3 to 20 pixels, wavelet scales ranging from 3 to 5, and the fractional measure of frequency spread ranging from 0.4 to 0.6. Based upon our observations, we decided that the parameters listed below best accomplished our goal of detecting edges in ultrasound. Kovesi did not test his algorithm on ultrasound images and Mulet-Prada et al. [31] tested Kovesi's algorithm on echocardiographic data in the temporal domain. We investigated Kovesi's algorithm and reported on its robustness to high speckle noise content and low contrast in ultrasound liver data in a non-realtime manner. Our stages were as follows:

- We determined an initial filter wavelength of 10 pixels which were analyzed in multiples of the scaling factor (3) between successive filters
- We analyzed 4 wavelet scales
- Six orientations between 0 and 180 degrees were analyzed in increments of 30 degrees
- The fractional measure of the frequency spread was analyzed at 0.4

- The number of standard deviations of the noise energy beyond the mean used to set the noise threshold was 2
- The ratio of angular interval between filter orientations was 1.2
- The ratio of the standard deviation of the Gaussian describing the log Gabor filter's transfer function in the frequency domain to the filter center frequency was set to 0.55
- A factor of 10 was used to control the sharpness in the sigmoid function used to weight phase congruency and frequency spread

Upon computing the phase congruency at each filter orientation, non-maximal suppression was applied to the image in order to thin the ridges of the gradient magnitude by suppressing all values along the line of the gradient that were not peak values of the ridge. Non-maximal suppression takes a radius as input whose distance in pixel units are viewed on each side of each pixel when determining whether it is a local maxima or not. Kovesi suggested using non-maximal suppression values between 1.2 and 1.5 and we chose a value of 1.3 for our results. Finally, hysteresis thresholding was applied to the ultrasound image to obtain more prominent organ boundaries. We opted to provide only the best algorithm results from our analysis of ultrasound liver data. Figure 3-6 illustrates results of implementing algorithm with the same parameters used by Kovesi on non-medical data. The parameters remained as defined above with the exception of a minimum λ of 3, scaling factor between successive filters of 2, and non-maximal suppression of 1.5. We concluded that these parameters were not optimal for edge detection in our ultrasound liver data due to an insufficient filter λ and over thresholding.

As a result, we investigated various parameter changes in Kovesi's algorithm and provided our best edge extraction results. The energy computations of the ultrasound liver data is shown in Figure 3-7. Figures 3-8 and 3-9 provide results of computing phase congruency and performing non-maximal suppression on the ultrasound image shown in Figure 3-6. In addition, we provide the mean squared energy values of the

smallest scale filter at each orientation in Table 3.1. The final edge extraction result after applying hysteresis thresholding values with an upper limit of 0.25 and lower limit of 0.075 is shown in Figure 3-10.

We concluded that Kovesi's algorithm provided a substantial improvement for ultrasound liver data edge detection compared to the Canny detector (Figure 3-3). The speckle content was sufficiently minimized and organ boundaries that were not visible using the Canny method became apparent. Almost the entire boundary of the liver is detected in Figure 3-10 using our defined parameters including the λ of 10. The liver is the large elliptical shaped object in the lower left hand area of the ultrasound images and it encompasses approximately one quarter of the entire ultrasound image. We note that small segments of the liver boundary are missing in the edge results which may have resulted from over thresholding the image at the cost of reducing speckle noise detection. Furthermore, we show the result of using a λ of 8 and threshold values of 0.3 and 0.1 in Figure 3-11, which resulted in failed detection of a significant part of the liver edge but, an insignificant noise presence. The result further validates our choice of a λ of 10 to obtain a reasonable compromise between edge detection and reduction of speckle noise.

Mean Energy Squared Values for Orientation Angles						
Angle (degrees)	0-30	30-60	60-90	90-120	120-150	150-180
Mean Energy Squared	0.1435	0.1242	0.2161	0.3127	0.2215	0.1246

Table 3.1: Mean Energy squared values recorded with smallest scale filter at each orientation

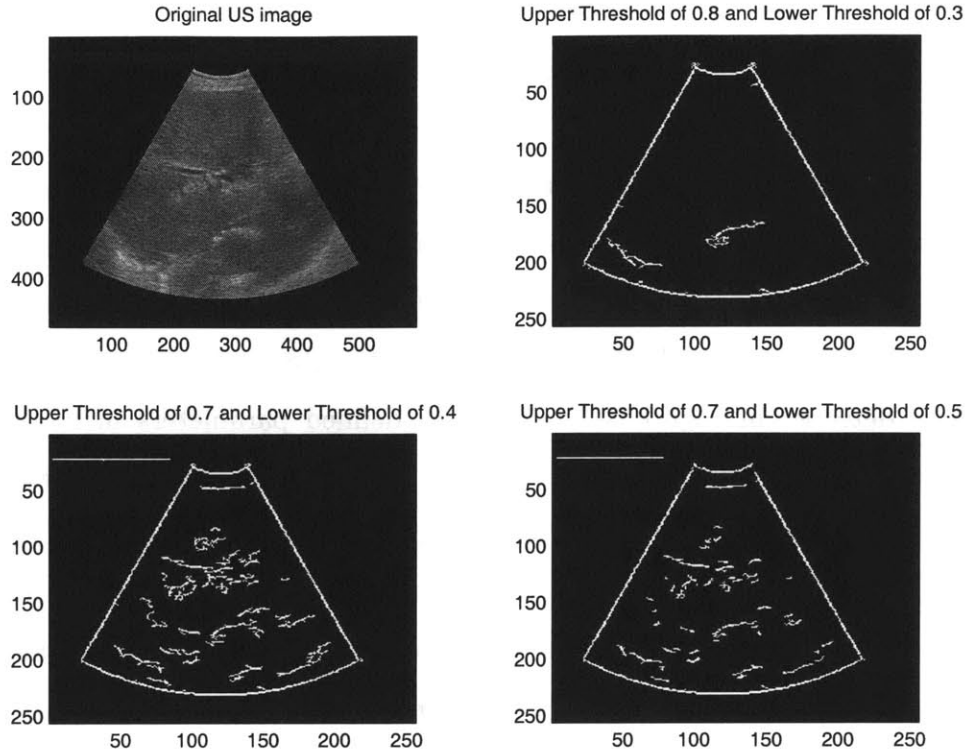
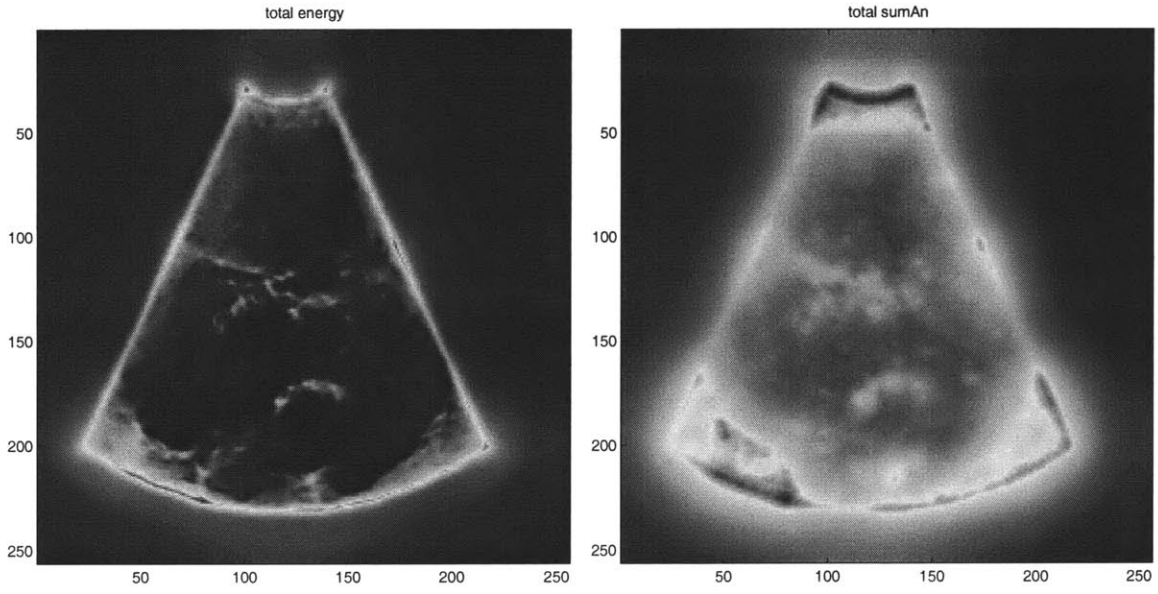


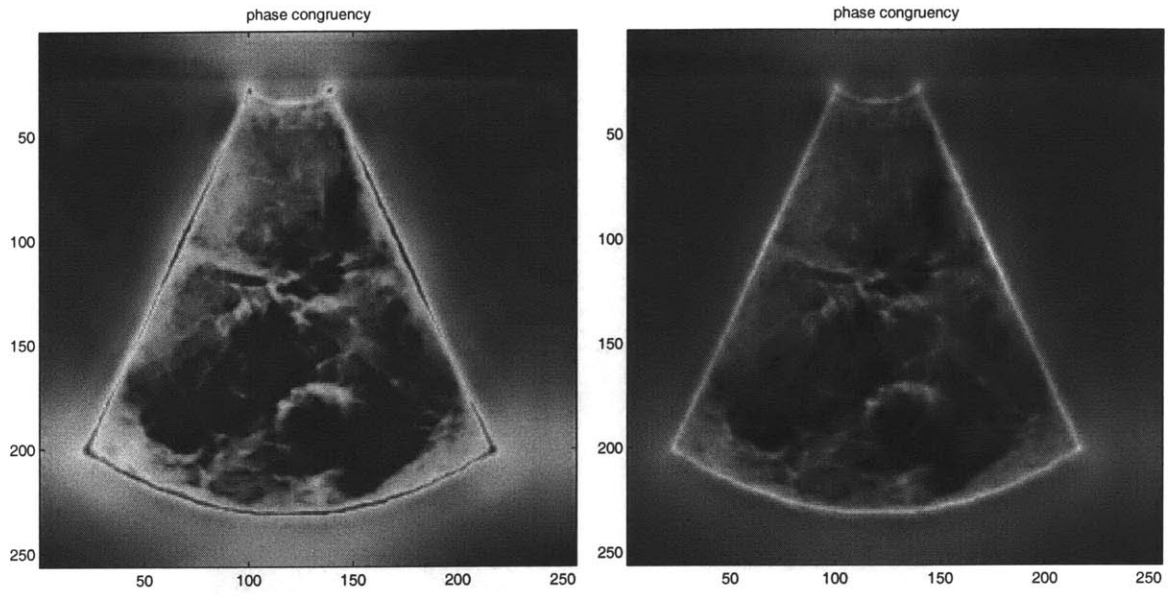
Figure 3-6: Edge results from phase congruency with hysteresis thresholding of an ultrasound image



(a) Total energy in ultrasound

(b) Energy in ultrasound proportional to phase congruency

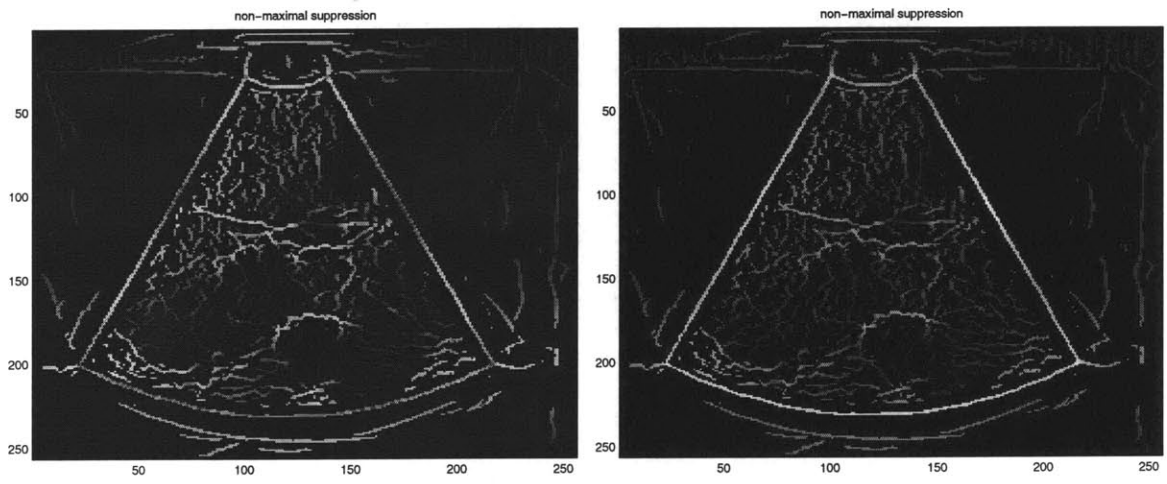
Figure 3-7: Energy calculations in ultrasound



(a) Phase congruency with fractional measure of frequency spread = 0.4

(b) Phase congruency with fractional measure of frequency. spread = 0.6

Figure 3-8: Phase Congruency Plots



(a) Non-Maximal Suppression with 0.4 cutoff

(b) Non-Maximal Suppression with 0.6 cutoff

Figure 3-9: Effects of NMS on Phase Congruency Plots

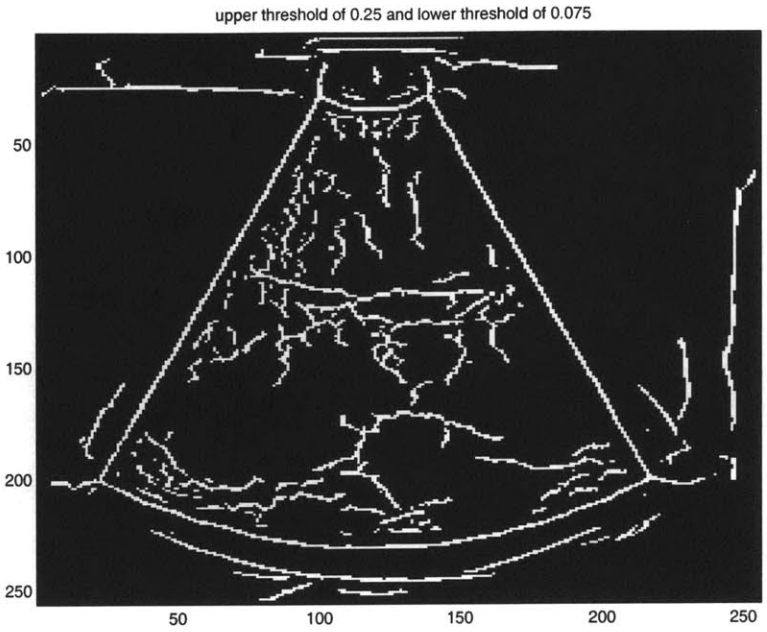


Figure 3-10: Ultrasound edges from phase congruency using $\lambda = 10$

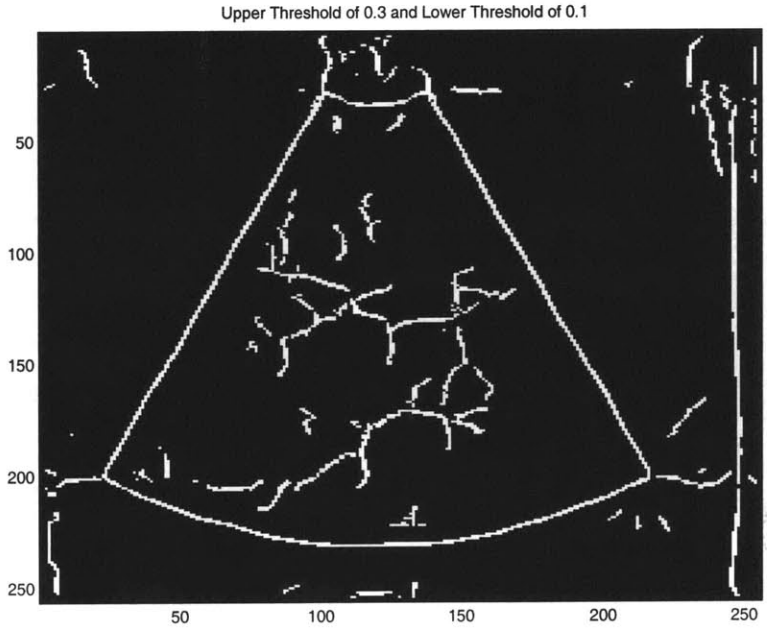


Figure 3-11: Ultrasound edges from phase congruency using $\lambda = 8$

Chapter 4

Correspondence

4.1 Point Mapping

The point- or landmark-mapping technique is the primary approach currently taken to match two images whose type of misalignment is unknown. The general method for point mapping consists of three stages. In the first stage, features in the image are computed. In the second stage, feature points in the reference image, often referred to as *control points*, are corresponded with feature points in the data image. In the last stage, a spatial mapping, usually two 2-D polynomial functions of a specified order (one for each coordinate in the registered image) are determined using these matched feature points. Resampling of one image onto the other is performed by applying spatial mapping and interpolation. However, there is another group of point-mapping methods used for images whose misalignment is a small rigid or affine transformation, but which contain significant amounts of local uncorrected variations. In this case, the similarity measures between the possible matches become unreliable. Point-mapping methods can overcome this problem by the use of feedback between the stages of finding the correspondence between control points and finding the optimal transformation [12].

4.1.1 Control Points

Control points can either be extrinsic or intrinsic. Intrinsic control points are markers in the image which are not relevant to the data itself. In medical imaging, identifiable structures, called fiducial markers, are placed in known positions in the patients to act as reference points. Although intrinsic points are preferable, they cannot always be used. For example, precisely placing markers internally is not always possible in diagnostic images [40]. Control points that are extrinsic are determined from the data, either manually or automatically. Manual control points are recognized by human intervention, such as identifiable landmarks or anatomical structures, and have several advantages. Features chosen by a medical expert are likely to be uniquely found in both images and more tolerant of local distortions. Figure 4-1 shows 534 manually chosen corresponding point sets in CT and MR data, which were completed by Julian Rathert, M.D.

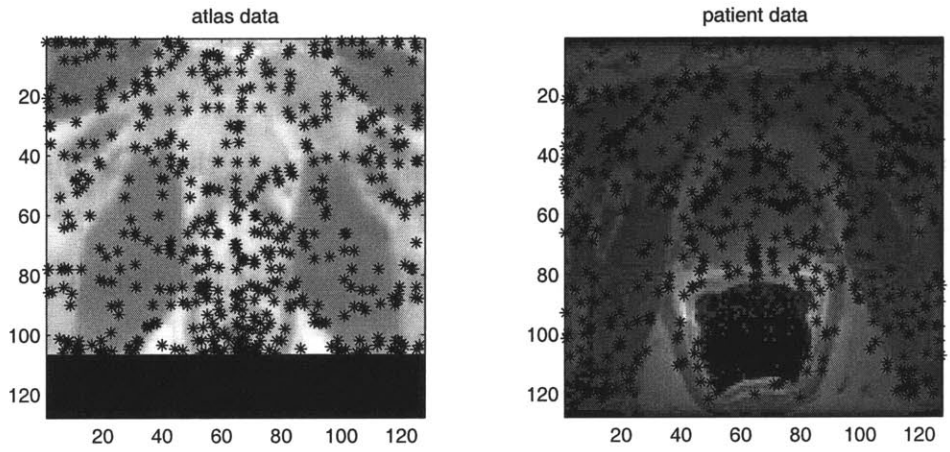


Figure 4-1: Manually selected corresponding points in CT/MR

In cases where there is a large amount of data, manual selection is not feasible. However, a sufficient number must be detected to calculate the transformation. Therefore, many applications use automatic location of control points. Techniques for matching automatically determined points combine the matching of features and the determination of the optimal transformation, which may involve, clustering, relaxation, matching of convex hull edges of the two sets, and matching of minimum

spanning trees [11]. Instead of mapping each point individually, these techniques map the set of points in one image onto the corresponding set in the second image. Consequently the matching solution uses the information from all points and their relative locations which can turn out to be computationally expensive.

4.2 Interpretation Tree Search

Certainly, manual selection of corresponding points can become an impractical task in clinical applications. In our case, surface points from CT/MR segmentation results should be matched to edge point extracted from ultrasound. Because correspondence quantities can reach orders of 100 or more for accurate registration computations, an automated correspondence procedure is worthy of investigation. In the search for correspondence between localized model and data features, one central issue is cost manageability in developing an effective approach to feature-based recognition and localization. Grimson [13] suggested using geometric constraints to reduce the search and explore the correspondence space of data-model pairings.

The correspondence space consists of all possible pairings between the data and model features. To search the entire space for correct interpretations would be both tedious and inefficient. One way to direct the search is to transform the space into a tree of interpretations. The tree is started by arbitrarily ordering the data features and hypothesizing that the first data feature, f_1, is in correspondence with the first model feature, F_1. This becomes the first point on the first axis of the search space and subsequently, a node on the tree. Keep in mind that each data feature is compared to each model feature, $F_j, j = 1, ..., m$, which introduces a set of nodes at the same level of the tree. The set of nodes at this level corresponds to the points along the first axis of the search space. Each pairing also defines a rough pose for the object. The second axis of the search space consists of the second data feature f_2 to all possible model features and the tree continues to grow until the last model feature is compared to each data feature. A node at level n of the interpretation tree describes a partial n-interpretation, in that nodes lying between the current node and the root of the

tree identify an assignment of model features to the first n data features. Any leaf of the tree defines a complete s-interpretation, where s is the number of sensor features.

In an effort to find consistent s-interpretations with as little effort as possible, constraints had to be determined that would eliminate the search of entire sub-trees, and hence entire subspaces of the search space without having to explicitly explore them. The advantage of using a tree is that we can start at the root and test interpretations as we move downward in the tree while in the process terminating the search when we find an inconsistent node, i.e. for which no rigid transformation will correctly align the model and data feature.

4.2.1 Consistency

The user has two options in testing for consistency at a node. One could explicitly solve for the best rigid transformation and test that the model features do in fact get mapped to their corresponding data features. One of the drawbacks of this method is that computation of the transformations is relatively expensive. In addition, computation of the transformation would require at least k data-model pairs, where k depends on the characteristics of the features. This implies that one must wait until at least k levels of depth before applying any consistency test thus increasing the amount of work to be done.

The second option for testing consistency is to seek less complete models. The goal of this method is to seek constraints that can be applied at any node of the tree, with the property that while no single constraint can uniquely guarantee consistency, some interpretations can be ruled out. The hope is that if enough independent constraints can be combined together, their aggregation will prove powerful in determining consistency constraints.

4.3 Constraints to Reduce the Search

Unary constraints apply to single pairings of a data and model feature. If a unary constraint applied to such a pairing and determined to be true then that data-model

pairing may be part of a consistent interpretation. However, a false determination implies that the pairing cannot possibly be part of the interpretation. Binary constraints follow a similar logic and the advantages of binary are that they can be quite simple, while retaining power to separate consistent from inconsistent interpretations, and they can be applied at any node in the tree. These observations formulate the approach to recognition as a problem of constraint satisfaction, or consistent labeling.

A general approach to exploring the tree is to use backtracking depth-first search as shown in Figure 4-2. The search begins at the root of the tree and continues downwards along the first branch. At the first node, we check the consistency of that node by examining the truth value of the unary and available binary constraints of the model-data pairing. The search continues downward as long as all of the constraints are satisfied. If a branch is reached in which a constraint does not hold, the remaining subtree is abandoned and we backtrack to the previous node. The next branch of that node is explored or if no branches exist, we backtrack again. This process continues until the entire tree has been explored and all possible interpretations have been found. The interpretations are subject to additional testing.

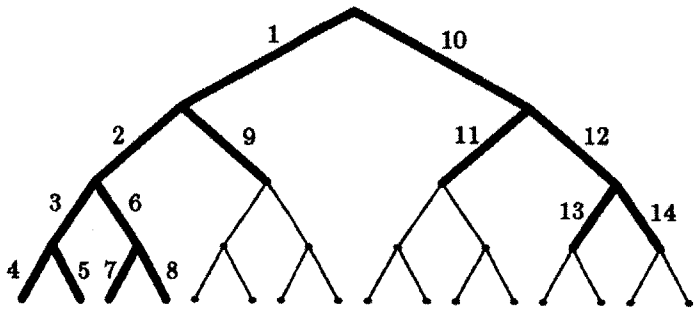


Figure 4-2: The darker edges in the diagram indicate on example of a backtracking search. the numbers on the edges indicate the order in which they are searched; courtesy of Grimson [13]

4.4 Model Tests to Verify Hypothesis

Although the interpretation tree search accounts for all of the data features, the interpretation defined by the node need not be a globally consistent one. This is because unary and binary constraints only allow us to infer that any two of the data-model pairings of the interpretation are consistent. This implies that we must check each leaf of the tree reached by the constrained search process to verify the interpretation at that leaf is globally valid. We can do this by solving for a rigid transformation mapping points $\mathbf{V}$ in model coordinates into points $\mathbf{v}$ in sensor coordinates. The general case is given by

$$\mathbf{v} = sR\mathbf{V} + \mathbf{v}_0 \tag{4.1}$$

where s is the scale factor, R is the rotation matrix, and $\mathbf{v}_0$ is the translation vector. An example is shown in Figure 4-3. Least-squares can be used to find the transformation that minimizes the error between the model and data features.

4.5 Finding Pose to Verify Hypothesis

In the case of alignment techniques, we only explore small dimensional subspaces of the correspondence space, then switch to a guided exploration of pose space to verify and refine the poses by searching for additional evidence. We can consider nodes of correspondence space as hypothesizing an interpretation, which must be verified to ensure global consistency. To do this, we solve for a rigid transformation as described in Equation 4.1. One approach is to consider a least squares computation of the transformation associated with the pose, which is the same as finding the transformation that minimizes the deviation of the transformed model features from their matching data features. The resulting transformation is used to project the model features into the data, and verify that each such transformed feature has a maximum deviation from its corresponding feature that is less than some predefined bound.

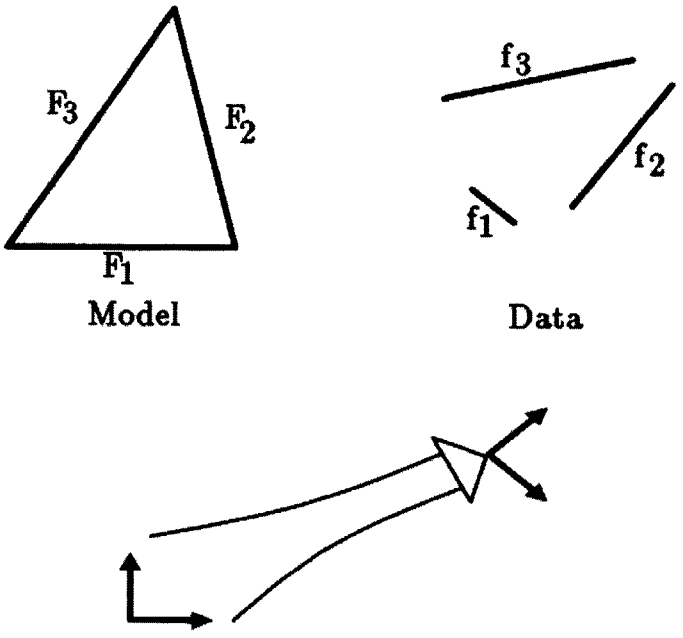


Figure 4-3: Given the data and model segments, one feasible interpretation associates $f_1 : F_1, f_2 : F_2, f_3 : F_3$. This is a consistent interpretation since there exists a rigid coordinate frame transformation that maps each model segment into agreement with its corresponding data segment; courtesy of Grimson [13]

4.5.1 Least Squares Estimation of Transform Parameters

A common theme in computer vision problems entails finding the similarity transformation parameters (R: rotation, t translation, and s: scaling) that gives the minimum value of the mean squared error $e^2(R, t, s)$ between two point sets x_i and y_i; $i = 1, 2, \ldots, n$ in m-dimensional space as proposed by Umeyama [42].

$$e^2(R, t, s) = \frac{1}{n} \sum_{i=1}^{n} \|y_i - (sRx_i + t)\|^2 \tag{4.2}$$

Arun, et al. [1] and Horn [14] presented a solution to the problem based on the singular value decomposition (SVD) of a covariance matrix of the data. As Umeyama pointed out, their solution sometimes failed to give a correct rotation matrix and gave a reflection instead when the data was severely corrupted thus, he presented a refinement to Arun's result.

Umeyama presented a theorem which gives the least squares estimation of similarity transformation parameters between two point patterns. The main result of his paper is shown in the lemma below which gives the least-squares estimation of rotation parameters.

Lemma 1 *Let A and B be $m \times n$ matrices and R an $m \times m$ rotation matrix, and UDV^T a singular value decomposition of AB^T ($UU^T = VV^T = I, D = diag(d_i), d_1 \geq d_2 \geq \ldots \geq d_m \geq 0$). Then the minimum value of $\|A - RB\|^2$ with respect to R is*

$$\min_R \|A - RB\|^2 = \|A\|^2 + \|B\|^2 - 2tr(DS) \tag{4.3}$$

where

$$S = \begin{cases} I & \text{if } \det(AB^T) \geq 0 \\ diag(1, 1, \ldots, 1, -1) & \text{if } \det(AB^T) < 0. \end{cases} \tag{4.4}$$

When rank $(AB^T \geq m - 1)$, the optimum rotation matrix R which achieves the above minimum value is uniquely determined.

$$R = USV^T \tag{4.5}$$

where S in Equation 4.5 must be chosen as

$$S = \begin{cases} I & \text{if } \det(U)\det(V) = 1 \\ diag(1, 1, \ldots, 1, -1) & \text{if } \det(U)\det(V) = -1 \end{cases} \tag{4.6}$$

when $\det(AB^T) = 0$ $(rank(AB^T) = m - 1)$.

The next step is to find the minimum value ε^2 of the mean squared error

$$e^2(R, t, s) = \frac{1}{n} \sum_{i=1}^{n} \|y_i - (sRx_i + t)\|^2 \tag{4.7}$$

of the two point sets with respect to the similarity transformation parameters which

is given as follows:

$$\varepsilon^2 = \sigma_y^2 - \frac{tr(DS)^2}{\sigma_x^2} \tag{4.8}$$

where σ_x^2 and σ_y^2 are the variances around the mean vectors of X and Y, respectively. The covariance matrix will be useful and is represented by

$$\Sigma_{xy} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \mu_y)(x_i - \mu_x)^T \tag{4.9}$$

where μ_x and μ_y are mean vectors of X and Y and a SVD of Σ_{xy} is UDV^T. Also,

$$S = \begin{cases} I & \text{if } \det(\Sigma_{xy}) \geq 0 \\ diag(1, 1, \ldots, 1, -1) & \text{if } \det(\Sigma_{xy}) < 0. \end{cases} \tag{4.10}$$

When the $rank(\Sigma_{xy}) \geq m - 1$, the optimum transformation parameters are determined uniquely as follows:

$$R = USV^T \tag{4.11}$$

$$t = \mu_y - sR\mu_x \tag{4.12}$$

$$s = \frac{1}{\sigma_x^2} tr(DS) \tag{4.13}$$

where the S in Equation 4.11 must be chosen as in Equation 4.6 when $rank(\Sigma_{xy}) = m - 1$. This analysis provides a closed form solution of the least squares problem of the similarity transformation parameter estimation using the SVD of a covariance matrix of the data.

4.5.2 Moment-Generating Functions

In medical image analysis, it is important to determine the relative positions of organs in an image. One method for accomplishing this goal is to find corresponding points of a particular organ in both modalities and compute an initial alignment using moments of inertia. Calculating the moments of a figure can be viewed as an alternative to computing pose versus using a simple Taylor series expansion, which doesn't work well for figures. Although the Taylor series has jump discontinuities, its Fourier transform is smooth which can be expanded in a power series. Similar to the Fourier transform, the moment-generating functions allow one to construct as informative a figure description as desired by using more coefficients of its expansion.

As stated in Duda and Hart [8], the moment generating function *M(u,v)* of a picture function $g(x, y)$ can be defined by

$$M(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} exp^{ux+vy} g(x, y) dxdy \tag{4.14}$$

If we were to take the $(p, q)^{th}$ partial derivative of M, we would get

$$\frac{\partial^{p+q} M(u, v)}{\partial u^p \partial v^q} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q exp^{ux+vy} g(x, y) dxdy \tag{4.15}$$

Note that when u and v are equal to 0, we generate the zeroth order moment m_{pq}, of the image which corresponds to the area of the figure, m_{00}. In the statistical case, the function $g(x, y)$ represents the probability density function and m_{01} and m_{10} would represent the first order moments which are the X and Y values of the mean of the probability distribution function. In terms of figures, if we divided m_{01} and m_{10} by m_{00}, the area of the figure would be normalized and yield the X and Y values of the figures' centroid.

A related set of moments are the *central moments* of the function $g(x, y)$, defined by

$$\mu_{pg} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \left(x - \frac{m_{01}}{m_{00}} \right)^p \left(y - \frac{m_{01}}{m_{00}} \right)^q g(x, y) dxdy. \tag{4.16}$$

The *central moments* provide for a change of coordinates systems that centers the X and Y axes at the centroid of the figure. The first central moments of a figure are zero where the second moments μ_{20}, μ_{02}, and μ_{11} are moments of inertia and are analogous to the variances and covariance of a bivariate probability function. The eigenvectors of the matrix of second order central moments are directions about which the figure has maximum and minimum amounts of inertia. The eigenvalues are the principle moments whose ratio define the *fatness* or *thinness* of a figure. An illustration of moments is given in Figure 4-4.

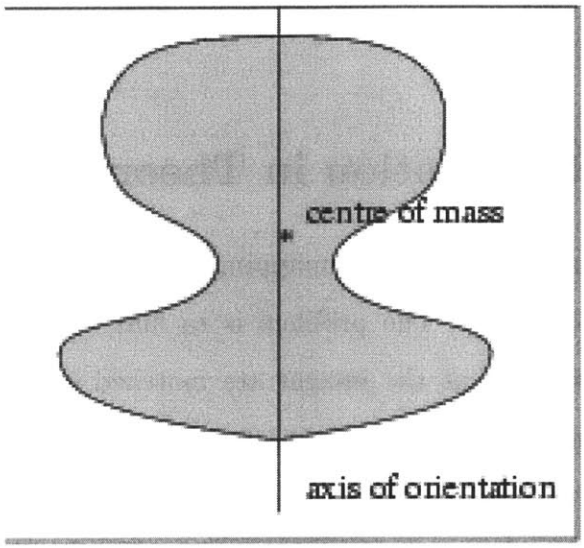


Figure 4-4: The 0^{th} moment is the area of the object; the 1^{st} moment gives the center of mass; and the 2^{nd} moments give the axes of orientation; courtesy of Kovesi [22]

Chapter 5

Image Registration

5.1 Image Registration in Theory

Image registration can be defined as a mapping between two images both spatially and with respect to intensity. The problem is to find the optimal spatial and intensity transformations so that the images are matched either for the purposes of determining the parameters of the matching transformation or to expose differences of interest between the images. Two of the most common general transformations are rigid and affine. Rigid transformations account for object or sensor movement in which objects in the images retain their relative shape and size. A rigid-body transformation is composed of a combination of a rotation, a translation, and scale change. Affine transformations are more general than rigid and can therefore tolerate more complicated distortions, such as shear and changes in aspect ratio, while still maintaining some nice mathematical properties [12]. Given the intrinsic nature of imagery of non-rigid objects, we concentrate our analysis on methods which are more likely to be applicable in medical diagnosis.

A standard approach to the registration of two corresponding medical images is as follows: the images can be reduced to binary images by detecting the edges or regions of highest contrast using a standard edge detection scheme. This removes extraneous information and reduces the amount of data to be analyzed [15]. Because we are interested in extracting non-rigid features in ultrasound, we know that the

search space, or class of transformations capable of aligning the images, will include translation, rotation, scaling, and shearing. For each of these operations, a similarity measure has to be computed between the edges of the ultrasound image to the edges of the CT/MR image. In integrating the information taken from the two different sensors, the problem of registration goes far beyond overlaying the two images. For example, the CT image of a patient's liver contains structural information that will be used to plan a surgery while the ultrasound image contains unintuitive content that contains some structural information.

Regarding realtime registration involving ultrasound, researchers have found that spatial methods ignore temporal continuity, which can potentially be used to improve the reliability of feature detection. Mulet-Parada and Noble [31] determined that acoustic image feature detection needs to be robust to speckle noise and attenuation imaging artifacts. Speckle noise corrupts the data by introducing sharp changes in the image intensity profile, while attenuation artifacts alter the intensity of equally significant cardiac structures depending on their orientation with respect to the ultrasound beam. This implies that measurements based on intensity values are not appropriate for acoustic feature detection which was validated in Chapter 3 on ultrasound data results.

5.2 Registration of Free Form Curves and Surfaces

One aspect of our registration goal requires the extraction of surface data from CT/MR as well as edge data in ultrasound. Our specific interests lie in the abdominal region, which consists of organs such as the liver that can take on various shapes at different time instances. The point correspondences used in the registration will come from organ boundaries, which can take the form of various curves. A study is deemed necessary to estimate 3-D motion from these point correspondences for registration of intra-operative ultrasound with pre-operative CT/MR data. Realtime estimates of such parameters will become useful in future research endeavors. In Zhang's [45] search to register free form curves and surfaces, curve data was treated in the same

way as surface data (a set of points). The word *shape* (S) refers to curves $S = \mathcal{C}$ or surfaces, $S = \mathcal{S}$. Curves were represented by chained points and surfaces were point sets. The points in the first 3-D map are noted by $x_i (i = 1, \ldots, m)$ and those in the second map are noted by $x'_j (j = 1, \ldots, n)$. The objective of the registration is to find the motion between two frames, i.e., $\mathbf{R}$ for rotation and $\mathbf{t}$ for translation, such that the following criterion

$$\begin{aligned} \mathcal{F}(\mathbf{R}, \mathbf{t}) = & \frac{1}{\sum_{i=1}^{m} p_i} \sum_{i=1}^{m} p_i d^2(\mathbf{R}\mathbf{x_i} + \mathbf{t}, \mathbf{S}') \\ & + \frac{1}{\sum_{j=1}^{n} q_j} \sum_{j=1}^{n} q_j d^2(\mathbf{R^T}\mathbf{x'_j} - \mathbf{R^T}\mathbf{t}, \mathbf{S}) \end{aligned} \tag{5.1}$$

is minimized, where $d(x, S)$ denotes the distance of the point x to S and p_i (resp. q_i) takes value 1 if the point x_i (resp. x'_j) can be matched to a point on S' in the second frame and takes value 0 otherwise.

The criteria were symmetric in the sense that neither of the two frames prevails over the other. To economize computation, the objective function to be minimized was

$$\mathcal{F}(\mathbf{R}, \mathbf{t}) = \frac{1}{\sum_{\mathbf{i=1}}^{\mathbf{m}} \mathbf{p_i}} \sum_{\mathbf{i=1}}^{\mathbf{m}} \mathbf{p_i} \mathbf{d^2}(\mathbf{R}\mathbf{x_i} + \mathbf{t}, \mathbf{S}'). \tag{5.2}$$

As a result of using this simplification, Zhang found that the the algorithm using the symmetric criterion yielded better motion estimates than that using the non-symmetric one. This was expected since the data in both frames contribute to the motion estimation and neither of the frames prevails over the other. However, the execution time using the symmetric criterion was twice as long.

5.2.1 Finding Closest Points

The distance between point x and shape S' was defined as

$$d(x, S') = \min_{x' \in S'} d(x, x') \tag{5.3}$$

where $d(x_1, x_2)$ is the Euclidean distance between the two points x_1 and x_2. We could use the following simplification:

$$d(x, S') = \min_{j \in \{1,\dots,n\}} d(x, x'_j). \tag{5.4}$$

The closest point y in the second frame to a given point x is the one satisfying

$$d(x, y) \leq d(x, z), \forall z \in S'. \tag{5.5}$$

As stated, the minimum of all distances from a given point to each *sample* point on the shape is calculated. This simplified version of the distance and different sampling of the shape affects the final estimate of the motion. The *real* closest points should ideally be used instead of the closest *sample* points ,however, the efficiency achieved with the sample points is lost. An easy way to overcome the sampling problem while maintaining efficiency is to increase the number of sample points through interpolation. However, the increase in required memory and increase in search time become problems requiring a tradeoff to be found.

5.2.2 Pseudo Point Matching

Problems which may occur due to sensor error or object motion can lead to the existence of spurious points in both image frames. To combat this problem, constraints may be imposed to remove spurious point pairings. One proposal is distance continuity in a neighborhood which could be useful in discarding false matches. Zhang exploits two heuristics which were all unary.

The first was the maximum tolerance for distance. If the distance between a point x_i and its closest one y_i is greater than the maximum tolerable distance D_{max}, then we set $p_i = 0$, which means that we could not pair a reasonable point in the second frame with the point x_i. As we will see later, D_{max} is set robustly during each iteration by analyzing distance statistics.

The second is orientation consistency. We can estimate the surface normal or

the curve tangent, both orientation vectors, at each point. The angle between the orientation vector at point x and its corresponding point y in the second frame cannot go beyond the rotation angle between the two frames. Thus, the angle between the orientation vectors at two paired points cannot be greater than a prefixed value, Θ, which is the maximum of the rotation angle expected between the two frames. This constraint is especially useful when the motion is relatively large.

5.2.3 Updating the Matching

We now exploit distance statistics between the point pairings. The idea is that distances between reasonably paired points should not be very different from each other. One parameter, $\mathcal{D}$, needs to be set by the user, which indicates when the registration between two frames is good. The value of $\mathcal{D}$ has an impact on the convergence of the algorithm. If $\mathcal{D}$ is smaller than necessary, then more iterations are required for the algorithm to converge because many good matches will be discarded at the step of the matching update. On the other hand, if $\mathcal{D}$ is much larger than necessary, it is possible for the algorithm not to converge to the correct solution because possibly many false matches will not be discarded. Thus, it is usually better to choose a small value for $\mathcal{D}$.

Let D_{max}^{I} denote the maximum tolerable distance in iteration I. At this point, each point in the first frame whose distance to its closest point is less than D_{max}^{I-1} is retained, together with its closest point and their distance. Let $\{x_i\}$, $\{y_i\}$, and $\{d_i\}$ respectively, represent the resulting set of original points, closest points, and their distances after the pseudo point matching and let N represent the cardinality of the sets. The mean, μ, and sample deviation, σ, of the distances is given by

$$\mu = \frac{1}{N}\sum_{i=1}^{N} d_i \tag{5.6}$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(d_i - \mu)^2}. \tag{5.7}$$

Various thresholds are set to determine a range of registration qualities from quite good to really bad. At that point, the newly set D^I_{max} is used to update the matching previously recovered: a pairing between x_i and y_i is removed if their distance d_i is greater than D^I_{max}. The remaining pairs are used to compute the motion between the two frames.

5.2.4 Computing Motion

Because the number of point pairs, N, will mostly be a large number, it is necessary to devise a procedure for computing the motion by minimizing the following mean-squares objective function

$$\mathcal{F}(\mathbf{R}, \mathbf{t}) = \frac{\mathbf{1}}{\mathbf{N}}\sum_{\mathbf{i=1}}^{\mathbf{N}} \|\mathbf{R}\mathbf{x_i} + \mathbf{t} - \mathbf{y_i}\|^{\mathbf{2}}, \tag{5.8}$$

which is the direct result of 5.2. The least squares rotation and translation parameters can also be found using Arun's [1] singular value decomposition method or Horn's [14] quaternion method.

5.2.5 Iterative Pseudo Point Matching Algorithm

Zhang's [45] iterative pseudo point matching algorithm can be summarized as follows:

- **input: Two 2-D frames containing m and n 2-D points, respectively**
- **output: The optimal motion (or transformation) between the two frames**
- **procedure:**
  1. **initialization: D^0_{max} is set to $20\mathcal{D}$, which implies that every point in the first frame whose distance to its closest point in the second**

frame is bigger than D^0_{max} is discarded from consideration during the first iteration. The number 20 can be replaced by a larger value.

2. **preprocessing:**
   (a) **Compute the tangent at each point of the two frames (only for curves)**
   (b) **Build the k-D tree representation of the second frame**
3. **iteration: until conversion of the computed motion**
   (a) **Find the closest points satisfying the distance and orientation constraints**
   (b) **Update the recovered matches through statistical analysis of distances**
   (c) **Compute the motion between the two frames from the updated matches**
   (d) **Apply the motion to all points (and their tangents for curves) in the first frame**

5.3 Iterative Closest Point Algorithm

The ICP procedure stems from a problem that was formalized by Besl [2] in which he described global and local shape matching metrics for free form curves and surfaces as well as point sets in an attempt to formalize and unify a key computer vision problem: Given 3-D data in a sensor coordinate system and given a model shape in a model coordinate system in a different geometric shape representation, estimate the optimal rotation and translation that registers the model shape and data shape minimizing the distance between the shapes and allowing the determination of the equivalence of the shapes via a mean-square distance metric. A method proposed by Besl and McKay [3] provides a solution to this free-form surface matching problem as a special case of a simple, unified approach, which generalizes to n dimensions and provides

solutions to 1) the point-set matching problem without correspondence and 2) the free-form curve matching problem. The algorithm requires no extracted features, no curve or surface derivatives, and no pre-processing of 3-D data, except for the removal of statistical outliers and can be used with the following representations of geometric data:

1. point sets
2. line segment sets (polylines)
3. implicit curves: $\overrightarrow{g}(x, y, z) = 0$
4. parametric curves:$(x(u), y(u), z(u))$
5. triangle sets (faceted surfaces)
6. implicit surfaces: $g(x, y, z) = 0$
7. parametric surfaces: $(x(u, v), y(u, v), z(u, v))$

We will restrict our discussion and analysis to point sets for our initial registration interests. The Euclidean distance $d(\overrightarrow{r_1}, \overrightarrow{r_2})$ between the two points $\overrightarrow{r_1} = (x_1, y_1)$ and $\overrightarrow{r_2} = (x_2, y_2)$ is $d(\overrightarrow{r_1}, \overrightarrow{r_2}) = \|\overrightarrow{r_1} - \overrightarrow{r_2}\| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. Let A be a point set with N_a points denoted $\overrightarrow{a_i} : A = \{\overrightarrow{a_i}\}$ for $i = 1, \ldots, N_a$. The distance between the point $\overrightarrow{p}$ and the point set A is

$$d(\overrightarrow{p}, A) = \min_{i \in \{1, \ldots, N_a\}} d(\overrightarrow{p}, \overrightarrow{a_i}) \tag{5.9}$$

The closest point $\overrightarrow{a_j}$ of A satisfies the equality $d(\overrightarrow{p}, \overrightarrow{a_j}) = d(\overrightarrow{p}, A)$.

5.3.1 Corresponding Point Set Registration

A procedure for yielding the least squares rotation and translation is reviewed by Besl and McKay in which they determined that the quaternion approach to find the closest point (minimum distance) is better in 2-D and 3-D than the singular

value decomposition (SVD) method because reflections are not desired. However, the cross-covariance matrix in SVD of two-point distributions easily generalizes to the n dimensional case and would be the method of choice for $n > 3$ in any n-dimensional applications. The basic solution of Horn is described in [15]. Besl and McKay stressed the role of the SVD cross-covariance matrix, which is an important relationship not discussed in Horn's work.

5.3.2 ICP Algorithm Description

The ICP algorithm can be described in terms of an abstract geometric shape X whose internal representation must be known in order to execute the algorithm. A "data" shape P is registered to be in best alignment with a "model" shape X. The data must be decomposed in point set form in preparation for registration if it is not already in point set form. These points are usually the vertices in a triangle or the endpoints of a line and if the data comes in curve or surface form, points from a line or triangle approximation are used. Let N_p represent the number of points in the data shape and N_x represent the number of points in the model shape.

The distance metric d between an individual data point $\overrightarrow{p}$ and a model shape X will be denoted

$$d(\overrightarrow{p}, X) = min_{\overrightarrow{x} \in X} \| \overrightarrow{x} - \overrightarrow{p} \|. \tag{5.10}$$

The closest point in X that yields the minimum distance is denoted $\overrightarrow{y}$ such that $d(\overrightarrow{p}, \overrightarrow{y}) = d(\overrightarrow{p}, X)$, where $\overrightarrow{y} \in X$. Computing the closest point is $O(N_x)$ worst case with expected cost $\log(N_x)$. When the closest point computation from $\overrightarrow{p} to X$ is performed for each point in P, that process is worst case $O(N_x N_p)$. Let Y denote the resulting set of closest points, and let $\mathcal{C}$ be the closest point operator:

$$Y = \mathcal{C}(P, X) \tag{5.11}$$

Given the resultant corresponding point set Y, the least squares registration is com-

puted as described above:

$$(\overrightarrow{q}, d) = \mathcal{Q}(P, Y). \tag{5.12}$$

The positions of the data shape point set are then updated via $P = \overrightarrow{q}(P)$.

5.3.3 ICP Algorithm Statement

The ICP Algorithm is stated in [3] as follows:

- The point set P with N_p points $\{\overrightarrow{p_i}\}$ from the data shape and the model shape X (with N_x supporting geometric primitives: points, lines, or triangles) are given.
- The iteration is started by setting $P_0 = P$, $\overrightarrow{q_0} = [1, 0, 0, 0, 0, 0, 0]^t$ (if using quaternions for rotation) and $k = 0$. The registration vectors are defined relative to the initial data set P_0 so that the final registration represents the complete transformation. Steps 1 through 4 are applied until convergence within a tolerance τ. The computational cost of each operation is given in brackets.
    1. Compute the closest points:$Y_k = \mathcal{C}(Pk, X)$ [cost: $O(N_x N_p)$ worst case, $O(N_p log N_x)$ average].
    2. Compute the registration: $(\overrightarrow{q_k}, d_k) = \mathcal{Q}(P_0, Y_k)$ [cost: $O(N_p)$].
    3. Apply the registration: $P_{k+1} = \overrightarrow{q_k}(P_0)$ [cost: $O(N_p)$].
    4. Terminate the registration when the change in mean-square error falls below a preset threshold $\tau > 0$ specifying the desired precision of the registration: $d_k - d_{k+1} < \tau$. If a dimensionless threshold is desired, one can replace τ with $\tau\sqrt{tr(\Sigma_x)}$, where the square root of the trace of the covariance of the model shape indicates the rough size of the model shape.

A convergence theorem is stated that is based on two key ideas: 1) least squares registration generally reduces the average distance between corresponding points during

each iteration, whereas 2) the closest point determination generically reduces the distance for each point individually.

1 Theorem

The iterative closest point algorithm always converges monotonically to a local minimum with respect to the mean-square distance objective function.

5.3.4 ICP Results on Rigid Data Sets

We tested a simple version of the ICP algorithm on simulated point sets of 2-D triangles and squares. In this section we demonstrate how ICP can find the closest points between the two data sets and accurately compute the registration.

In Figures 5-1 and 5-2, we show two point sets that were rotated by 15 and 45 degrees, respectively. The points marked as " * " are the original points and the points noted by " + " are the target points after rotation. ICP does converge instantaneously to its registration result however, a trace from the original to the target points is shown for visualization of the convergence via iterations, 50 of which were observed in this case. We used small angle approximations to estimate the rotation for robustness in anticipated image analysis. Instead of using non-linear optimization methods such as Powell's method, the rotation was linearized so that the least squares error could be computed and those values are provided in the respective figures' caption. Figure 5-3 shows the ICP result of a translated point set with additional random noise. Fifty iterations were initially given as input to the algorithm, but iterations ended when the norm of the mean square error of the distance between each point was less than a threshold of 0.1. In this particular case, the registration was completed by the 35^{th} iteration.

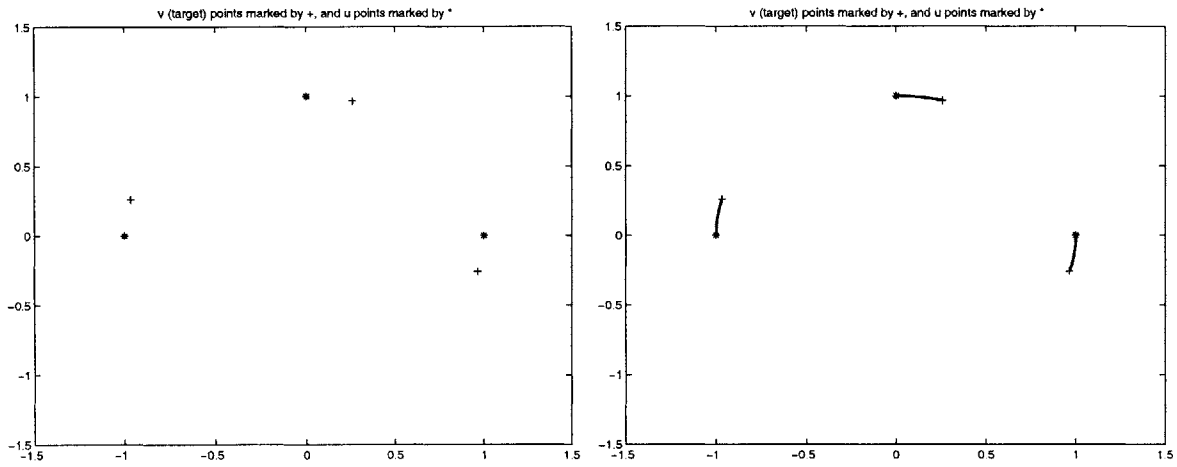
Figure 5-4 shows an example of initial transform computation in which an ultrasound image was scaled by 0.5 and shifted in the x and y direction, respectively by the vector [60, 30]. We allow user input in this particular demonstration in which one is prompted to select a number of corresponding points in the two upper images and a transformation is computed and displayed in the lower right-hand corner. In this example, 6 points were selected and the *estimated* scale and translation vector were

0.5 and [64.9, 40.6]. The scale and shift were estimated with standard deviation and mean values as in the following:

$$estimScale = \frac{std(v)}{std(u)} \tag{5.13}$$

$$estimShift = mean(v) - mean(u) \times estimScale. \tag{5.14}$$

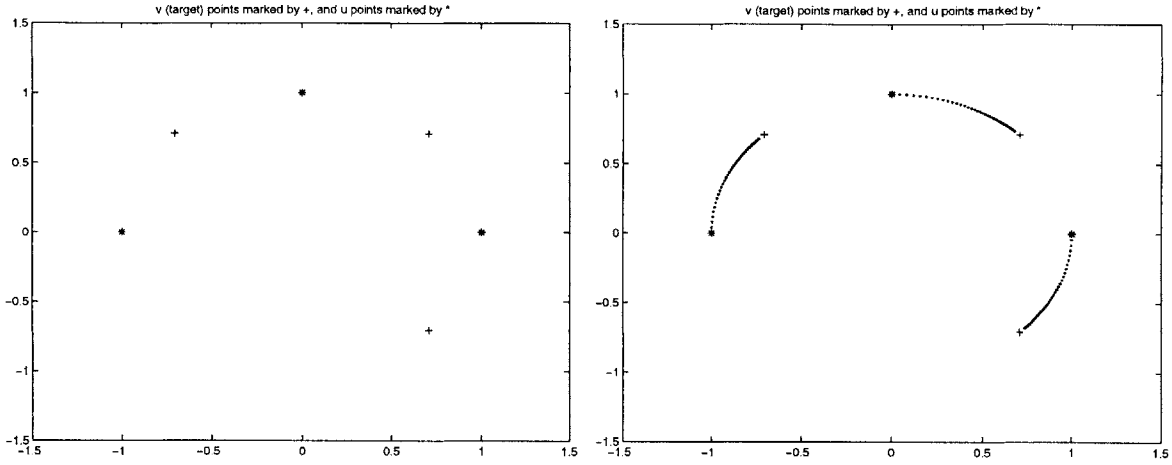
We will continue our investigation into initial transform computation. This initial transform will provide correspondences between CT/MR surface points and ultrasound edge points for refinement of the registration using polynomial warps discussed in Chapter6.



(a) ICP Results: θ=15 degrees

(b) ICP Results θ_{error} = 0.46 degrees

Figure 5-1: Test of ICP Algorithm with only Rotation



(a) ICP Results: θ=45 degrees

(b) ICP Results $\theta_{error} = 1.29$ degrees

Figure 5-2: Test of ICP Algorithm with only Rotation

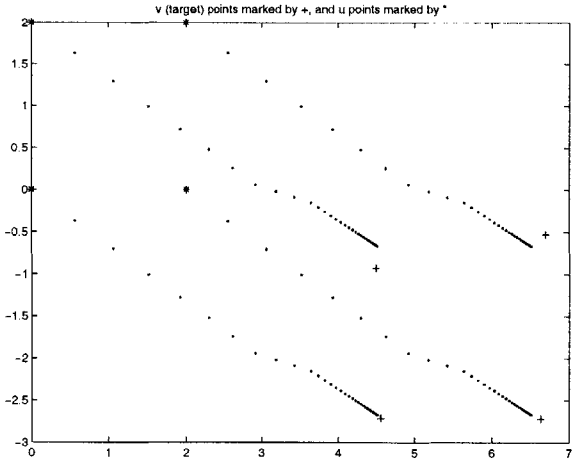


Figure 5-3: Test of ICP Algorithm with only Translation

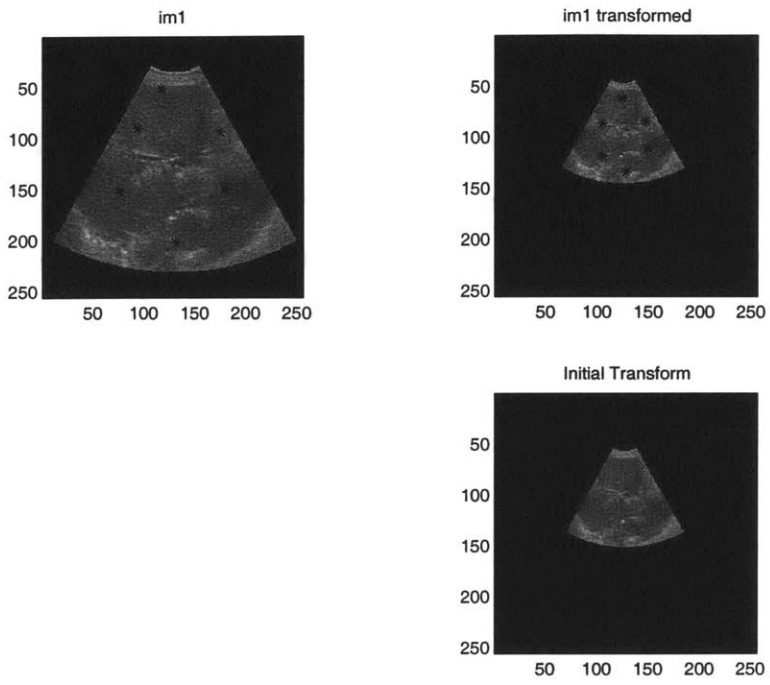


Figure 5-4: Initial Transform Computation

5.3.5 ICP Results on Non-Rigid Data Sets

Chui and Rangarajan [5] proposed an alternative to ICP called the Mixture Point Matching Algorithm (MPM), which can solve for both rigid and high-dimensional (thin-plate spline-based) non-rigid transformations between points sets in the presence of noise and outliers. The algorithm is similar to the Expectation Maximization algorithm and jointly solves for feature correspondence as well as geometric transformations. The mixture point matching framework is designed to be general and can be applied to both rigid and non-rigid point sets. The algorithm is similar to ICP however, ICP treats the corresponding points as binary rather than probabilistic variables. We plan to investigate and test MPM's performance on ultrasound and CT data in future analysis however, we will continue with ICP for results in this thesis.

Using Chui and Rangarajan's demo intended to compare ICP and MPM results on simulated and sulcal point sets, we tested ICP on various corresponding point sets in the abdominal area. Figures 5-5 through 5-14 contain corresponding points from ultrasound and CT along with ICP results (ultrasound (open areas) and CT (closed areas)) and thin-plate spline warping results, which are located directly below the

corresponding ultrasound and CT image. A rather detailed discussion on thin-plate spline warping is given in [6] for further reference.

We concluded from the non-rigid point experiments that ICP did encounter problems when attempting to find the non-rigid transformation between multi-modal image data. In each of the experiments, ICP did not always find the correct match between each point in the ultrasound and CT point sets. For instance, Figure 5-14 resulted from numerous ultrasound points determining that a single CT point was the corresponding point, which did not satisfy our requirement for 1-to-1 point correspondence. This observation calls for a further investigation into ICP, MPM, and possibly other algorithms that will produce the most accurate correspondence between the data sets.

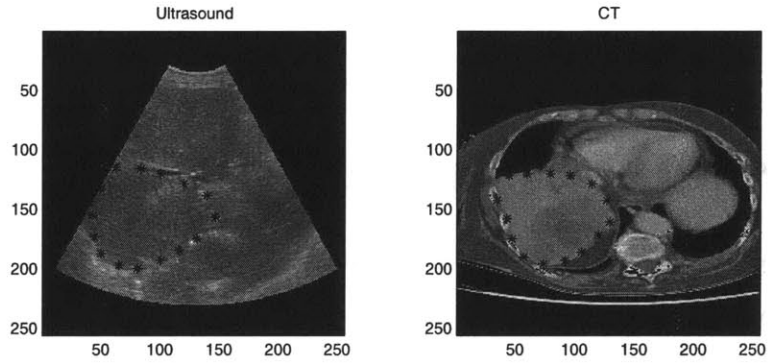


Figure 5-5: 15 corresponding points in ultrasound and CT

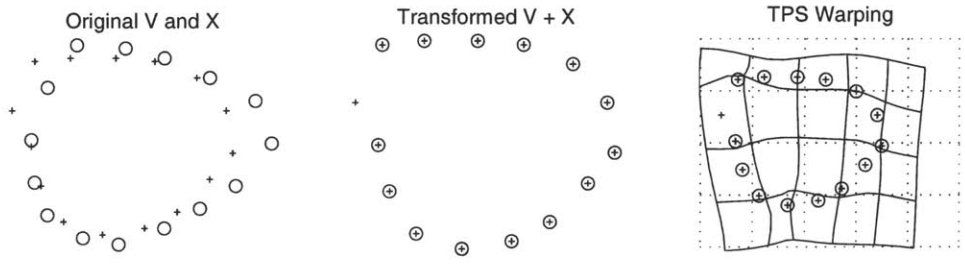


Figure 5-6: Test of ICP algorithm using indicated points

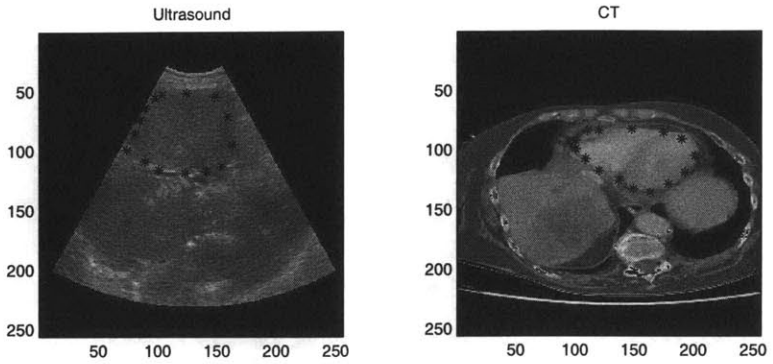


Figure 5-7: 15 corresponding points in ultrasound and CT

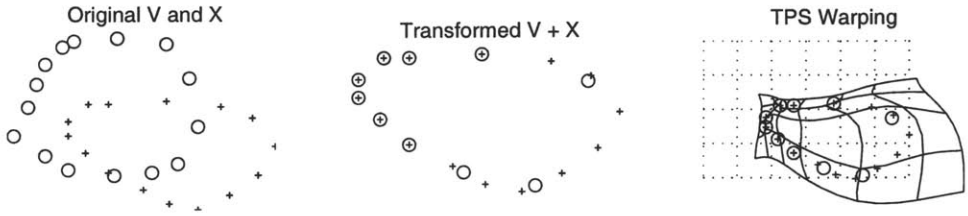


Figure 5-8: Test of ICP algorithm using indicated points

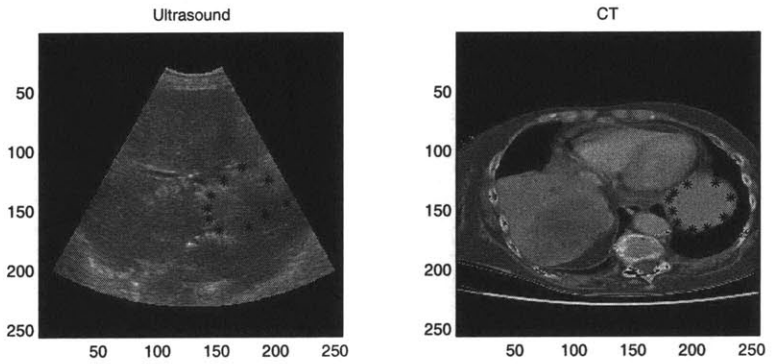


Figure 5-9: 10 corresponding points in ultrasound and CT

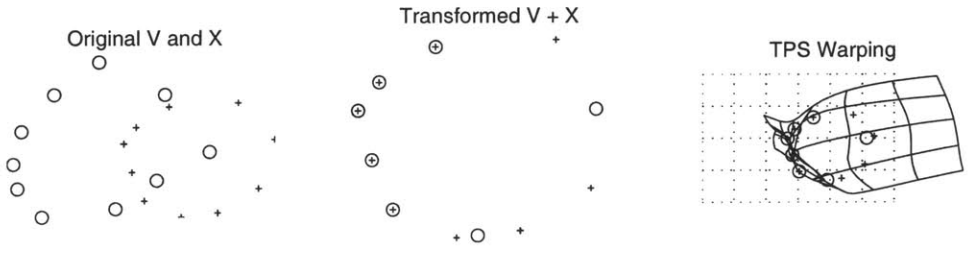


Figure 5-10: Test of ICP algorithm using indicated points

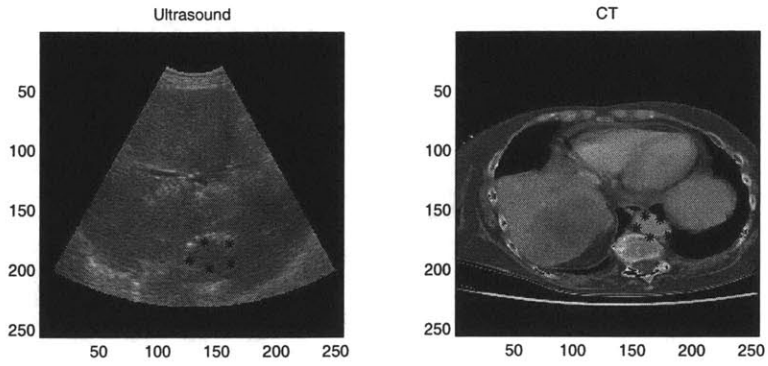


Figure 5-11: 5 corresponding points in ultrasound and CT

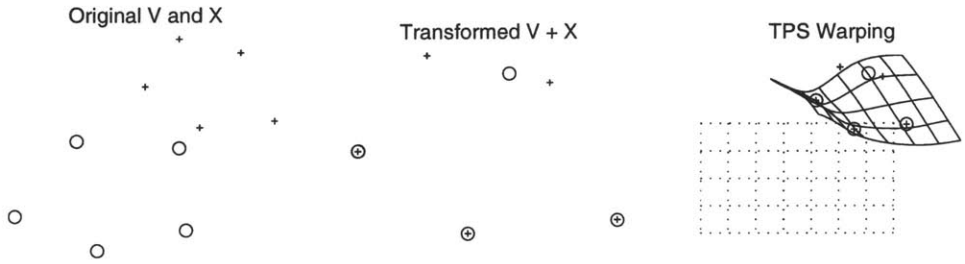


Figure 5-12: Test of ICP algorithm using indicated points

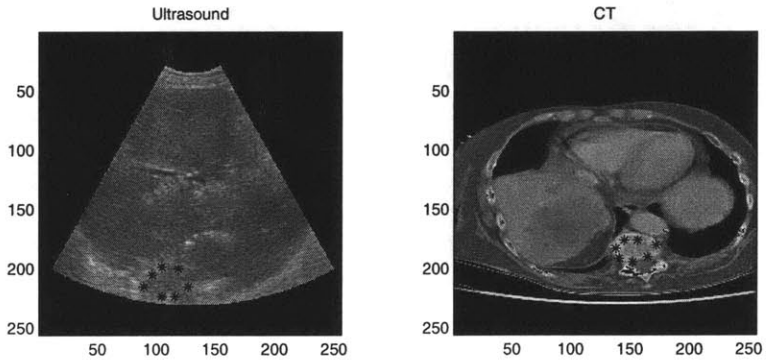


Figure 5-13: 7 corresponding points in ultrasound and CT

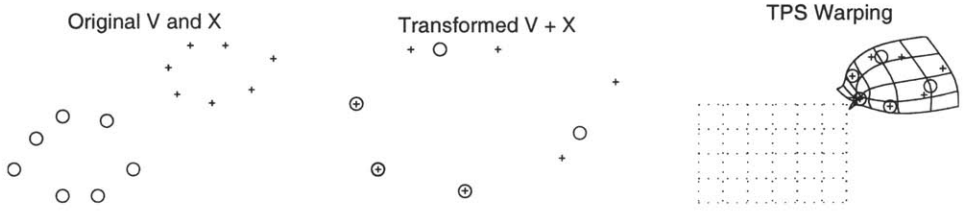


Figure 5-14: Test of ICP algorithm using indicated points

5.4 Nonlinear Registration Approach

A common registration approach is to use a reference data set and to compare it with the patient data by finding a transformation that aligns the reference data set with the patient. This procedure should be automatic, reliable, repeatable and robust. The first component of automatic nonlinear registration is to establish a relationship between the model (atlas or canonical data set) and the patient specific data. The second component is the actual warping procedure used to establish the transform between the patient and atlas. A 3-D geometric transformation maps an image from the coordinate system defined by u,v,w into a new image defined by the coordinate system (x,y,z).

$$T : (u, v, w) \longrightarrow (x, y, z) \tag{5.15}$$

where the mapping functions are $x = x(u, v, w), y = y(u, v, w), z = z(u, v, w)$

The nonlinear registration techniques described in [41] aimed to compute high order mapping functions using local information and constrain the mapping functions based on a physical model of elastic materials. The elastic matching involves the construction of a high-order nonlinear transform describing how the template must be deformed to maximize the match to the patient scan. The template can be a normal anatomy atlas (deterministic or probabilistic) or it can be a scan from a different modality or from the same modality. The matching is usually a two-step process: 1) determine a global registration between the model and patient scan, thus accounting for translation, rotation, and scale differences among the corresponding regions of both data sets and 2) determine a nonlinear transform that maximizes the similarity of these regions.

Most elastic matching techniques can be defined in the following framework: for a template data set T (such as an atlas) and a subset data set S (such as an MRI scan) defined on a rectangular lattice $x = (x, y, z)^T \in \mathcal{Z}^3$, a deformation field which matches data set T to data set S is estimated. A vector describing the deformation at location x is $U = (u, v, w)^T = U(u(x), v(x), w(x))^T \in \mathbf{R^3}$. The goal of the elastic

registration is to compute deformation field U, such that the sum of local similarities between the subject,S, and template, T, under the deformation field is maximized,

$$\int similarity(S(x), T(x - U(x)))dV \tag{5.16}$$

subject to some constraint upon the deformation field–the regularization (or "elasticity") constraint,

$$\int D(U)dV. \tag{5.17}$$

Attempts have been made to use only the surfaces of objects to compute the transformation between objects. A recently used scheme provided for the computation of the elastic transformation by an iterative refinement approach [30]. First, contours of the surface of each object were extracted from a 3-D volume using a semi-automatic contour model. A surface was formed by triangulating between the contours of neighboring slices. A combination of Euclidean distance and similarity of surface orientation described the distance function used to show how well local patches of the two surfaces matched. Surfaces of the objects were matched by computing a displacement vector for each triangle vertex on the basis of this distance function. To capture the interaction between neighboring vertices, a deformation field is computed by Gaussian smoothing of the displacement vectors. Iterations follow with decreasing stiffness of the surfaces. The transformation is computed by interpolation of the deformation field over the entire volume.

5.5 Elastic Registration

Nonlinear registration is the technique used to align data that is mismatched in a non-uniform or nonlinear manner. Such misalignments can be caused by intrinsic shape differences, such as motion of non-rigid organs in the abdomen, or physical deformation such as those caused by opening the skull for neurosurgery. In Toga's study [41], the term *matching* is used to refer to any process that determines correspondences

between point sets and *warping* refers to matching techniques which require deformation field computation between corresponding points. *Elastic matching* is used to refer to a subset of the available nonlinear registration techniques that operate by computing a deformation field through the minimization of a functional consisting of a term measuring local similarity and a regularization constraint based on a physical model of an elastic material. Even models that don't have the properties of an elastic body are used in elastic matching if they make use of the regularization model.

One characteristic of nonlinear registration techniques that poses problems in clinical settings is computational cost. The computation of higher order nonlinear transformations can require multiple 3-D (floating point) data sets be held in working memory while a lengthy nonlinear optimization process is performed. The long running times on workstations and sometimes even supercomputers are acceptable for research but not for clinical applications. Thus, a rapid and robust nonlinear registration algorithm is deemed necessary for applications in image-guided procedures.

5.5.1 Splines

Some registration techniques have been developed that exploit elastic models. Instead of directly applying piecewise interpolation to compute a transformation that maps the control points of one image onto another, these methods model the distortion in the image as the deformation of an elastic material. In other words, the registration transformation is the result of the deformation of an elastic material with the minimal amount of bending and stretching. The amount of bending and stretching is characterized by the energy state of the elastic material. Nevertheless, the methods of piecewise interpolation are closely related since the energy minimization needed to satisfy the constraints of the elastic model can be solved using splines [12].

Piecewise interpolation requires that a spatial mapping transformation for each coordinate point be specified which interpolates between the matched coordinate value. Most of the methods evaluated by Franke [10] use the general spline approach to piecewise interpolation. This requires the selection of a set of basis functions, $B_{i,j}$ and a set of constraints to be satisfied so that solving a set of linear equations will

specify the interpolation function. The spline surface $S(x, y)$ can be defined as

$$S(x, y) = \sum_{i,j} V_{i,j} B_{i,j}(x, y) \tag{5.18}$$

where $V_{i,j}$ are the control points. These local interpolations, which perfectly match the control points, act as a refinement after a global match has been performed using the best rigid transformation [33].

5.5.2 Elastic Warping Algorithms

Warping algorithms calculate a 3-D deformation field which can be used to non-linearly register one image with another image (or atlas). Instead of warping the atlas into the configuration of a new subject's anatomy, the new subject's anatomy can be warped into the configuration of the atlas anatomy, removing subject-specific shape differences. The resultant deformation fields can subsequently be used to transfer physiologic data from different individuals to a single anatomic template, enabling functional data from different subjects to be compared and integrated in a context where confounding effects of anatomical shape differences are factored out [39].

Chapter 6

Results of CT/MR Elastic Warping

6.1 Registration Error Analysis

A precise quantitative study of registration results is deemed necessary for most clinical applications. Julian Rathert, M.D. manually located 534 point correspondences in a 3-D CT/MR data set and together with C.F. Westin of Brigham and Women's Hospital, they used an elastic warping algorithm to register the two modalities. A sparse deformation field, calculated from the corresponding points, was resampled by a factor of 8 to reduce computation time. We found that the resampling factor had no significant effect on the registration results. The elastic transformation was computed by interpolating the sparse deformation field over the entire volume using polynomials. Additional linear interpolation was applied to produce a dense deformation field the same size as the atlas image. In this analysis, we sought to explore numerous variations on polynomial warping methods which included the following: variation of the number of corresponding points; variation of polynomial order; variation of the point set distribution; and error verification to measure sensitivity in our computed registration errors.

6.1.1 Error Analysis Algorithm Statement

Our algorithm for computing registration errors was as follows:

1. Calculate the dense deformation field from a select group of 3-D point correspondences in CT (Image 1) and MR (Image 2), which we call the *original* point sets $p1$ and $p2$, respectively.

2. Choose a group of corresponding points mutually exclusive from the *original* set and call these the *reserve* point sets $p1_r$ and $p2_r$.

3. For both the *original* and *reserve* point sets in MR ($p2$ and $p2_r$), find the corresponding values in the deformation field. We will call these sampled versions of the deformation field, $deform_{orig}$ from the *original* points and $deform_{res}$ from the *reserve* points.

4. Use $deform_{orig}$ to transform the *original* point set $p1$ and $deform_{res}$ to transform the *reserve* point set $p1_r$ by subtracting $deform_{res}$ from $p2_r$ and $deform_{orig}$ from $p2$. We denote these transformations as $p1'$ and $p1'_r$.

5. Measure the Euclidean distance between $p1'_r$ and $p1_r$, which gives the primary results relevant to the accuracy of our algorithm. Repeat the process for the *original* points for further analysis.

6. Calculate the mean, standard deviation, maximum, and median of the Euclidean distances measured above. Note that all error measurements are in voxels, however, the voxel dimensions are $0.975 \times 0.975 \times 1.5$ mm for those interested in metric conversions.

Standard Polynomial Information					
poly. order	1	2	3	4	5
poly. fit	4	10	20	35	56

Table 6.1: Relationship Between Polynomial Order and Fit

6.2 Investigation of Polynomial Order and Point Correspondence Quantities

For each of the following experiments, we used 526 of the 534 manually chosen corresponding points. All points which had a 0 coordinate were considered erroneous and eliminated from the point set because valid pixel values were only in the range from 1 to 128. The displayed CT atlas image is from the 80^{th} slice of the volume while the MR image is from the 79^{th} slice and the transformed atlas image is from the 79^{th} slice. Figures 6-1 through 6-4 show CT/MR with the corresponding point pairs used in the following four experiments. Tables 6.2 through 6.5 contain the quantity and sampling rates of the *original* and *reserve* points. Recall that the deformation field was calculated from points in the *original* point set and was applied to a separate set of *reserve* points that were not included in the *original* point set. Using our Error Analysis Algorithm, we computed the mean (μ_{error}), standard deviation (σ_{error}), maximum (max_{error}), and median (med_{error}) of the Euclidean distances between the transformed points and their actual location in the image. We observed polynomial orders 1 through 5 for *original* point sets consisting of 263, 176, 132, and 106 points and the respective *reserve* point sets consisting of 132, 88, 66, and 53 points for each of the four experiments. Error bars are provided in Figure 6-5 to compliment the tables for visualization of the errors. Table 6.1 provides the relationship between polynomial order and the number of polynomials used to fit the points in a distribution.

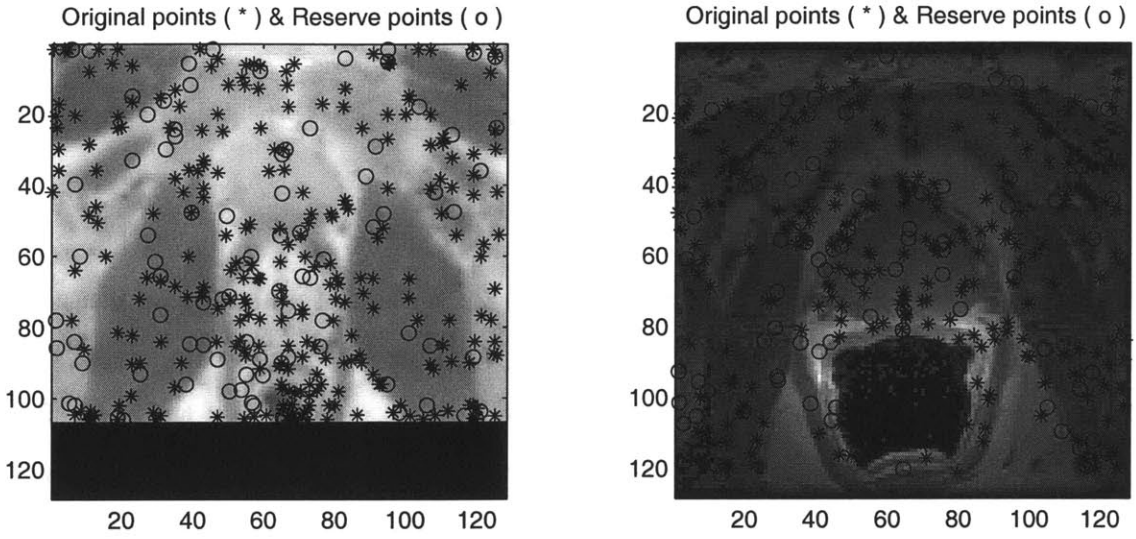


Figure 6-1: CT/MR shown with 263 *original* points (*) and 132 *reserve* points (o)

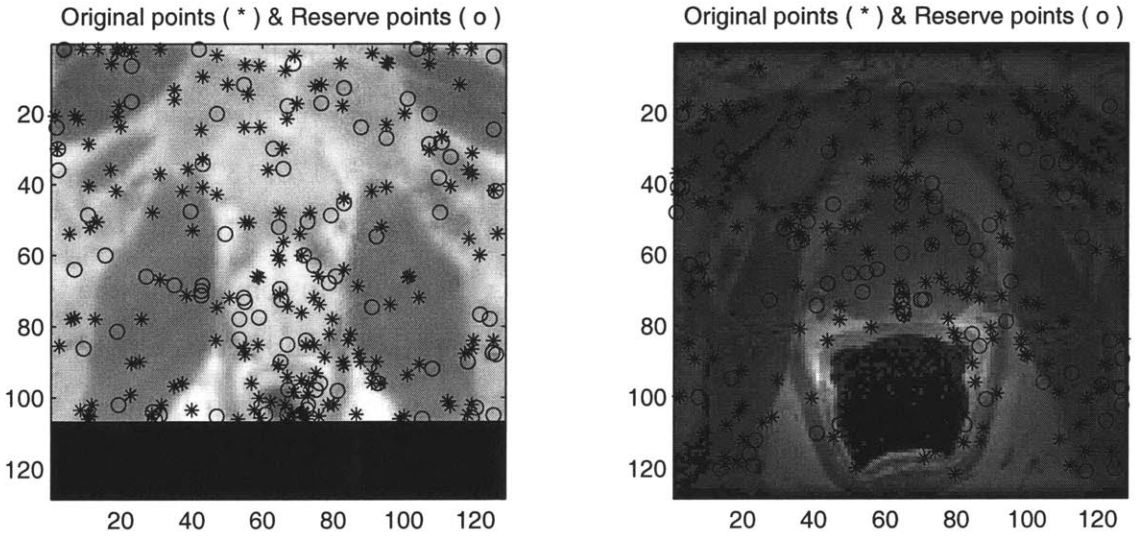


Figure 6-2: CT/MR shown with 176 *original* points (*) and 88 *reserve* points (o)

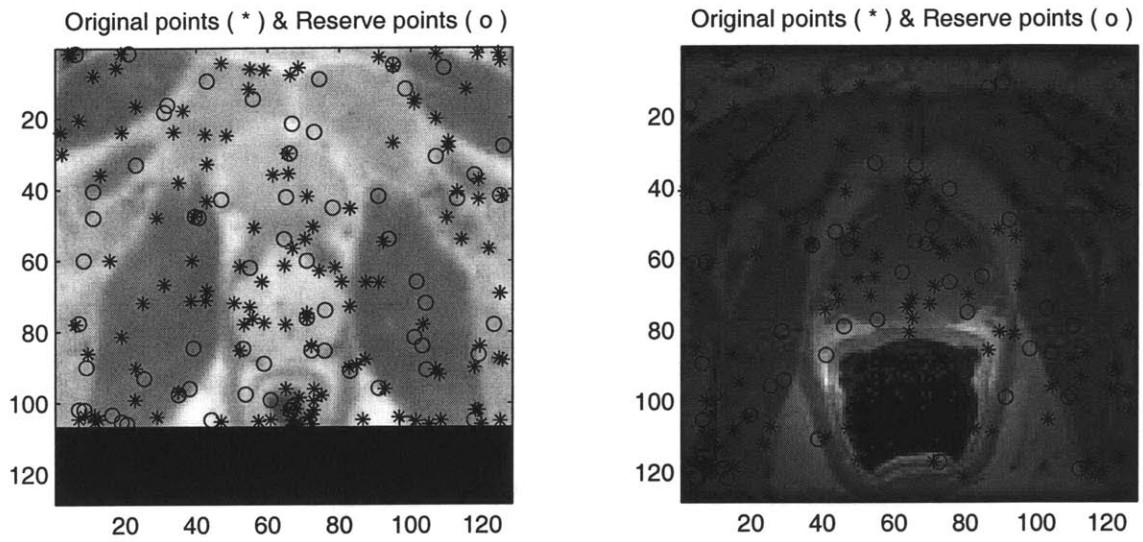


Figure 6-3: CT/MR shown with 132 *original* points (*) and 66 *reserve* points (o)

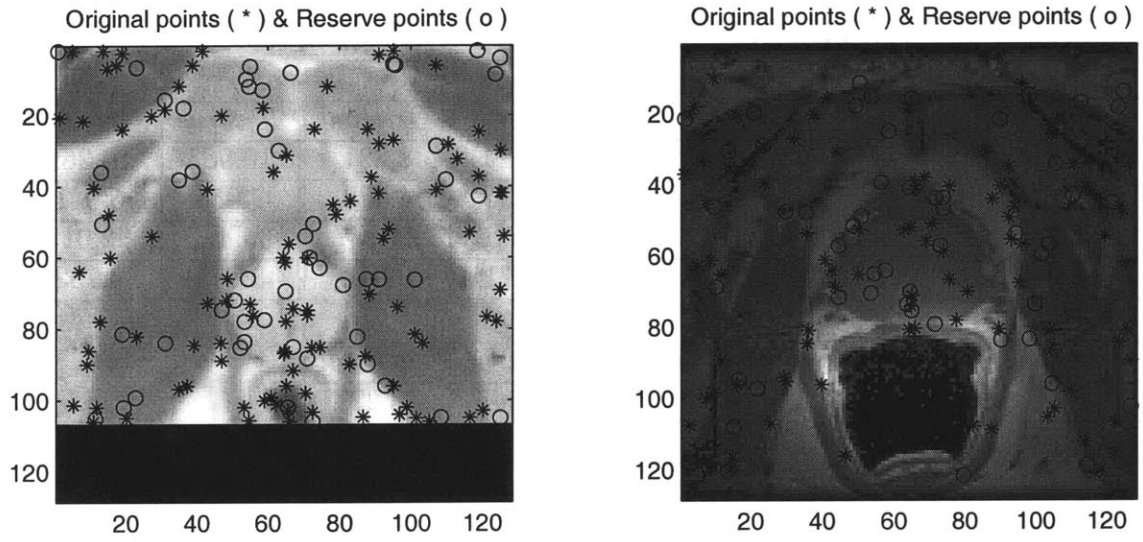


Figure 6-4: CT/MR shown with 106 *original* points (*) and 53 *reserve* points (o)

263 Original Points and 132 Reserve Points						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error} (voxels)	10.706	11.040	9.147	8.932	10.382	10.263
σ_{error} (voxels)	4.776	5.599	4.283	4.999	5.322	6.410
max_{error} (voxels)	24.040	33.238	27.033	31.615	31.299	39.294
med_{error} (voxels)	10.797	10.533	8.572	8.001	10.059	8.867
poly. order	4		5			
	Reserve	*Original*	*Reserve*	*Original*		
μ_{error} (voxels)	11.209	11.488	12.527	13.726		
σ_{error} (voxels)	6.401	7.852	8.442	10.985		
max_{error} (voxels)	35.435	45.667	51.149	57.607		
med_{error} (voxels)	9.869	9.691	10.752	10.547		

Table 6.2: Experiment 1: Original Sampling Rate of 2; Reserve Sampling Rate of 2

176 Original Points and 88 Reserve Points						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error} (voxels)	10.686	9.654	8.639	8.821	10.963	10.269
σ_{error} (voxels)	6.209	5.219	5.235	4.423	7.055	5.941
max_{error} (voxels)	29.603	23.736	27.716	24.147	35.427	32.017
med_{error} (voxels)	9.720	8.681	7.671	8.013	8.755	9.732
poly. order	4		5			
	Reserve	*Original*	*Reserve*	*Original*		
μ_{error} (voxels)	13.478	11.552	15.545	13.277		
σ_{error} (voxels)	10.728	7.985	13.983	10.877		
max_{error} (voxels)	66.706	48.443	66.054	60.576		
med_{error} (voxels)	10.043	9.724	10.567	9.568		

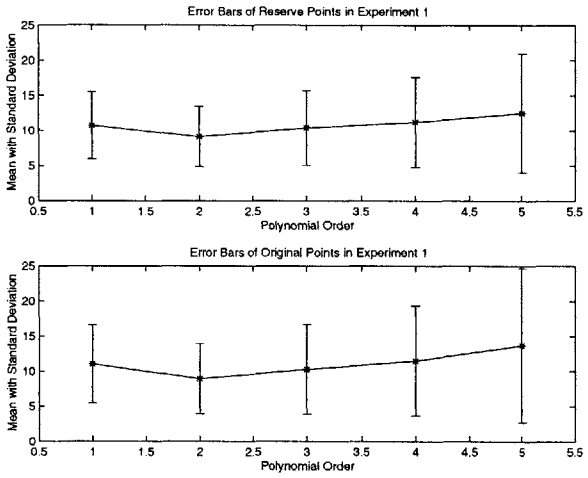
Table 6.3: Experiment 2: Original Sampling Rate of 3; Reserve Sampling Rate of 2

132 Original Points and 66 Reserve Points						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error} (voxels)	9.722	10.635	8.588	8.801	10.460	10.244
σ_{error} (voxels)	4.788	5.584	4.365	5.037	6.573	6.405
max_{error} (voxels)	20.452	30.122	20.874	27.146	39.475	29.442
med_{error} (voxels)	9.329	10.230	7.672	7.883	9.369	8.572
poly. order	4		5			
	Reserve	*Original*	*Reserve*	*Original*		
μ_{error} (voxels)	10.595	12.023	17.731	19.433		
σ_{error} (voxels)	5.867	8.807	21.413	21.009		
max_{error} (voxels)	26.495	48.671	144.760	112.643		
med_{error} (voxels)	10.137	10.305	11.167	13.114		

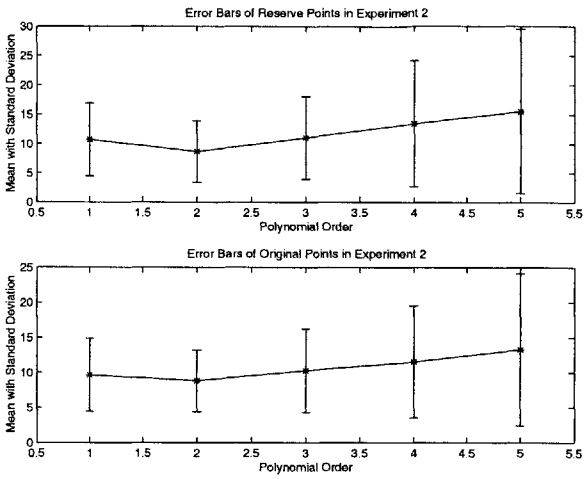
Table 6.4: Experiment 3: Original Sampling Rate of 4; Reserve Sampling Rate of 2

106 Original Points and 53 Reserve Points						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error} (voxels)	11.371	10.822	9.257	8.934	11.619	10.834
σ_{error} (voxels)	6.163	4.694	4.785	4.052	7.602	5.406
max_{error} (voxels)	25.666	22.194	22.971	20.571	30.544	23.980
med_{error} (voxels)	10.923	11.283	8.340	8.774	8.337	10.575
poly. order	4		5			
	Reserve	*Original*	*Reserve*	*Original*		
μ_{error} (voxels)	14.080	13.402	26.416	22.842		
σ_{error} (voxels)	12.629	10.816	50.339	40.068		
max_{error} (voxels)	62.826	82.081	244.690	329.679		
med_{error} (voxels)	9.288	12.308	9.880	13.898		

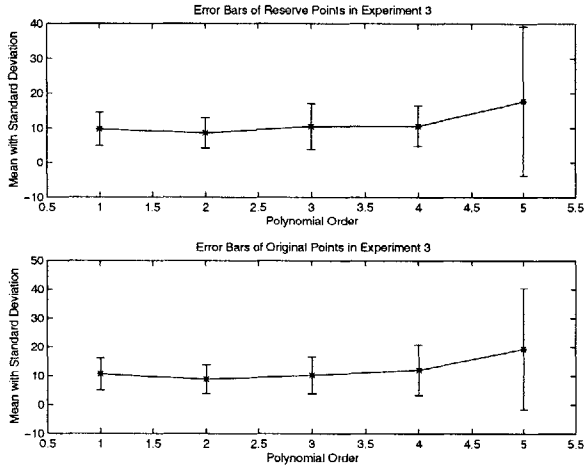
Table 6.5: Experiment 4: Original Sampling Rate of 5; Reserve Sampling Rate of 2



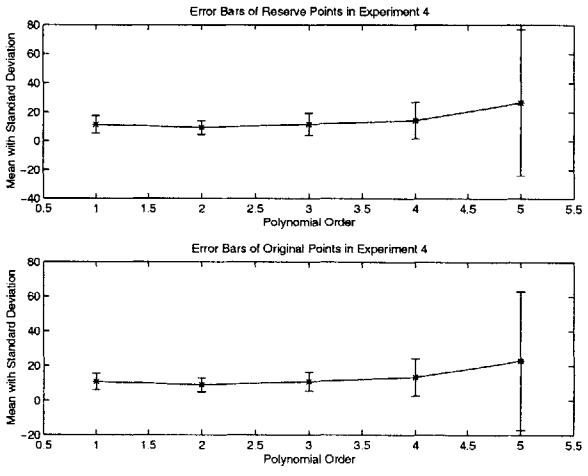
(a) Experiment 1 (refer to Table 6.2)



(b) Experiment 2 (refer to Table 6.3)

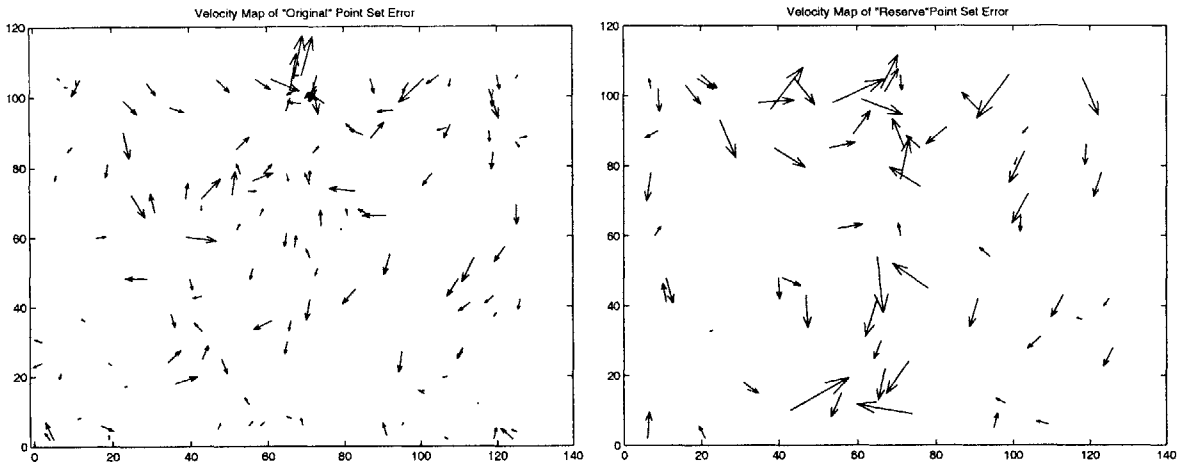


(c) Experiment 3 (refer to Table 6.4)



(d) Experiment 4 (refer to Table 6.5)

Figure 6-5: Error Bars illustrating μ_{error} (*) and σ_{error}



(a) Original Point Error (132 points)

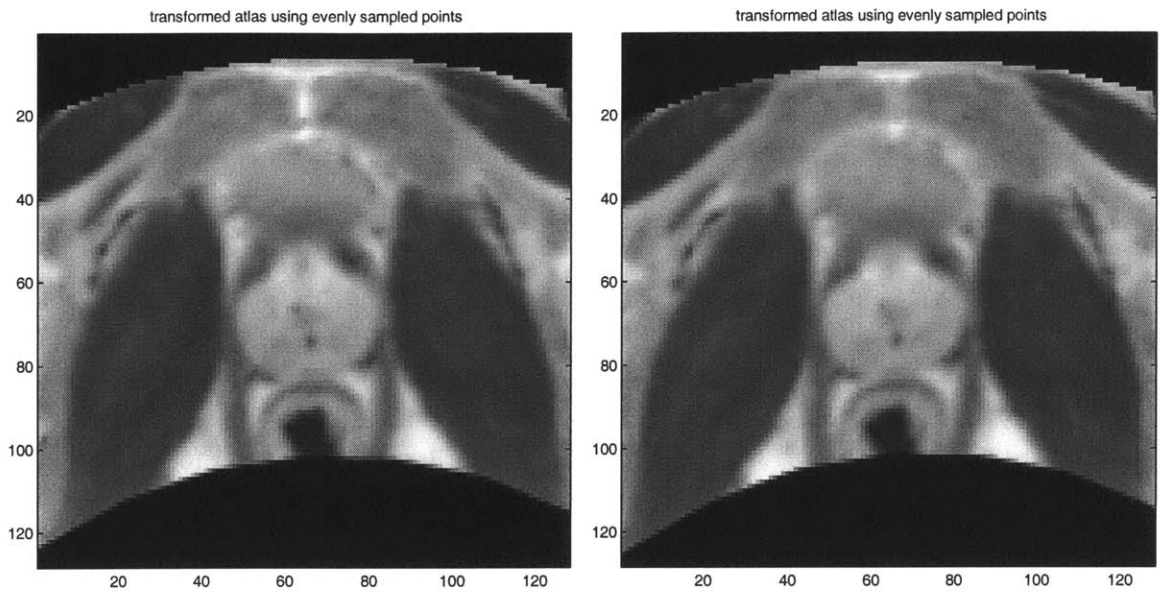
(b) Reserve Point Error (66 points)

Figure 6-6: Error maps for *original* and *reserve* points in Experiment 3

6.2.1 Results and Conclusions

Because the least error in our calculations stemmed from computing the 2^{nd} order polynomial with 66 *reserve* points and 132 *original* points (Table 6.4), we decided to compare those results with a "ground truth" experiment. This "ground truth" result was used to compare results of computing the deformation field with all 526 corresponding points versus a sample consisting of 132 points. We proceeded with a 2^{nd} order polynomial calculation of a data set with no sampling, which produced μ_{error} and σ_{error} values of 9.301 and 4.793 voxels. We provide error maps for Experiment 3, which involved 132 *original* points and 66 *reserve* points, to show the difference in the transformed point location and its actual location as seen in Figure 6-6.

A comparison of the registration results of the two data sets is provided in Figure 6-7 along with a difference of the images in Figure 6-8. The difference image confirms that the registration results involving 526 *original* points versus 132 *original* points are not equivalent as expected from the calculated error results but, are quite similar. This result implies that fewer points can be used to compute the deformation field for registration of CT/MR.



(a) Registration results using 132 points

(b) Registration results using 526 points

Figure 6-7: Registration results using 132 and 526 points

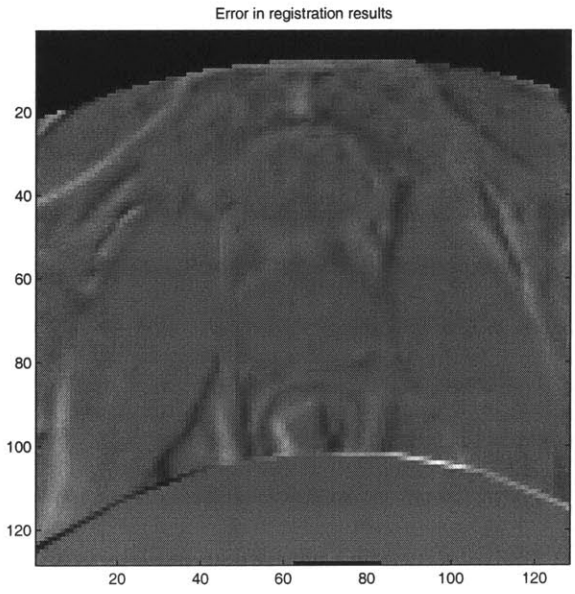


Figure 6-8: Difference between registration results using 132 and 526 points

6.3 Investigation of Clustered Point Set Distributions

In the following experiments, we aimed to calculate and visualize errors for which registration was computed from clustered point distributions versus the evenly sampled point distributions evaluated in previous experiments. We analyzed registration results for various corresponding point clusters throughout the entire volume of CT and MR images, which both had dimensions of $128 \times 128 \times 149$. For each of the following experiments involving the clustered distributions, we used the registration results from the point set with no sampling to compare with the registration results using the point clusters. The registration results are provided in the lower left hand corner of the provided figures while the differences between the CT and MR are provided in the lower right hand corner of each figure. Note that our error analysis algorithm is not utilized in these experiments thus, the overlap of the 526 points and the clustered points does not affect our analysis. We provided errors between the actual and transformed point locations for the non-sampled point set in Table 6.6 and errors for the clustered distributions were tabulated in Table 6.7. The polynomial order for these experiments was evaluated at 2.

In the first set of experiments, we divided the entire point set into quadrants (127 points for Quadrant 1; 124 points for Quadrant 2; 118 points for Quadrant 3; 145 points for Quadrant 4) by defining bounds between 1, 64, and 128 for the entire $128 \times 128 \times 149$ image. These clustered points were used to compute the deformation field that registered CT/MR. Because corresponding points in the CT and MR were not congruent with respect to the boundary locations, some residual MR image points were located outside of the quadrant boundary while the CT image remained within the defined boundary as shown in Figures 6-9 through 6-12. The figures also display the registration results of the clustered data sets along with comparisons to registration that involved no sampling of points.

In the second set of experiments, we sought more realistic examples in which point correspondences were chosen in a particular region of interest. We chose correspond-

ing points on the prostate of CT/MR in Figures 6-13 and in the center of the images in Figure 6-14. Furthermore, we experimented with point clusters along the outer edges of the CT and MR images. Registration results using points particularly in the left and right regions of CT and MR are provided in Figure 6-15 while results using points in the top and bottom regions of the images are shown in Figure 6-16. Each of the point clusters varied in quantity as indicated in Table 6.8, which also provides error results for each of the four clustered "region of interest" experiments.

6.3.1 Results and Conclusions

Regarding the experiments involving quadrants of points, the expected results were produced in which registration accuracy was directly dependent on point distribution in deformation field calculation. Because the deformation field was based only on particular points in a respective quadrant or region, we expected accurate registration only in that particular area. We illustrated in Figures 6-9 through 6-12 that the algorithm worked to accurately register only particular quadrants in each case. Further confirmation is provided in Table 6.7 where we present comparable errors to results in which evenly sampled points were used instead of clusters.

The second set of experiments, containing more realistic chosen point distributions, further supported the accuracy in our registration algorithm. It important to note that different point samples were used in each case. In Figure 6-13, 110 corresponding points were chosen around the prostate. We observed poor registration results in this experiments as can be seen in the figure. The registration result was compressed and was a warped version of the original CT/MR image. We concluded that using correspondences in an isolated prostate area is a bad choice for registration purposes.

Compared to the prostrate example, point correspondences in the center of the images were relatively better. However, as shown in Figure 6-14, the registration is definitely not acceptable for clinical purposes due to warping in the outer regions of the registration result. The results are only slightly comparable to the results using an evenly sampled set of points as seen in the difference image in the lower right

hand corner of the figure. However, quantitatively the error results in the center are quite similar to evenly sampled point sets with μ_{error} values both of approximately 9 voxels.

The last two experiments with outer area point distributions gave the most satisfactory results of all of the experiments involving various point distributions. In Figure 6-15, the correspondences were all of the points with x-coordinates less than 35 and greater than 100 and all of their relative y-coordinates from 1 to 128. The difference image (lower right corner) produced more accurate results in the right-hand side of image compared to the left-hand side. We expect the registration results to balance out if more correspondences are chosen in the left hand side of the image. A total of 225 points were used to compute the results for this experiment. We were pleased with the μ_{error} value of 7.682 voxels, which was lower that any of the experiments involving evenly sampled points and polynomial order 2 in Tables 6.2 through 6.5.

The lowest μ_{error} of 6.954 voxels was produced from the experiment involving points chosen from the top and bottom areas of the CT/MR images to compute the registration. The point quantity was 312 and provided a registration error image that was the best result from all of the point distribution experiments. Only minor offsets were apparent when compared to registration results involving 526 points. The bounds were chosen by defining the x-coordinates to include points ranging from 1 to 128 and the y-coordinates to include points less than 30 and greater than 85. There were no compressed or asymmetrical registration characteristics as in the previous experiments involving quadrants or other point distributions. We gather from these results that the algorithm most accurately computes the registration from the corresponding points in clusters with a significant horizontal distribution.

Errors without Sampling (voxels)			
μ_{error}	σ_{error}	max_{error}	med_{error}
9.316	4.785	31.243	8.539

Table 6.6: Registration error results from 526 point set

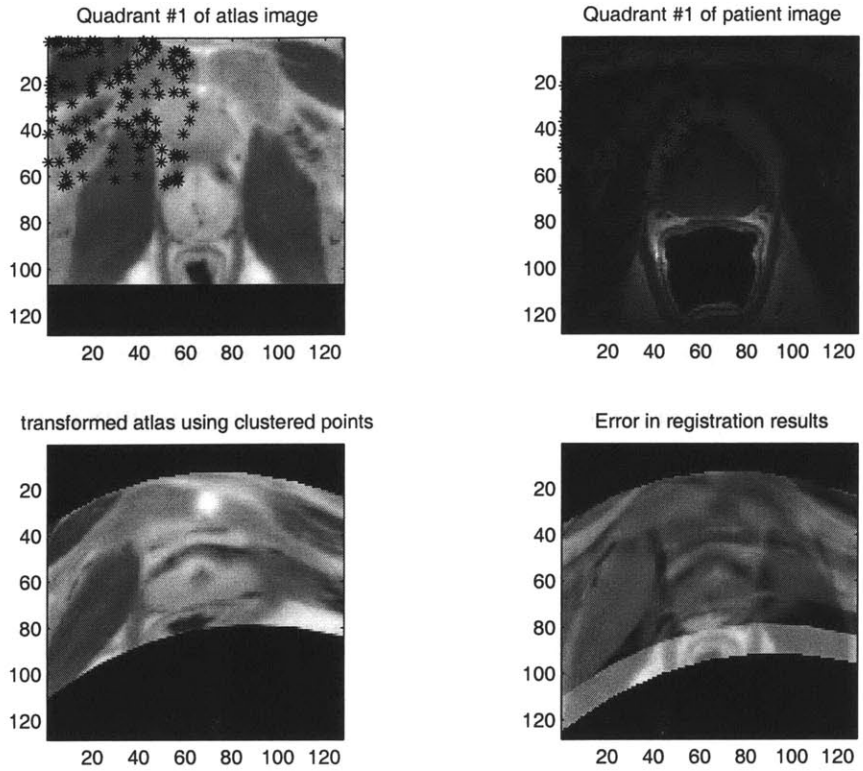


Figure 6-9: Quadrant 1 Registration Results

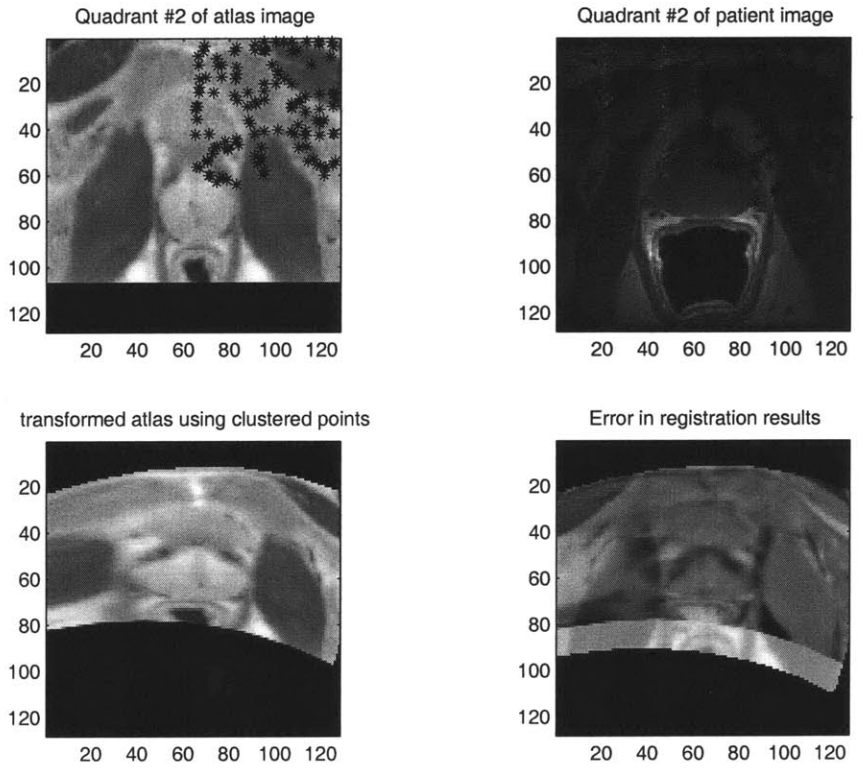


Figure 6-10: Quadrant 2 Registration Results

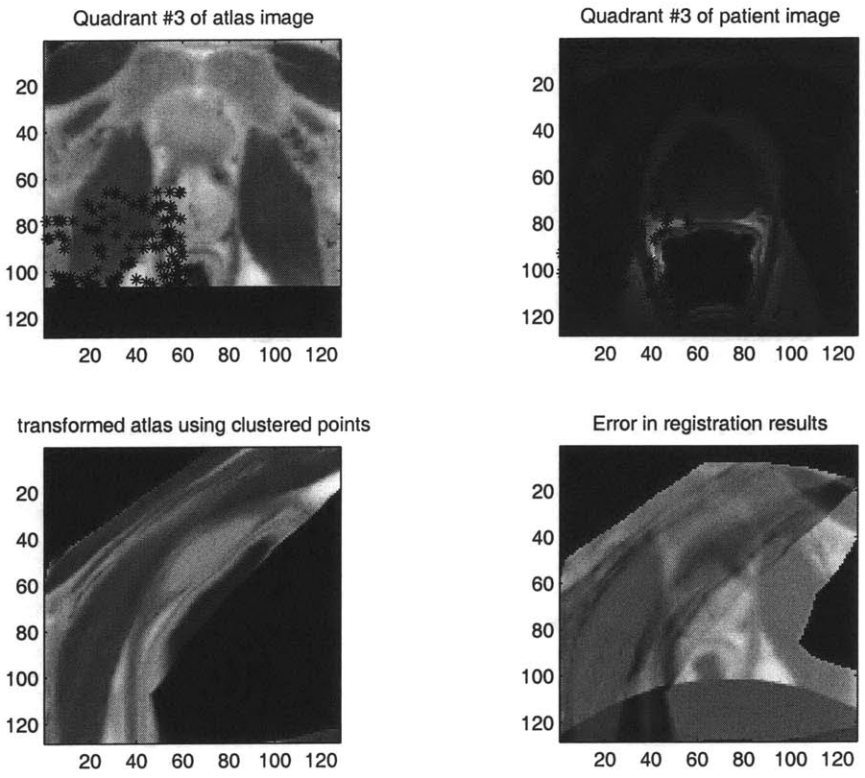


Figure 6-11: Quadrant 3 Registration Results

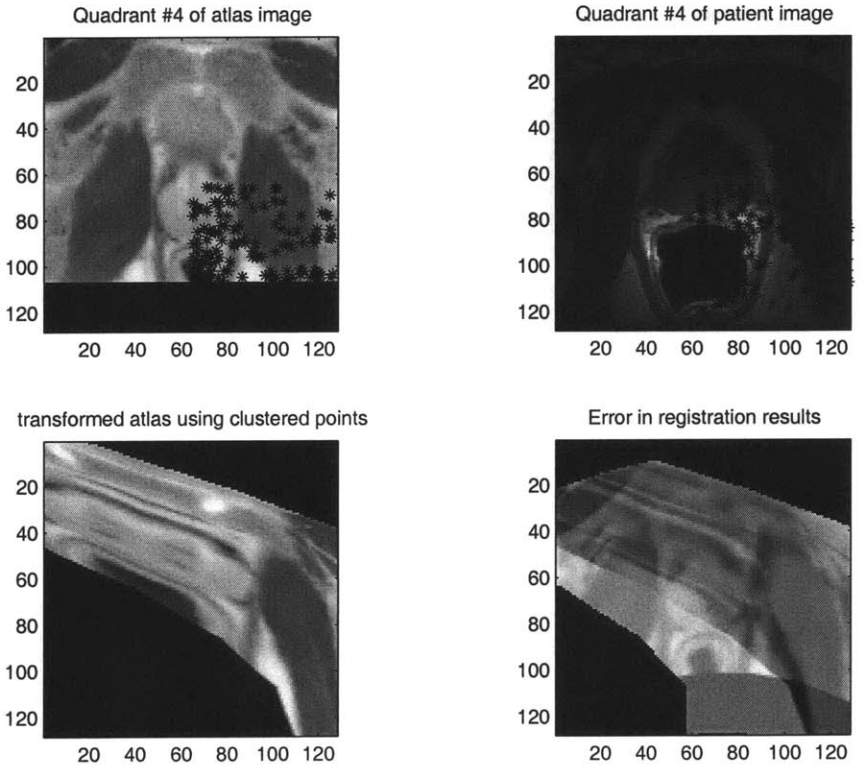


Figure 6-12: Quadrant 4 Registration Results

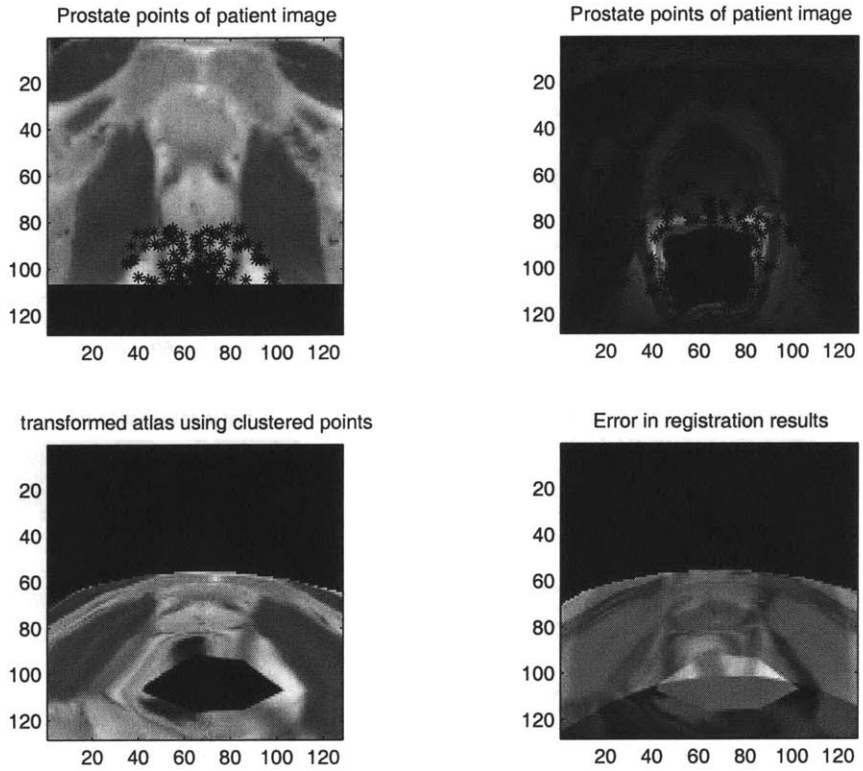


Figure 6-13: Prostate Only Registration Results

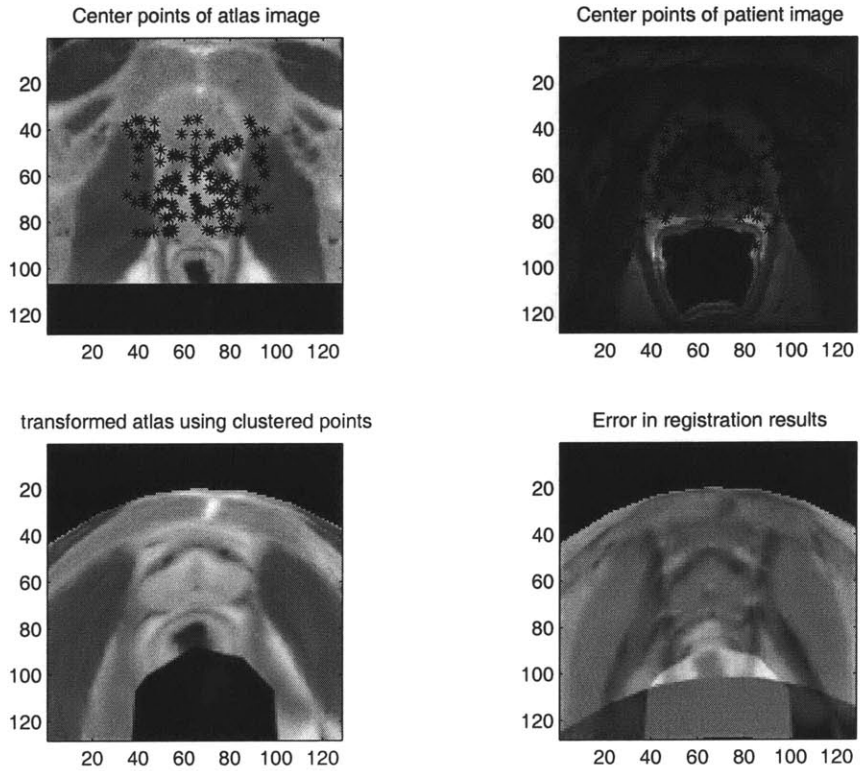


Figure 6-14: Center Points Registration Results

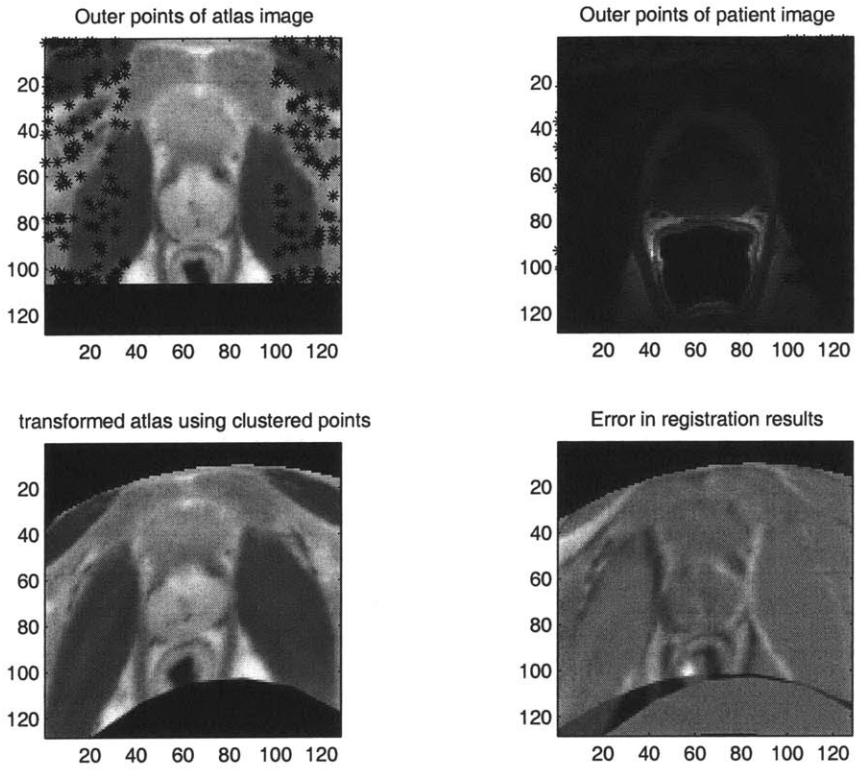


Figure 6-15: Left/Right Border Registration Results

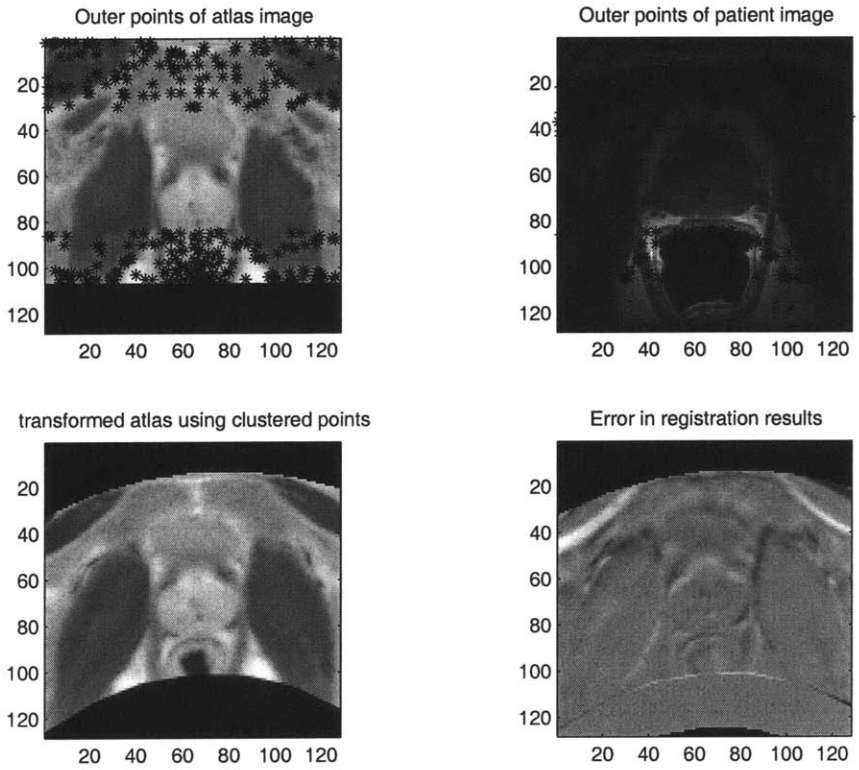


Figure 6-16: Top/Bottom Border Registration Results

Quadrant 1 Errors (voxels)	
μ_{error}	8.328
σ_{error}	4.747
max_{error}	31.663
med_{error}	7.133

Quadrant 2 Errors (voxels)	
μ_{error}	9.327
σ_{error}	4.721
max_{error}	26.944
med_{error}	8.239

Quadrant 3 Errors (voxels)	
μ_{error}	10.513
σ_{error}	4.524
max_{error}	25.499
med_{error}	10.784

Quadrant 4 Errors (voxels)	
μ_{error}	9.191
σ_{error}	4.928
max_{error}	26.845
med_{error}	8.270

Table 6.7: Quadrant Point Set Distributions

Prost. Errors–110 points	
μ_{error}	11.247
σ_{error}	5.151
max_{error}	26.845
med_{error}	10.823

Center Errors–137 points	
μ_{error}	9.156
σ_{error}	4.736
max_{error}	31.663
med_{error}	8.656

Lft/Rt Errors–225 points	
μ_{error}	7.682
σ_{error}	4.659
max_{error}	28.091
med_{error}	6.420

Tp/Bot Errors–312 points	
μ_{error}	6.954
σ_{error}	4.143
max_{error}	26.944
med_{error}	5.966

Table 6.8: Various Point Set Distribution Errors (voxels)

6.4 Error Validation

To validate the error, we decided to measure errors using known deformation fields. We defined a $64 \times 64 \times 64$ "cube" of random numbers. We then created a random point set, which we denoted $p1$. We performed the following operations on $p1$ to create the 2^{nd} point set $p2$: rotation by 5 degrees; translation of 5 voxels, and scaled by 0.5. We then used our algorithm on four experiments using 50,100, 150, and 200 points. The *original* points were sampled by 2 in each experiment and the *reserve* points were chosen as all points not included in the *original* set. This in effect evenly divided the number of points between the two point sets. We analyzed results for polynomial orders 1, 2, and 3. The resulting error calculations are shown in Tables 6.9 through 6.12. We concluded that our error results were accurate because the values were close to zero as expected, especially given the complexity of the transformation.

Exact error values of 0 voxels were produced in trials involving only a translated 2^{nd} point set.

Most importantly, we wanted to validate our error analysis algorithm which produced the results for the variation in point set quantity and distribution and polynomial order. By showing that minimal errors can be achieved in the simple case using known deformation fields, we offer support for our registration results of CT/MR. We have demonstrated that accurate results can be achieved by using clusters of points versus a point set consisting of evenly sampled correspondences throughout the entire CT/MR images. Due to the fact that error values are expected to be of the order of 2 voxels or less, we realize that a different approach must be taken regarding our registration calculations. We expect the results to improve if we were to use non-linear interpolation functions in future computations instead of linear interpolation functions used to compute a dense deformation field in these results.

Error Validation – 50 points (voxels)						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error}	0.600	0.745	0.942	0.829	2.122	2.028
σ_{error}	0.332	0.391	0.530	0.470	0.586	0.658

Table 6.9: Error Validation–25 *original* points and 25 *reserve* points

Error Validation – 100 points (voxels)						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error}	0.651	0.794	0.848	0.773	0.871	0.830
σ_{error}	0.354	0.464	0.518	0.501	0.418	0.439

Table 6.10: Error Validation–50 *original* points and 50 *reserve* points

Error Validation – 150 points (voxels)						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error}	0.809	0.792	0.873	0.898	0.743	0.730
σ_{error}	0.456	0.452	0.591	0.573	0.348	0.409

Table 6.11: Error Validation–75 *original* points and 75 *reserve* points

Error Validation – 200 points (voxels)						
poly. order	1		2		3	
	Reserve	*Original*	*Reserve*	*Original*	*Reserve*	*Original*
μ_{error}	0.788	0.839	0.773	0.769	0.911	0.864
σ_{error}	0.536	0.510	0.449	0.456	0.446	0.477

Table 6.12: Error Validation–100 *original* points and 100 *reserve* points

6.5 Conclusions

Based upon our calculations, we concluded that the 2^{nd} order polynomial was sufficient for minimizing the error in the registration calculations. Note that for each experiment involving polynomial orders 1 through 4, the greatest μ_{error} and σ_{error} decrease occurred in the transition from order 1 to 2 and greatest increase from order 2 to 3. In a 2^{nd} order polynomial there are three points through which the polynomials must pass, and we conclude from our error analysis that it is the best polynomial fit given the particular sparsity of our data. We realize that polynomial order 1 is not expected to properly fit sparse data while higher order polynomials such as 3 through 5 in our case, produce large errors due to overfitting the data.

It is important to note that the results in Figure 6-7 are quite similar despite the fact that there are approximately 4 times as fewer points in Figures 6-7(a). This implies that fewer corresponding points can be used to compute an accurate registration. Also, we showed that a horizontally inclined correspondence distribution produced the minimum μ_{error} calculation compared to using any of the distributions tested in this thesis. Investigations into other interpolation functions must be completed before further conclusions can be drawn.

6.6 Future Work

We realize that caution must apply when using polynomial interpolators, particularly with very sparse data points. In some cases the fitting polynomial can introduce wild variations above or below the actual data points. The fitting polynomial does actually pass through all the data points, but by forcing it to be of the minimal degree, the polynomial may become a very poor representation of the data between the given data points. In these cases it is frequently better to use other techniques, such as a spline curve through the data points. In this case the data is represented by a succession of cubic polynomials chosen so that there are no kinks, in the sense that that the derivative is continuous. In addition, these cubic spline pieces are constrained to be the straightest pieces possible. Thus these spline curves are what one would get by threading a reasonably stiff piece of wire (a spline) through all the data points [34].

In addition to investigating different interpolation functions, we would like to observe results from more randomized point distributions than the clustering presented. Particularly, we will investigate more cases of horizontal point set distributions with fewer points than the 312 points for which we provide results. We will investigate various methods for calculating the deformation field. Ideally, we would like to have errors that have a μ_{error} and σ_{error} of no more than 2 voxels, which would be appropriate in clinical applications.

We will return to the problem of feature detection in ultrasound in anticipation that more prominent edges can be detected for organ boundary isolation. Currently, we have corresponding CT and ultrasound liver data, but no corresponding MR data. In addition, we have corresponding CT/MR prostate data but, no corresponding prostate ultrasound data. Thus far, our plan is to compute an initial alignment between our CT and ultrasound liver data until we can obtain CT with corresponding MR data. We will investigate algorithms such as interpretation tree search, ICP, and Mixture Point Matching (MPM) to determine automatic correspondences to replace our current manual method. A segmentation algorithm for surface point extraction of CT/MR is necessary so that we can use ICP, for instance, to match the surface points

to edge points in ultrasound. This will provide an initial correspondence between the two data sets. Our polynomial warping algorithm will then be used to refine the matching, taking the correspondence returned by ICP as the set of matching points. Finally, we will warp the entire CT/MR data set to the ultrasound for our registration results which will eventually be extended to real-time.

Bibliography

[1] K.S. Arun, T.S. Huang, and S.D. Blostein. Least-squares fitting of two 3-D point sets. *IEEE Trans. Pattern Anal. Machine Intell.*, (5):698–700, 1987.

[2] P.J. Besl. Geometric modeling and computer vision. In *Proc. IEEE*, volume 76, pages 936–958, 1988.

[3] P.J. Besl and N.D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[4] J.F. Canny. Finding edges and lines in images. Master's thesis, MIT AI Lab, 1983.

[5] H. Chui and A. Rangarajan. A feature registration framework using mixture models. page 190, 2000.

[6] H. Chui and A. Rangarajan. A new algorithm for non-rigid point matching. volume 2, page 190, 2000.

[7] E. Cosman. Rigid Registration of MR and Biplanar Fluoroscopy. Master's thesis, MIT AI Lab, 2000.

[8] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* Wiley and Sons, New York, NY, 1973.

[9] D.J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of The Optical Society of America A*, 4(12):2379–2394, 1987.

[10] R. Franke. A critical comparison of some methods for interpolation of scattered data. Technical report, Naval Postgraduate School, 1988.

[11] A. Goshtasby. Image registration by local approximation. *Image Vision Comput.*, 6(4):255–261, 1988.

[12] L. Gottesfield-Brown. A survey of image registration. *ACM Computing Surveys*, 24(4):325–376, 1994.

[13] W.E.L. Grimson. *Object Recognition by Computer: The Role of Geometric Constraints.* The MIT Press, Cambridge, 1990.

[14] B.K.P. Horn. Closed-form solution of absolute orientation using orthonormal matrices. *J. Opt. Soc. Amer. A.*, 5(7):1127–1135, 1987.

[15] B.K.P. Horn. Closed-form solution of absolute orientation using unit quaternions. *Comput. Vision Graphics Image Processing*, 1989.

[16] J.P. Hornak. The Basics of MRI. *http://www.cis.rit.edu/htbooks/mri/inside.htm.*

[17] F. Hottier and A.C. Billon. *3D Imaging in Medicine: Algorithms, Systems, Applications.* NATO ASI Series, 1990.

[18] F.A. Jolesz. Image-guided procedures and the operating room of the future. *Radiology*, 204:601, 1997.

[19] M. Kass, A. Witkin, and D.Terzopoulos. Snakes: active contour models. *Intl. J. Computer Vision*, 2, 1988.

[20] D.L. King and P.P. Lele. *Diagnostic Ultrasound.* The C.V. Mosby Co., 1974.

[21] H. Knutsson and G.H. Granlund. Fourier domain design of line and edge detectors. In *Proceedings of the 5th International Conference on Pattern Recognition*, Miami, Florida, December 1980.

[22] P. Kovesi. Binary images: Thresholding, Moments, Topology. *http://www.cs.uwa.edu.au/undergraduate/courses/233.412/Lectures/Lecture2/lecture2.html.*

[23] P. Kovesi. *Invariant Measures of Image Features from Phase Information.* PhD thesis, University of Western Australia, May 1996.

[24] P. Kovesi. Image features from phase congruency. *Videre: Journal of Computer Vision Research*, 1(3), 1999.

[25] F.W. Kremkau. *Diagnostic Ultrasound: Principles and Instruments, 4th ed.* W.B. Saunders Co., 1983.

[26] J.F. Lehman and A.W. Guy. Ultrasound therapy, interaction of ultrasound and biological tissues. In *Workshop Proceedings, U.S. Department of Health, Education, and Welfare Publication*, Washington, D.C., 1973.

[27] J. Morlet, G. Arens, E. Fourgeau, and D. Giard. Wave propagation and sampling theory-part II: sampling theory and complex waves. *Geophysics*, 47(2):222–236, 1982.

[28] M.C. Morrone and R.A. Owens. Feature detection from local energy. *Pattern Recognition Letters*, 6, 1987.

[29] M.C. Morrone, R. Ross, D.C. Burr, and R.A. Owens. Mach bands are phase dependent. *Nature*, 324, 1986.

[30] M. Moshfeghi, S. Ranganath, and K. Nawyn. Three-dimensional elastic matching of volumes. *IEEE Transactions on Image Processing*, 3(2):128–138, 1994.

[31] M. Mulet-Parada and J.A. Noble. 2D+T acoustic boundary detection in echocardiography. In *Medical Image Computing and Computer-Assisted Intervention-MICCAI '98*, 1998.

[32] N.Ayache, J.D. Boissonnat, L. Cohen, B. Geiger, J. Levy-Vehel, O. Monga, and P. Sande. *3D Imaging in Medicine: Algorithms, Systems, Applications.* NATO ASI Series, 1990.

[33] H.R. Schelbert O. Ratib, L. Bidaut and M.E. Phelps. A new technique for elastic registration of tomographic images. In *IEEE Proceedings of SPIE: Medical Imaging II*, volume 914, pages 452–455, 1988.

[34] University of Toronto. Interpolating Data. *http://www.scar.utoronto.ca/ dyer/csca57/interpolate/interpolate.html.*

[35] P. Perona and J. Malik. Detecting and localizing edges composed of steps, peaks, and roofs. volume 24, pages 52–57, Osaka, 1990.

[36] J.P.W. Pluim, J.B. Maintz, and M.Viergever. Mutual information matching in multiresolution contexts. Technical report.

[37] B.A. Porter, W. Hastrup, and M.L. Richardson. Classification and investigation of artifacts in Magnetic Resonance Imaging. *RadioGraphics*, 7(2):271, 1987.

[38] R. Prager and A. Gee. Three-Dimensional Ultrasound Imaging. *http://www.quns.cam.ac.uk/Queens/Record/2000/.*

[39] R.J. Seitz, C. Bohm, T. Greitz, P.E. Roland, L. Eriksson, G. Blomqvist, and G. Nordell. Accuracy and precision of the computerized brain atlas programme for localization and quantification in positron emission tomography. *J. Cereb. Blood Flow. Metab.*, 10:443–457, 1990.

[40] M. Singh, W. Frei, T. Shibata, G. C. Huth, and N.E. Telfer. A digital technique for accurate change detection in nuclear medical images–with application to myocardial perfusion studies using thallium-201. *IEEE Trans. Nuclear Sci.*, 26(1), 1979.

[41] Arthur W. Toga. *Brain Warping.* Academic Press, California, 1999.

[42] S. Umeyama. Least squares estimation of transformation parameters between two point patterns. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 13(4):376–380, 1991.

[43] S. Venkatesh and R. Owens. An energy feature detection scheme. In *The Intl. Conference on Image Processing*, Singapore, 1989.

[44] S. Web, editor. *Medical Sciences Series.* Institute of Physics Pub. Ltd., Bristol and Philadelphia, 1998.

[45] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International Journal of Computer Vision*, 13(2):119–152, 1994.