

Abstract

*Statistical Shape Analysis for Image Segmentation
and Physical Model-Based Non-Rigid Registration*

Yongmei Wang

1999

This dissertation presents novel statistical shape analysis methods for both segmentation and non-rigid registration, which are two of the most important topics in medical image analysis.

For the proposed approaches to boundary finding, the correspondence of a subset of boundary points to a model is simultaneously determined. Global shape parameters derived from the statistical variation of object boundary points in a training set are used to model the object. A Bayesian formulation, based on this prior knowledge and the edge information of the input image, is employed to find the object boundary with its subset points in correspondence with boundaries in the training set or the mean boundary. In order to demonstrate the power of this statistical information, the use of a generic smoothness prior and a uniform independent prior are compared with the training set prior. An integrated approach is also described and validated which uses a combined prior of the smoothness and statistical variation modes when few training example shapes are available. This approach adapts gradually to use more statistical modes of variation as larger data sets are available.

The resulting corresponding boundary points derived from the segmentation are then incorporated into our physical model-based non-rigid registration. The two new atlas-based methods of 2D single modality non-rigid registration proposed in this work

use the combined power of physical and statistical shape models. A Bayesian formulation, based on each physical model (elastic solid and viscous fluid), an intensity similarity measure, and statistical shape information embedded in corresponding boundary points, is employed to derive more accurate and robust approaches to non-rigid registration.

Finally, the 3D generalization to volumetric segmentation is addressed with emphasis on the new techniques required, which include the identification of corresponding surface points in the training set and 3D surface triangulation. They are efficiently computed together in a new hierarchical approach.

Throughout all the work in this thesis, the key link is **statistical shape**, which is the prior model in segmentation, as well as the extra source of information in non-rigid registration.

Statistical Shape Analysis for Image Segmentation
and Physical Model-Based Non-Rigid Registration

A Dissertation

Presented to the Faculty of the Graduate School

of

Yale University

in Candidacy for the Degree of

Doctor of Philosophy

by

Yongmei Wang

Dissertation Director: Lawrence Hamilton Staib

May 1999

© 2000 by Yongmei Wang

ALL RIGHTS RESERVED

Acknowledgments

My thanks go to the faculty and students of the Image Processing and Analysis Group / Department of Electrical Engineering for their help during my years at Yale University.

First, I am especially grateful to my mentor and academic advisor, Lawrence Staib, for his invaluable guidance, encouragement, patience, ideas and support throughout my thesis research. I am thankful to James Duncan for not only serving on my committee, but also for introducing me to the areas of Image Processing & Computer Vision. I also thank Roman Kuc and Peter Belhumeur for serving on my committee, and giving me helpful knowledge in Signal Processing and Pattern Recognition, respectively. Thanks to Gábor Székely for being my external reader.

This work was made enjoyable by the congenial, open and egalitarian atmosphere of the Image Processing and Analysis Group. I thank all the past and present members who made this possible: Amir Amini, Ravi Bansal, Amit Chakraborty, Haili Chui, John McEachen, Francois Meyer, Wiro Neissen, Xenios Papademetris, Suguna Pappu, N. Rajeevan, Jim Rambo, Anand Rangarajan, Pengcheng Shi, Oskar Skrinjar, Rik Stokking, Colin Studholme, Hemant Tagare, Lawrence Win, Xiaolan Zeng, Lei Zhang, and George Zubal. Additional thanks go to Carolyn Meloling for providing administrative assistance.

I thank my parents for their support, attention and moral guidance throughout my life. I will always be indebted to my aunt for the intellectual influence she had on me.

I dedicate this thesis to my son, Elbert, who was born two years ago and has brought me lots of enjoyable memory, in addition to the hardship.

This work was supported in part by a grant from the Whitaker Foundation.

List of Figures

| | | |
|------|---|----|
| 1.1 | Process of non-rigid registration | 3 |
| 1.2 | Diagram for our statistical shape analysis methods | 5 |
| 1.3 | Overview of the 2D system framework | 7 |
| 3.1 | Synthetic shape model | 30 |
| 3.2 | Heart shape model | 34 |
| 3.3 | Effects of $C_{identity}$ with the synthetic model | 36 |
| 3.4 | Effects of C_{smooth} with the synthetic model | 39 |
| 3.5 | Comparison of eigenvalues of C_{smooth} at three different scales for a 24 point model | 40 |
| 3.6 | Effects of C_{smooth} with the square & rectangle model | 44 |
| 3.7 | Effects of C_{mix} with the square & rectangle model | 45 |
| 3.8 | Synthetic image example | 56 |
| 3.9 | Sensitivity to noise experiment | 57 |
| 3.10 | Sensitivity to initialization experiment | 59 |
| 3.11 | Bias experiment | 60 |
| 3.12 | Prior experiment for synthetic image | 61 |
| 3.13 | Corpus callosum shape model | 63 |

| | | |
|------|---|----|
| 3.14 | MR sagittal brain example | 64 |
| 3.15 | MR axial brain example | 65 |
| 3.16 | MR heart example and prior experiment | 66 |
| 3.17 | MR heart images used to evaluate the performance of the combined model C_{mix} | 68 |
| 3.18 | Relationship between the error, the proportion of C_{smooth} included (w) and the number of training examples (m) | 69 |
| 3.19 | Performance comparison of $C_{training}$ and C_{mix} with varying number of examples (m) | 70 |
| 4.1 | Comparison of synthetic image non-rigid registration by Christensen’s elas- tic and fluid methods and our fluid method | 74 |
| 4.2 | Comparison of synthetic image non-rigid registration by Christensen&Miller’s and our elastic method | 75 |
| 4.3 | Comparison of synthetic multi-object image non-rigid registration by Chris- tensen’s and our fluid methods | 76 |
| 4.4 | Comparison of synthetic multi-gradient image non-rigid registration by Christensen&Miller’s, Davatzikos’ and our elastic methods | 77 |
| 4.5 | Synthetic images and displacement vectors by our and Christensen&Miller’s elastic methods | 94 |
| 4.6 | Noisy images for synthetic image sensitivity-to-noise experiment by our and Christensen&Miller’s elastic methods | 95 |
| 4.7 | Sensitivity to noise experiment for synthetic images by our and Christensen&Miller’s elastic methods | 96 |

| | | |
|------|--|-----|
| 4.8 | MR sagittal corpus callosum images and synthetic displacement vectors by our and Christensen&Miller’s elastic methods | 97 |
| 4.9 | MR sagittal corpus callosum image and synthetic displacement vectors by Christensen’s and our fluid methods, and our elastic method | 98 |
| 4.10 | Axial brain shape model | 101 |
| 4.11 | MR axial brain images and displacement vectors | 102 |
| 4.12 | MR heart images and displacement vectors | 103 |
| 4.13 | Demonstration of boundary finding with correspondence influenced by non-rigid registration (corpus callosum example (100×64)) | 108 |
| 5.1 | Diagram for corresponding surface points identification and surface triangulation in our hierarchical approach | 112 |
| 5.2 | Gradient calculation within a triangle | 114 |
| 5.3 | Shortest paths on a synthetic surface | 116 |
| 5.4 | 3D point set alignment process | 118 |
| 5.5 | 3D synthetic training set construction | 126 |
| 5.6 | Principal mode a_1 of the 3D synthetic model | 128 |
| 6.1 | Statistical shape analysis framework for segmentation and registration . . | 130 |

List of Tables

| | | |
|-----|---|-----|
| 3.1 | Error measure for the synthetic image in the prior experiment with different covariance matrices | 62 |
| 3.2 | Error measure for the heart image with different covariance matrices and different methods | 67 |
| 4.1 | Error measure for MR sagittal corpus callosum image with known warping (Figure 4.8) showing the improvement of our elastic method | 99 |
| 4.2 | Error measure for MR sagittal corpus callosum image with known warping (Figure 4.9) showing the improvement of our fluid method | 99 |
| 4.3 | Error measure E_{ba} and approximate execution time for MR brain (axial) and heart images | 104 |

Contents

| | |
|--|------------|
| Acknowledgements | iii |
| List of Figures | iv |
| List of Tables | vii |
| 1 Introduction | 1 |
| 1.1 Introduction to the Problem | 1 |
| 1.2 Overview of the Framework | 4 |
| 1.3 Main Contributions | 6 |
| 2 Background and Related Work | 9 |
| 2.1 Related Work in Image Segmentation | 9 |
| 2.1.1 Region Based Methods | 9 |
| 2.1.2 Boundary Methods | 10 |
| 2.1.2.1 Edge Detection | 11 |
| 2.1.2.2 Whole Boundary Methods | 12 |
| 2.2 Related Work in Image Registration | 17 |
| 2.2.1 Rigid Registration | 18 |

| | | |
|----------|--|-----------|
| 2.2.2 | Non-Rigid Registration | 19 |
| 2.2.2.1 | Affine, Projective and Curved Transformations | 20 |
| 2.2.2.2 | Bookstein’s Thin-plate Splines | 22 |
| 2.2.2.3 | Physical Models | 22 |
| 2.2.2.4 | Physical Model-Based Non-rigid Registration vs. Optical Flow | 24 |
| 2.3 | Related Work in Shape Matching (for Segmentation or Registration) . . . | 24 |
| 2.3.1 | Rigid Matching | 25 |
| 2.3.2 | Unconstrained Deformable Models | 25 |
| 2.3.3 | Parametric Deformable Models | 26 |
| 3 | Statistical Shape Analysis for Boundary Finding with Correspondence | 28 |
| 3.1 | Introduction | 28 |
| 3.2 | Statistical Shape Analysis | 29 |
| 3.2.1 | Statistical Point Model — $C_{training}$ | 29 |
| 3.2.1.1 | Labeling and Aligning the Training Set | 29 |
| 3.2.1.2 | Capturing the Statistics | 31 |
| 3.2.2 | Identity Covariance Matrix — $C_{identity}$ | 35 |
| 3.2.3 | Smoothness Covariance Matrix — C_{smooth} | 37 |
| 3.2.4 | Combined Model — C_{mix} | 41 |
| 3.3 | Bayesian Matching | 46 |
| 3.3.1 | Maximum <i>a Posteriori</i> Match | 47 |
| 3.3.1.1 | Prior Probability Density | 48 |
| 3.3.1.2 | Likelihood | 49 |
| 3.3.2 | Simplified Boundary Objective Function | 50 |

| | | |
|----------|--|-----------|
| 3.4 | Implementation | 53 |
| 3.4.1 | Optimization Algorithm | 53 |
| 3.4.2 | Gradient Formulation | 53 |
| 3.4.3 | Comparison Method and Computation Time | 55 |
| 3.5 | Experimental Results | 55 |
| 3.5.1 | Evaluation Criteria | 55 |
| 3.5.2 | Synthetic Images | 56 |
| 3.5.3 | Real Images | 62 |
| 3.6 | Conclusions | 68 |
| 4 | Physical Model-Based Non-Rigid Registration Incorporating Statistical Shape Information | 72 |
| 4.1 | Introduction | 72 |
| 4.2 | Physical Models | 79 |
| 4.2.1 | Reference Frames and Problem Statement | 79 |
| 4.2.2 | Elastic Model | 81 |
| 4.2.3 | Fluid Model | 82 |
| 4.2.4 | Viscous Fluid vs. Linear Elasticity | 83 |
| 4.3 | Integration | 84 |
| 4.4 | Implementation | 88 |
| 4.4.1 | Linear Elastic Model Algorithm — Algorithm 1 | 88 |
| 4.4.2 | Viscous Fluid Model Algorithm — Algorithm 2 | 88 |
| 4.4.3 | Successive Over-Relaxation (SOR) Algorithm | 90 |
| 4.4.4 | Different Methods and Computation Time Comparison | 91 |
| 4.5 | Experimental Results | 92 |

| | | |
|----------|---|------------|
| 4.5.1 | Evaluation Criterion | 92 |
| 4.5.2 | Synthetic Images with Known Warping | 93 |
| 4.5.3 | Real Image with Known Warping | 95 |
| 4.5.4 | Real Atlas and Study Images | 100 |
| 4.6 | Segmentation Influenced by Non-Rigid Registration | 105 |
| 4.7 | Conclusions | 107 |
| 5 | 3D Generalization to Volumetric Segmentation | 109 |
| 5.1 | Introduction | 109 |
| 5.2 | Training Set Labeling and Surface Triangulation — New Techniques Required | 110 |
| 5.2.1 | Step 1: Critical points labeling and initial triangle construction . . | 111 |
| 5.2.2 | Step 2: Shortest paths finding | 111 |
| 5.2.3 | Step 3: Mid-point selection and triangle subdivision | 115 |
| 5.2.4 | Step 4: Repeating step 2 and step 3 | 115 |
| 5.3 | Aligning the 3D Labeled Training Set Points | 116 |
| 5.4 | Surface Matching Objective Function | 117 |
| 5.4.1 | Transformation Matrix | 117 |
| 5.4.2 | Point Representation | 120 |
| 5.4.3 | Bayesian Objective Function | 122 |
| 5.4.4 | Gradient Formulation | 123 |
| 5.5 | Initial Synthetic Results | 125 |
| 5.6 | Conclusions | 127 |
| 6 | Summary | 129 |
| | Bibliography | 132 |

Chapter 1

Introduction

1.1 Introduction to the Problem

The goal of this work is to first develop statistical shape model-based segmentation and then use the resulting corresponding points which contain statistical shape information for physical model-based non-rigid registration.

Locating the boundaries of structures in an image is of great importance in a variety of image analysis and computer vision applications including robot vision, pattern recognition and biomedical image processing. Numerous boundary finding methods have been proposed [2, 23, 25, 32, 38, 45, 73, 78, 88, 92, 117, 119, 143]. However, these methods (except to some extent, [38, 45]) do not provide any notion of correspondence. Correspondence is a key step in a number of computer vision applications especially non-rigid registration. There has been much work on determining boundary correspondence using local shape features [121, 94]. In this work, however, we are interested in both determining an object's boundary and simultaneously determining spatial correspondence between similar structures over different subjects. Prior shape information is quite helpful

in constraining both the object boundary and point correspondence. Our first goal is to use statistical prior shape models with shape and shape variations generated from a training set to find the object boundary. In addition, since the shape model is a point based statistical model, the correspondence of boundary points to the model is simultaneously determined as a by-product of boundary finding. A Bayesian objective function based on this model and image derived information is employed in this thesis for segmentation. The resulting corresponding boundary points are then used in the physical model-based non-rigid registration, whose aim is to provide representations which allow for the generalization of a single ideal anatomical atlas to the individual.

An anatomical atlas is a registered set of images generated from typical healthy anatomy within a population. It consists of two types of images, measured and labeled. The measured set of images can consist of MR images, CT images, digitized photographs of histological sections, etc. The labeled images contain information about the measured images, consisting of structure names, sizes, locations, functionality, dependence on other structures, etc. The goal of our non-rigid registration is to estimate the transformation that accommodates the shape differences between the measured images of the atlas and measured images of an individual image, or study image (Figure 1.1). Once this transformation is known, all the information contained in the labeled images of the atlas can be mapped onto the measured images of the study.

Non-rigid registration allows point-by-point comparison between individuals of structure and function. Some of the applications of non-rigid registration are: comparing shape between individuals, comparing function between individuals, automatic segmentation, statistical characterization of shape variability, surgery, etc.

There have been many approaches to non-rigid registration in recent years [8,

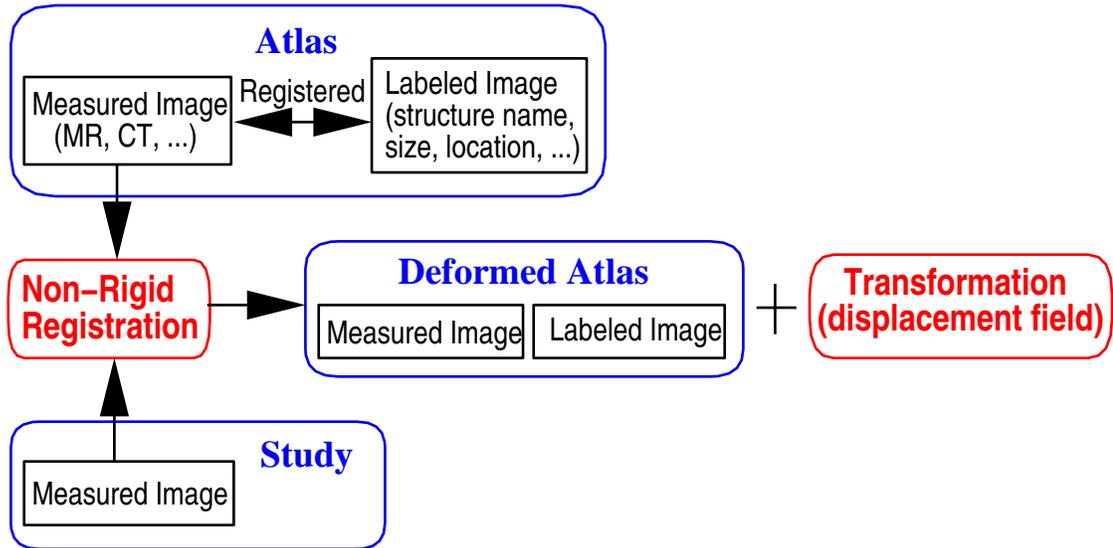


Figure 1.1: Process of non-rigid registration.

14, 29, 33, 40, 53, 96, 123]. Usually, the transformation is constrained in some way because of the ill-posedness (i.e. in this case, the existence of many possible solutions) of the problem. Physical models, for example, linear elastic and viscous fluid models, are widely used to enforce topological properties on the deformation and then constrain the enormous solution space [8, 29, 31, 40, 53, 96]. Such models are limited by themselves since the physical analogy is artificial and sometimes they are under-constrained. With statistical information, we have a stronger bias for the physical model. Statistical shape models, instead of physical models, can be powerful tools to directly capture the character of the variability of the individuals being modeled. This idea has been used successfully for segmentation but has not yet been effectively applied to registration. Instead of relying on a physical model to guide the deformation in a roughly plausible way, the statistics of a sample of images can be used to guide the deformation in a way governed by the measured variation of individuals. Also, we believe that much of the useful information

in medical images is carried by the shape of features in the image rather than by other properties such as area or intensity. Therefore, our second goal here is to incorporate statistical shape information into this type of physical model-based registration and to develop a more accurate and robust algorithm.

In conclusion, our aim in this thesis is to first develop statistical shape analysis methods for segmentation and then use the resulting corresponding boundary points which contain statistical shape information for physical model-based non-rigid registration.

1.2 Overview of the Framework

A complete system of statistical shape analysis for segmentation and non-rigid registration has been developed in this dissertation.

Chapter 2 reviews the previous research in the field of segmentation and registration.

In Chapter 3, the statistical point model derived from the training set by principal component analysis of the covariance matrix is described. Also the independent model, the smoothness model and the combined model are formulated and compared with the statistical point model. A diagram of our statistical shape analysis is shown in Figure 1.2. Then, a Bayesian formulation for 2D boundary finding with the different prior models is described. Finally, a number of experiments are performed on both synthetic and real medical images of the brain and heart to evaluate the approaches, including the validation of the dependence of the method on image quality, initialization and prior information.

Chapter 4 first presents the physical models used in non-rigid registration, including elastic solids and viscous fluids. Then the integration of statistical shape information

Statistical Shape Analysis

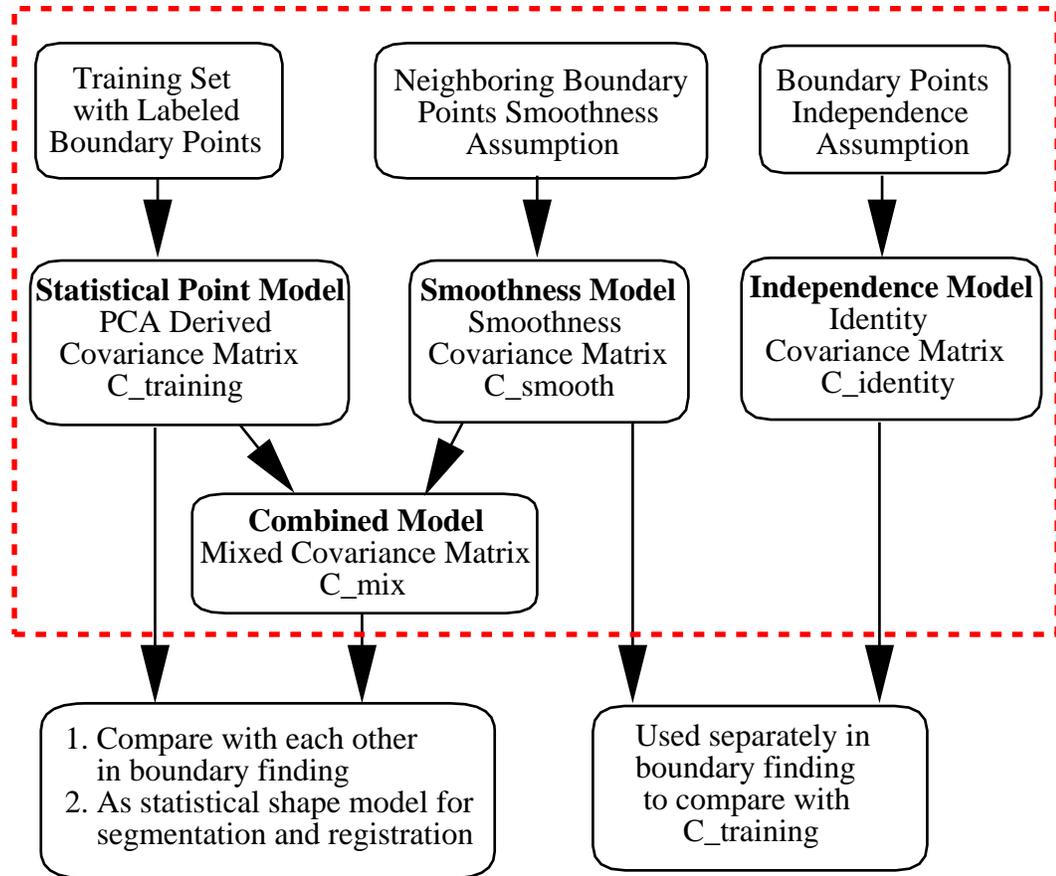


Figure 1.2: Diagram for our statistical shape analysis methods.

into the physical model-based non-rigid registration is formulated. A number of experiments are performed on both synthetic and real medical images of the brain and heart to evaluate the approaches. It is shown that statistical boundary shape information significantly augments and improves physical model-based non-rigid registration and the two methods we present each have advantages under different conditions. Also, the proposed non-rigid registration is shown to be helpful for our boundary finding.

In Chapter 5, we extend the proposed boundary finding algorithm to 3D volumetric segmentation. The new techniques required that are totally different from that in 2D are addressed and described, as well as some initial synthetic results.

A diagram outlining the complete 2D system is shown in Figure 1.3.

1.3 Main Contributions

In this thesis, statistical point models with shape and shape variation generated from sets of examples by principal component analysis of the covariance matrix are used for boundary finding. This idea has been used previously for boundary determination (e.g. [38, 119]). Also, the Bayesian formulation of the objective function is akin to the work of Staib and Duncan [117]. The primary contribution of this dissertation is that it extends these ideas to provide a more robust method for both boundary and correspondence finding. Moreover, in order to show the important role of the statistical prior shape model, we demonstrate the use of two other kinds of generic prior information: an independence model and a smoothness model [126]. In addition, we consider prior shape models trained on small sets which may not allow enough variation to adequately span the space of plausible shapes. For this case, we further propose the use of an integrated prior model-based on a combination of our proposed smoothness covariance matrix and the traditional training set covariance matrix [129].

The key benefits of our boundary finding approaches include the following. During the optimization, we search for the shape and pose parameters together directly in the parameter space, resulting in a better optimum. Continuous penalizing criteria based on a training set derived Gaussian distribution for the prior shape and pose parameters are used during the search. When few training set examples are available, our combined

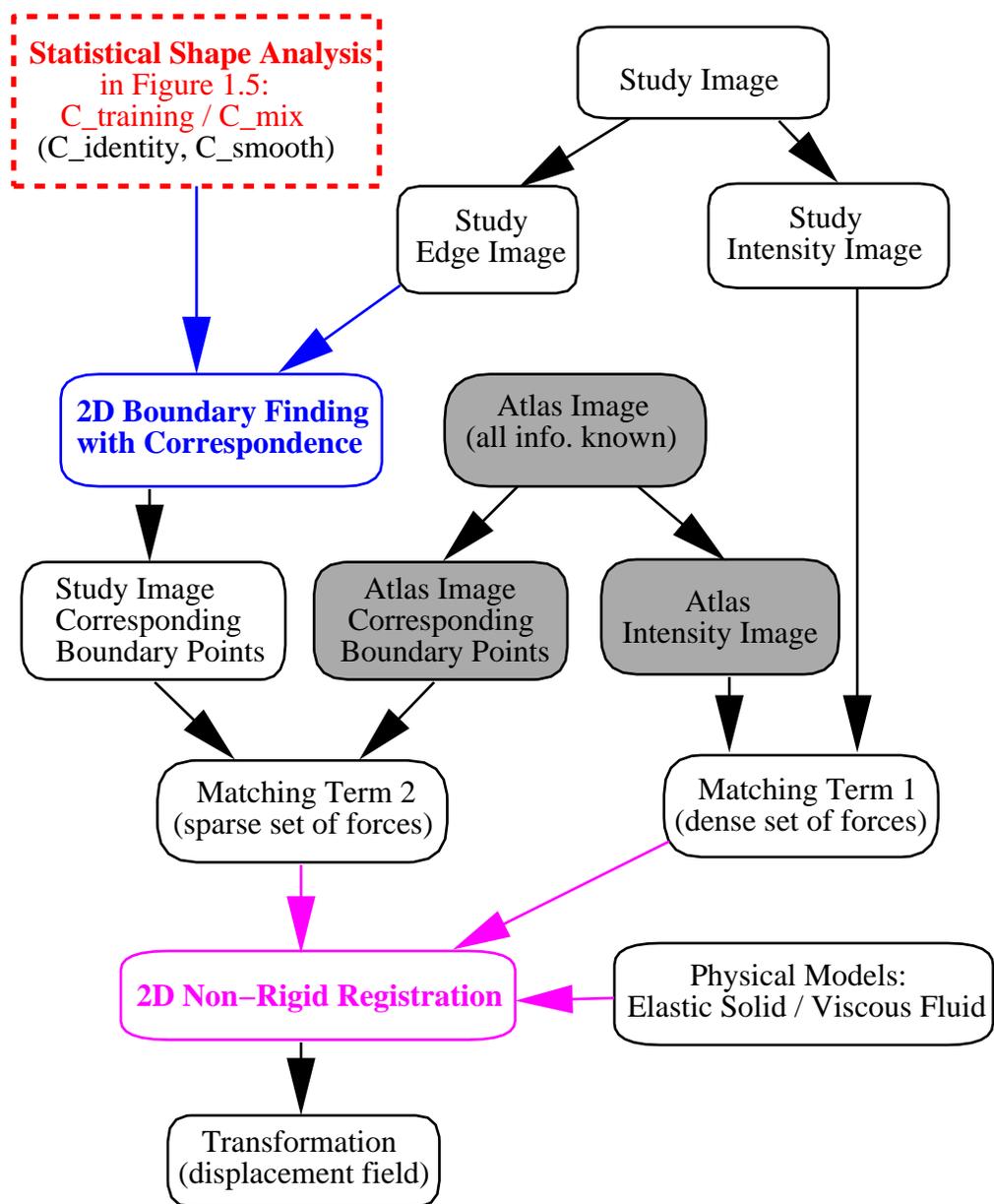


Figure 1.3: Overview of the 2D system framework.

model is more straightforward and simple than that proposed by Cootes *et al.* [39]. In addition, the optimization algorithm is faster and more robust than the image search of Cootes *et al.* [38]. Moreover, using a Bayesian framework allows us to adjust the weighting between the statistical prior knowledge and the image information based on the image quality and the reliability of the training set.

Our main contribution to non-rigid registration is to incorporate statistical shape information into physical model-based registration (linear elastic solid [127] and viscous fluid [128]) and to develop more accurate and robust algorithms. The power of physical and statistical shape models are combined in our approaches using a Bayesian framework [130].

Finally, we have developed a new approach to 3D deformable surface statistical model construction, where the corresponding surface points are identified in a hierarchical way based on geodesics. Meanwhile, the surface is triangulated based on the labeled surface points, which results in an efficient algorithm for our model construction [131].

Chapter 2

Background and Related Work

In this chapter, we describe briefly the available literature on the topics that are relevant to this work. Over the years a number of different approaches has been developed for image segmentation and registration. Excellent reviews of segmentation [52, 66, 99] and registration [47] survey these techniques. Here, we first describe some of those approaches. Next, we will describe briefly the available work on rigid or deformable models for shape matching, which includes both segmentation and registration.

2.1 Related Work in Image Segmentation

2.1.1 Region Based Methods

We classify region-based methods as those that exploit homogeneity of spatially dense information, e.g. pixel-wise grey level values, texture properties etc., to produce the segmented image. These methods primarily depend on the underlying consistency of any relevant feature in the different regions of an image.

Thresholding [109] and region growing [21] are two of the oldest and widely used

tools for image segmentation. While region growing methods perform better than thresholding procedures, they continue to have similar problems, producing artificial and poorly localized boundaries. Thus, to produce reliable boundary estimates, further processing often becomes necessary. This kind of information can, however, be incorporated into boundary methods which have better localization properties but suffer from high noise sensitivity, as in [24].

Region-based image segmentation techniques using spatial interaction models like Markov Random Field (MRF) or Gibbs random fields to model the image [54, 43, 44] have become very popular. They assume that the probability of a point being in a particular state is entirely dependent on the probabilities of the states of its neighbors. MRF models allow the incorporation of statistical information about the local variation of gray levels. Whereas the statistical model we use are based on shape.

Non-linear diffusion methods [102, 56, 1] are used for doing *edge preserved smoothing*. Non-linear diffusion can be considered to be a natural extension of scale space filtering. Linear scale space methods solve the linear diffusion equation and were first developed by Witkin [133] and Koenderink [83] and further developed by others [6, 140, 71]. The essential idea here is to produce a multiscale description of the image. Both isotropic and anisotropic diffusion equations [102, 1] have been developed.

In the next section, we will discuss boundary methods, which have better localization properties than region methods and are more related to the work here.

2.1.2 Boundary Methods

Boundary based methods rely on the pixel-wise difference to guide the process of segmentation. Thus, they try to locate points of abrupt changes in grey tone images. There are a number of approaches that use boundary information some of which are discussed

below.

2.1.2.1 Edge Detection

Edge detection is a picture segmentation technique based on the detection of discontinuity where there is a more or less abrupt change in gray level. There are a variety of edge operators such as Robert's, Sobel, Prewitt, etc. [108]. A good edge detector should be a differential operator, taking either a first or second spatial derivative of the image. Second, it should be capable of being tuned to act at any desired scale, so that large filters can be used to detect blurry shadow edges, and small ones to detect sharply focused fine details. One of the most satisfactory operator fulfilling these conditions is the Laplacian of Gaussian operator [90]. It has the desirable property of being smooth and localized in both spatial and frequency domains. However, since it is a second derivative operator, it is quite sensitive to noise. Also, it is difficult to distinguish between more and less probable edges and the only way of discarding false edges would be to do so interactively by human operator.

The Canny edge detector [22] seeks to optimize by making the best trade-off between detection and localization of edges, the two main yet opposing goals of edge detection, in addition to producing one and only one response for a single edge point. Canny showed that the ideal edge detector can be approximated by first taking the gradient of the image convolved with the Gaussian kernel and then choosing the maxima of the gradient image in the direction of the gradient. However, except under ideal circumstances, it results in broken edges and thus identifying objects would require grouping those edges either interactively or by using some other algorithm [108]. The Canny edge detector is utilized as a preprocessing step in our segmentation, where the resulting edge images instead of the original grey level images are used as the input features for image

segmentation.

Next, we focus on the whole boundary methods which are most related to this dissertation.

2.1.2.2 Whole Boundary Methods

Whole boundary methods [78, 2, 38, 117] rely mainly on gradient features at a subset of the spatial positions of an image (near an object boundary) for segmentation of structures from an image. By considering the boundary as a whole, a global shape is imposed on the problem that simplifies the task. Thus, gaps are prohibited and overall consistency is emphasized. Once again, there are a number of approaches, some of which will be described in this section and Section 2.3 — *Related Work in Shape Matching*. More details can be found in the book by Blake & Yuille [11] which contains an excellent collection of papers on deformable contour methods. Here, we will also briefly discuss some of the most recent work on this topic using the level set method [88].

Snakes: One of the most popular methods of detecting whole boundaries is the snakes approach due to Kass *et al.* [78]. A snake is a continuously deformable curve that can be used as a mechanism to locate features of interest in an image. Its shape is controlled both by internal forces (the implicit model) and the external or image forces.

The internal energy represents the smoothing forces on the curve, and the image forces represent the image-derived forces that contain the curve to take the shape of the features present in the image.

The smoothness model proposed in this work is actually similar to that of the internal smoothness forces in Kass’s snakes. However, it is expected that our image search algorithm with this smoothness constraint would be more efficient than Kass’s because we select the eigenvectors of the smoothness covariance matrix corresponding to

the largest eigenvalues and thus reduce the dimensionality of the search space.

Also, the problem with snakes initial version was that if the initial curve was not close enough to an edge, it had difficulty in being attracted by it. Also, the curves in the original snakes formulation had a tendency to shrink on themselves. To improve the convergence properties, dynamic programming can be used for energy minimization [2]. This also allows the objective function to include hard local constraints. Some robustness to the problem of finding a good initial guess was achieved by the addition of an extra force which made the contour have a more dynamic behavior. In other works, the curve was considered to be a balloon [32] (in 2D) that was being inflated. From an initial oriented curve, an extra pressure force was added that pushed the curve out as if air was being introduced inside the closed contour. However, despite all these improvements, this direct representation still had the problem of having too many parameters (here the position vectors along the curve) to optimize over. And more importantly, the parameters are typically still free to take almost any smooth boundary with no constraints on the overall shape. Model-based snakes [105] allow deformations based on a template model but take its shape information into account only in a very limited and general way.

Shape-constrained deformable models: As an alternative to the snakes method, orthogonal representations for curves were proposed by Staib and Duncan [116, 117] where elliptic Fourier descriptors are used as model parameters to represent open and closed boundaries. An orthonormal set is desirable because it makes the parameters uncorrelated. This reduction in redundancy becomes particularly helpful during the optimization process. The Fourier coefficients are used to bias the boundary toward a range of shapes about a mean using a Gaussian distribution on the parameters as a prior probability. A Bayesian approach is then used to obtain the maximum *a posteriori*

estimate of the boundary. Chakraborty *et al.* [25] extended this approach to incorporate region homogeneity. Fourier descriptors are somewhat limited because they are not suitable for describing some shapes, such as those with morphological convolutions or corners. In addition, these methods are designed for boundary finding, without regard to correspondence, which is one of our goals here.

Cootes *et al.* [38] combined deformable shape descriptors with statistical modal analysis built from a training set of annotated images. Object shapes are represented by a subset of boundary points, and a correspondence is established between these points from the different images of the training set. The deformations are modeled using linear combinations of the eigenvectors of the variations from the mean shape, thus defining the characteristic pattern of a shape class and allowing deformation reflecting the variations in the training set. In Cootes' image search algorithm, the model is adjusted by searching a region in the image space around each model point for an improved displacement. These local deformations are transformed into adjustments to the pose parameters, and followed by adjustments to the shape parameters of the point model in the parameter space. By projecting the shape onto the shape parameters and enforcing limits, global shape constraints are applied ensuring that the current shape remains similar to the training set. The estimation of the displacement, however, is determined by a search only in the normal direction toward the strongest image edge [38]. In some situations, errors in the image search at certain points require adjustments of the pose and shape parameters in the parameter space which results in a misadjusted boundary. Shape parameters are accepted unless they are beyond three standard deviations. A more continuous penalty, however, is likely to be more effective, which is the case in our segmentation approaches. In addition, the influence of the training set on the results is always fixed, and is not

adjusted according to image quality.

An improvement of the active shape procedure introduced by Cootes *et al.* to find new examples of previous learned shapes using the point distribution models is presented by Duta [45]. Three additional features characteristic to MR brain images were included in the model: gray-level appearance, border strength, and average position. The search procedure is based on a model fitting strategy, which consists of the following steps: 1) model fitting using linear transforms, 2) model fitting using piecewise linear transforms, 3) outlier removal, 4) final point adjustment, 5) final outlier removal [45]. Although for some images, this strategy may improve the results, the excessive procedures and associated parameters make the algorithm too complicated to be suitable in general.

A similar modal analysis scheme is proposed by Pentland and Sclaroff [101] which gives a set of linear deformations of the shape equivalent to the modes of vibration of the original shape. However, the modes of vibration are based on an arbitrary elastic model that is not likely to be representative of the real variations which occur in a class of shapes. This approach is similar to the proposed smoothness model in this work. Cootes and Taylor [39] incorporated this idea by combining the shape variability determined statistically from a training set with that generated artificially by using a physically based model [101] of the objects of interest. These processes require a computationally intensive finite element analysis to solve for the modes of vibration. While the combined model proposed in this work achieves the similar goal in a much simpler way.

Székely *et al.* [119] describes a model-based segmentation technique combining desirable properties of physical models (snakes) [78], shape representation by Fourier parameterization [117], and modeling of natural shape variability [38]. They performed a principal component analysis of the covariance matrix of the normalized Fourier coeffi-

cients. Elastic fit of the mean model in the subspace of eigenmodes restricts possible deformations and finds an optimal match between the model surface and boundary candidates. This approach combines the descriptive power of Fourier coefficients with the efficiency of restriction based on natural variability. However the use of Fourier coefficients makes it suitable primarily to relatively smooth shapes.

Level set method: While such deformable boundary finding approaches have been used successfully in computer vision applications, one important limitation of these methods is that they are unable to handle topological changes of the deforming contour. This limitation is removed by the level set technique, due to Osher & Sethian [98, 112, 88, 142].

The level set method is an approach that can be applied for recovering boundaries of objects in two and three dimensions from various types of image data. The model consists of a moving front and may be modeled into any desired shape by externally applied halting criteria synthesized from the image data. The moving front $\gamma(t)$ is represented as the zero level set of a higher dimensional function Ψ given by

$$\Psi(\mathbf{x}, t = 0) = d \tag{2.1}$$

where d is the distance from \mathbf{x} to $\gamma(t = 0)$.

Given a moving closed boundary $\gamma(t)$, an Eulerian formulation for the motion of the boundary is produced, propagating along its normal direction with speed F , where F can be a function of various arguments, including the curvature, normal directions, etc. The evolution equation for Ψ is given by

$$\Psi_t + F |\nabla\Psi| = 0 \tag{2.2}$$

The speed function F from the image data is applied to the propagating front and is split into two components F_A and F_G . The term F_A , referred to as the advection term, is independent of the moving front's geometry. The front uniformly expands or contracts with speed F_A depending on its sign. The second term F_G depends on the geometry of the front, such as its local curvature. This term can be used, for example, to smooth out high curvature regions of the front.

McInerney and Terzopoulos [92] presented a variation of the original snakes approach which can handle changes in the topology of the contour by embedding the snakes in a simplicial domain decomposition. This formulation handles the splitting or merging of contours in a topologically consistent way. Recently, Vemuri and Guo proposed a new geometric shape modeling scheme using snake pedals [125] which allows for the representation of both global and local shape characteristics of an object, as well as topological changes.

However, topology changes are not necessary in this work and it is still not clear if it is possible to incorporate global shape information into the level set formulation. Our segmentation approaches are more related to the *shape-constrained deformable models* instead of the *level set method*.

2.2 Related Work in Image Registration

Any registration method will produce a set of equations that transforms the coordinates of each point in one image into the coordinates of the corresponding point in the other image. Registration methods may be classified according to various criteria, such as dimensionality (1D/2D/3D/...), origin of image properties (intrinsic/extrinsic), domain of the transformation (global/local), elasticity of the transformation (rigid/affine/projective/curved),

etc. Here, we are more interested in the last criteria. A general discussion of the classes of rigid and non-rigid transformations of images is given by Wolberg [135].

The registration in this work is non-rigid. However, some ideas of rigid registration are relevant. Thus, a brief review of rigid registration is also given as follows.

2.2.1 Rigid Registration

A transformation is called rigid if the distance between any two points in the first image is preserved when these two points are mapped onto the second image. Rigid image registration is the task of bringing two different images into alignment so that pixel positions correspond to equivalent points in the space of the object imaged. Rigid registration can also be used as the initial alignment for non-rigid registration.

Rigid transformation can be decomposed into translation and rotation. In 2D, the point (x, y) is transformed into the point (x', y') using the formula:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (2.3)$$

where α denotes the rotation angle, and $\begin{pmatrix} t_x \\ t_y \end{pmatrix}$ the translation vector.

Many approaches to rigid registration have been developed that rely exclusively on image information [69, 74, 100, 118, 46, 137], such as features [100, 103, 74, 107, 46], or grey levels [137, 138, 86, 132, 12]. Grey level properties, unlike features, avoid losing any valuable information by using all of it. The most basic technique of this kind is correlation, which will try to match the grey levels, pixel by pixel. This method has problems when the grey levels do not really correspond, such as when the images are from

different modalities, have inhomogeneities, use different pulse sequences, etc. Registration by mutual information [86, 132, 12] allows the comparison of images at a raw pixel level, but is independent of the problems of scaling, level and other variations of gray level. This is a powerful technique that avoids many of the problems of other gray-level techniques. An essential part of feature methods is the establishment of a correspondence between features in the two images. From this correspondence, an error of total distance can be calculated and then minimized over the space of possible transformations. A well-known technique to solve a 3D rigid point-to-point match, is the use of Singular Value Decomposition (SVD) to find a least-squares fit [4, 59]. Evans *et al.* [48] apply this method to PET and MR brain images, while it is used by Hill *et al.* [68] to register CT and MR skull base images. This method is also used in our 3D training set alignment for segmentation (Section 5.3).

2.2.2 Non-Rigid Registration

Comparing function or morphology between individuals requires non-rigid registration, because the detailed structure of the brain differs, sometimes greatly, between individuals. In addition, there is functional variability between individuals. The goal of non-rigid registration is to remove the structural variation between individuals by matching an atlas image to each individual or study image, in order to have a common coordinate system for comparison. Shape differences between the atlas and study's anatomy are contained in the non-rigid transformation. Functional differences will remain and can then be studied without the confounding structural variations. There have been many approaches to non-rigid registration and most of the work has occurred in the last two years [19, 87, 7, 8, 96, 120, 40, 97, 41, 31, 28, 53, 33, 50, 123, 5, 34, 114]. Usually, the transformation is constrained in some way because of the ill-posedness (i.e. in this

case, the existence of many possible solutions) of the problem. Mathematically, non-rigid transformations can be classified as affine, projective, curved and by other models, or they may be general, usually at least preserving continuity.

2.2.2.1 Affine, Projective and Curved Transformations

A transformation is called affine when any straight line in the first image is mapped onto a straight line in the second image, while parallelism is preserved. An affine transformation can be decomposed into a linear (matrix) transformation and a translation. In 2D, the coordinate transformation can be formulated as follows:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \quad (2.4)$$

where $\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$ denotes any real-valued matrix.

An affine variation of the above SVD point match that determines scaling from the principal axes is described by Marrett *et al.* [91], for matching of MRI patient data with an MRI brain atlas.

A projective (or perspective) transformation maps any straight line in the first image onto a straight line in the second image. It is almost exclusively used to register projective (X-ray) images to 3D tomographic images, employing a 3D-to-2D variant of

the projective formula as follows:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \frac{u}{w} \\ \frac{v}{w} \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (2.5)$$

where w represents the extra homogeneous coordinate.

Curved transformations may map a straight line onto a curve. In 2D, this is formulated as follows:

$$(x', y') = \mathcal{F}(x, y) \quad (2.6)$$

where \mathcal{F} denotes any function mapping the coordinates in the first image onto coordinates in the second image. An example of this approach used in the determination of mapping functions for the registration of two images is given by Goshtasby [62]. Given the coordinates of corresponding control points in two images of the same scene, first the images are divided into triangular regions by triangulating the control points. Then, a linear mapping function is obtained by registering each pair of corresponding triangular regions in the images. The overall mapping function is then obtained by piecing together the linear functions. We have generalized this approach to 3D, and it could be used as the initialization of our non-rigid registration. Goshtasby also extends this technique to include piecewise nonlinear functions as mapping functions [63]. This nonlinear mapping function, in addition to providing a continuous mapping, provides a smooth transition from one local function to another. The obtained mapping function which is a piecewise combination of the local mapping function is therefore continuous and smooth all over.

2.2.2.2 Bookstein's Thin-plate Splines

The model of deformation, based on the physics of thin-plate splines, is used by Bookstein [14, 13]. This method requires the identification of a large number of homologous pairs of anatomic point landmarks. Unfortunately, after the first few obvious points in the brain, this becomes difficult because there are not enough true landmark points that can be accurately marked. Landmarks are typically needed throughout the image because local changes to the spline have global effects. These landmarks must be very reliable, because the thin-plate spline forces these points exactly into correspondence. Also, while the transformation has interesting statistical properties, especially in terms of decomposing the warp, it can result in impossible warpings. For example, the space can fold over itself, resulting in a non-homeomorphic warping, when the geometry is particularly warped. This technique is intended primarily as a method for the statistical comparison of shape through the location of homologous landmarks, and not as a general technique for image warping [13].

2.2.2.3 Physical Models

Another classification of non-rigidity comes from who use analogous physical models, for example, linear elastic [96] or viscous [28] models, to control the deformation. Here, the image or underlying structure is considered to be composed of a physical material of known properties of deformation, and is deformed in accordance with those properties. The validity of these techniques is predicated on the applicability of such a model to this kind of deformation. Of course, there is no true physical model, because each individual anatomy does not literally result from the deformation of another anatomy. However, such models are a powerful technique to constrain the enormous solution space. The

non-rigid registration approaches developed in this dissertation are based on the physical models.

Gee *et al.* [53] use a linear elastic strain model and a Markov random field to model a displacement field for non-rigid registration. The elastic model, and other related models, penalize deformation in proportion to the displacement distance. This captures the idea that the deformation should be as small as possible and thus that the necessary deformation is not large. This is too strict a model because of the large variation between brains and does not allow for the complete deformation necessary.

A more relaxed model is the viscous fluid model used by Christensen *et al.* [28]. Christensen transforms a generic digital neuroanatomical atlas to closely correspond with a specific individual's data set, thus creating an individualized atlas. The transformations are constrained to be consistent with the physical properties of deformable elastic solids in the first method and those of viscous fluids in the second [29]. The fluid model has the advantage of tracking long-distance, nonlinear deformations of small subregions. The viscous model is used in conjunction with the topologic constraint that the transformation must be homeomorphic. This technique uses landmark-based matching to initialize, followed by gray level intensity-based matching with a viscous fluid-flow model. Viscous fluid models are designed to be less constraining than elastic models. Deformation occurs in accordance with the physical properties of a flowing liquid, however, this may not be a strong enough constraint. The homeomorphic condition is an important one. This ensures that the deformation is continuous and one-to-one. Thus, structures do not split, merge, fold over, etc. It is a very reasonable assumption that different individual brains are topologically equivalent, except for severe disease and abnormal variation, such as the absence of a corpus callosum. Joshi *et al.* [77] presents a coarse-to-fine approach for

the transformation of an atlas from the ideal to the individual that combines landmark deformation [14] and volume-based transformations [96, 28, 65, 49, 3]. Rabbitt *et al.* [104] use finite deformation continuum mechanics to develop a new finite element tool that combines image-based data with mechanics. They use a hyperelastic material model valid for large-magnitude nonlinear deformations.

2.2.2.4 Physical Model-Based Non-rigid Registration vs. Optical Flow

In Christensen, Miller *et al.*'s elastic registration [96, 29], the transformations are constrained to be consistent with the physical properties of deformable elastic solids to maintain smoothness and continuity. The matching term is the intensity difference between the two images. Therefore, the resulting displacement field is essentially optical flow, which is the apparent motion of the brightness pattern and is solved from the optical flow constraint equation (intensity match term) and the smoothness constraint. Also, both of them assume small deformation displacement. However, Christensen's fluid model [28, 29] allows long-distance nonlinear deformations, which is quite different from optical flow field, although the intensity matching term is similar.

2.3 Related Work in Shape Matching (for Segmentation or Registration)

There is also a rich collection of publications on shape matching using either rigid or deformable templates.

2.3.1 Rigid Matching

Hough Transform (HT) [70] is a rigid template matching scheme, which transforms points in the spatial feature space into a parameter space, and maxima in the parameter space correspond to possible shape instances. It was originally formulated for lines and other simply parameterizable shapes, but was extended later to generalized shapes [9, 10]. While relatively insensitive to noise and gaps, the computational requirement goes up geometrically with the number of parameters. The complete surveys on different variations of the HT technique can be found in [72, 84].

Different from the rigid matching, the deformable models are capable of accommodating the variability of the objects and then fit the data by deforming itself. There are two classes of the deformable models — **unconstrained** and **parametric**.

2.3.2 Unconstrained Deformable Models

For unconstrained deformable models, the template is only constrained by local continuity and general smoothness constraints without any global structure of the template. The template can be deformed to match image features, such as edges, using the energy function produced by those features. The unconstrained deformable models have the advantage of allowing great variability and can represent an arbitrary shape as long as the continuity and smoothness constraints are satisfied. A typical example of segmentation with unconstrained deformable model is the active contour model proposed by Kass *et al.* [78] which has been reviewed in Section 2.1.2.2. The smoothness model and the independence model proposed in the segmentation part of this dissertation (Chapter 3) also belong to the application of this kind of unconstrained deformable model. The elastic model [20, 96, 97] or fluid model [31] proposed in the atlas registration problem is

another example of a unconstrained deformable model. The atlas deforms and matches itself onto corresponding regions in the study image in response to local forces derived from images based on the established elastic or fluid model. The physical model-based non-rigid registration approaches presented in this work falls into this category.

2.3.3 Parametric Deformable Models

When prior knowledge about the geometric shape, which can be embedded in a small number of parameters, is available, a parametric deformable model can be used.

In addition to the shape-constrained deformable models for whole boundary segmentation reviewed in Section 2.1.2.2, some researchers have incorporated information of object shape into deformable models by using deformable shape templates. These models usually use hand-crafted global shape parameters to embody *a priori* knowledge of expected shape and shape variation of the structures. An excellent example is the work of Yuille *et al.* [141] who construct deformable templates for detecting and describing features of faces, such as the eyes and mouth. The parameters which control the shape of a template are the center and the radius of a circle, etc. The eyes and mouths in real images can be accurately located if the initialization is close to the desired objects.

Deformable models based on superquadrics are another example of parametric deformable models. Metaxas and Terzopoulos [95] employ a dynamic deformable superquadric model [122] to reconstruct and track human limbs from 3D biokinetic data. Their models can deform both locally and globally by incorporating the global shape parameters of a superellipsoid with the local degrees of freedom of membrane spline deformation in a Lagrangian dynamics formulation. The global parameters capture the overall shape of the data, while the local deformation parameters reconstruct the details. Vemuri *et al.* [124] construct a deformable superquadric model in an orthonormal

wavelet basis. This multi-resolution basis provides the model with the ability to continuously transform from local to global shape deformations thereby allowing a continuum of shape models to be created and to be represented with relatively few parameters. They apply the model to segment and reconstruct anatomical structures in the human brain from MRI data.

A parametric deformable template can also be represented by using a prototype template under a parameter mapping. Grenander [64] described a systematic framework to represent shape classes of a characteristic structure. A model template is based on prior knowledge of the objects obtained from the training samples and describes the overall shape. A parametric statistical mapping controls the random variations of the shape [3, 26, 65, 96] and reflects the allowable deformations according to the application domain. Such a framework can be very versatile depending on the choice of template and deformation. A similar method is used in active shape models proposed by Cootes *et al.* [37], which was reviewed in detail in Section 2.1.2.2. As we saw, active shape models are able to learn the characteristic pattern of a shape class and can deform in a way that reflects the variation in the training set. Jain *et al.* [73] represent the prototype template in the form of a bitmap which consists of the representative contour/edges of an object shape. Their deformation model is then deformed to fit salient edges in the input image by applying a probabilistic transformation on the prototype contour which maintains smoothness and connectedness. Note that our statistical shape analysis for boundary finding with correspondence described in Chapter 3 is also related to this category with the mean shape being the prototype template.

Chapter 3

Statistical Shape Analysis for Boundary Finding with Correspondence

3.1 Introduction

As introduced in Chapter 1, we are interested in both determining an object boundary and simultaneously determining spatial correspondence between similar structures over different subjects. The complete statistical shape analysis methodology (Figure 1.2) is provided in this chapter which includes the traditional statistical point model, our original independence model and smoothness model, and the newly formulated combined model which is intended for situations where there are small training sets without enough variation to adequately span the space of plausible shapes. A Bayesian formulation, based on each of the proposed prior models and the edge information of the input image, is employed to find the object boundary with its subset points in correspondence with

boundaries in the training set or the mean boundary. A number of experiments were performed on both synthetic and real images of the brain and heart to evaluate and compare the approaches with different prior models. The key benefits of our approaches has been summarized in Section 1.3.

3.2 Statistical Shape Analysis

3.2.1 Statistical Point Model — $C_{training}$

Suppose we want to derive a model to represent the shapes whose boundaries are shown in Figure 3.1(b). We can represent each example shape as a set of hand-labeled landmark points. The model is then based on the mean positions of the points on the aligned shapes and the main variation of the points from the mean.

3.2.1.1 Labeling and Aligning the Training Set

The critical points on the boundary are usually easily identified features, such as high curvature points, sharp corners, etc. Equally spaced points are interpolated between the critical points along the boundary (Figure 3.1(a)).

The training set points are first aligned by scaling, rotation and translation. The aim is to minimize a weighted sum of squared distances between equivalent points on different shapes. Each shape is aligned with the first shape. The mean shape and pose are calculated from the aligned shapes. Every shape is realigned with the current mean and the standard deviation of the pose (scale, rotation, translation) is calculated.

While the resulting pose depends on the chosen first shape, as long as this shape is not at an extreme pose, the difference is insignificant. We search for both the shape parameters and the pose parameters together and experiments show that the results are

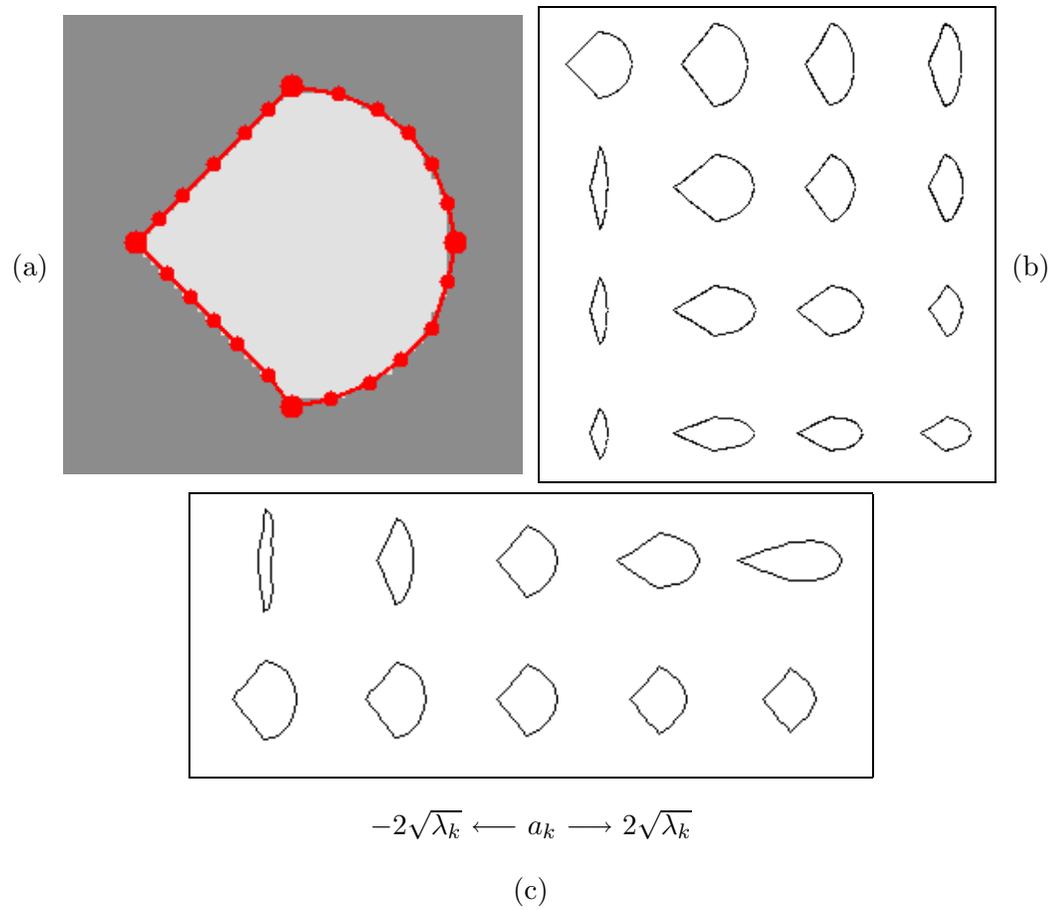


Figure 3.1: Synthetic shape model. (a): synthetic image (64×64) with its 24 point model of the boundary (4 critical points, (large dots), shown with equally spaced points (small dots)); (b): 16 examples of synthetic shapes from a training set; (c): effects of varying each of the first two shape parameters of the synthetic model.

independent of the first shape.

3.2.1.2 Capturing the Statistics

The following 2D formulation is similar to that of Cootes *et al.* [38]. Given a set of m aligned examples and each example of a set of N aligned labeled points, each example can be represented by a single point in a $2N$ dimensional space,

$$\mathbf{L}_i = (x_i(1), y_i(1), x_i(2), y_i(2), \dots, x_i(N), y_i(N))^T \quad (i = 1, 2, \dots, m). \quad (3.1)$$

Thus, a set of m example shapes gives a cloud of m points in this $2N$ dimensional space. We assume that these points lie within some region of the space, and that the points give an indication of the shape and size of this region. Every $2N$ - D point within this domain gives a set of landmarks whose shape is broadly similar to that of those in the original training set. Thus, by moving in this region, we can generate new shapes in this high dimensional cloud, and hence capture the relationships between the positions of the individual landmark points.

For the given set of m aligned shapes, the mean shape $\bar{\mathbf{L}}$ is calculated by

$$\bar{\mathbf{L}} = \frac{1}{m} \sum_{i=1}^m \mathbf{L}_i \quad (3.2)$$

The principal axes of a $2N$ - D cloud fitted to the data can be calculated by applying a principal component analysis (PCA) to the data [75, 76]. Each axis gives a “mode of variation”: a way in which the landmark points tend to move together as the shape varies. For the traditional statistical problem, in order to get the unbiased estimate, the variance

s_X^2 of variable X and the covariance s_{XY} of variables X and Y are defined as [57]:

$$s_X^2 = \frac{\sum_i (X_i - \bar{X})^2}{\text{degrees of freedom}}; \quad s_{XY} = \frac{\sum_i (X_i - \bar{X})(Y_i - \bar{Y})}{\text{degrees of freedom}}$$

where *degrees of freedom* is one less than the sample size. Hence, we calculate the $2N \times 2N$ covariance matrix $C_{training}$ of random vector \mathbf{L} (with entries of variance and covariance) about the mean using

$$C_{training} = \frac{1}{m-1} \sum_{i=1}^m (\mathbf{L}_i - \bar{\mathbf{L}})(\mathbf{L}_i - \bar{\mathbf{L}})^T \quad (3.3)$$

When m is large enough, whether “ $m-1$ ” or “ m ” is used in the denominator of Equation (3.3) makes negligible difference in the $C_{training}$ calculation.

The principal axes of the cloud, giving the modes of variation of the points of the shape, are described by \mathbf{p}_k ($k = 1, 2, \dots, 2N$), the unit eigenvectors of $C_{training}$ such that

$$C_{training} \mathbf{p}_k = \lambda_k \mathbf{p}_k \quad (3.4)$$

(where λ_k is the k th eigenvalue of $C_{training}$, $\lambda_{k-1} \geq \lambda_k$).

It can be shown that the eigenvectors of the $2N \times 2N$ covariance matrix, $C_{training}$, corresponding to the largest eigenvalues describe the most significant modes of variation in the variables used to derive the covariance matrix. The variance explained by each eigenvector is equal to the corresponding eigenvalue λ_k [75, 76]. Typically, most of the variation can be explained by a small number of modes, t ($< 2N$). The space of variation is then restricted to a subspace of eigenvectors without a significant loss of descriptive power. We will select the appropriate number of vectors, t , using the criterion that the

sum of the t eigenvalues (ordered from largest to smallest) is greater than some proportion, d (say, $0.8 \sim 0.9$), of the total variance. That is, choose the minimum t such that:

$$\sum_{k=1}^t \lambda_k > d \sum_{n=1}^{2N} \lambda_n \quad (3.5)$$

Any point in this region can be reached by taking the mean and adding a linear combination of the eigenvectors. Any shape in the training set can be approximated using the mean shape and a weighted sum of deviations obtained from the first t modes:

$$\mathbf{L} = \bar{\mathbf{L}} + Q\mathbf{a} \quad (3.6)$$

where $Q = (\mathbf{q}_1 \mid \mathbf{q}_2 \mid \cdots \mid \mathbf{q}_t)$ is the matrix of the first t eigenvectors, and $\mathbf{a} = (a_1 a_2 \cdots a_t)^T$ is a vector of weights, which is also the set of t shape parameters to be optimized later.

This equation allows us to generate new examples of shapes by varying the parameter \mathbf{a} so that the new shapes will be similar to those in the training set.

We have used the techniques described above to generate statistical point models for both synthetic objects (e.g. Figure 3.1) and biological objects (e.g. Figure 3.2 for the heart).

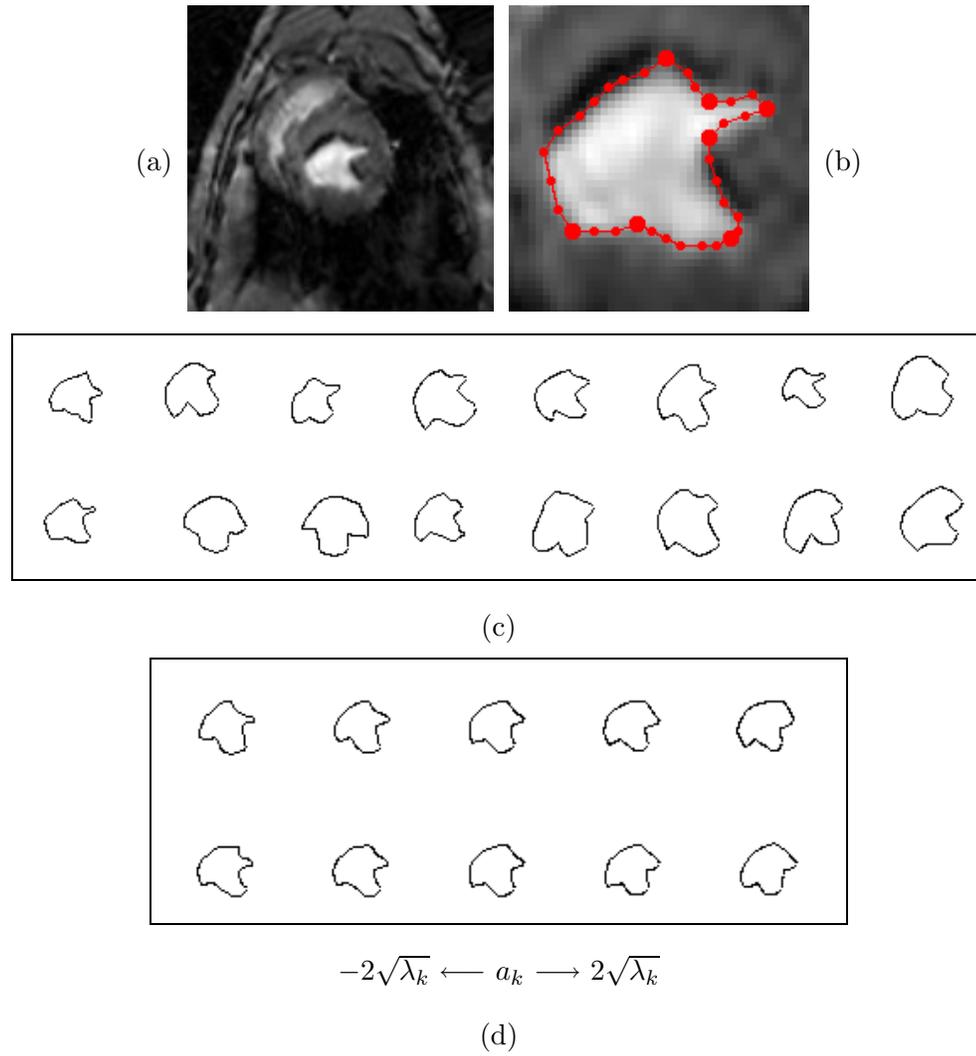


Figure 3.2: Heart shape model. (a): MR heart image (150×150); (b): 34 point model of the endocardium boundary (7 critical points, large dots) (cropped); (c): 16 examples of heart shapes from a training set; (d): effects of varying each of the first two shape parameters of the heart model.

3.2.2 Identity Covariance Matrix — $C_{identity}$

Consider the use of a $2N \times 2N$ identity covariance matrix $C_{identity}$ (Equation (3.7)) instead of the covariance derived from the training set.

$$C_{identity} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \quad (3.7)$$

This means that all points (x and y coordinates) are uncorrelated. The eigenvectors of $C_{identity}$, \mathbf{q}_k (size $2N$), for $k = 1, 2, \dots, 2N$, are of the form:

$$\begin{aligned} \mathbf{q}_1 &= (1, 0, 0, 0, \dots, 0, 0, 0)^T \\ \mathbf{q}_2 &= (0, 1, 0, 0, \dots, 0, 0, 0)^T \\ &\vdots \\ \mathbf{q}_{2N} &= (0, 0, 0, 0, \dots, 0, 0, 1)^T \end{aligned} \quad (3.8)$$

with the corresponding eigenvalues $\lambda_k = 1$. If k is even, the k th eigenvector moves point $k/2$ in the y direction; if k is odd, the k th eigenvector moves point $(k + 1)/2$ in the x direction. Combinations of vectors, one for each mode, can move the modeled landmark points anywhere in the image. Any shape can also be approximated using the mean shape and a weighted sum of deviations obtained from the $2N$ modes. Since the eigenvalues here are very small, a large range is used ($-20\sqrt{\lambda_k} \leq a_k \leq 20\sqrt{\lambda_k}$) to demonstrate the generation of new shapes, as shown in Figure 3.3.

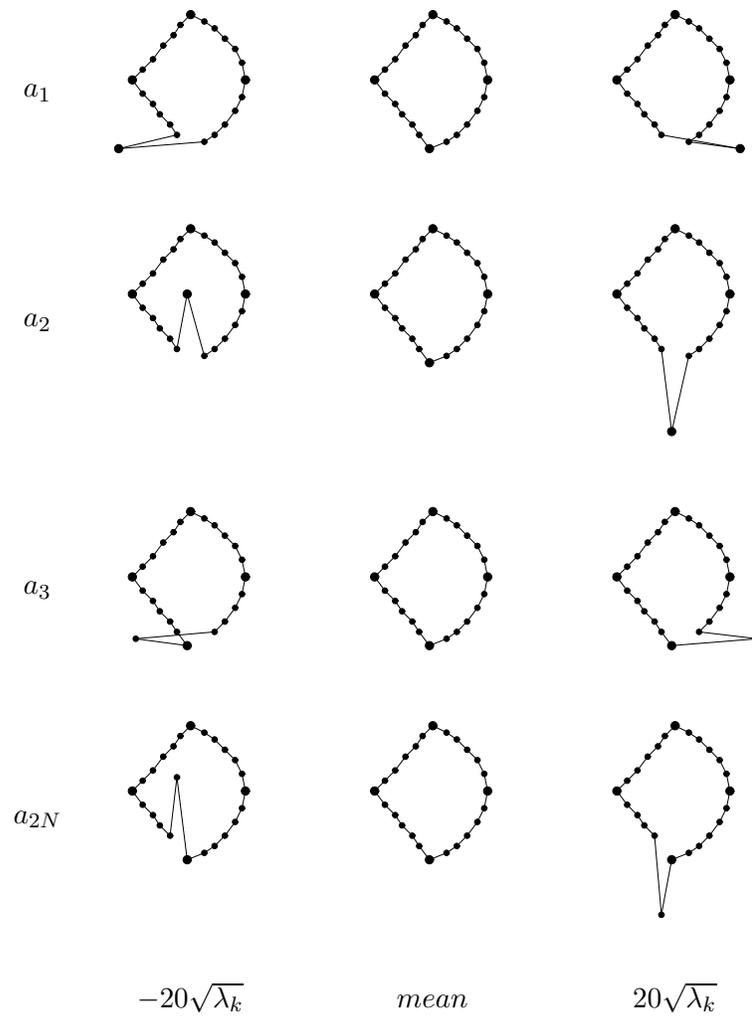


Figure 3.3: Effects of varying each of the 1st, the 2nd, the 3rd and the $2N$ th shape parameters of the synthetic model with the identity covariance matrix. (The mean shape here is the mean of the shapes in Figure 3.1(b).)

3.2.3 Smoothness Covariance Matrix — C_{smooth}

The identity covariance matrix is completely unconstrained; therefore, consider the incorporation of a type of smoothness constraint into the covariance matrix where neighboring points are correlated. That is:

$$C_{smooth} = \begin{bmatrix} 1 & 0 & 0.5 & 0 & 0 & \cdots & 0.5 & 0 \\ 0 & 1 & 0 & 0.5 & 0 & \cdots & 0 & 0.5 \\ 0.5 & 0 & 1 & 0 & 0.5 & \cdots & 0 & 0 \\ 0 & 0.5 & 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0.5 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0.5 & 0 & 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \quad (3.9)$$

where C_{smooth} is a $2N \times 2N$ matrix¹. Here, neighboring points are more likely to move together than if they were uncorrelated. The first two eigenvectors of C_{smooth} (with equal eigenvalues) allow for a rigid translation. The other eigenvectors correspond to increasing frequency variation. Combinations of these eigenvectors, one for each mode, can move the modeled landmark points anywhere in the image with the neighboring points moving together. The shapes generated (Figure 3.4) are smoother than those using the identity covariance matrix. Shapes are not restricted by a training set and thus the model is not

¹The C_{smooth} here is only valid for general single closed boundary. Specific C_{smooth} matrices need to be generated for multiple boundaries or open boundaries.

specific, but allows smooth variability. Also, critical points are unlikely to correspond when using this model. Eigenvectors corresponding to higher frequencies have lower eigenvalues and will have less influence.

Note that the degree or scale of smoothing can be controlled by changing the coefficients along the diagonals. This is equivalent to changing the covariance values for the boundary neighbor points. The identity covariance matrix, $C_{identity}$ in Equation (3.7), can be considered as an extreme case of C_{smooth} with scale zero (the smallest scale by considering only zero order boundary neighbor smoothness). The C_{smooth} in Equation (3.9) has a medium scale by including first order boundary neighbors with covariance 0.5. A larger scale C_{smooth} could be constructed by using higher order boundary neighbor smoothness with covariance decreasing gradually between 1.0 and 0.0 as the order of boundary neighbor increases from zero. Another way to construct C_{smooth} is to set the covariance values for the boundary neighbor points according to a Gaussian, with larger standard deviation corresponding to larger smoothing scale and smaller standard deviation corresponding to smaller scale. In either case, the larger the smoothing scale, the larger the eigenvalues corresponding to the low frequency eigenvectors are and the lower the eigenvalues corresponding to the high frequency eigenvectors are (Figure 3.5). However, because most of the variation can be explained by a small (or medium) number of significant modes, the cut-off of the high frequency components makes the scale of C_{smooth} a relatively unimportant factor during optimization. For the example shown in Figure 3.5, if the number of modes used is half of the number of model points, (i.e. 12), then the only difference is $\sqrt{\lambda_k}$, the standard deviation of the Gaussian prior density (σ_j in Equation (3.28)). In practice, results are robust to a wide range of scales. Hence, in the later experiments, we always use C_{smooth} as given by Equation (3.9).

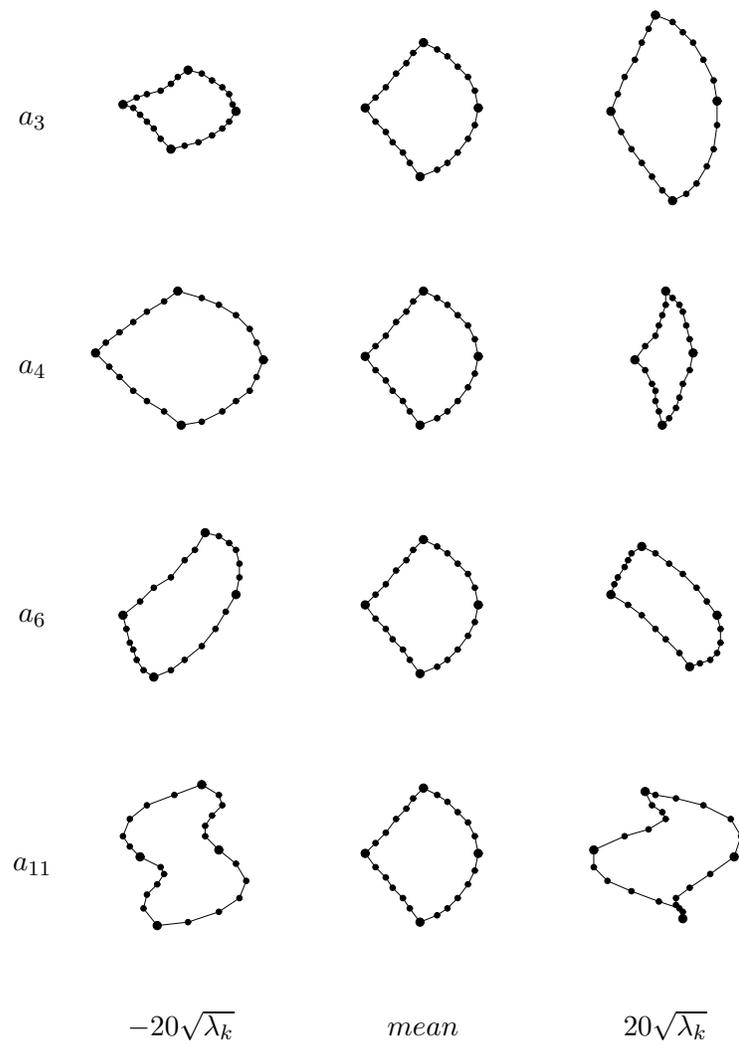


Figure 3.4: Effects of varying each of the 3rd, the 4th, the 6th and 11th shape parameters of the synthetic model with the smoothness covariance matrix. (The mean shape here is the mean of the shapes in Figure 3.1(b).)

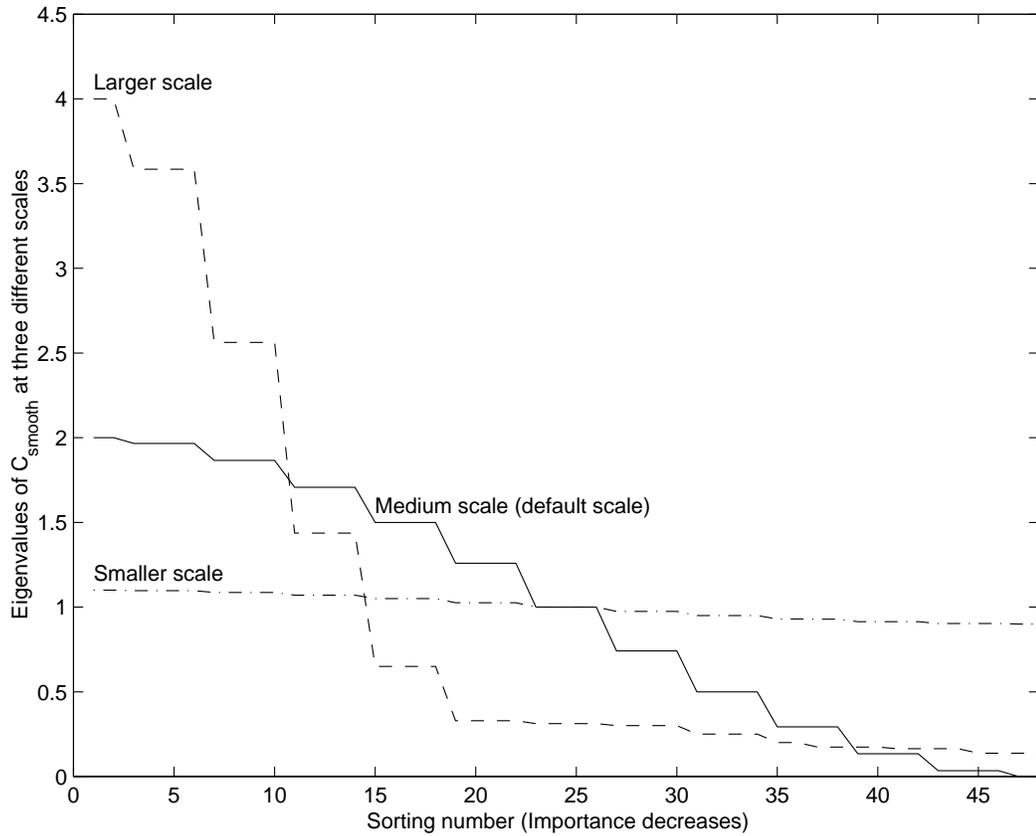


Figure 3.5: Comparison of eigenvalues of C_{smooth} at three different scales for a 24 point model. Note: Medium scale C_{smooth} (Equation (3.9)) with boundary points variance 1.0, 1st order boundary neighbor covariance 0.5 and all the other higher order boundary neighbor covariance 0.0; diagonal entries for larger scale are respectively 1.0, 0.7, 0.45, 0.25, 0.10, 0.0, ... (approximately Gaussian with standard deviation 1.2); diagonal entries for smaller scale are respectively 1.0, 0.05, 0.0, ... (approximately Gaussian with standard deviation 0.4).

If we examine C_{smooth} , we see that it is not only symmetric, a characteristic of covariance matrices, it is also circulant. The circulant structure explains the frequency interpretation, because the eigenvectors will be sinusoidal [108]. Since the C_{smooth} we constructed is a symmetric matrix, we always have real eigenvalues and eigenvectors. The real eigenvectors we get are linear combinations of the complex vectors, corresponding to multiple eigenvalues [103]. This is why we do not have complex eigenvectors as predicted by the matrix’s circulant structure.

The motivation for the use of this smoothness covariance matrix is actually similar to that of the internal smoothness forces in Kass’s snakes [78]. However, it is expected that our image search algorithm with this smoothness constraint would be more efficient than Kass’s because we select the eigenvectors of C_{smooth} corresponding to the largest eigenvalues and thus reduce the dimensionality of the search space. The use of this smoothness covariance matrix is also similar to Staib’s Fourier descriptors [117] and Pentland’s Modal Analysis scheme [101] since all of them describe shapes based on frequency components.

3.2.4 Combined Model — C_{mix}

For a small training set, $C_{training}$ is unlikely to contain adequate statistics to characterize valid distortions. The generic constraint of C_{smooth} can help augment the available statistics. In order to build a training set covariance matrix, $C_{training}$, we need at least two examples. Given two examples, however, there is only one mode of variation which interpolates between the two. C_{smooth} , however, can generate many modes which allow smooth deformation, but they allow variability without much specificity. The combination of $C_{training}$ and C_{smooth} results in a much better model allowing variability due to C_{smooth} and specificity due to $C_{training}$ when few training set examples are available.

Cootes *et al.* [39] also combined the shape variability determined statistically

from a training set with that generated artificially by using a physically based model proposed by Pentland *et al.* [101]. These processes require a computationally intensive finite element analysis to solve for the modes of variation. The combined model proposed in this work achieves the similar goal in a different and much simpler way.

In addition to the true training set we introduced in Section 2.1, suppose we also have a smoothness set with m' aligned examples and each example has a set of N aligned labeled points, $\mathbf{L}'_i = (x'_i(1), y'_i(1), x'_i(2), y'_i(2), \dots, x'_i(N), y'_i(N))^T$ ($i = 1, 2, \dots, m'$). The corresponding mean shape and covariance about the mean can be denoted as $\bar{\mathbf{L}}'$ and C_{smooth} . As shown in Equation (3.3),

$$C_{training} = \frac{1}{m-1} \sum_{i=1}^m (\mathbf{L}_i - \bar{\mathbf{L}})(\mathbf{L}_i - \bar{\mathbf{L}})^T \quad (3.10)$$

Similarly,

$$C_{smooth} = \frac{1}{m'-1} \sum_{i=1}^{m'} (\mathbf{L}'_i - \bar{\mathbf{L}}')(\mathbf{L}'_i - \bar{\mathbf{L}}')^T \quad (3.11)$$

Now we want to derive a pooled estimate of the covariance matrices $C_{training}$ and C_{smooth} , which is denoted as C_{mix} . A pooled estimate of the variance from s_1^2 and s_2^2 is given by *adding the sums of squares of deviations in the numerators of s_1^2 and s_2^2 and dividing by the sum of their degrees of freedom, which is one less than the total sample size* [115].

Therefore, C_{mix} can be calculated as:

$$\begin{aligned} C_{mix} &= \frac{\sum_{i=1}^m (\mathbf{L}_i - \bar{\mathbf{L}})(\mathbf{L}_i - \bar{\mathbf{L}})^T + \sum_{i=1}^{m'} (\mathbf{L}'_i - \bar{\mathbf{L}}')(\mathbf{L}'_i - \bar{\mathbf{L}}')^T}{(m-1) + (m'-1)} \\ &= \frac{m-1}{(m-1) + (m'-1)} C_{training} + \frac{m'-1}{(m-1) + (m'-1)} C_{smooth} \end{aligned}$$

$$= \left[C_{training} + \frac{m' - 1}{m - 1} C_{smooth} \right] \frac{m - 1}{(m - 1) + (m' - 1)} \quad (3.12)$$

Now, let:

$$w = \frac{m' - 1}{m - 1} \implies (m' - 1) = w(m - 1) \quad (3.13)$$

We have:

$$\begin{aligned} C_{mix} &= [C_{training} + w \cdot C_{smooth}] \frac{m - 1}{(m - 1)(1 + w)} \\ &= [C_{training} + w \cdot C_{smooth}] \frac{1}{1 + w} \end{aligned} \quad (3.14)$$

In this way, we can build a model by calculating the eigenvectors and eigenvalues of matrix C_{mix} when few true training set examples are available.

For $w > 0$, C_{mix} will include both the training set statistics and the smoothness model. $C_{training}$ will have only $m - 1$ non-zero eigenvalues (for $m < 2N$). Using C_{mix} , however, gives the full $2N$ non-zero eigenvalues no matter what the training set size and allows for greater flexibility in the model consistent with the additional smoothness statistics. For larger training sets, the statistics of the available data become more dependable and thus we can reduce w as the number of samples m increases. Given a large number of examples, our model is more likely to describe examples from a class of shapes by linear combinations of the training set, without the need for additional smooth deformations. By Equation (3.13), w tends to decrease as $(m - 1)^{-1}$. If m' is chosen as a constant, the number of smoothness set examples is treated as fixed no matter the number of true training examples. Therefore, when fewer training examples are available, the smooth-

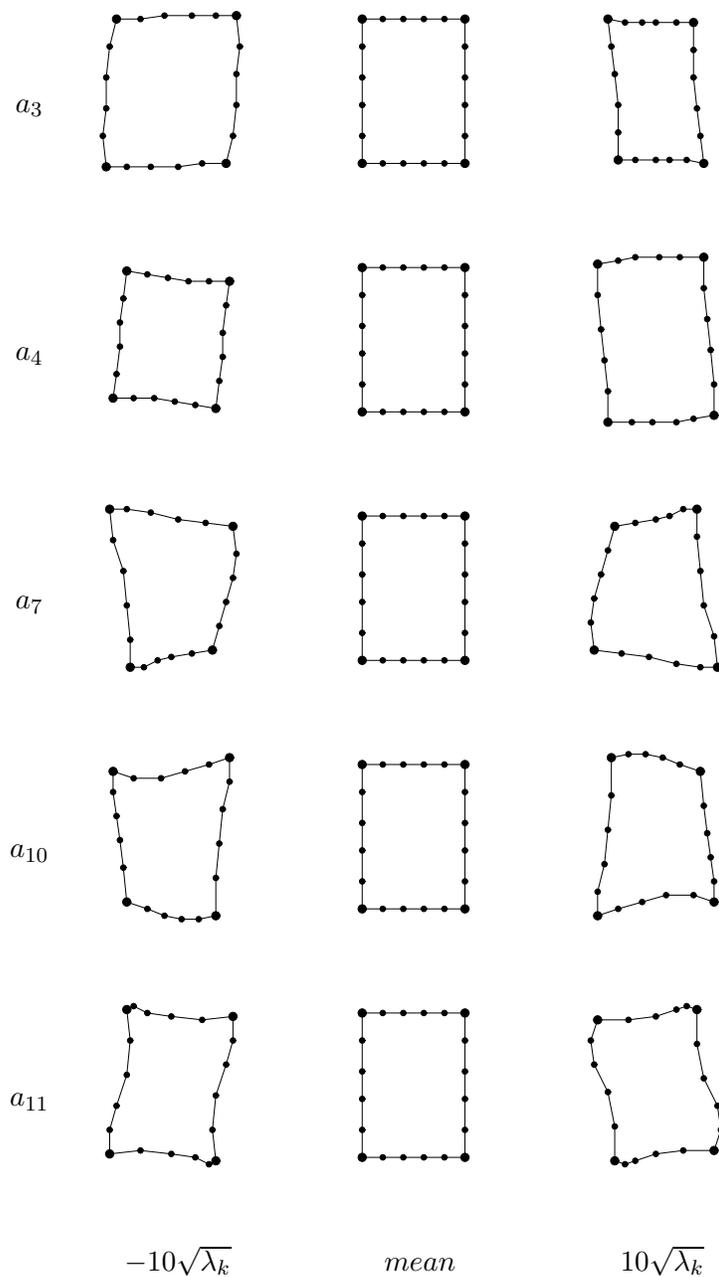


Figure 3.6: Effects of varying each of the 3rd, 4th, 7th, 10th and 11th shape parameters of the square and rectangle shape model with smoothness covariance matrix C_{smooth} . (The first two shape parameters correspond to rigid translation.)

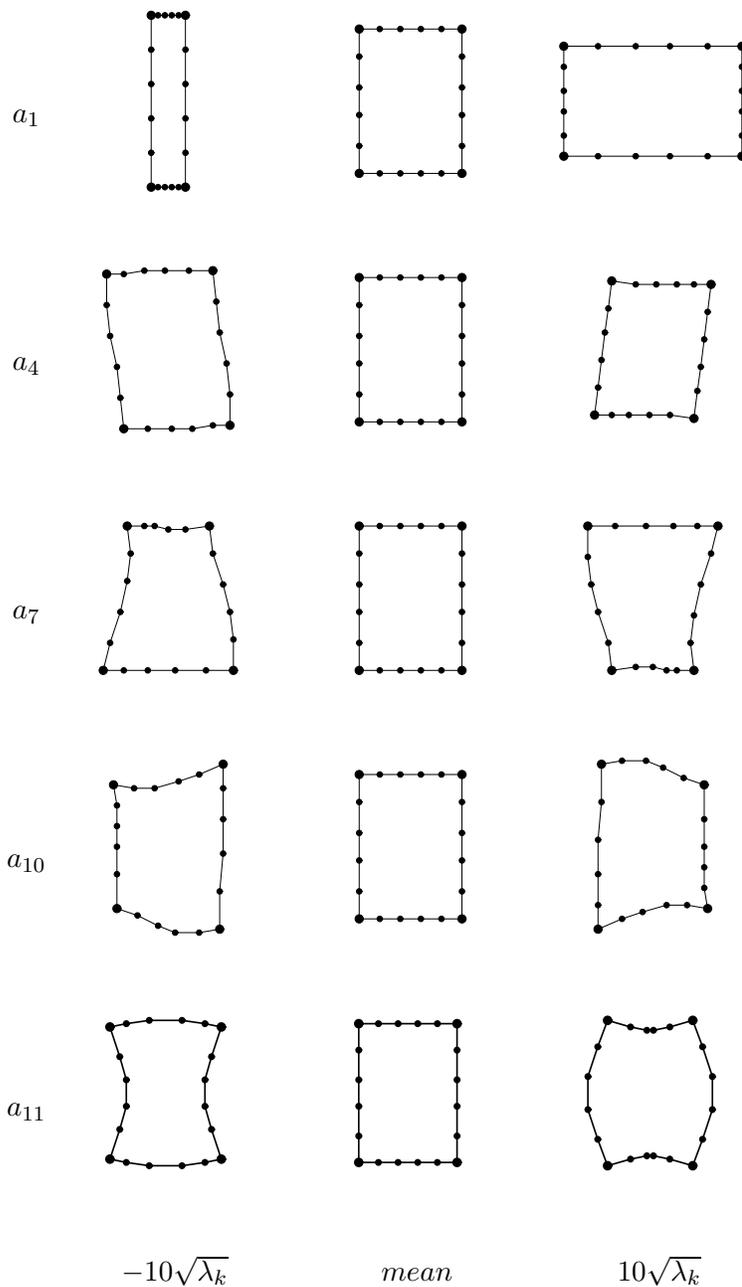


Figure 3.7: Effects of varying each of the 1st, 4th, 7th, 10th and 11th shape parameters of the square and rectangle shape model with combined covariance matrix C_{mix} . (The 2nd and 3rd shape parameters correspond to rigid translation.)

ness examples play a role of increasing the variability of the model; when more training examples are available, the relative role played by the smoothness set is decreased because we have more reliable variation modes derived from the true training set.

As an example, two sets of 20 points were generated: a square and a rectangle. Figure 3.6 shows the modes of variation generated by C_{smooth} (Equation (3.9)) allowing smooth changes to the mean shape. Figure 3.7 shows the modes of variation generated by C_{mix} (Equation (3.14)). Here, in addition to smooth variation, the principal mode a_1 changes the aspect ratio reflecting the variation in the training set. In fact, this would be the only mode of $C_{training}$.

3.3 Bayesian Matching

Given the statistical models, our aim is to use them to model particular examples of structure in individual images, and then to find the shape parameters $\mathbf{a} = (a_1 a_2 \cdots a_t)^T$, and pose parameters: scale s , rotation θ , translation T_x, T_y . The combined pose and shape parameter vector to be determined is represented by $\mathbf{p} = (s, \theta, T_x, T_y, a_1, a_2, \cdots, a_t)^T$. The point representation of the n th boundary point ($n = 0, 1, \dots, N - 1$) is

$$\begin{cases} x(\mathbf{p}, n) = s \cos \theta \left\{ \bar{x}(n) + \sum_{k=1}^t Q_{2n,k} a_k - \frac{S_x}{2} \right\} \\ \quad - s \sin \theta \left\{ \bar{y}(n) + \sum_{k=1}^t Q_{2n+1,k} a_k - \frac{S_y}{2} \right\} + T_x + \frac{S_x}{2} \\ y(\mathbf{p}, n) = s \sin \theta \left\{ \bar{x}(n) + \sum_{k=1}^t Q_{2n,k} a_k - \frac{S_x}{2} \right\} \\ \quad + s \cos \theta \left\{ \bar{y}(n) + \sum_{k=1}^t Q_{2n+1,k} a_k - \frac{S_y}{2} \right\} + T_y + \frac{S_y}{2} \end{cases} \quad (3.15)$$

where $\bar{x}(n)$ and $\bar{y}(n)$ are the mean shape of the n th point, and S_x, S_y are the image sizes in x, y directions respectively.

In order to apply the prior knowledge of the shape model to the problem of bound-

ary and correspondence determination, we pose the problem in a maximum *a posteriori* Bayesian formulation. The prior information of the object shape comes from the statistical shape analysis described in Section 3.2. We propose an energy function based on the image boundary strength and the deformed boundary in order to arrive at the likelihood. The likelihood is then combined with the prior using Baye’s rule to obtain the *a posteriori* probability density of the deformations of the boundary given the input image. The boundary of the object and the point-by-point boundary correspondence are determined when the *a posteriori* probability density is maximized.

3.3.1 Maximum *a Posteriori* Match

The estimation process involves finding optimum values of the parameters that describe the boundary given the image data.

This is based on the expectation that the target object can be differentiated from the background by some measure of the boundary strength and direction (if available), computed from the image. This section described the objective function that is being optimized based on a maximum *a posteriori* (MAP) formulation after [116].

A maximum *a posteriori* formulation is used in order to incorporate prior shape knowledge while estimating the boundaries from the true image data. Let $E(x, y)$ be the smoothed edge image intensity and $t_{\mathbf{p}}(x, y)$ be an image template corresponding to the parameter vector \mathbf{p} , which needs to be estimated. The goal is to detect the object boundary, which in turn is given by the most probable such boundary given the prior shape knowledge and the image information. This is done by maximizing $\Pr(t_{\mathbf{p}} | E)$, the probability of the template given the edge image, where the maximization is done over

\mathbf{p} . Using Bayes rule, this is equivalent to:

$$\arg \max_{\mathbf{p}} \Pr(t_{\mathbf{p}} | E) = \arg \max_{\mathbf{p}} \frac{\Pr(E | t_{\mathbf{p}}) \Pr(t_{\mathbf{p}})}{\Pr(E)} \quad (3.16)$$

where in the above MAP formulation, $\Pr(t_{\mathbf{p}})$ is the prior probability of the template and $\Pr(E | t_{\mathbf{p}})$ is the likelihood that the template conforms to the cues available from the image. Ignoring the denominator of the RHS(which is not a function of \mathbf{p} , with respect to which the optimization is carried out), and taking the logarithm of the above gives,

$$\arg \max_{\mathbf{p}} O(E, t_{\mathbf{p}}) = \arg \max_{\mathbf{p}} [\ln \Pr(t_{\mathbf{p}}) + \ln \Pr(E | t_{\mathbf{p}})] \quad (3.17)$$

In the above, $O(\cdot)$ is the general form of the objective function that is being optimized. This basic form shows the tradeoff or compromise that will be made between the prior information, $\Pr(t_{\mathbf{p}})$ (which will be formulated in the Section 3.3.1.1), and the data-driven likelihood information, $\Pr(E | t_{\mathbf{p}})$ (which will be discussed in Section 3.3.1.2).

3.3.1.1 Prior Probability Density

In order to use prior information, probability distributions are associated with the parameters. This prior information can then bias the boundary finder to search for a particular range of shapes. In this work, the statistical point model and the combined model derived from the training set, or the smoothness model are used as the prior model to reduce the search space. When prior information is not available, the proposed independent model is employed.

Instead of using the original covariance matrices corresponding to each of the four models, we derive the mean and the variance of the shape parameters through the eigen-

decomposition (Section 3.2.1.2). Then, an independent, multivariate Gaussian density $\Pr(\mathbf{p})$ is used for the $t + 4$ shape and pose parameters (as in [117]):

$$\Pr(\mathbf{p}) = \prod_{j=1}^{t+4} \Pr(p_j) = \prod_{j=1}^{t+4} \frac{1}{\sqrt{2\pi}\sigma_j} e^{-\frac{(p_j - m_j)^2}{2\sigma_j^2}} \quad (3.18)$$

Here, m_j is the mean of p_j and σ_j^2 is the variance for each of the parameters. For the shape parameters, translation and rotation, the means are defined to be zero relative to the mean point configuration. The mean scale (m_1) is defined to be 1. The variance for each shape parameter is the eigenvector's corresponding eigenvalue. For the pose parameters, the variance is calculated from the training set alignment (see Section 3.2.1.1).

3.3.1.2 Likelihood

The likelihood is a measure of the similarity between the deformed template and the object present in the image. Similar as that in [116], a Gaussian noise model assumption is used. Consequently, the smoothed edge image E is assumed to consist of one of the deformed templates, $t_{\mathbf{p}}$, corrupted by additive white zero mean Gaussian noise, i.e. $E = t_{\mathbf{p}} + n$. Hence,

$$\Pr(E | t_{\mathbf{p}}) \equiv \Pr(E = t_{\mathbf{p}} + n) \equiv \Pr(n = E - t_{\mathbf{p}}) \quad (3.19)$$

and the noise at each pixel $n(x, y)$ is given by:

$$n(x, y) = E(x, y) - t_{\mathbf{p}}(x, y) \quad (3.20)$$

Due to the white noise assumption (which implies independence between the pixels), the joint probability over the entire area \mathcal{A} is identical to the product of their individual probabilities. The conditional probability of obtaining E given the underlying structure is that of the template is then given by

$$\Pr(E | t_{\mathbf{p}}) = \prod_{\mathcal{A}} \Pr(n(x, y)) \quad (3.21)$$

If the mean of the Gaussian is zero and the standard deviation is σ_n , then

$$\Pr(E | \mathbf{p}) = \prod_{\mathcal{A}} \frac{1}{\sqrt{2\pi}\sigma_n} e^{-\frac{(E(x,y)-t_{\mathbf{p}}(x,y))^2}{2\sigma_n^2}} \quad (3.22)$$

By taking the logarithm and substituting the result in Equation (3.17), the expanded objective function becomes

$$O(E, t_{\mathbf{p}}) = \ln \Pr(t_{\mathbf{p}}) + \sum_{\mathcal{A}} \ln \frac{1}{\sqrt{2\pi}\sigma_n} - \sum_{\mathcal{A}} \frac{(E(x, y) - t_{\mathbf{p}}(x, y))^2}{2\sigma_n^2} \quad (3.23)$$

This represents the MAP function for the images with the assumption of independent Gaussian noise at each pixel. The first term is the prior term, the second one is a constant, and the last one represents the data likelihood term. It is easy to see that as long as $\sum_{\mathcal{A}} t_{\mathbf{p}}^2(x, y)$ does not vary much, the last term in Equation (3.23) is similar to a correlation term.

3.3.2 Simplified Boundary Objective Function

As already indicated before, the boundary of the target object is represented by the template $t_{\mathbf{p}}(x, y)$. The templates are assumed to form a continuum, each having a cor-

responding value of the parameter vector \mathbf{p} . Essentially, the boundary is 1D, but it is embedded in a 2D image by assuming that $t_{\mathbf{p}}(x, y)$ is constant along the contour and is zero everywhere else. Thus, $E(x, y)$ is considered to be a boundary measure applied to the raw image data, i.e. $E(x, y) = E(I(x, y))$. Generally, the boundary measure is given by the gradient magnitude, or, by both the gradient magnitude and the direction of the original image. In this study, the Canny edge operator is used, but other features can easily be incorporated. Only points that lie on the contour are considered because those are the only points at which the template is non-zero. Now, Equation (3.23) becomes:

$$O(E, t_{\mathbf{p}}) = \ln \Pr(t_{\mathbf{p}}) + \sum_{\mathcal{A}} \ln \frac{1}{\sqrt{2\pi}\sigma_n} - \frac{1}{2\sigma_n^2} \left[\sum_{\mathcal{A}} E^2(x, y) + \sum_{\mathcal{C}_{\mathbf{p}}} (-2E(x, y)t_{\mathbf{p}}(x, y) + t_{\mathbf{p}}^2(x, y)) \right] \quad (3.24)$$

where $\mathcal{C}_{\mathbf{p}}$ is the curve defined by the boundary $(x(\mathbf{p}), y(\mathbf{p}))$ in the template $t_{\mathbf{p}}$. Since $t_{\mathbf{p}}(x, y)$ is constant over the curve, and the curve consists of N discrete points, then:

$$O(E, t_{\mathbf{p}}) = \ln \Pr(t_{\mathbf{p}}) + \sum_{\mathcal{A}} \left[\ln \frac{1}{\sqrt{2\pi}\sigma_n} - \frac{E^2(x, y)}{2\sigma_n^2} \right] + \frac{1}{2\sigma_n^2} \sum_{n=1}^N (2E(x(\mathbf{p}, n), y(\mathbf{p}, n))k - k^2) \quad (3.25)$$

where k is the template magnitude at any point which is assumed to be a constant and is chosen to be the maximum boundary response. The first summation term in the above does not change with respect to variations in \mathbf{p} . Further, the last term that involves k^2 , is proportional to the length of the contour which is assumed to not change appreciably when compared to changes in the boundary measure. In view of that, we ignore that

term, and redefine (after simplification) $O(\cdot)$ as:

$$O(E, \mathbf{p}) = \ln \Pr(t_{\mathbf{p}}) + \frac{1}{\sigma_n^2} \sum_{n=1}^N E(x(\mathbf{p}, n), y(\mathbf{p}, n))k \quad (3.26)$$

Now, we expand the first term in Equation (3.26) using the Gaussian distribution shown in Equation (3.18) to get

$$O(\mathbf{p}) = \sum_{j=1}^{t+4} \left[-\frac{(p_j - m_j)^2}{2\sigma_j^2} \right] + \frac{1}{\sigma_n^2} \sum_{n=1}^N E(x(\mathbf{p}, n), y(\mathbf{p}, n))k \quad (3.27)$$

In practical implementation, we use the following equation:

$$O(\mathbf{p}) = c \sum_{j=1}^{t+4} \left[-\frac{(p_j - m_j)^2}{2\sigma_j^2} \right] + \sum_{n=1}^N E(x(\mathbf{p}, n), y(\mathbf{p}, n)) \quad (3.28)$$

where c is a constant coefficient. It is fixed empirically for different training examples within the range $0.1 \sim 1.0$.

This equation (Equation (3.28)) is the maximum *a posteriori* objective incorporating a prior bias to likely shapes and poses (first term) and match to the edges in the image by maximizing the sum of the smoothed edge image intensity at the boundary points defined by vector \mathbf{p} (second term). The influence of the prior term is controlled by the variance of the prior probability. If the variance of the prior is greater, the influence of this term is smaller.

3.4 Implementation

3.4.1 Optimization Algorithm

There are many optimization algorithms that can be used to maximize the objective function $O(\mathbf{p})$. Some methods need only the evaluation of the function to be minimized, while others also require evaluations of the derivative of that function. We optimize the objective function $O(\mathbf{p})$ using the conjugate gradient method, which takes small steps in the direction of the gradient. If the gradient computation is comparable with the function computation, the overall computation can be greatly reduced by the use of the gradient. We can efficiently compute the gradient from an analytic formulation which is described below.

3.4.2 Gradient Formulation

A practical gradient calculation may be obtained by using a numerical approximation based directly on the objective function calculation. First, differentiate Equation (3.28) to get

$$\begin{aligned} \nabla O(\mathbf{p}) = & c \sum_{j=1}^{t+4} \left[-\frac{p_j - m_j}{\sigma_j^2} \right] + \sum_{n=1}^N \left[\frac{\partial E(x(\mathbf{p}, n), y(\mathbf{p}, n))}{\partial x} \frac{\partial x(\mathbf{p}, n)}{\partial \mathbf{p}} \right. \\ & \left. + \frac{\partial E(x(\mathbf{p}, n), y(\mathbf{p}, n))}{\partial y} \frac{\partial y(\mathbf{p}, n)}{\partial \mathbf{p}} \right] \end{aligned} \quad (3.29)$$

The partials of E , the edge image, with respect to x and y can be calculated using a central discrete divided difference approximation. The partials of $x(\mathbf{p}, n)$ with respect

to \mathbf{p} can be calculated from Equation (3.15) by:

$$\begin{aligned}
\frac{\partial x(\mathbf{p}, n)}{\partial s} &= \cos \theta \left[\bar{x}(n) + \sum_{k=1}^t Q_{2n,k} a_k - \frac{S_x}{2} \right] \\
&\quad - \sin \theta \left[\bar{y}(n) + \sum_{k=1}^t Q_{2n+1,k} a_k - \frac{S_y}{2} \right]; \\
\frac{\partial x(\mathbf{p}, n)}{\partial \theta} &= -s(\sin \theta) \left[\bar{x}(n) + \sum_{k=1}^t Q_{2n,k} a_k - \frac{S_x}{2} \right] \\
&\quad - s(\cos \theta) \left[\bar{y}(n) + \sum_{k=1}^t Q_{2n+1,k} a_k - \frac{S_y}{2} \right]; \\
\frac{\partial x(\mathbf{p}, n)}{\partial T_x} &= 1; \\
\frac{\partial x(\mathbf{p}, n)}{\partial T_y} &= 0; \\
\frac{\partial x(\mathbf{p}, n)}{\partial a_j} &= s(\cos \theta) Q_{2n,j} - s(\sin \theta) Q_{2n+1,j};
\end{aligned} \tag{3.30}$$

and similarly for $y(\mathbf{p}, n)$:

$$\begin{aligned}
\frac{\partial y(\mathbf{p}, n)}{\partial s} &= \sin \theta \left[\bar{x}(n) + \sum_{k=1}^t Q_{2n,k} a_k - \frac{S_x}{2} \right] \\
&\quad + \cos \theta \left[\bar{y}(n) + \sum_{k=1}^t Q_{2n+1,k} a_k - \frac{S_y}{2} \right]; \\
\frac{\partial y(\mathbf{p}, n)}{\partial \theta} &= s(\cos \theta) \left[\bar{x}(n) + \sum_{k=1}^t Q_{2n,k} a_k - \frac{S_x}{2} \right] \\
&\quad - s(\sin \theta) \left[\bar{y}(n) + \sum_{k=1}^t Q_{2n+1,k} a_k - \frac{S_y}{2} \right]; \\
\frac{\partial y(\mathbf{p}, n)}{\partial T_x} &= 0; \\
\frac{\partial y(\mathbf{p}, n)}{\partial T_y} &= 1; \\
\frac{\partial y(\mathbf{p}, n)}{\partial a_j} &= s(\sin \theta) Q_{2n,j} + s(\cos \theta) Q_{2n+1,j}.
\end{aligned} \tag{3.31}$$

3.4.3 Comparison Method and Computation Time

For comparison, we also implemented a version of the image search method of Cootes *et al.* [38], which determines a suggested movement for each model point based only on image edges. This does not include, however, a model of the expected gray levels around each model point [35, 67]. In this case, during image search, each point is moved toward the nearby area which best matches its local gray level model. It is also possible for us to incorporate this local gray level model into our method. However, in this work, only edge information is used. In order to get a rough idea about the difference between our method and that of Cootes, we show some real image results of Cootes' image search method using only edge information. On an SGI Indy 133MHZ (MIPS R4600), the convergence time for our boundary finding method is about 5 seconds compared with 50 seconds for theirs for an average of 36 points, a speed up by a factor of ten.

3.5 Experimental Results

3.5.1 Evaluation Criteria

The error of each labeled boundary point on the final boundary is calculated by finding the distance to the closest point on the true boundary. We use both average, E_{b_a} , and maximum, E_{b_m} , boundary error measures. The correspondence error of each critical point on the final boundary is the distance between this point and its corresponding critical point on the true boundary. The average error of the correspondence is denoted E_{c_a} .

3.5.2 Synthetic Images

The image shown in Figure 3.8 is a simple synthetic image where the target object (the brightest) belongs to the training set family shown in Figure 3.1, but is not among that training set. The initial curve position is defined by the mean of the training set. The edge map of the input image is calculated by the Canny edge detector [22]. Other appropriate features are possible, as well. The final curve position accurately finds the target object. The following are four sets of experiments testing the effect of noise, initialization, prior probability bias and the form of prior.

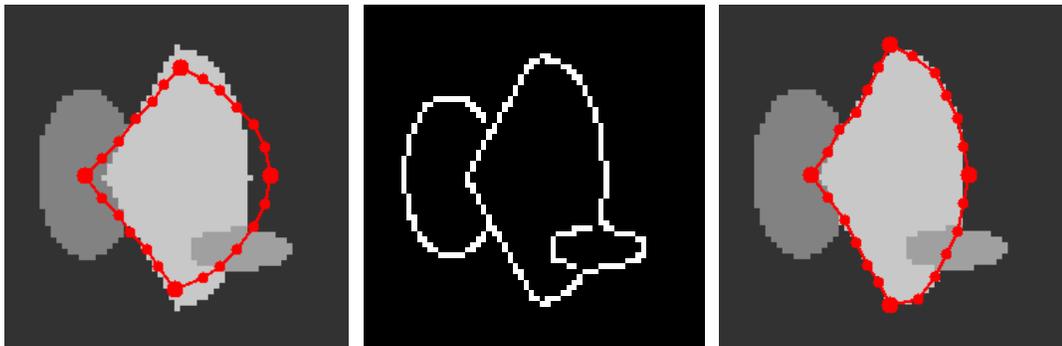


Figure 3.8: Synthetic image example. Left: initial contour on the synthetic image (64×64); Middle: Canny edge image (scale: 1.0); Right: final contour on target shape.

The first experiment, shown in Figure 3.9, demonstrates the effect of noise on the method, by adding different amounts of zero mean Gaussian noise to the synthetic image shown in Figure 3.8 and measuring the boundary and correspondence error. Not all the Canny scales we chose here are optimal so that we can test our algorithm's tolerance to noise and to spurious and broken edges. The initial curve position is the mean of the

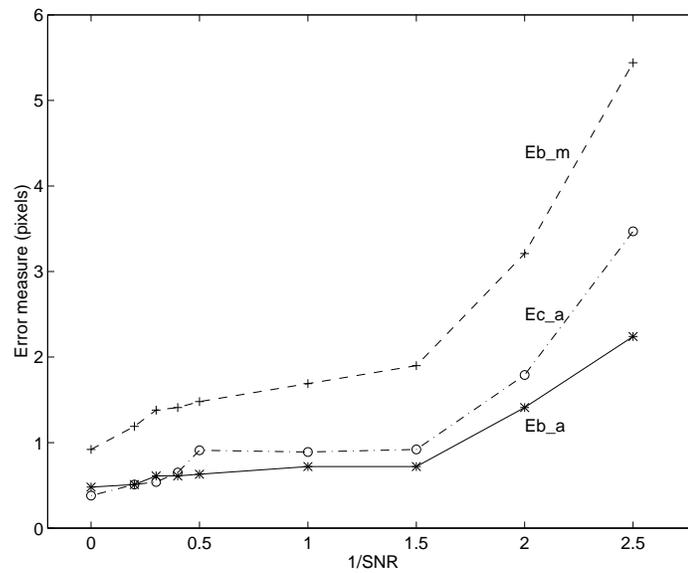
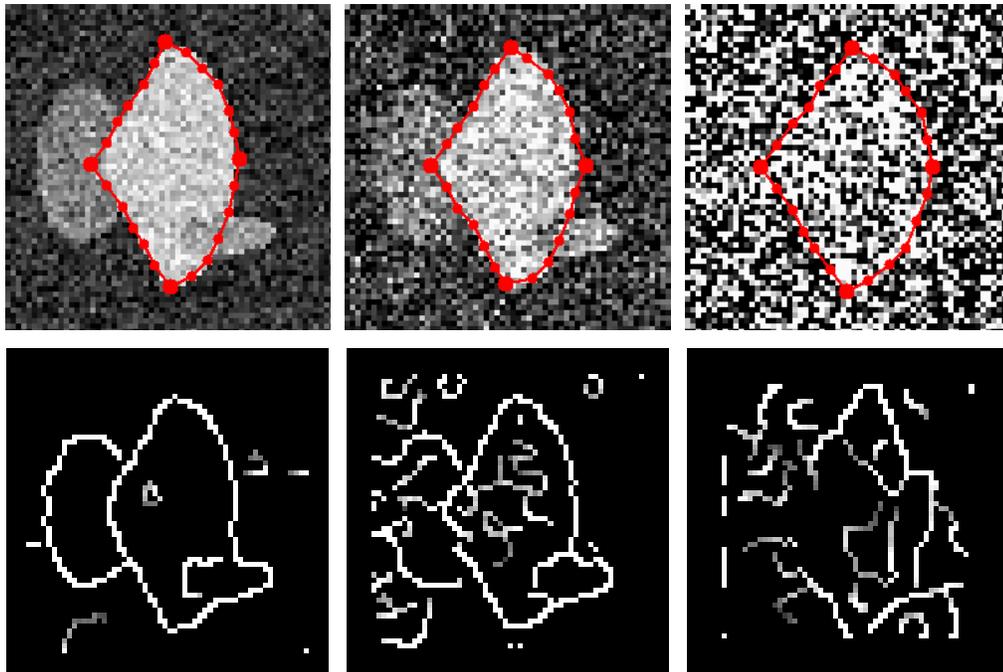


Figure 3.9: Sensitivity to noise experiment. Top, left to right: image from Figure 3.8 with Gaussian noise added with SNR of 5.0, 2.5, 0.5 (each shown with final contour); Middle, left to right: corresponding Canny edge image with scale of 1.2, 1.4, 2.0; Bottom, error measures (Eb_a — boundary average error; Eb_m — boundary maximum error; Ec_a — correspondence average error).

training set for all testing. Signal-to-noise ratio (SNR) is defined here as the ratio of gray-level contrast between the target object and the background to the standard deviation of the Gaussian noise. This example shows the method to be very robust to noise up to $1/\text{SNR}= 1.5$.

The second experiment, shown in Figure 3.10, examines the effect of different initializations. We average three results for each initialization using the synthetic image from Figure 3.8 with SNR of 5.0, 2.5 and 1.0. Each of the different initializations, which is also used as the new mean in Equation (3.6) during optimization, is derived by individually varying the first shape parameter, scale, horizontal translation or rotation while holding the other parameters constant. This experiment tests the robustness of the algorithm when the object in the image is far away or quite different from the mean. All results are good within a reasonable range. Note, some error curves are not symmetric due to the non-symmetric test image with respect to the mean or to itself. For all the initial parameters, when they are too far away from the true boundary, the optimization may be trapped by local maxima corresponding to nearby edges.

In the third experiment, we demonstrate the effect of different prior probability densities. The shape model for the training set is shown in the top of Figure 3.11. A synthetic image, shown in the bottom of Figure 3.11, was designed containing two objects. The light object corresponds to the shape with the first shape parameter at two standard deviations from the mean. The dark object underneath it is the shape with the second shape parameter at two standard deviations from the mean. Using densities with the same mean, but different variances, we can demonstrate different results that are completely due to the prior bias density. The prior can be biased towards finding the light object by having a wide distribution on the first shape parameter and a narrow distribution on the

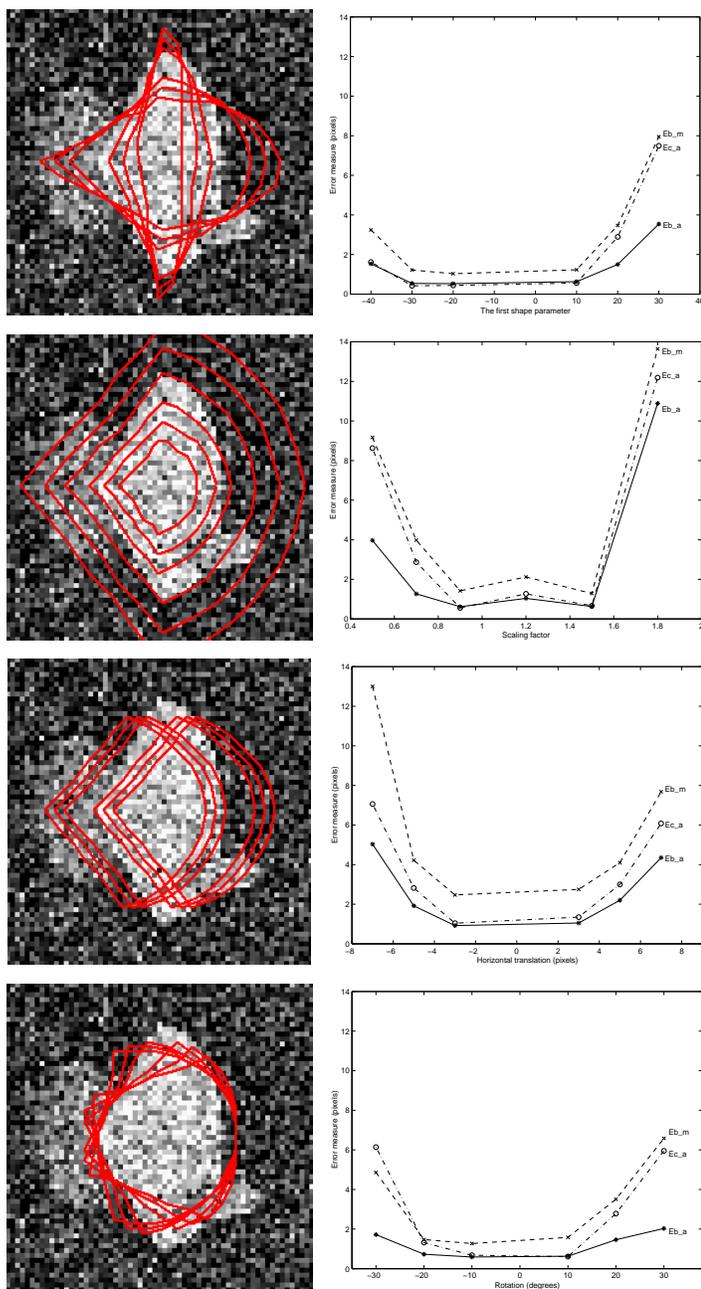


Figure 3.10: Sensitivity to initialization experiment. Left, top to bottom: image from Figure 3.8 with Gaussian noise added (SNR = 2.5) shown with initial contours for the range of the first shape parameter tested, the range of scale tested, the range of horizontal translation tested, and the range of rotation tested; Right: corresponding error measures respectively ($E_{b,a}$ — boundary average error; $E_{b,m}$ — boundary maximum error; $E_{c,a}$ — correspondence average error).

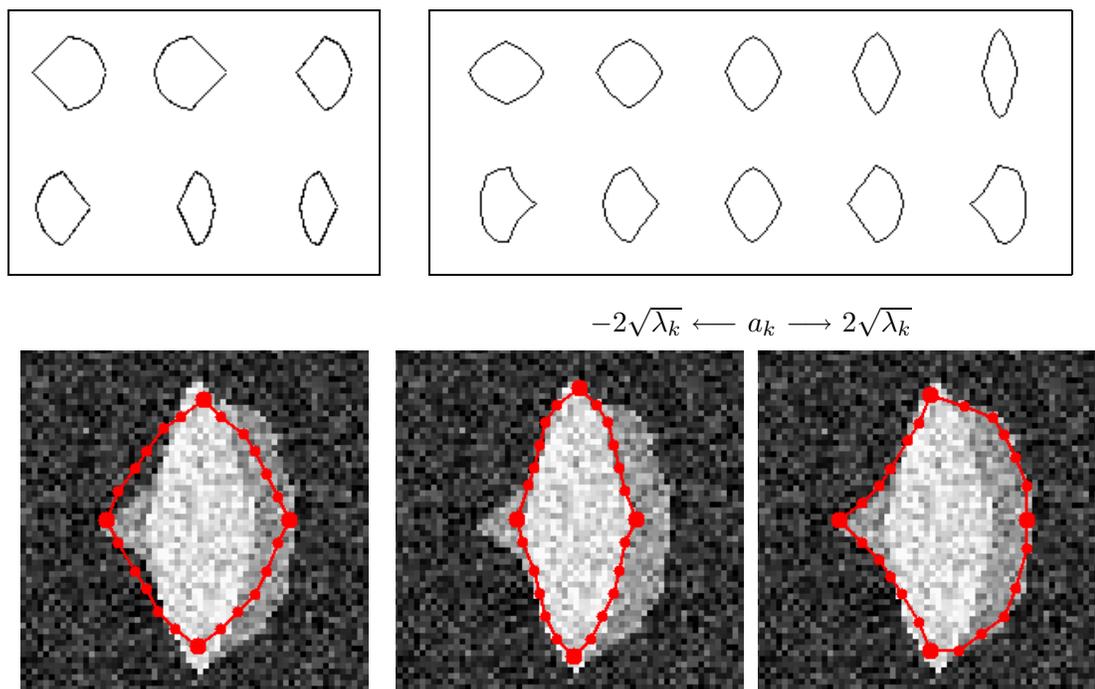


Figure 3.11: Bias experiment. Top left: 6 examples of synthetic shapes from a training set, each containing 24 points; Top right: effects of varying each of the first two shape parameters of the synthetic model; Bottom left: initial contour on the synthetic image (64×64); Bottom middle: final contour, biased to the brighter target shape; Bottom right: final contour, biased to the darker target shape.

second parameter and vice versa (the bottom of Figure 3.11). This would not be possible without the continuous bias of the prior term.

In the fourth experiment, we compare different prior models to illustrate the appropriateness of our prior model. If there is no prior information, using an identity covariance matrix, each point on the boundary can move independently. The objective function only includes the likelihood term. The model will try to match to edges without regard to shape or correspondence. With the smoothness covariance of Equation (3.9), neighboring points on the boundary will tend to move together. The resulting boundary

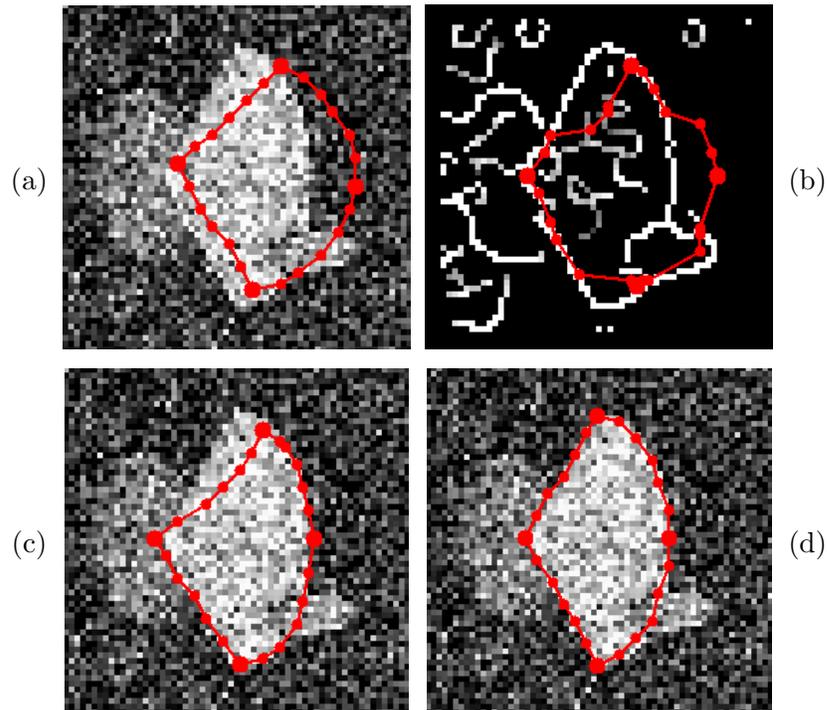


Figure 3.12: Prior experiment for synthetic image. (a): initial contour; (b): final contour on Canny edge image using identity covariance matrix $C_{identity}$; (c): final contour on target shape using smoothness covariance matrix C_{smooth} ; (d): final contour using training set covariance matrix $C_{training}$.

points will try to match edges while maintaining smoothness, but the correspondence of the points may not be maintained. Using the training set covariance matrix, both the boundary and correspondence are found correctly. Here, we used the same synthetic image with $SNR=2.5$ and the same Canny edge image at scale 1.4 (the middle of Figure 3.9). The results with different prior models are shown in Figure 3.12 and Table 3.1. The training set covariance $C_{training}$ works best, as expected, while the smoothness covariance may be suitable in situations where there is no training set and correspondence is not needed. The identity matrix fails to locate the boundary, as expected in this noisy image.

| | $C_{identity}$ | C_{smooth} | $C_{training}$ |
|------|----------------|--------------|----------------|
| Eb_a | 2.58 | 1.45 | 0.57 |
| Eb_m | 7.23 | 5.90 | 1.52 |
| Ec_a | 5.04 | 1.86 | 0.36 |

Table 3.1: Error measure for the synthetic image in the prior experiment with different covariance matrices: Eb_a — boundary average error; Eb_m — boundary maximum error; Ec_a — correspondence average error.

3.5.3 Real Images

Here, all real images (in the training sets and for testing) are 2D slices that roughly correspond from different brains and hearts for demonstration purposes.

The result of the method applied to a sagittal magnetic resonance (MR) image of the human brain is shown in Figure 3.14. Here, we used a 49 point model derived from a set of 12 corpus callosum shapes (Figure 3.13). Not only is the final contour delineated successfully, but also the correspondence of the points is established accurately.

For an axial MR image of the human brain, a 93 point model derived from a set of 12 basal ganglia and ventricle boundaries is used. Figure 3.15 shows that our Bayesian formulation and conjugate gradient optimization gives a quite good final contour and correspondence, although we do not explicitly match features for correspondence, while using Cootes’ image search method [38] the result does not converge. Their algorithm only searches along the normal direction toward the strongest image edge for an estimate of an improved displacement. For the brain image here, this estimation is trapped by nearby edges of other structures, and results in the wrong adjustment of pose and shape parameters. As the iterations continue, the result gets worse.

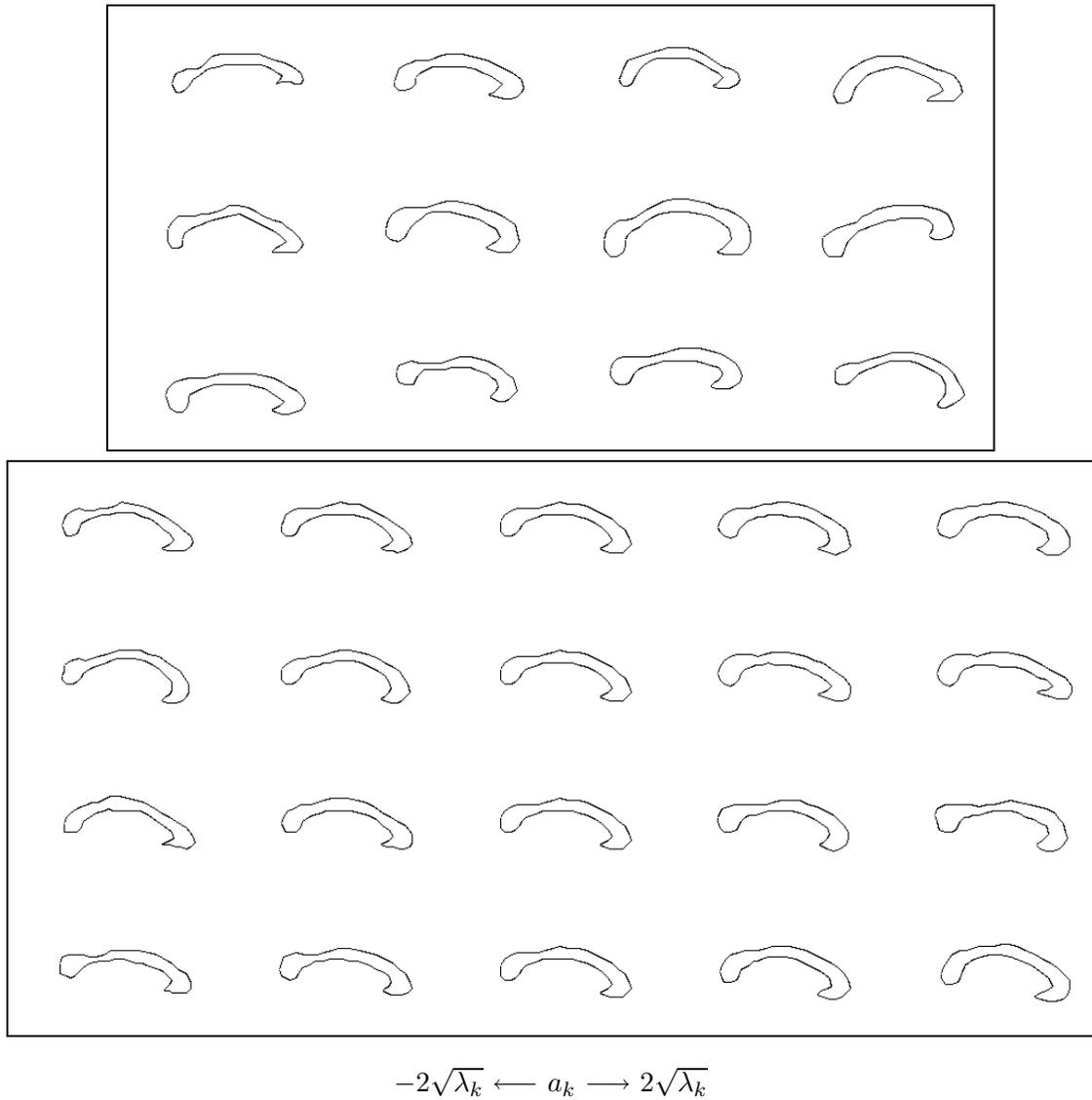


Figure 3.13: Corpus callosum shape model. Top: 12 examples of corpus callosum shapes from a training set; Bottom: effects of varying each of the first four shape parameters of the corpus callosum model.

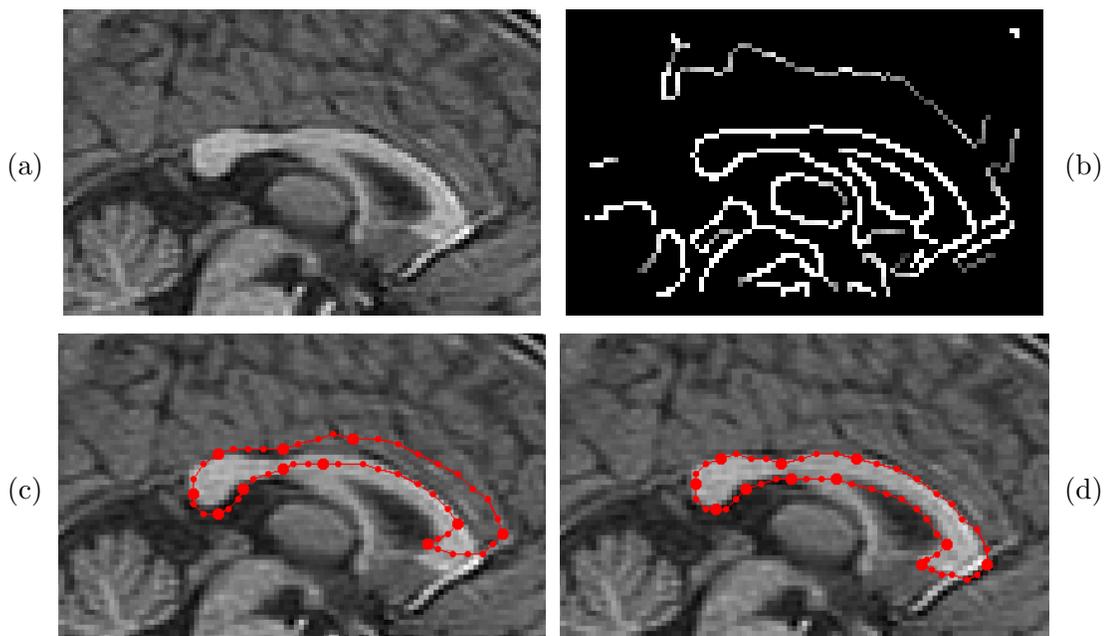


Figure 3.14: MR sagittal brain example. (a): original MR image (100×64); (b): Canny edge image (scale: 1.2); (c): initial contour (mean curve); (d): final contour on corpus callosum.

Figure 3.16 shows our method applied to the endocardium in an MR image of a dog heart. We compared three different prior models for this heart image with the same initial position. With the identity covariance, the boundary cannot be found since the shape is complicated and the points move independently. With the smoothness covariance of Equation (3.9), although the resulting boundary is similar to the true boundary, the correspondence of the points is not determined. By using the heart shape model in Figure 3.2, both the boundary and correspondence are found correctly by our Bayesian method. For comparison, we see that if Cootes' image search is used, the result is not as good. Since the initialization here is quite far away from the true boundary, the normal direction estimation of the displacement is not accurate. The adjustment of the pose and shape parameters can not accommodate the inaccuracies this causes resulting in an inaccurate

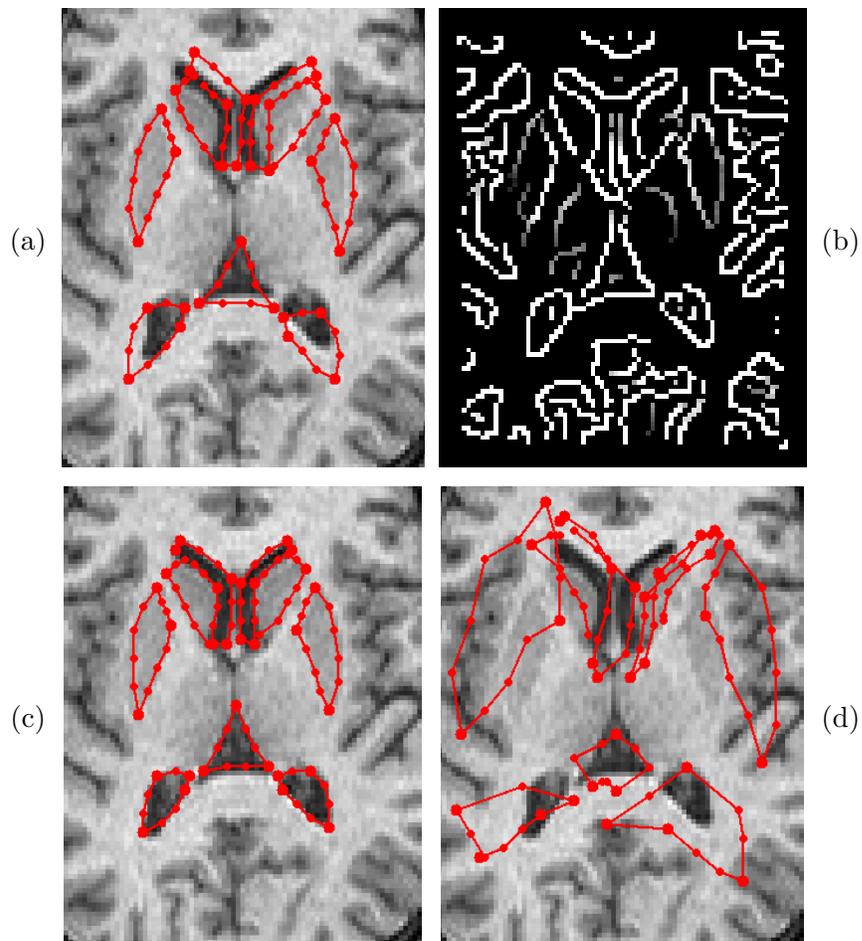


Figure 3.15: MR axial brain example. (a): initial contour (mean curve) on the original MR image (100×80); (b): Canny edge image (scale: 1.2); (c): final contour on basal ganglia and ventricle boundaries by our method; (d): contour at iteration 20 on original image by Cootes' method [38].

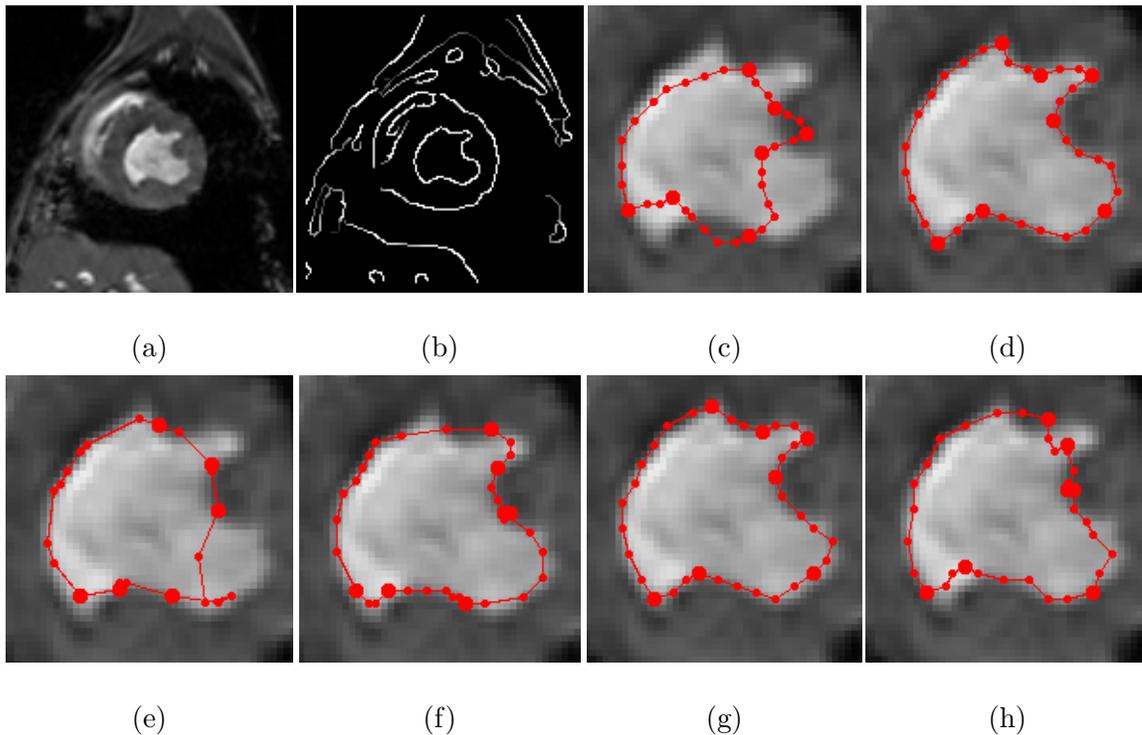


Figure 3.16: MR heart image example and prior experiment. (a): original MR image (150×150); (b): Canny edge image (scale: 1.5); (c): initial contour (mean curve) on endocardium (cropped); (d): contour on endocardium drawn by an expert; (e): final contour on the endocardium using the identity covariance matrix $C_{identity}$ by our method; (f): final contour using smoothness covariance matrix C_{smooth} by our method; (g): final contour using training set covariance matrix $C_{training}$ by our method; (h): final contour by Cootes' method [38].

boundary and correspondence. Table 3.2 shows the quantitative comparison with an expert-drawn endocardium. For this type of heart image, a shape model is a necessity for finding the endocardium and its critical points, and our Bayesian formulation and optimization leads to a much better result than Cootes' image search.

When few training set examples are available, we would like to show the advantage of the combined prior model over the regular statistical point prior model. The training set of varying size, is taken from the set shown in Figure 3.2(c). The performance is evaluated

| | Our method | | | Cootes' method |
|------|----------------|--------------|----------------|----------------|
| | $C_{identity}$ | C_{smooth} | $C_{training}$ | $C_{training}$ |
| Eb_m | 6.12 | 1.79 | 1.32 | 2.44 |
| Ec_a | 5.57 | 7.19 | 0.95 | 4.68 |

Table 3.2: Error measure for the heart image with different covariance matrices and different methods: Eb_m — boundary maximum error; Ec_a — correspondence average error.

by averaging the results on eight test images, shown in Figure 3.17. As mentioned in Section 2.4, the weighting for C_{smooth} , w , tends to decrease as $(m - 1)^{-1}$ (Equation (3.13)) and a reasonable choice of w is to set $(m' - 1)$, or $w(m - 1)$, to a constant from experiment. Experiments show that the error measure of the combined model is stable over a very wide range of $w(m - 1)$. In order to get a better idea of the error versus $w(m - 1)$, Figure 3.18 shows the error measure of the combined model plotted against $\sqrt{w(m - 1)}$. When $w = 0$, we have a pure statistical point model. The figure shows that our mixed model performs well and has a significant improvement over the pure statistical point model when $2 \leq \sqrt{w(m - 1)} \leq 5$ (or $4 \leq w(m - 1) \leq 25$). This wide range means that our mixed model is not very sensitive to the parameter w . The difference between the error at $\sqrt{w(m - 1)} = 0$ and that at $\sqrt{w(m - 1)} = 2 \sim 5$ is an indication of the benefit of adding smoothness variation to the model. Of course, this improvement is not as large for larger training sets. Figure 3.19 shows the variation of error with training set size for the two models. Equation (3.13) was used to set w for the mixed model with $w(m - 1) = 3^2 = 9$, which is equivalent to pooling 10 smoothness examples into the true heart training examples when few training set examples are available. The smaller the training set size, the greater the improvement of the combined model. As the training

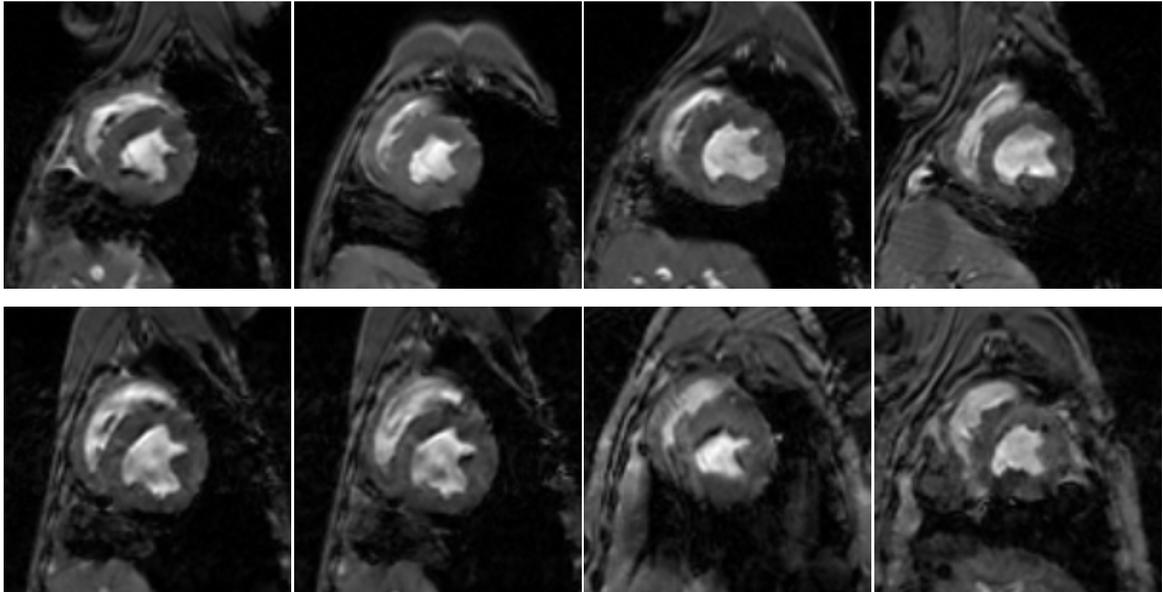


Figure 3.17: MR heart images used to evaluate the performance of the combined model.

set size increases (>14), the improvement is negligible due to the improved quality of the statistical information.

3.6 Conclusions

This chapter presents a systematic approach to the determination of an object's boundary, as well as the correspondence of boundary points to a model. The statistical point models derived from a training set by principal component analysis are used as the prior probability in a Bayesian scheme, capturing prior knowledge of the shape. The structure is delineated and the spatial correspondence of these points to the model is established as a by-product when the *a posteriori* probability is maximized using conjugate gradient optimization. From experimental results, it was found that this method performs well and

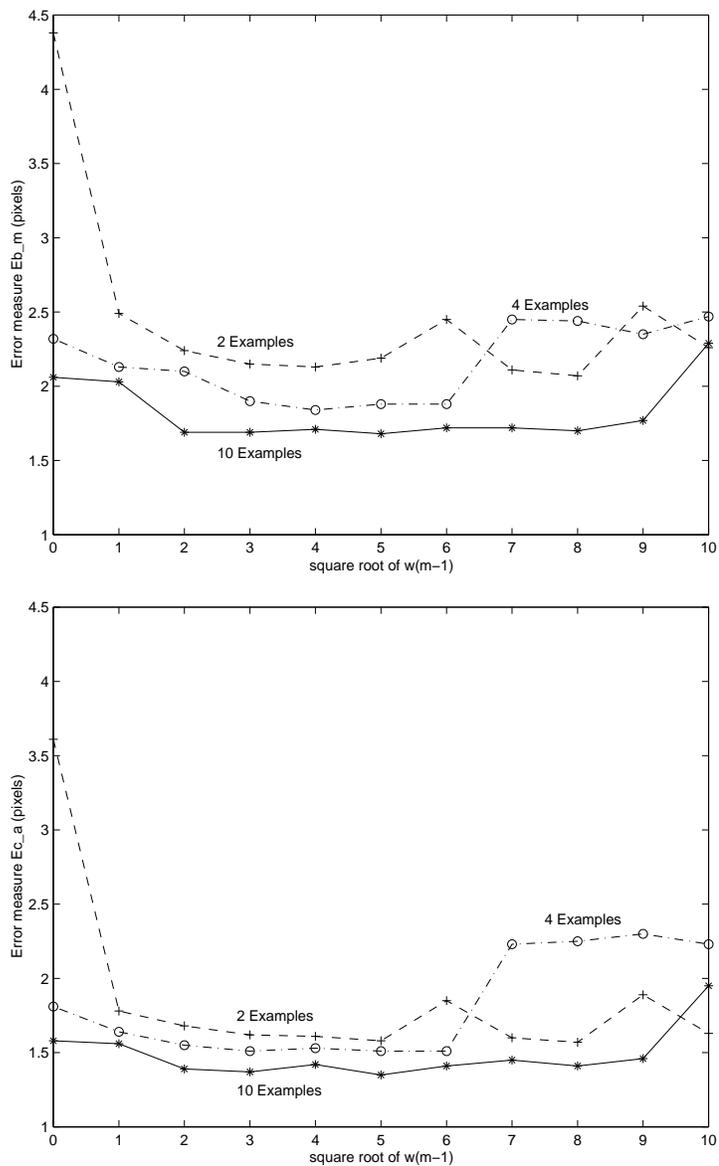


Figure 3.18: Relationship between the error (averaged over the eight test images), the proportion of C_{smooth} included (w) and the number of examples used to train the model (m). ($E_{b,m}$ — boundary maximum error; $E_{c,a}$ — correspondence average error.)

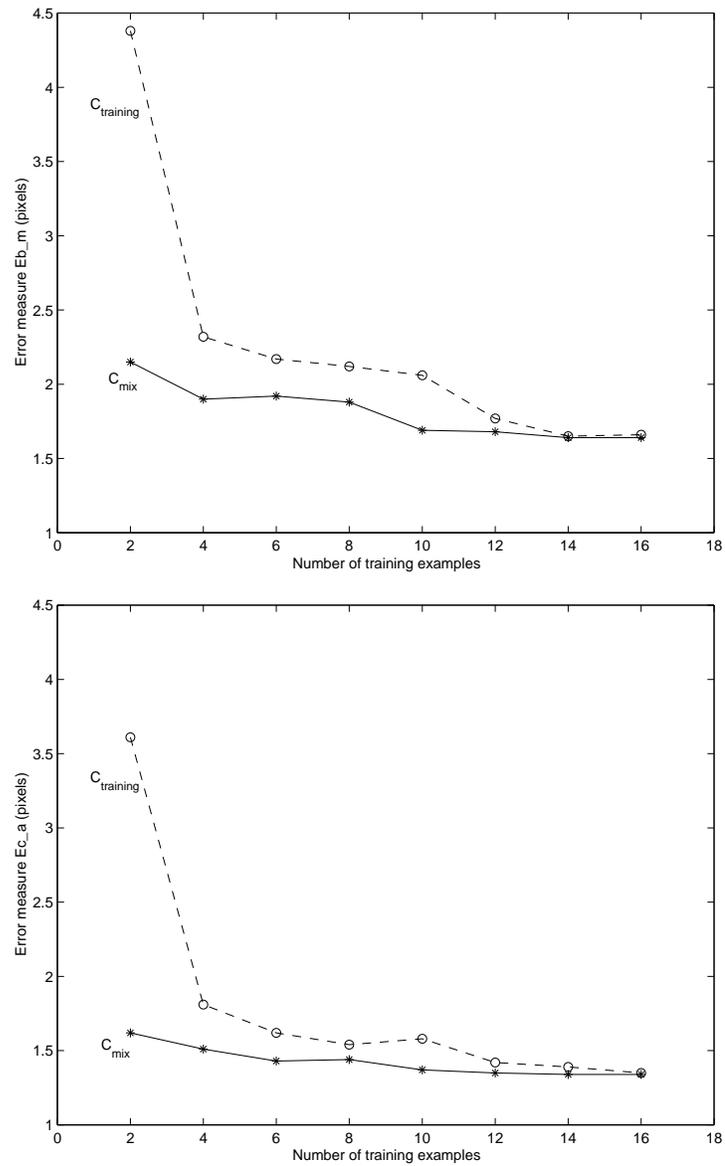


Figure 3.19: The performance (averaged over the eight test images) of $C_{training}$ and C_{mix} training with varying number of examples. (Eb_m — boundary maximum error; Ec_a — correspondence average error.)

is also relatively insensitive to noise and initialization. The prior model testing showed that the statistical shape models are crucial for both boundary and correspondence finding. When few example shapes are available, a combined model of the statistical point model and smoothness model is proposed and adjusts smoothly to use more statistical modes of variation as more and more data is presented. The validation of the combined model training with varying numbers of examples further demonstrated that it improves the results significantly when insufficient shape examples are available.

Chapter 4

Physical Model-Based Non-Rigid Registration Incorporating Statistical Shape Information

4.1 Introduction

As mentioned in Chapter 1, for many approaches to non-rigid registration [8, 14, 29, 33, 40, 53, 96, 123], the transformation is usually constrained in some way because of the ill-posedness (i.e. in this case, the existence of many possible solutions) of the problem. Physical models, for example, linear elastic and viscous fluid models, are widely used to enforce topological properties on the deformation and then constrain the enormous solution space [8, 29, 31, 40, 53, 96]. As with rigid registration, features or gray-level properties can be used for matching. Non-rigid registration requires such detail locally throughout the image that it is hard to avoid using gray-level properties. Structural features may be too sparse for a fine-grain warping. Curves corresponding to structure have,

however, been used successfully to warp images [40, 97, 41, 123]. Use of the continuous gray-level structure within the image allows for a much richer set of data to be exploited for matching purposes. Within medical images, there is an enormous amount of detail that might be lost by a reduction to particular features. On the other hand, strong features are a powerful constraint and may be best used in conjunction with gray level data. Therefore, in this work, we are particularly interested in intensity based deformation using elastic or fluid models. Our goal is to incorporate statistical shape information into this type of physical model-based registration and to develop a more accurate and robust algorithm.

Christensen *et al.* [29] present two physical models for non-rigid registration of the brain. The transformations are constrained to be consistent with an elastic model in the first method and a fluid model in the second. The elastic model requires less computation compared to the fluid model and penalizes deformation in proportion to the deformed distance. This is often too strict a model because of the large variation between anatomical structures and does not allow for the complete deformation necessary. Viscous fluid models are less constraining than elastic models and allow long-distance, nonlinear deformations of small subregions (Figure 4.1, (a) to (c) columns). However, no matter what model is used, elastic solid [96], viscous fluid [31], or other physics model such as hyperelasticity [104], in these cases, the deformed configuration of the atlas is always determined by driving the deformation using only pixel-by-pixel intensity differences between images. In many applications, however, this kind of warping is under-constrained and yields to unreasonable registrations. Corresponding anatomical structure may shift or twist away from one position to another (Figure 4.2(a)(b)), and very large volumes of matter may stream through very small areas from one region to another (Figure 4.3(a)(b)). Even

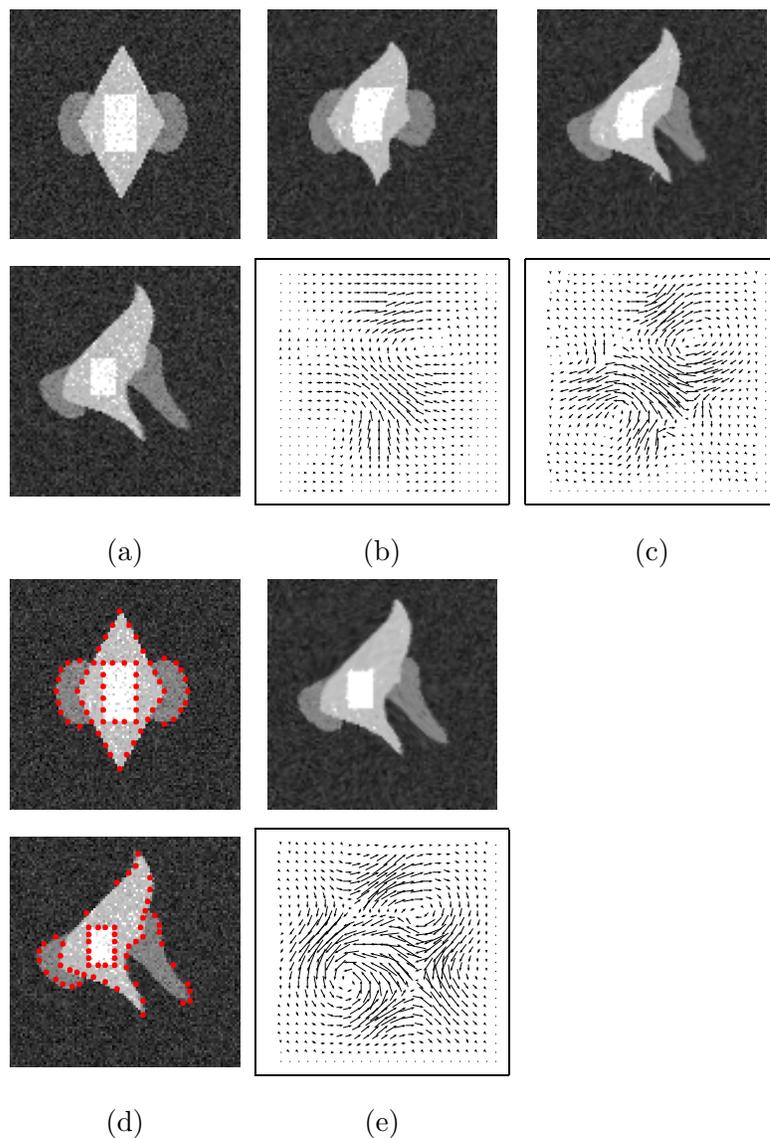


Figure 4.1: Comparison of synthetic image (100×100) non-rigid registration by Christensen's elastic and fluid methods and our fluid method. (a): the atlas image (top) and the study image (bottom); (b): the final deformed atlas by Christensen&Miller's elastic model [29, 96] (top), showing large distance deformations prevented using the appropriate elasticity parameter (large μ) to just ensure a globally positive Jacobian, and the corresponding vector map (bottom); (c): the final deformed atlas by Christensen's fluid model [31] (top), showing that large deformations can be accommodated (compare to (b)), but the shape of the white rectangle is not maintained (compare to (e)); (d): the atlas image with control points (top) and the study image control points (bottom); (e): the final deformed atlas using our viscous-fluid mapping (top), showing appropriate deformation, and the corresponding vector map (compare to (c)).

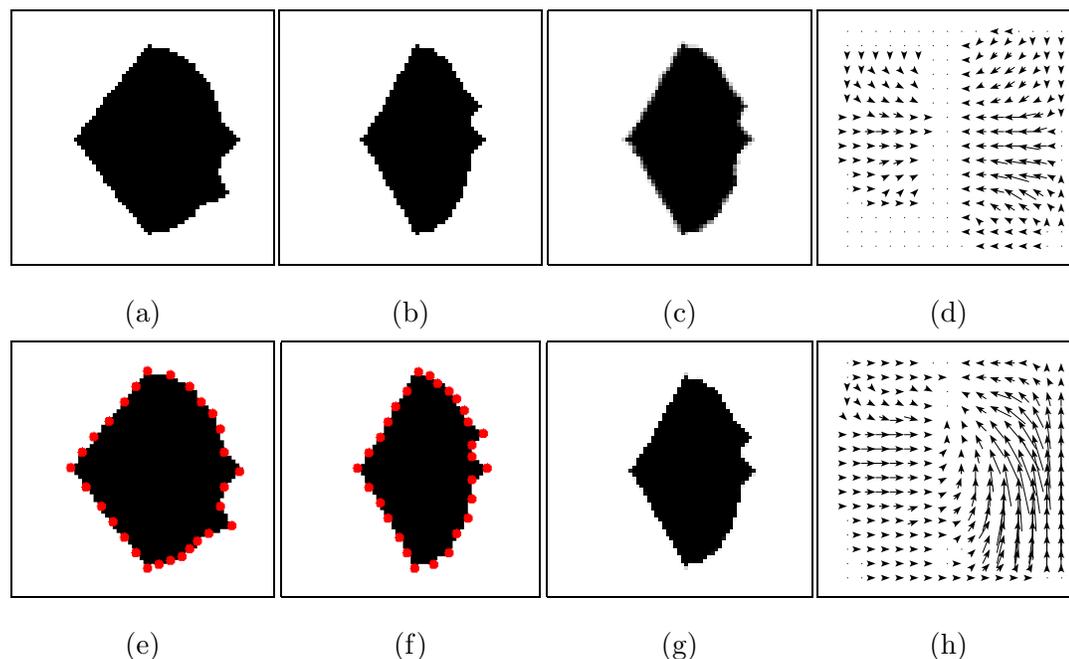


Figure 4.2: Comparison of synthetic image (64×64) non-rigid registration by Christensen&Miller's and our elastic methods. (a): atlas image; (b): study image; (c): deformed atlas image by Christensen&Miller's elastic method [29] [96] (based on our implementation, discussed in Section 5); (d): corresponding unreasonable vector map of (c) (The two bumps on the right side do not slide up to the corresponding bumps, as desired); (e): atlas image with control points; (f): study image with control points; (g): deformed atlas image by our method; (h): vector map of our elastic transformation showing correct tracking of features.

if the driving force is very small, the transformation may not be accurate enough, or may even be completely wrong, *even though* the deformed atlas and study appear similar (Figure 4.2(b)(c)(d) and 4.3(b)(c)(d)). In these circumstances, if shape information embedded in corresponding boundary points (Figure 4.2(e)(f) and 4.3(e)(f)) had been included, the correct mapping or registration could have been found (Figure 4.2(g)(h) and 4.3(g)(h)). In addition, due to the use of the gray-level gradient of the deformed atlas in the body force formulation [29], lower contrast objects deform much slower than

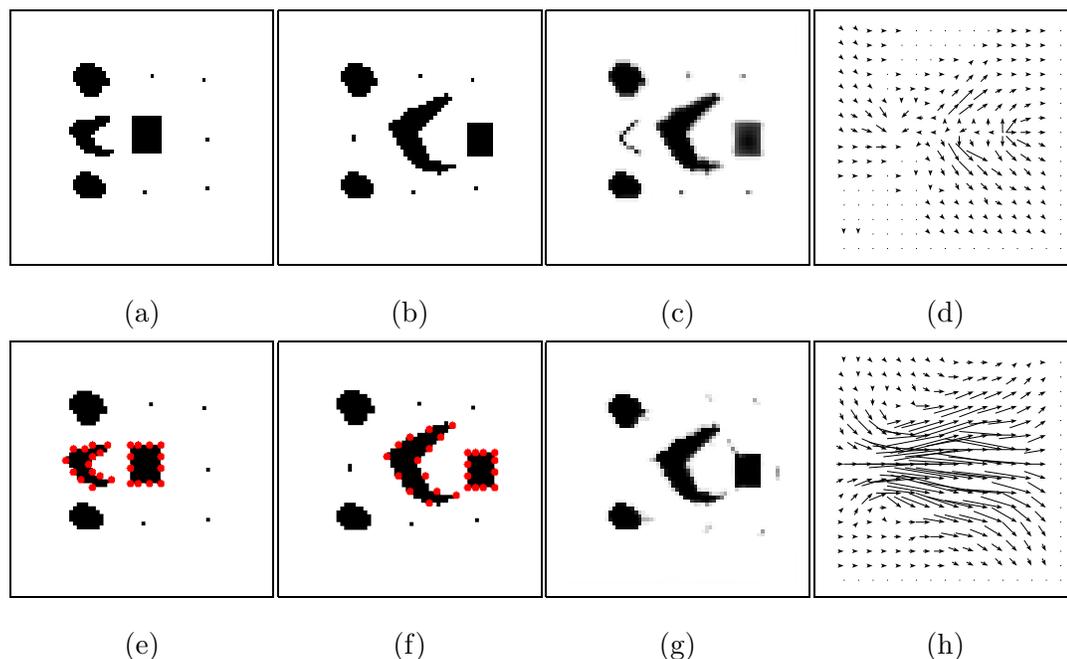


Figure 4.3: Comparison of synthetic multi-object image (64×64) non-rigid registration by Christensen's and our fluid methods. (a): atlas image; (b): study image; (c): final fluidly deformed atlas by Christensen's fluid model [31]; (d): unreasonable vector map of the final viscous-fluid transformation by Christensen's fluid model: objects deform rather than shifting to match shape (compare to (h)); (e): atlas image with control points; (f): study image with control points; (g): final deformed atlas using our non-rigid registration method; (h): vector map of our final fluid transformation showing appropriate displacement of objects (compare to (d)).

high-contrast objects, independent of their importance. Sometimes objects do not deform correctly because their gradient is too low compared to high-gradient objects (Figure 4.4 top) and the smoothness ensured by the physical models dominates the deformation (Figure 4.1, third column top). With the incorporation of the shape information embedded in corresponding boundary points (Figure 4.4 bottom and Figure 4.1 the fourth column), the result is improved (Figure 4.4 bottom and Figure 4.1 the fifth column).

Davatzikos and Prince [40] propose a method that deforms the boundaries in one

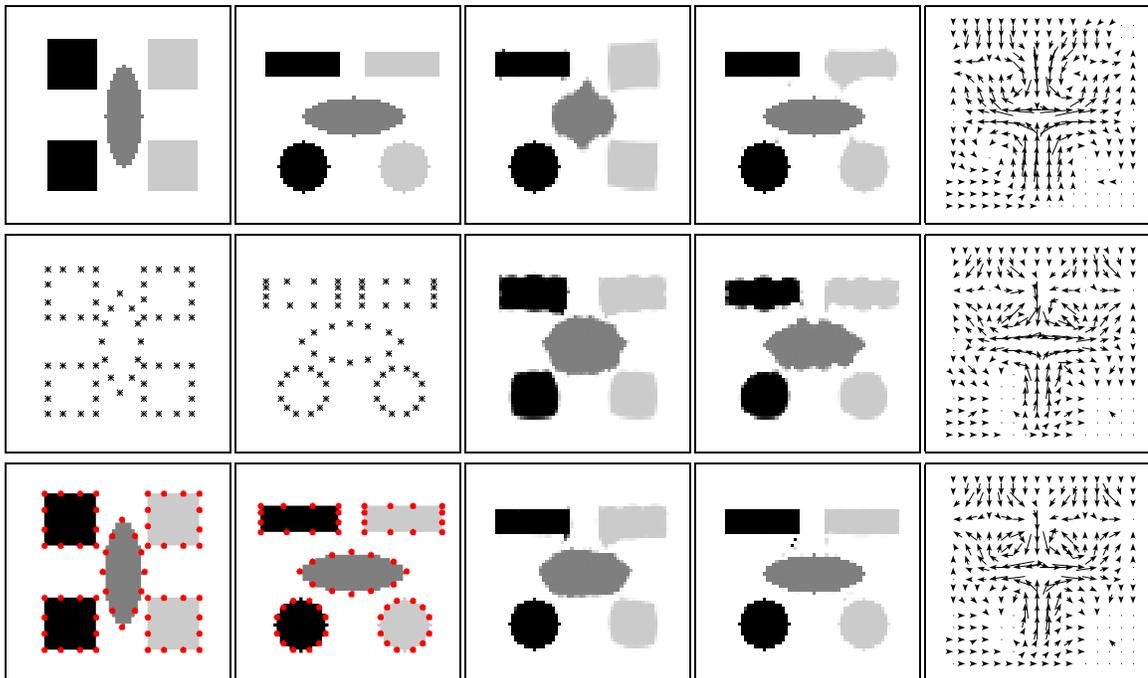


Figure 4.4: Comparison of synthetic multi-gradient image (64×64) non-rigid registration by Christensen&Miller's, Davatzikos' and our elastic methods. Top: results of Christensen&Miller's elastic method [29, 96], and from left to right: atlas image; study image; deformed atlas at iteration 20; deformed atlas at iteration 150; vector map at iteration 150; Middle: results of Davatzikos' elastic method [40] (with body force solely determined by corresponding pairs of control points) and from left to right: atlas image control points; study image control points; deformed atlas at iteration 20; deformed atlas at iteration 150; vector map at iteration 150; Bottom: results of our elastic method, and from left to right: atlas image with control points; study image with control points; deformed atlas at iteration 20; deformed atlas at iteration 150, showing good convergence without any boundary jiggling effect; vector map at iteration 150.

image into those in another image based on a one-to-one mapping they established. The rest of the image is deformed by solving the equations describing the deformation of an elastic body using the boundary deformation as input. In this approach, although the mapping may be accurate on the boundary, the farther away the structure is from the boundary, the more error there is, because only information from object boundaries is used for registration. Also, the localization of the object boundary depends on the density of the boundary control points. The resulting boundaries can have quite severe errors producing a jiggling effect when the boundary control points are not dense enough, as shown in Figure 4.4 middle row. In our approach, we also use intensity information and thus the density of the control points is not as important (Figure 4.4 bottom). In addition, the boundary information used in their approach is derived by an active contour algorithm [40] which does not guarantee true correspondence, and it does not include any shape information which we believe is crucial in non-rigid registration for medical images.

The algorithm we employed is based on a physical model (linear elastic [127] or viscous fluid [128]), a gray level similarity measure and a consistency measure between corresponding boundary points. As shown in Figure 1.3, a dense set of forces arises from the intensity similarity measure to accommodate complex anatomical details. A sparse set of forces constrains consistency with statistical shape models derived from a training set. The statistical shape information is embedded in the boundary finding with correspondence process [126] applied to the study. This method uses statistical point models with shape, and shape variation, generated from sets of examples by principal component analysis of the covariance matrix. The power of physical and statistical shape models are combined in our approach using a Bayesian framework. For small deformations, our elastic model is used because both of our methods would perform similarly and the elastic

model requires less computation. While for large deformations, we use our fluid model-based algorithm which can track long-distance, nonlinear deformations. Also note that all the 2D formulation in this chapter can be directly generalize to 3D.

4.2 Physical Models

There is no true physical model for deformation between individuals because, for example, one individual's anatomical structure does not literally result from the deformation of another individual's. We use analogous physical models to enforce topological properties on the deformation. Without them, the results could be almost completely arbitrary.

4.2.1 Reference Frames and Problem Statement

The task of non-rigid registration is to track the course of every material particle of the atlas from its original to its final position. Each particle can be identified on the basis of either its original or final coordinates. If a particle is tracked with respect to its original coordinates, the reference frame is called *Lagrangian*; otherwise, if the particle is tracked with respect to its final coordinates, the reference frame is called *Eulerian*. In an Eulerian reference frame, the state of the deformed atlas is tracked with respect to a fixed voxel lattice. Therefore, there is no extra processing required to display the deformed atlas or to compare it to a study image. However, in a Lagrangian coordinate system, the trajectories of individual material particles of the atlas are tracked over time. As the atlas deforms the particles being tracked move to locations that do not necessary correspond to pixel locations. Therefore, an extra step is required to project the deformed continuum onto a voxel lattice for comparison with a study and for display.

In this work, both Eulerian and Lagrangian reference frames are used to track the

deformation of the atlas. Our overall systems – physical models (elastic and fluid models) – are Eulerian reference based because of its advantage described above. However, when the statistical shape information which is embedded in the corresponding boundary points is incorporated, we need to track the deformed atlas boundary points position so that a sparse set of forces constrains consistency between corresponding boundary points can be applied. A Lagrangian reference frame is used in this case. We search for the deformed atlas boundary points based on the best consistency with the calculated displacement field.

The non-rigid registration is defined by the transformation corresponding to a homeomorphic mapping of the coordinate system, $\vec{h} : \Omega \rightarrow \Omega$, defined in 2D by

$$\vec{h} : \vec{w} = (x, y) \rightarrow (x - u_x(\vec{w}), y - u_y(\vec{w})) \quad (4.1)$$

for points $\vec{w} \in \Omega$. The vector field $\vec{u}(\vec{w}) = [u_x(\vec{w}), u_y(\vec{w})]^T$ is called the Eulerian displacement field. A time dependent displacement field is denoted by $\vec{u}(\vec{w}, t)$. It is defined as a map from points in the atlas to fixed observation points in the deforming continuum such that a mass particle instantaneously located at \vec{w} at time t originated at point $\vec{w} - \vec{u}(\vec{w}, t)$. The time t can correspond to real or simulation time. This mapping allows for the detailed local transformation into the specific anatomy of the individual, or study.

Note that a homeomorphic transformation is defined to be a continuous, 1-to-1, onto mapping. A necessary condition for a transformation \vec{h} to be globally 1-to-1 is that it is locally 1-to-1, which is satisfied if its Jacobian at any point $\vec{w} \in \Omega$ is positive [27], i.e.

$$J(\vec{w}) = \det(\vec{\nabla} \vec{h}(\vec{w})) > 0 \quad (4.2)$$

where

$$J(\vec{w}) = \begin{bmatrix} \frac{\partial h_x(\vec{w})}{\partial x} & \frac{\partial h_x(\vec{w})}{\partial y} \\ \frac{\partial h_y(\vec{w})}{\partial x} & \frac{\partial h_y(\vec{w})}{\partial y} \end{bmatrix}. \quad (4.3)$$

Our goal here is to find the transformation \vec{h} that best matches the atlas with the study, constrained by the physical models. The following formulations of the physical models are similar to that in Christensen *et al.* [29].

4.2.2 Elastic Model

For linear elastic solids, the restoring force holding the template together grows proportionately with the displacement from the original configuration of the atlas. The force is proportional to the displacement. The spatial transformation satisfies the partial differential equation (PDE):

$$\mu \nabla^2 \vec{u} + (\mu + \beta) \vec{\nabla}(\vec{\nabla} \cdot \vec{u}) = \vec{F}(\vec{u}) \quad (4.4)$$

with boundary conditions such as that $\vec{u}(\vec{w}) = 0$ for \vec{w} on the image boundary. In this equation, the divergence and Laplacian operators are:

$$\vec{\nabla} = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right)^T \quad (4.5)$$

and

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad (4.6)$$

and μ and β are the Lamé constants. Normally, we choose them with $\mu = 1.0$ and $\beta = 0.0$. However, in order to guarantee a homeomorphic transformation for large deformations, we use large μ for strong elasticity. Also note that $\beta = 0.0$ corresponds to the elastic solid whose volume is not conserved during the deformation. The body force, $\vec{F}(\vec{u})$, drives the deformation of the atlas into the study, and will be formulated in detail in Section 4.3.

4.2.3 Fluid Model

Transformations such as those based on the theory of elasticity develop restoring forces which are proportional to the deformed distances. Except for the smallest deformations, such elastic transformations prevent the atlas from being fully deformed into the shape of the study. The short-comings of the elasticity model can be overcome by a viscous fluid which allows the restoring forces to relax over time.

For viscous fluids, the force is proportional to the time rate of change in displacement. The PDE describing the fluid transformation of the atlas is given by:

$$\mu \nabla^2 \vec{v} + (\mu + \beta) \vec{\nabla}(\vec{\nabla} \cdot \vec{v}) = \vec{F}(\vec{u}) \quad (4.7)$$

where $\vec{v} = [v_x(\vec{w}, t), v_y(\vec{w}, t)]^T$ is the instantaneous velocity of the deformation field \vec{u} . It is related to its displacement, \vec{u} , by

$$\vec{v}(\vec{w}, t) = \frac{\partial \vec{u}(\vec{w}, t)}{\partial t} + \vec{v}(\vec{w}, t)^T \vec{\nabla} \vec{u}(\vec{w}, t) \quad (4.8)$$

The $\nabla^2 \vec{v}$ term is the viscous term of the PDE. This term constrains the velocity of neighboring particles of the displacement field to vary smoothly. The term $\vec{\nabla}(\vec{\nabla} \cdot \vec{v})$ is the mass source term and it allows structures in the atlas to change in mass. The coefficients

μ and β are the viscosity coefficients, and they are chosen as $\mu = 1.0$, $\beta = 0.0$, where β controls the rate of growth or shrinkage of local region within the deformable atlas. For the work herein, the boundary conditions are $\vec{v}(\vec{w}) = 0$ for \vec{w} on the image boundary. The body force, $\vec{F}(\vec{u})$, will be formulated in detail in Section 4.3. The term $\vec{v}(\vec{w}, t)^T \vec{\nabla} \vec{u}(\vec{w}, t)$ accounts for the kinematic nonlinearities of the displacement field \vec{u} [27, 89].

4.2.4 Viscous Fluid vs. Linear Elasticity

The fluid PDE (Equation (4.7)) is similar in form to the elastic PDE (Equation (4.4)) except that the displacement field \vec{u} is replaced by the velocity field \vec{v} . The resulting behavior of the fluid is very different due to the nonlinear relationship between \vec{v} and \vec{u} (Equation (4.8)) and allows long-distance, nonlinear deformations.

The linear elastic model is derived assuming small angles of rotation and small linear deformations. Large deformations can not be accommodated with this linear PDE. However, even though linear elasticity does not guarantee a homeomorphic transformation, in practice a homeomorphic transformation can be generated using strong elasticity (large μ). The trade-off is that only small deformations can be generated [31]. This limitation of linear elasticity is removed by using the viscous model because the restoring forces relax over time and then account for the large-distance kinematic nonlinearities, while ensuring a homeomorphic transformation (globally positive Jacobian). A detailed comparison of the elastic and fluid models can be found in [27].

Also, note that it might be possible to use a more complex elastic model instead of a fluid model to allow large deformation while ensuring a homeomorphic transformation. However, it is not the goal in this work.

4.3 Integration

While physical models are useful in non-rigid registration, they are limited by themselves because they are too generic. Instead, the statistics of a sample of images can be used to guide the deformation in a way governed by the measured variation of individuals. Thus, this work proposes algorithms which use physical models, yet incorporate a statistical shape model to constrain solutions to more anatomically consistent deformations.

We pose the displacement estimation problem in a maximum *a posteriori* framework. As input to the problem, we have both the intensity image of the study (individual), $I_s(\vec{w})$, and the boundary points of the study $\vec{b}_s(\mathbf{p}, n) = (x_s(\mathbf{p}, n), y_s(\mathbf{p}, n))$, for $n = 1, 2, \dots, N$, given the shape and pose parameters, \mathbf{p} , which are derived from the statistical shape model-based boundary finding described in Chapter 3. Thus, we want to maximize:

$$\Pr(\vec{u}|I_s, \vec{b}_s(\mathbf{p})) = \frac{\Pr(\vec{u}, I_s, \vec{b}_s(\mathbf{p}))}{\Pr(I_s, \vec{b}_s(\mathbf{p}))} \quad (4.9)$$

Ignoring the denominator, which does not change with \vec{u} , and by using Bayes rule, our aim is to find:

$$\begin{aligned} & \arg \max_{\vec{u}} \Pr(\vec{u}|I_s, \vec{b}_s(\mathbf{p})) \\ & \equiv \arg \max_{\vec{u}} \Pr(\vec{b}_s(\mathbf{p})|\vec{u}, I_s) \Pr(I_s|\vec{u}) \Pr(\vec{u}) \\ & \propto \arg \max_{\vec{u}} \left[\ln \Pr(\vec{u}) + \ln \Pr(I_s|\vec{u}) + \ln \Pr(\vec{b}_s(\mathbf{p})|\vec{u}) \right] \end{aligned} \quad (4.10)$$

where we ignore the dependence of $\vec{b}_s(\mathbf{p})$ on I_s (because $\vec{b}_s(\mathbf{p})$ is obtained as a prior here and is not modified in this formulation), and take the logarithm.

The Bayesian posterior can be directly connected to the PDE in Equation (4.4) or Equations (4.7)&(4.8) based on a variational principle from which the PDE can be derived [31]. Such principles are well known in mechanics [106] and link the PDE formulation as the minimizer of some potential. For example, the PDE for the linear elastic model, which is given in Equation (4.4), is produced by setting the variation of the *generalized Lagrangian* energy density associated with constraints imposed by the linearized mechanics equal to zero [31]. The forcing function in the PDE (Equation (4.4)) is then the variation of the likelihood function with respect to the vector displacement field [3] [96].

The first term in Equation (4.10) corresponds to the transformation prior term, which gives high probability to transformations consistent with a physical model (elastic solid or viscous fluid) and low probability to all other transformations. As mentioned above, they are given by Equation (4.4) for the elastic model, and Equations (4.7)&(4.8) for the fluid model.

The second term in Equation (4.10) represents the likelihood which depends on the study image. Let $I_a(\vec{w})$ be the intensity image of the atlas. We model the study image as a Gaussian process with mean given by the deformed atlas image, $I_a(\vec{w} - u(\vec{w}))$ [31] (since an Eulerian reference frame is used here, a mass particle instantaneously located at \vec{w} originated from point $\vec{w} - \vec{u}(\vec{w})$). That is,

$$\ln \Pr(I_s | \vec{u}) = -\frac{1}{2\sigma_1^2} \int_{\Omega} [I_s(\vec{w}) - I_a(\vec{w} - u(\vec{w}))]^2 d\vec{w}$$

where σ_1 is the standard deviation of the Gaussian process.

The first body force, \vec{F}_1 , is the gradient of this likelihood term with respect to \vec{u}

at each \vec{w} [31]:

$$\vec{F}_1(\vec{u}) = -\frac{1}{\sigma_1^2} [I_s(\vec{w}) - I_a(\vec{w} - u(\vec{w}))] \nabla I_a(\vec{w} - u(\vec{w})) \quad (4.11)$$

This force is a combination of $I_s(\vec{w}) - I_a(\vec{w} - u(\vec{w}))$, the difference in intensity between the study and the deformed atlas, and $\nabla I_a(\vec{w} - u(\vec{w}))$, the gradient of the deformed atlas. The gradient term determines the directions of the local deformation forces applied to the atlas. As explained in the introduction, this kind of forcing by itself is often under-constrained.

The main contribution of this work lies in the last term of Equation (4.10), which incorporates statistical shape information into the non-rigid registration framework. The extra constraint of corresponding boundary points is used as an additional matching criterion. The boundary point positions are the result of the deformation of the model to fit the data in ways consistent with the statistical shape models derived from the training set, as described in Chapter 3. Let $\vec{b}_a(n) = (x_a(n), y_a(n))$, for $n = 1, 2, \dots, N$, denote the atlas boundary points positions, which are known since we have full information about the atlas. We now model $\vec{b}_s(\mathbf{p})$ as a Gaussian process with mean given by the deformed atlas boundary position, expressed as $\vec{b}_a(n) + \vec{u}(\vec{w})$, for pixels \vec{w} on the deformed atlas boundary points. Then,

$$\ln \Pr(\vec{b}_s(\mathbf{p}) | \vec{u}) = -\frac{1}{2\sigma_2^2} \sum_{n=1}^N \| \vec{b}_s(\mathbf{p}, n) - [\vec{b}_a(n) + \vec{u}(\vec{w})] \|^2$$

where σ_2 is again the standard deviation of the Gaussian process.

The second body force, \vec{F}_2 , is then the gradient of the above equation with respect

to \vec{u} for pixels \vec{w} on the deformed atlas boundary points:

$$\vec{F}_2(\vec{u}) = \frac{1}{\sigma_2^2} \|\vec{b}_s(\mathbf{p}, n) - [\vec{b}_a(n) + \vec{u}(\vec{w})]\| \quad (4.12)$$

$\vec{F}_2(\vec{u})$ is zero for pixels \vec{w} not on the deformed atlas boundary points.

From Equation (4.12), we can see that the calculated displacements at the sparse boundary points are constrained to match the vector difference of the corresponding atlas and study boundary point positions. This kind of forcing contains information from the statistical shape model. The result will match shape features of the atlas and the study, such as high curvature points and important anatomical landmarks, in addition to the intensity measure given by $\vec{F}_1(\vec{u})$.

The total force term, $\vec{F}(\vec{u})$, in Equation (4.4) and Equation (4.7) is then the weighted sum of $\vec{F}_1(\vec{u})$ in Equation (4.11) and $\vec{F}_2(\vec{u})$ in Equation (4.12), that is, for each \vec{w} ,

$$\vec{F}(\vec{u}) = c_1 \vec{F}_1(\vec{u}) + c_2 \vec{F}_2(\vec{u}) \quad (4.13)$$

The two coefficients, c_1 and c_2 , can be related to the image contrast and the deformation between the atlas and study image. If c_2 is too large, $\vec{F}_2(\vec{u})$ will play a dominant role by matching only boundary points, which may cause discontinuity when the boundary points are not dense enough. On the other hand, if c_1 is too large, $\vec{F}_1(\vec{u})$ will have almost no effect and the algorithm is then an elastic or a fluid regularization method (as [96, 31]) without statistical information. For the time being, they are fixed empirically so that $\vec{F}_1(\vec{u})$ and $\vec{F}_2(\vec{u})$ are of the same order.

Our approaches that incorporate statistical shape information are then composed

of Equations (4.13)&(4.4) for the elastic model, and Equations (4.13),(4.7)&(4.8) for the fluid model.

4.4 Implementation

4.4.1 Linear Elastic Model Algorithm — Algorithm 1

The complete algorithm for solving our elastic model-based non-rigid registration is as follows:

Algorithm 1:

1. From boundary and correspondence finding methods developed in Chapter 3 calculate $\vec{b}_s(\vec{p})$ by using $C_{training}$ in Equation (3.3) or C_{mix} in Equation (3.14).
2. Initialize $t = 0$ and $\vec{u}(\vec{w}, 0) = 0$.
3. Calculate the body force $F(\vec{w}, \vec{u}(\vec{w}, t))$ using Equation (4.13).
4. If $F(\vec{w}, \vec{u}(\vec{w}, t))$ is below a threshold for all w , then STOP.
5. Solve the linear PDE Equation (4.4) for the displacement $\vec{u}(\vec{w}, t)$ using Successive over-relaxation algorithm (SOR) (see Section 4.4.3).
6. Let $t = t + 1$, and goto step 3.

4.4.2 Viscous Fluid Model Algorithm — Algorithm 2

The viscous fluid PDE is solved on a discrete lattice associated with Ω . For large curved deformations, the transformation evaluated at these spatial grid points becomes singular over time even though the transformation evaluated on the continuum would not. To circumvent this problem, a method of regriding the atlas is used by generating a new, or propagated atlas, whenever the discretized transformation approaches local singularity. This is accomplished as follows. The discretized displacement field is propagated through

time using a discretized version of Equation (4.8) until the Jacobian of the transformation indicates that the transformation is approaching singularity. When the magnitude of the Jacobian drops below a threshold (0.5 for the results shown later), the computation is stopped and a new propagated atlas is generated equal to the deformed atlas at the previous instant. The algorithm is restarted using the new atlas. When mapped back to the original atlas, this regridding approach is equivalent to defining a new nonhomogeneous computation grid on the undeformed continuum. Initial conditions for the propagated atlas are set to match the final state of the previous atlas, i.e., the instantaneous velocity remains the same and the displacement field corresponding to the new atlas is set to zero. The total transformation is continuously tracked via the concatenation of the displacement fields associated with all of the propagated atlases. Details about this regridding can be found in [27].

Thus, the complete algorithm for solving the viscous fluid registration is as follows:

Algorithm 2:

1. Same as step 1 in **Algorithm 1**.
2. Let $t = 0$, $i = 0$, $I_a^{(0)}(\vec{w}) = I_a(\vec{w})$, and $\vec{u}^{(0)}(\vec{w}, 0) = 0$. (Note: $I_a^{(i)}(\vec{w})$ for $i = 0, 1, 2, \dots$ denotes the sequence of propagated atlases with associated displacement field $\vec{u}^{(i)}(\vec{w}, t)$.)
3. Calculate the body force $F(\vec{w}, \vec{u}^{(i)}(\vec{w}, t))$ using Equation (4.13).
4. If $F(\vec{w}, \vec{u}^{(i)}(\vec{w}, t))$ is below a threshold for all w , then STOP.
5. Solve the linear PDE Equation (4.7) for instantaneously velocity $\vec{v}(\vec{w}, t)$ using Successive over-relaxation algorithm (SOR) (see Section 4.4.3).

6. Calculate the perturbation of the displacement field

$$\vec{R}(\vec{w}) = \vec{v}(\vec{w}, t) - \left[v_x(\vec{w}, t) \frac{\partial \vec{u}^{(i)}(\vec{w}, t)}{\partial x} + v_y(\vec{w}, t) \frac{\partial \vec{u}^{(i)}(\vec{w}, t)}{\partial y} \right]$$

7. Choose a real-time step size Δ which is a function of $\gamma = \max_{\vec{w} \in \Omega} \|\vec{R}(\vec{w})\|$.

(In practice, Δ is chosen so that the maximum displacement at each instant of real-time is 0.5.)

8.

- If the Jacobian of $\vec{w} - \vec{u}^{(i)}(\vec{w}, t) - \Delta \vec{R}(\vec{w})$ is less than 0.5 then propagate to atlas $i + 1$

$$- I_a^{(i+1)}(\vec{w}) = I_a^{(i)}(\vec{w} - \vec{u}(\vec{w}, t)), \quad \vec{u}^{(i+1)}(\vec{w}, t) = 0, \quad \text{and } i = i + 1;$$

- go to step 6.

- Else, update the i^{th} displacement field

$$- \vec{u}^{(i)}(\vec{w}, t + \Delta) = \vec{u}^{(i)}(\vec{w}, t) + \Delta \vec{R}(\vec{w}) \quad \text{and } t = t + \Delta;$$

- go to step 3.

4.4.3 Successive Over-Relaxation (SOR) Algorithm

Successive over-relaxation (SOR) [18] is used to solve Equation (4.4) for the displacement $u(\vec{w}, t)$ and Equation (4.7) for the instantaneous velocity $v(\vec{w}, t)$ of the displacement field at each grid point $\vec{w} \in \Omega$. The SOR formula used for solving Equation (4.4) at time t with grid spacing Δ is given by

$$u_x^{n+1}(i, j) = (1 - \omega)u_x^n(i, j) + \frac{\omega}{6\mu + \beta} \left\{ (2\mu + \beta) \left[u_x^n(i + 1, j) + u_x^{n+1}(i - 1, j) \right] \right\}$$

$$\begin{aligned}
& +\mu \left[u_x^n(i, j+1) + u_x^{n+1}(i, j-1) \right] \\
& + \frac{\mu + \beta}{4} \left[u_y^n(i+1, j+1) - u_y^n(i-1, j+1) \right. \\
& \quad \left. - u_y^n(i+1, j-1) + u_y^{n+1}(i-1, j-1) \right] - \Delta^2 F_x(i, j) \} \\
u_y^{n+1}(i, j) = & (1 - \omega)u_y^n(i, j) + \frac{\omega}{6\mu + \beta} \left\{ (2\mu + \beta) \left[u_y^n(i, j+1) + u_y^{n+1}(i, j-1) \right] \right. \\
& + \mu \left[u_y^n(i+1, j) + u_y^{n+1}(i-1, j) \right] \\
& + \frac{\mu + \beta}{4} \left[u_x^n(i+1, j+1) - u_x^n(i-1, j+1) \right. \\
& \quad \left. - u_x^n(i+1, j-1) + u_x^{n+1}(i-1, j-1) \right] - \Delta^2 F_y(i, j) \} \quad (4.14)
\end{aligned}$$

The ij in the brackets denote the ij^{th} pixel location in the study image, the superscript n denotes the relaxation iteration.

The SOR formula used for solving Equation (4.7) is similar to the above equation (Equation (4.14)) in format except that u_x is replaced by v_x and u_y is replaced by v_y .

4.4.4 Different Methods and Computation Time Comparison

For all of the experiments, we apply either Christensen&Miller's elastic registration [96, 29] or Christensen's fluid registration [31] for a direct comparison based on our own implementation. As to the computation time, while our method requires an extra force (\vec{F}_2) calculation at sparse boundary points, this leads to fast and accurate convergence. Also, since the boundary finding step takes only several seconds, the total convergence time of our elastic and fluid methods are usually a little faster than or similar to Christensen's elastic and fluid methods respectively. In general, the elastic methods take much less time than the fluid methods since no time integration is needed¹. An example

¹A detailed execution time investigation of Christensen's elastic and fluid models can be found in [29].

comparison of the different methods' approximate execution times on a Silicon Graphics Octane 250-MHZ MIPS R10000 is listed in Table 4.3.

4.5 Experimental Results

In addition to the demonstration of the previous synthetic image non-rigid registration methods (Figures 4.1, 4.2, 4.3 and 4.4), we also give the following quantitative validation of our methods.

4.5.1 Evaluation Criterion

To evaluate the methodology, we quantify errors in the displacement field over the objects of interest, since warping of the background is irrelevant.

Given a known warp, we can measure detailed displacement errors throughout the object. For testing purposes, we can define a particular warp and apply it to an image, generating a warped study image to which the algorithm can be applied. We use either of the following sinusoidal displacement fields for transforming the atlas image to a study image (see Figure 4.5 top right and Figure 4.9 top right respectively):

$$x_{new} = x_{old} + A_x \sin(\pi x_{old}/32); \quad y_{new} = y_{old} + A_y \sin(\pi y_{old}/32) \quad (4.15)$$

and

$$x_{new} = x_{old} + A_x \sin(\pi y_{old}/32); \quad y_{new} = y_{old} + A_y \sin(\pi x_{old}/32) \quad (4.16)$$

where x_{old} and y_{old} are coordinates of a point in the atlas image and x_{new} and y_{new} are

coordinates of the corresponding point in the transformed study image. A_x and A_y are the limits of the maximum displacement distances along the x and y directions.

For a known non-rigid warp, the average (E_{oa}) and maximum (E_{om}) differences between the estimated and actual displacement vectors over the objects are used to measure accuracy. We also use the average difference between the estimated and actual displacement vectors on the sparse boundary control points, E_{ba} . Since the control points are also derived from the known warp, all three measures only reflect the non-rigid registration, and do not include the boundary finding step. For true image pairs, we only use E_{ba} as an accuracy measure because we do not know the true warp, except at sparse boundary points determined by an expert. Since the study image boundary control points are derived by our statistical shape model-based boundary finding, the error, E_{ba} , for true image pairs includes both the boundary finding step and the non-rigid registration step.

4.5.2 Synthetic Images with Known Warping

The experiments in this subsection test the robustness of our elastic method compared with Christensen&Miller’s elastic method. The synthetic study image, Figure 4.5 top middle, is obtained by resampling the synthetic atlas image, Figure 4.5 top left, based on the predetermined displacement vectors (Figure 4.5 top right, Equation (4.15)). The study control points are also derived from the same predetermined displacement vectors. The atlas image is then registered to the synthetic study image using our image registration procedure. Although the resulting deformed atlases for our method and Christensen&Miller’s method are similar, the estimated displacement vectors are not. From the errors in the estimated vectors (differences between the estimated and true displacement vectors), we can see that our method has almost zero error over the objects, while there is significant error in their approach.

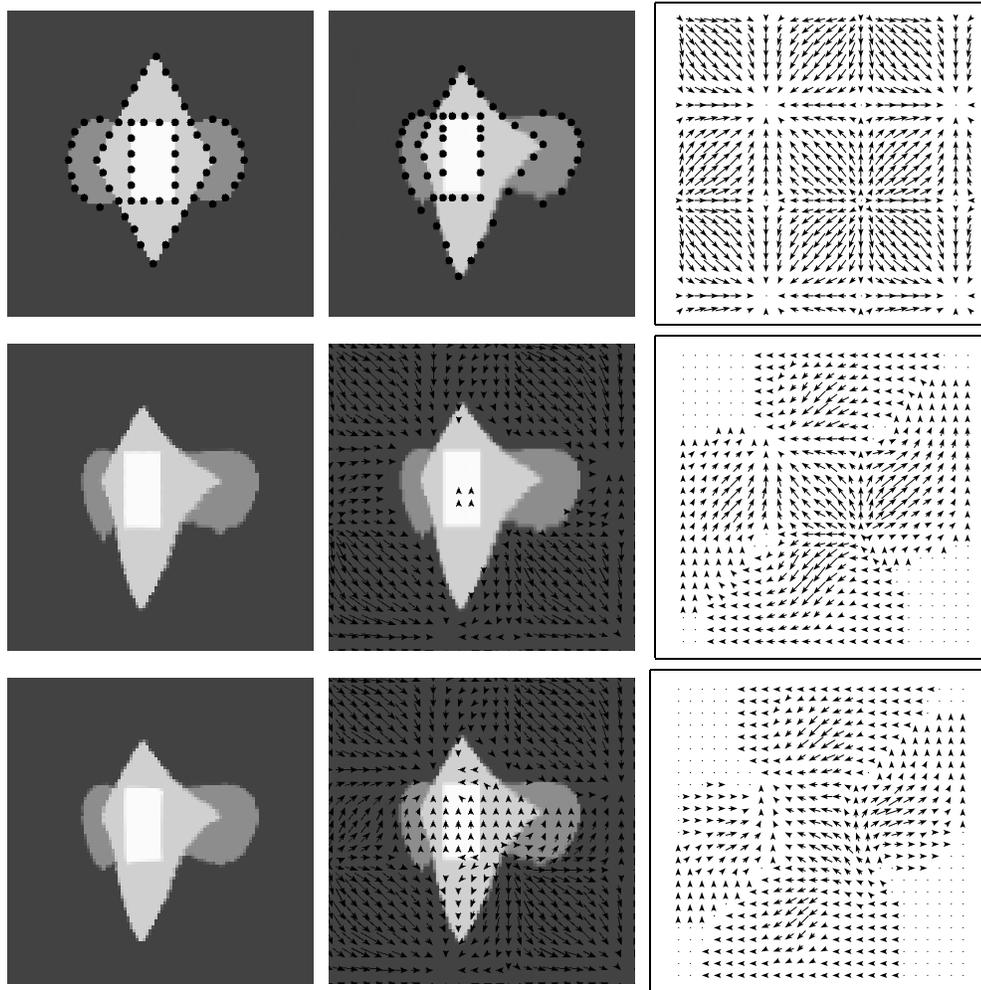


Figure 4.5: Synthetic images (100×100) and displacement vectors by our and Christensen&Miller's elastic methods. Top left: atlas image with its control points; Top middle: study image with its control points; Top right: true displacement vectors by Equation (4.15); The second and third row: The left column shows the deformed atlas by our elastic method (the second row) and by Christensen&Miller's elastic method (the third row); The middle column shows the corresponding errors in the estimated vectors on the study image, showing near perfect displacement for our method (second row); The right column shows the corresponding estimated vectors.

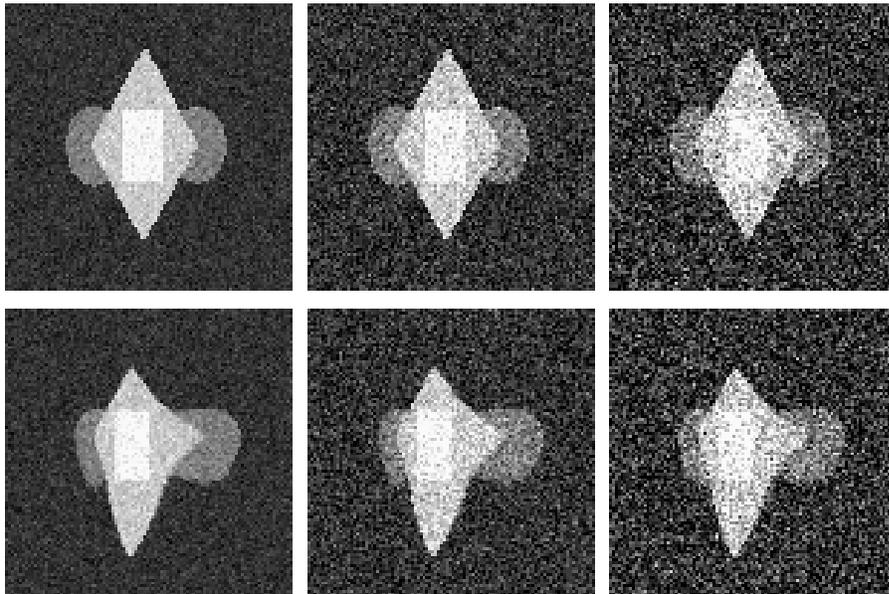


Figure 4.6: Noisy images for synthetic image sensitivity-to-noise experiment by our and Christensen&Miller’s elastic methods. Left to right, atlas image (top) and study image (bottom) from Figure 4.1 with zero mean Gaussian noise added with standard deviation of 15.0, 30.0, 45.0, respectively. Note: the intensity range in the atlas and study images in Figure 4.5 is 50 to 250.

The following experiment, shown in Figures 4.6 and 4.7, demonstrates the effect of noise on the two methods, by adding varying amounts of zero mean Gaussian noise to the synthetic images shown in Figure 4.5 top left and Figure 4.5 top middle. The error measures defined above are computed for our method ($E1_{oa}$, $E1_{om}$, $E1_{ba}$) and Christensen&Miller’s elastic approach ($E2_{oa}$, $E2_{om}$, $E2_{ba}$) and show consistent improvement for our method.

4.5.3 Real Image with Known Warping

In this section, we first apply a known warping (Figure 4.8 top right, Equation (4.15)) to a magnetic resonance (MR) sagittal brain image showing the corpus callosum to compare

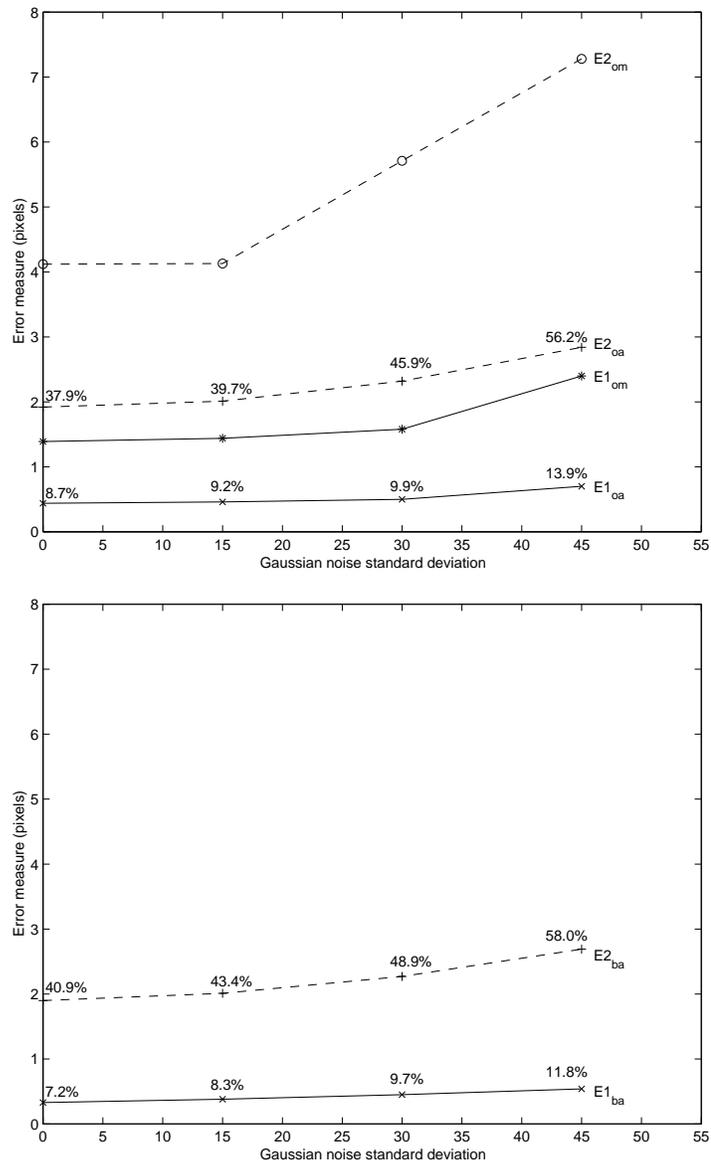


Figure 4.7: Sensitivity to noise experiment for synthetic images by our and Christensen&Miller's elastic methods. E_{om} : maximum displacement error over the deformed objects; E_{oa} : average displacement error over the deformed objects; E_{ba} : average displacement error on the sparse boundary points. Note: The error for our method ($E1$) is significantly better than that for Christensen&Miller's elastic method ($E2$); the percentages shown are the percentages of the average errors (E_{oa} , E_{ba}) relative to the true average displacements.

the performance of our elastic method with that of Christensen&Miller's elastic method. While the deformed atlases appear similar, the results (Fig.4.8 and Table 4.1) show that our elastic method leads to a much better registration in the object of interest than Christensen&Miller's elastic method.

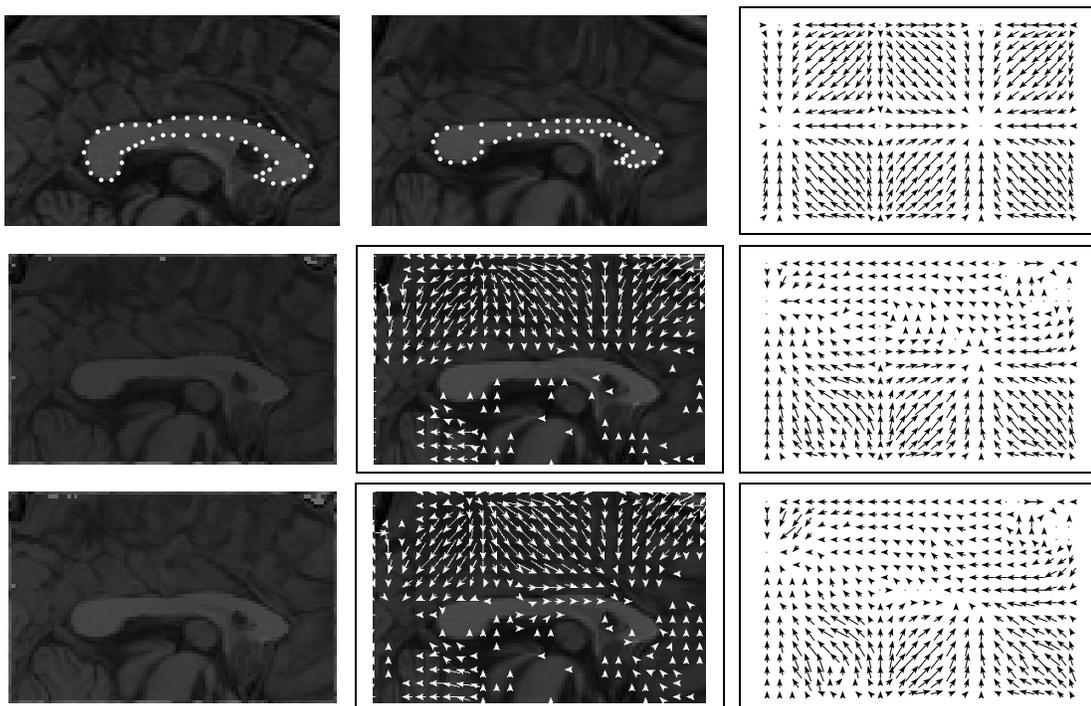


Figure 4.8: MR sagittal corpus callosum images (100×64) and synthetic displacement vectors by our and Christensen&Miller's elastic methods. Top left: atlas image with control points; Top middle: study image with control points; Top right: true displacement vectors by Equation (4.15); Second row left: our deformed atlas image; Second row middle: errors in our estimated vectors on study image; Second row right: our estimated vectors; Bottom left: deformed atlas image by Christensen&Miller's elastic method; Bottom middle: errors in their estimated vectors on study image. Bottom right: their estimated vectors.

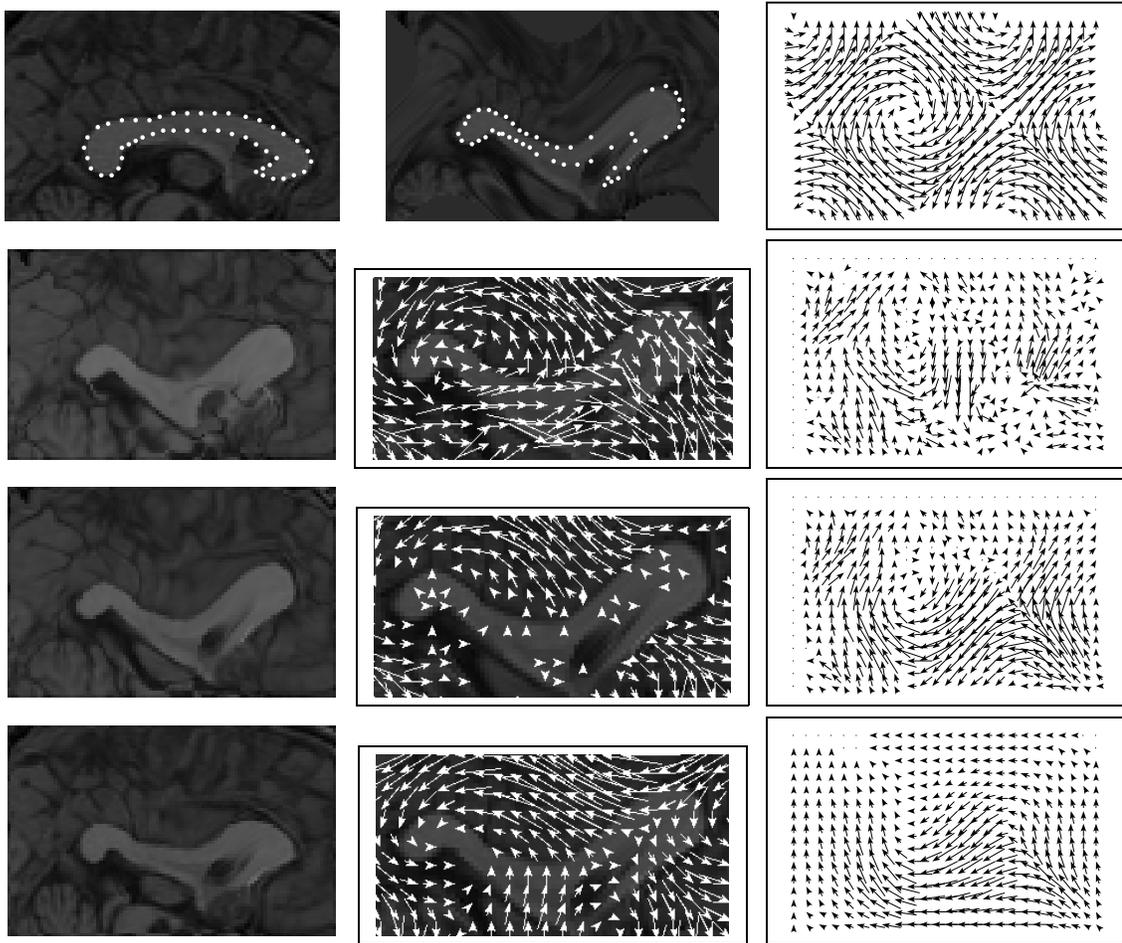


Figure 4.9: MR sagittal corpus callosum image (100×64) and synthetic displacement vectors by Christensen's and our fluid methods, and our elastic method. Top left: atlas image with control points; Top middle: study image with control points; Top right: true displacement vectors by Equation (4.16); From the second row to the bottom: The left column shows the deformed atlas by Christensen's fluid method (the second row), our fluid method (the third row), and our elastic method (bottom); The middle column shows the corresponding errors in estimated vectors on the study image (cropped), with significantly smaller error for our fluid method (third row); The right column shows the corresponding estimated vectors.

| <i>Method</i> | E_{oa} (%) | E_{om} | E_{ba} (%) |
|-------------------------------------|---------------------|-------------|---------------------|
| Christensen&Miller’s elastic method | 1.26 pixels (30.0%) | 3.56 pixels | 1.51 pixels (34.7%) |
| Our elastic method | 0.48 pixels (11.5%) | 1.33 pixels | 0.45 pixels (10.3%) |

Table 4.1: Error measure for MR sagittal corpus callosum image with known warping (Figure 4.8) showing the improvement of our elastic method. E_{oa} : average displacement error over corpus callosum; E_{om} : maximum displacement error over corpus callosum; E_{ba} : average displacement error on sparse boundary points. Note: the percentages shown with each average error are with respect to the true average displacement.

| <i>Method</i> | E_{oa} (%) | E_{om} | E_{ba} (%) |
|----------------------------|---------------------|--------------|---------------------|
| Christensen’s fluid method | 5.82 pixels (63.2%) | 13.82 pixels | 5.91 pixels (65.0%) |
| Our fluid method | 0.99 pixels (11.4%) | 2.29 pixels | 0.89 pixels (9.8%) |
| Our elastic method | 3.46 pixels (37.6%) | 10.54 pixels | 3.32 pixels (36.5%) |

Table 4.2: Error measure for MR sagittal corpus callosum image with known warping (Figure 4.9) showing the improvement of our fluid method. E_{oa} : average displacement error over corpus callosum; E_{om} : maximum displacement error over corpus callosum; E_{ba} : average displacement error on sparse boundary points. Note: the percentages shown with each average error are with respect to the true average displacement.

In this experiment (Figure 4.9), we apply another known warping (Figure 4.9 top right, Equation (4.16)) to a magnetic resonance (MR) sagittal brain image showing the corpus callosum. Because the deformation here between the atlas and study is quite large, we use our fluid method to register the atlas image to the study image. For comparison, we also register the two images using Christensen’s fluid method and our elastic method by choosing the appropriate elasticity parameter (large μ) so that a homeomorphic map is just ensured (globally positive Jacobian). The results (Figure 4.9 and Table 4.2) show

that our fluid method leads to a much better registration in the object of interest than Christensen&Miller’s fluid method, and our elastic method because of the large deformation of the images which can not be tracked by the elastic model. Using our elastic method, the atlas can not deform well (Figure 4.9 bottom left).

4.5.4 Real Atlas and Study Images

Results of the method applied to MR brain (axial) and heart image pairs are shown in Figures 4.11 and 4.12. These examples show 2D slices that roughly correspond from different brains and hearts for demonstration purposes. The control points of the study image are derived from statistical shape model-based boundary finding algorithm [126]. The shape model for the brain examples (Figure 4.10) incorporates multiple objects and thus also models the distance between the objects. The shape model used for the heart examples is shown in Figure 3.2. From the error measures shown in Table 4.3, we see that even with the error in the boundary finding step, the final error of our methods is still much better than Christensen’s methods. Specifically, for brain images, the final error of our elastic and fluid methods perform similarly because for small deformations both of our methods work well, and much better than Christensen&Miller’s elastic method. Note in particular, in Figure 4.11 the corner of the third ventricle in the study was not registered to the atlas correctly by Christensen&Miller’s elastic method (Figure 4.11(h)). The structures of the study are shifted away from the corresponding ones in the atlas based on gray level information. Our methods calculated the correct mapping (Figures 4.11(i) and 4.11(j)) by incorporating statistical shape information and using the corresponding boundary points as an extra constraint. Also, note that the putamen did not deform well in Christensen&Miller’s elastic method (Figure 4.11(k)) because the contrast of the putamen is too low compared to the contrast of the ventricles. In our approaches, the

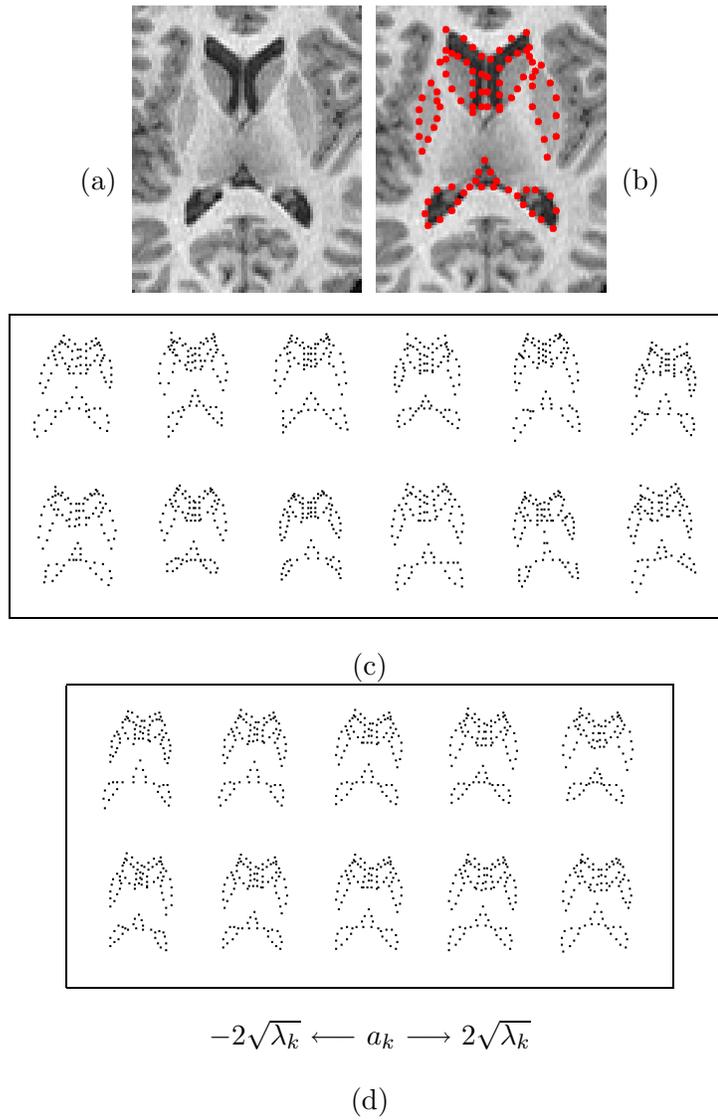


Figure 4.10: Axial brain shape model. (a): MR brain image (80×100); (b): 93 point model of basal ganglia and ventricle boundaries; (c): 12 examples of brain shapes from a training set; (d): effects of varying each of the first two shape parameters of the brain model.

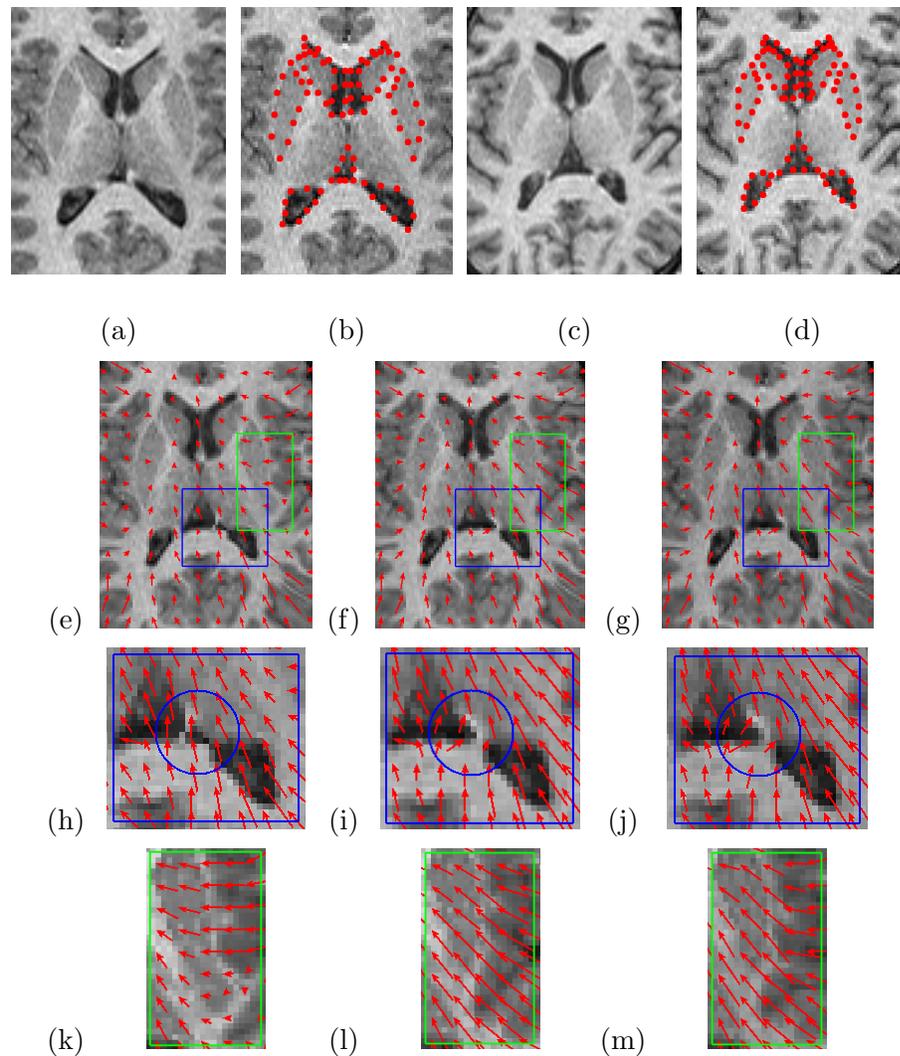


Figure 4.11: MR axial brain images (80×100) and displacement vectors. (a): atlas image; (b): atlas image with its control points; (c): study image; (d): study image with its control points derived from our boundary finding algorithm (Chapter 3); (e): estimated vectors by Christensen&Miller's elastic method over their deformed atlas; (f): our elastically estimated vectors over our elastically deformed atlas; (g): our fluidly estimated vectors over our fluidly deformed atlas; (h): enlargement of (e) showing mis-matching due to Christensen&Miller's elastic method; (i): correct mapping of the ventricle corners by our elastic method; (j): correct mapping of the ventricle corners by our fluid method; (k): poorly deformed putamen by Christensen&Miller's elastic method (cropped); (l): well deformed putamen by our elastic method; (m): well deformed putamen by our fluid method.

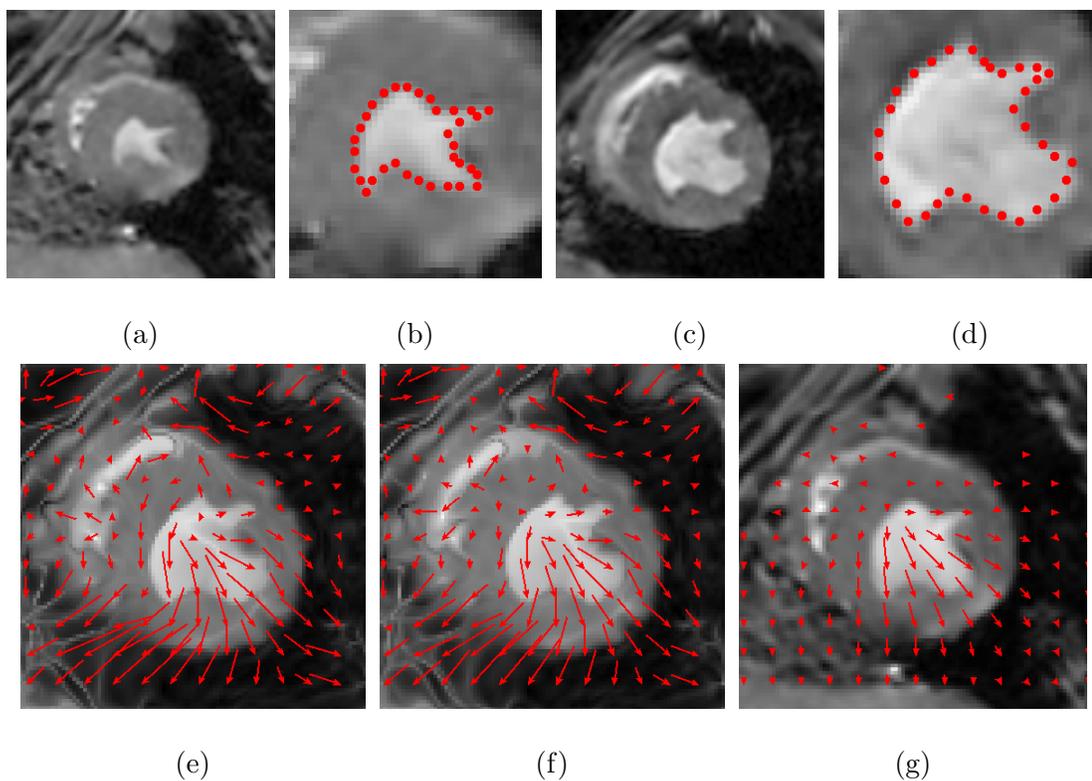


Figure 4.12: MR heart images (100×100) and displacement vectors. (a): atlas image; (b): atlas image with its control points on the endocardium (cropped); (c): study image; (d): study image with control points derived from our boundary finding algorithm; (e): estimated vectors by Christensen's fluid method over their deformed atlas; (f): vectors by our fluid method over our fluidly deformed atlas; (g): vectors by our elastic method over our elastically deformed atlas (appropriate elasticity parameter μ is used to just ensure globally positive Jacobian).

putamen deformed correctly (Figures 4.11(l) and 4.11(m)) since shape information of the putamen was included. For the heart images, because of the large deformation, our fluid model works better than our elastic model, by which the atlas can not deform well (Figure 4.12(g)). More importantly, our fluid method results in much smaller error than Christensen’s fluid method because of the included statistical shape information.

| <i>Image</i> | <i>Method</i> | E_{ba} (%) pixels (%) | <i>Time</i> (minutes) |
|---|-------------------------------------|----------------------------|--------------------------|
| MR Brain (80×100) with 93 control points (Figure 4.11) | Christensen&Miller’s elastic method | 2.04 (43.7%) | 15 |
| | Our elastic method | 0.75 (16.0%) | 13 |
| | Our fluid method | 0.76 (16.3%) | 28 |
| MR Heart (100×100) with 34 control points Figure 4.12 | Christensen’s fluid method | 2.08 (28.7%) | 60 |
| | Our fluid | 0.92 (12.6%) | 60 |
| | Our elastic method | 1.92 (26.4%) | 5 |

Table 4.3: Error measure E_{ba} and approximate execution time (Silicon Graphics Octane 250-MHZ MIPS R10000) for MR brain (axial) and heart images. E_{ba} : average displacement error on sparse boundary points. Note: the percentages shown with each average error are with respect to the true average displacement; the extremely short execution time for the heart images with our elastic method is due to pre-mature convergence.

4.6 Segmentation Influenced by Non-Rigid Registration

From the above results, we can see that accurate boundary finding can help us to get a more accurate and reasonable non-rigid registration. On the other hand, as mentioned in Chapter 1, image segmentation is a very useful application of non-rigid registration and therefore may be able to assist in the registration process. When a presegmented atlas image is warped to a study image, the warped segmentation can then be applied. In this section, we show an initial attempt at segmentation which is influenced by non-rigid registration.

Let E_s denote the Canny edge image of the study; B_a denote the true boundary image of the atlas; and \vec{u} denote the displacement vector between the atlas and study image. We want to find the boundary points of the study image $\vec{b}_s(\mathbf{p})$, defined by vector \mathbf{p} , given the three known inputs: E_s , B_a and \vec{u} .

As in Section 4.3 (using the same notation in Section 4.3),

$$\Pr(\vec{b}_s(\mathbf{p}) \mid E_s, \vec{u}, B_a) = \frac{\Pr(\vec{b}_s(\mathbf{p}), E_s, \vec{u}, B_a)}{\Pr(E_s, \vec{u}, B_a)} \quad (4.17)$$

Ignoring the denominator which does not change with \mathbf{p} and by using Bayes rule, our aim is to find:

$$\begin{aligned} & \arg \max_{\mathbf{p}} \Pr(\vec{b}_s(\mathbf{p}) \mid E_s, \vec{u}, B_a) \\ & \equiv \arg \max_{\mathbf{p}} \Pr(B_a \mid \vec{b}_s(\mathbf{p}), E_s, \vec{u}) \Pr(E_s \mid \vec{b}_s(\mathbf{p}), \vec{u}) \Pr(\vec{u} \mid \vec{b}_s(\mathbf{p})) \Pr(\vec{b}_s(\mathbf{p})) \\ & \propto \arg \max_{\mathbf{p}} \Pr(B_a \mid \vec{b}_s(\mathbf{p}), \vec{u}) \Pr(E_s \mid \vec{b}_s(\mathbf{p})) \Pr(\vec{u} \mid \vec{b}_s(\mathbf{p})) \Pr(\vec{b}_s(\mathbf{p})) \quad (4.18) \end{aligned}$$

$$\propto \arg \max_{\mathbf{p}} \Pr(\vec{b}_s(\mathbf{p})) \Pr(E_s \mid \vec{b}_s(\mathbf{p})) \Pr(B_a \mid \vec{b}_s(\mathbf{p}), \vec{u}) \quad (4.19)$$

where Equation (4.18) is valid if we ignore the dependence of B_a on E_s and E_s on \vec{u} ; Equation (4.19) is true if we ignore the dependence of \vec{u} on $\vec{b}_s(\mathbf{p})$ because \vec{u} is obtained as a prior here and is not modified in this formulation. Thus, we ignore this constant term $Pr(\vec{u})$ and reorder the remaining terms.

A Bayesian formulation (similar to that in Section 3.3) leads to the following objective function:

$$\begin{aligned}
O(\mathbf{p}) &= c_1 \ln \Pr(\mathbf{p}) + c_2 \sum_{n=1}^N E_s(x(\mathbf{p}, n), y(\mathbf{p}, n)) \\
&\quad + \sum_{n=1}^N B_a(x(\mathbf{p}, n) - u_x(\vec{b}_s(\mathbf{p}, n)), y(\mathbf{p}, n) - u_y(\vec{b}_s(\mathbf{p}, n))) \\
&= c_1 \sum_{j=1}^{t+4} \left[-\frac{(p_j - m_j)^2}{2\sigma_j^2} \right] + c_2 \sum_{n=1}^N E_s(x(\mathbf{p}, n), y(\mathbf{p}, n)) \\
&\quad + \sum_{n=1}^N B_a(x(\mathbf{p}, n) - u_x(\vec{b}_s(\mathbf{p}, n)), y(\mathbf{p}, n) - u_y(\vec{b}_s(\mathbf{p}, n))) \tag{4.20}
\end{aligned}$$

where c_1 and c_2 are two weighting coefficients.

As in Equation (3.28), the first term in Equation (4.20) is a prior bias to likely shapes and poses from the statistical shape analysis methods described in Section 3.2; the second term matches to the edges in the study image by maximizing the sum of the smoothed study edge image intensity at the boundary points defined by vector \mathbf{p} . However, one more term (the third term) is incorporated here which uses the known atlas true boundary information and the non-rigid registration results between the atlas and the study derived from our proposed algorithms. Such information will be helpful, for example, in situations where good quality study image edges are not available. This third term matches to the true boundary in the atlas image by maximizing the sum of the smoothed true atlas boundary points, which is defined by the calculated study boundary

points and the pre-calculated displacement vectors between the atlas and the study at the calculated study boundary points.

Figure 4.13 shows a comparison of the corpus callosum segmentation results by using our previous boundary finding with correspondence algorithm (described in Chapter 3) and by using the above proposed algorithm, which incorporates our non-rigid registration results. We can see that in this example, the non-rigid registration really helped the segmentation, since extra information from true atlas boundary image is included through the non-rigid mapping.

4.7 Conclusions

This work presents two systematic approaches for non-rigid registration. Transformations are constrained to be consistent with physical deformations of elastic solids in the first approach and viscous fluids in the second approach in order to maintain the topology, or integrity, of the anatomic structures while allowing detailed displacements to accommodate complex anatomical details. Both intensity information and statistical shape information are used as matching criteria in a Bayesian formulation. The incorporation of statistical shape information into the framework is the main contribution of our work. From the experimental results, statistical boundary shape information has been shown to augment physical model formulations for non-rigid registration, and both of our methods have their own advantages under different situations. On the other hand, our non-rigid registration has also been shown to have the potential of improving segmentation results.

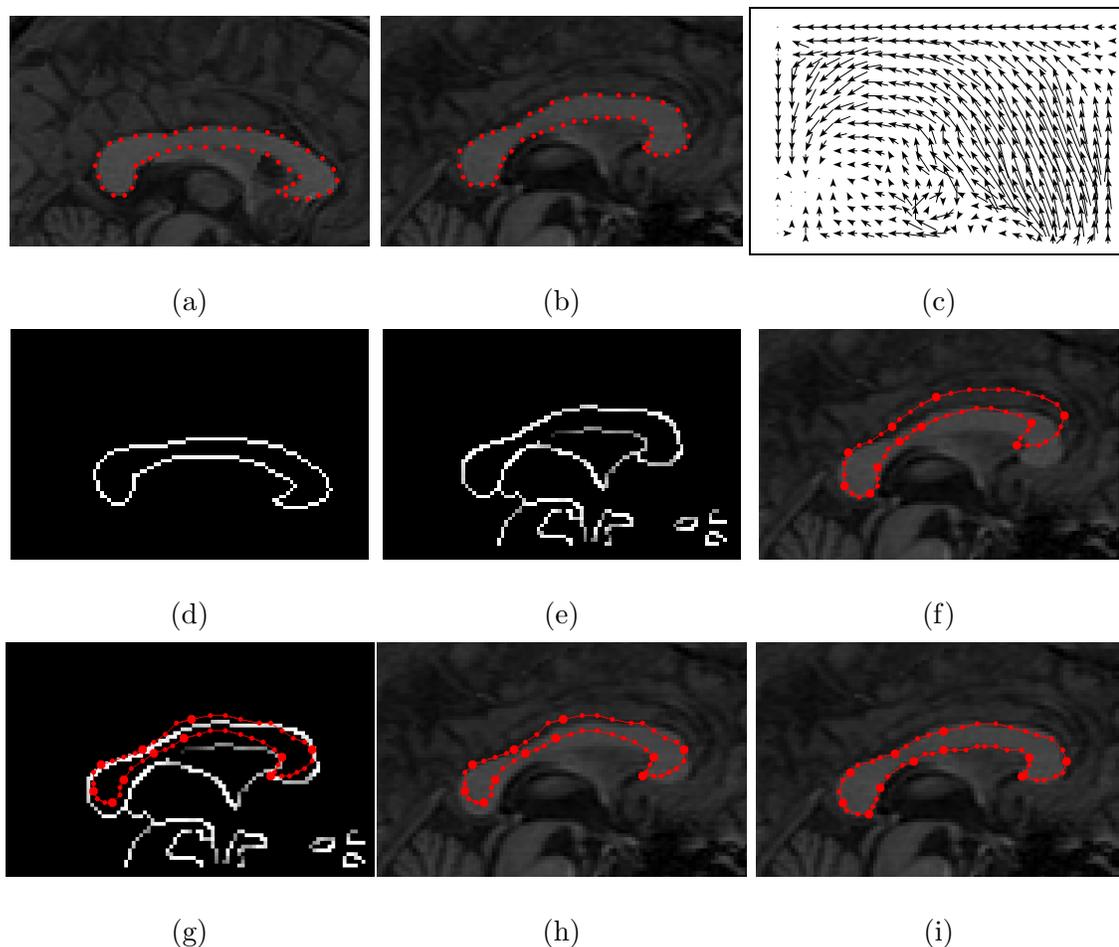


Figure 4.13: Demonstration of boundary finding with correspondence influenced by non-rigid registration (corpus callosum example (100×64)). (a): atlas image with its control points on the corpus callosum; (b): study image with its control points on the corpus callosum; (c): displacement vectors between the atlas and study by our elastic model-based non-rigid registration; (d): true atlas boundary image (known); (e): Canny edge image of the study; (f): initial contour (mean curve) on the study image; (g): inaccurate final contour on (e) by using objective function Equation (3.28), showing the final contour being trapped by the stronger upper edge without our registration result; (h): inaccurate final contour on the study image using objective function Equation (3.28); (i): accurate final contour on the study image by using objective function Equation (4.20), which incorporates our registration results and the true atlas boundary image information.

Chapter 5

3D Generalization to Volumetric Segmentation

The complete 2D statistical shape analysis framework for segmentation and registration has been finished and validated. The ideas that we have presented in this work can be extended to 3D as well. In this chapter, we give the 3D formulation for segmentation, which addresses the new techniques required for our 3D generalization to volumetric segmentation that are totally different from that in 2D: identification of corresponding surface points in the training set and 3D surface triangulation for visualization.

5.1 Introduction

Although Cootes' Active Shape Model has been well developed in 2D [35, 38, 39, 67], the 3D volumetric generalization has been hampered due to the difficulty in identification of corresponding surface points. As has been pointed out in [36]: the problem of devising ways of choosing suitable model points and placing them consistently on sets of examples

is the subject of continuing research.

Recently, both Gonzalez-Ballester [60, 61] and Fleute [51] developed a 3D version of Cootes' algorithm. The idea of their methods is that non-organized random point sets on the surface are identified first, and then these clouds of points are registered and matched to establish the points correspondence. The Principal Component Analysis and image search method they used are similar to Cootes' Active Shape Models. One disadvantage of their model construction is that the accuracy of the correspondence is very dependent on the registration process, which is very computationally expensive. The other disadvantage is that an extra method for surface triangulation from the unorganized points is needed. We use a totally different method to identify the corresponding surface points directly and simultaneously triangulate the surface based on the labeled points. In order to formulate the surface matching objective function, we directly generalize our proposed Bayesian method in 2D (Chapter 3).

5.2 Training Set Labeling and Surface Triangulation — New Techniques Required

In 2D, manual labeling of corresponding boundary points is relatively simple. First, critical points, which are usually easily identified features, such as high curvature points, sharp corners, etc, are labeled manually. Then, interpolated points are equally-spaced between the critical points along the boundary. This interpolation assumes that the deformation is approximately a uniform stretching between critical points. Locally uniform stretching is a reasonable assumption and can be satisfied by appropriately choosing the critical points. In 3D, critical points labeling would be similar to the 2D case since feature points would be visually identifiable from a 3D renderings. However, interpolation is

difficult because we have a smooth surface instead of a boundary. To solve this, we base the interpolation on geodesics, or shortest paths, between critical points on the surface. Again, the assumption is locally uniform stretching, or homothetic deformation [58] and will be satisfied, at least approximately, by the appropriate selection of critical points.

Therefore, in order to label the interpolated points in 3D, we find the shortest surface paths between the critical points and then label the interpolated points equally spaced along these paths. Once we get all the corresponding surface points, we also need to triangulate these 3D points to visualize and validate the surface, which is a traditionally difficult problem. Our model construction is a brand-new approach which can also triangulate the surface during the identification of corresponding surface points in a hierarchical way. The detailed steps of our approach can be described as follows.

5.2.1 Step 1: Critical points labeling and initial triangle construction

First, the critical points on the surface are labeled manually. Then, based on these points, we construct an initial triangle connection, which only involves limited number of points and thus can be done by hand (Figure 5.1(a)).

Eventually, we would like to automatically (or semi-automatically) identify these points and triangulate them.

5.2.2 Step 2: Shortest paths finding

The second step is to find the shortest paths between each pair of connected surface points (Figure 5.1(b)).

There are many methods to solve the shortest surface path finding problem [111, 136, 80, 81, 82]. The algorithm we used is based on Kimmel's two methods [80, 82], and is almost the same as his extended Fast Marching Method [82]. When we calculate the

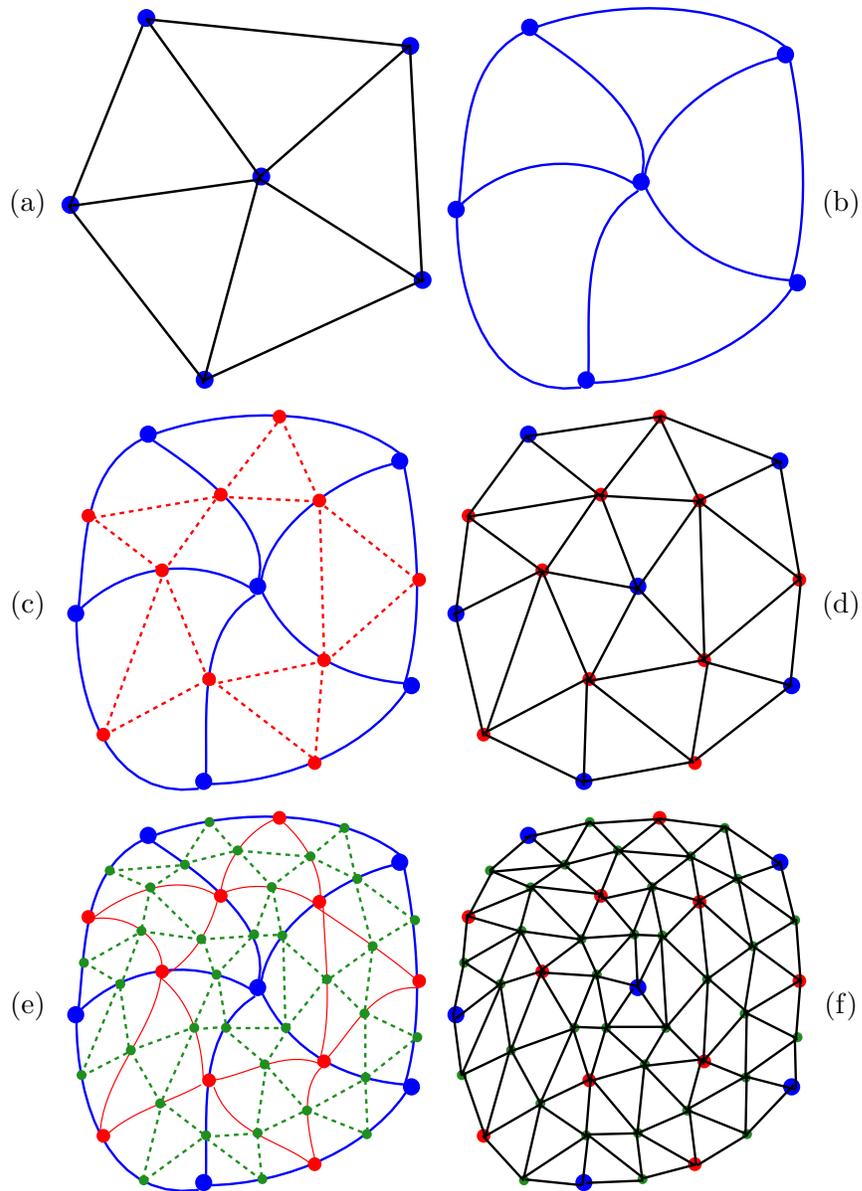


Figure 5.1: Diagram for corresponding surface points identification and surface triangulation in our hierarchical approach. (a): hand-labeled critical points and initial triangulation; (b): shortest paths between each pair of connected critical points; (c): selected mid-points on each shortest path (red dots) and triangle subdivision; (d): more dense triangulation (compare to (a)); (e): repeating of (c) in a hierarchical way; (f): even more dense triangulation (compare to (d)).

shortest path between two surface points, one is treated as source point, and the other as destination point.

The Fast Marching Method [113] is a numerical algorithm for solving the Eikonal equation $|\nabla T| = \mathcal{F}(x, y)$ on a rectangular orthogonal mesh in $O(M \log M)$ steps, where M is the total number of grid points. It has been extended to triangulated domains with the same computational complexity [82]. In our implementation, first, by using this extended Fast Marching Method, we solve the Eikonal equation with speed $\mathcal{F} = 1$ on the triangulated surface to compute the distance T from a source point. Note that the triangulated surface we used is computed using the Marching Cube algorithm [85]. Then, we backtrack along the gradient of the distance T from the destination point. That is, for each triangle, there is one gradient (this calculation is described in the following paragraph). Start from a point and “flow” inside the triangle which has the largest gradient according to the computed gradients. In this way, we get a sequence of straight segments, each segment corresponding to a path through one triangle. Thus, the shortest path between the source and destination points is constructed.

Gradient calculation for each triangle: Suppose we want to find the gradient direction AD for triangle ABC given that point A is the start point for the flow (Figure 5.2). Let T_A , T_B and T_C denote the known distance value at point A , B and C , respectively, calculated by the extended Fast Marching Method. Denote $a = |BC|$, $b = |AC|$, $c = |AB|$ and $a' = |BD|$. The gradient along AD , $g(a')$, is calculated as:

$$g(a') = \frac{T_A - T_D}{|AD|} \quad (5.1)$$

where T_D is the distance value at point D , which can be estimated as the linearly inter-

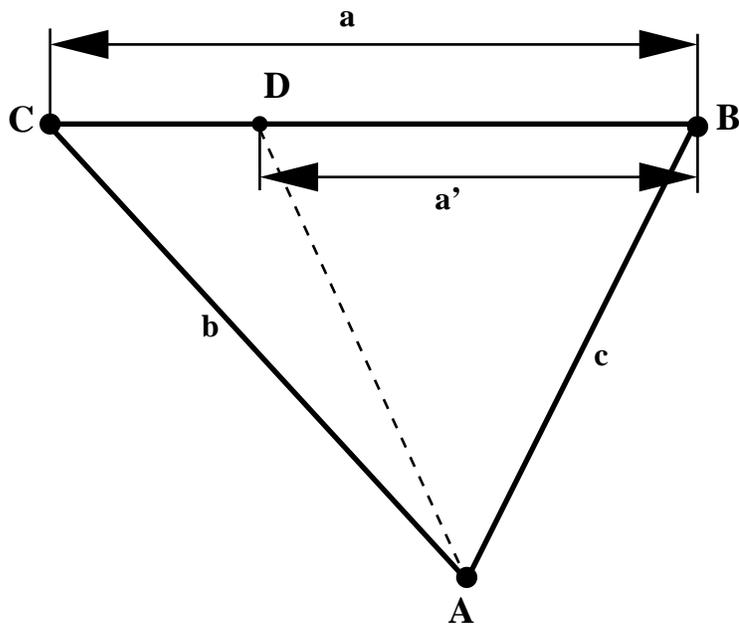


Figure 5.2: Gradient calculation within a triangle.

polated distance between point B and C :

$$T_D = T_B + \frac{a'(T_C - T_B)}{a} = \frac{a'T_C + (a - a')T_B}{a} \quad (5.2)$$

and

$$|AD| = \sqrt{a'^2 + c^2 - 2a'c \cos B} \quad (5.3)$$

Now, take the gradient of $g(a')$ in Equation (5.1) with respect to a' (using Equa-

tions (5.2)&(5.3)) and then set the numerator of $\frac{\partial g(a')}{\partial a'}$ equal to zero. Finally, we get:

$$a' = \frac{c^2(T_C - T_B) - ac(T_A - T_B) \cos B}{c(T_C - T_B) \cos B - a(T_A - T_B)} \quad (5.4)$$

where

$$\cos B = \frac{a^2 + c^2 - b^2}{2ac} \quad (5.5)$$

Therefore, the gradient direction for triangle ABC is AD with a' given by Equation (5.4); and the gradient for triangle ABC is calculated as Equation (5.1) with T_D , $|AD|$ and a' given by Equations (5.2), (5.3) and (5.4).

Figure 5.3 shows the Marching Cube triangulation and two shortest paths for a synthetic surface.

5.2.3 Step 3: Mid-point selection and triangle subdivision

The third step of our approach is selecting the mid-point on each shortest path, and decomposing each big triangle into four smaller ones (Figure 5.1(c)). In this way, a more dense triangulation than the initial one is derived (Figure 5.1(a) and 5.1(d)).

5.2.4 Step 4: Repeating step 2 and step 3

Now, we simply repeat the previous step 2 — shortest path calculation, and step 3 — mid-point selection & triangle sub-division, and thus generate even more dense surface points and triangulation. Figure 5.1(f) shows the resulting diagram using our hierarchical strategy: the blue dots are hand-labeled critical points; the red dots are interpolated points generated from the 1st iteration; and the green ones are generated from the 2nd

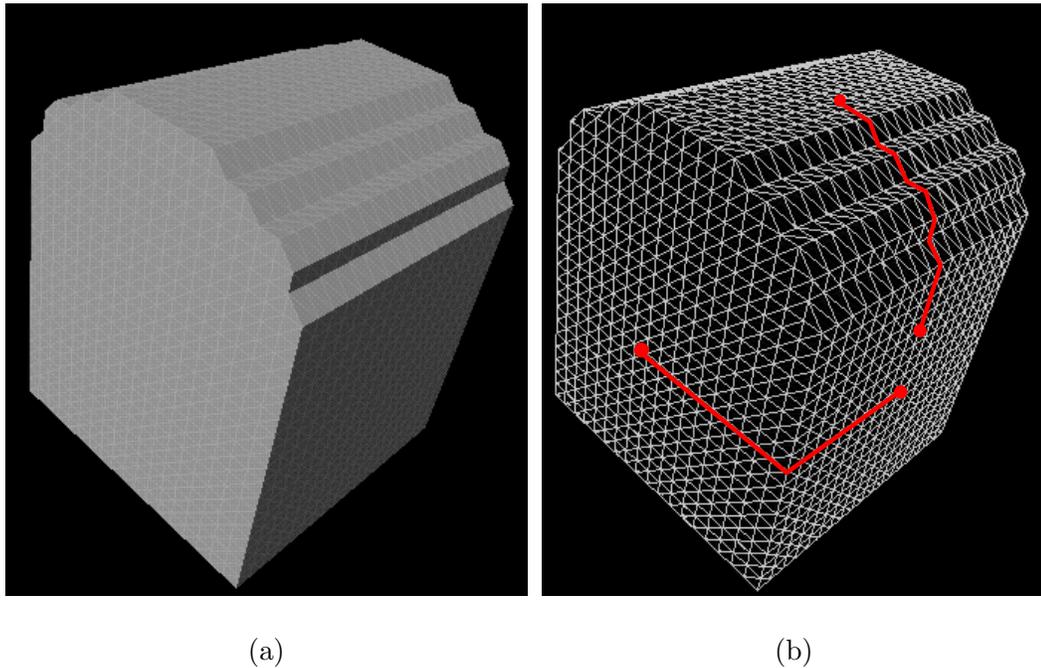


Figure 5.3: Shortest paths on a synthetic surface. (a): the synthetic surface; (b): two shortest paths on the Marching Cube triangulated surface of (a).

iteration, etc.

We can repeated this process until the triangles are small enough. In this way, we can identify corresponding surface points and construct a surface triangulation based on these points at the same time.

5.3 Aligning the 3D Labeled Training Set Points

Given the labeled training set points, we align them by scaling, translation and rotation. Because of the increased number of parameters (7) and the complexity of the 3D formulation, traditional least square matching which requires analytic derivative formulation is not convenient for us to use. One possible 3D rigid registration method uses only a

translation vector and a rotation matrix without the scale parameter [30]. We adopted a method similar to the Procrustes shape distance method [15].

Procrustes distance is the distance between each point on the shape and the centroid of the shape. We first calculate the centroid of each training example, and find the translation parameters so that the centroids of each training shape are superimposed. Then, the mean and scale of each set is calculated so that the sum of squares of the Procrustes distance of each training set is normalized with respect to that of the mean shape. After we compensate for translation and scale, singular value decomposition (SVD) [4] is employed to find the three rotation parameters. The whole process is repeated until convergence occurs. At last, we calculate the pose standard deviations of the training set with respect to the finally converged mean shape, which would be the standard deviations for the pose parameters of our prior model. The whole alignment process is shown in Figure 5.4. The criterion of convergence is as follows: if sum of the distance between all N points on the mean surface at time t and time $t - 1$ is smaller than a small number δ times N (say, $\delta = 0.15$), the whole process is regarded as converged at time t . That is,

$$\sum_{n=1}^N \|\mathbf{m}_t(n) - \mathbf{m}_{t-1}(n)\| < \delta N \quad (n = 1, 2, \dots, N) \quad (5.6)$$

5.4 Surface Matching Objective Function

5.4.1 Transformation Matrix

The Cartesian coordinate system is used in our formulation. Let $(x_o, y_o, z_o)^T$ be one of the original un-aligned points, and $(x, y, z)^T$ be the corresponding aligned point after the following transformations: Translation vector $(T_x, T_y, T_z)^T$; Scaling matrix S with scaling

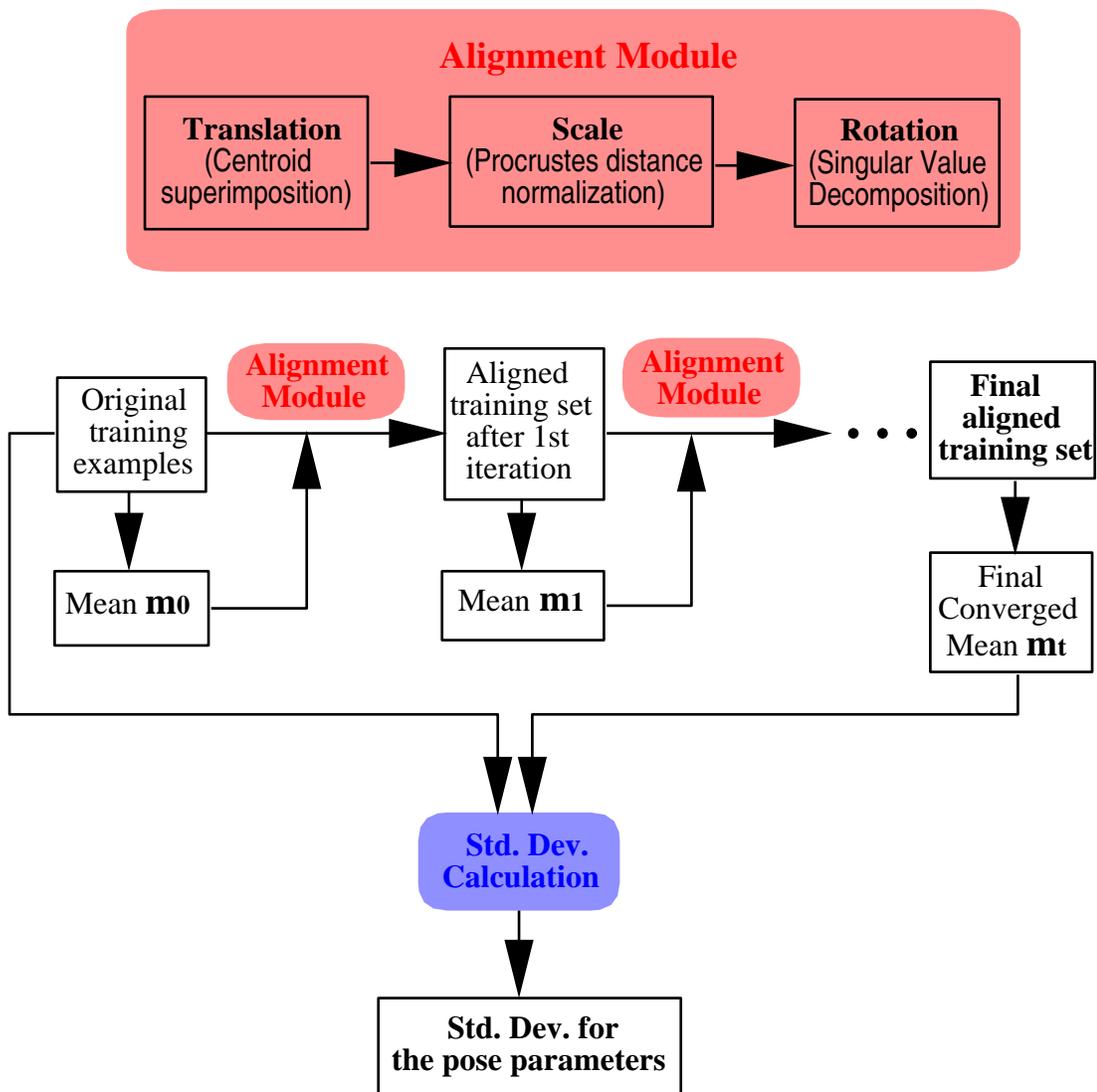


Figure 5.4: 3D point set alignment process.

factor about each axes $s_x = s_y = s_z = s$; Rotation about the y (R_y), then x (R_x), and then z (R_z) axes by degrees β , α , and γ degrees respectively. Then,

$$\begin{aligned} \begin{pmatrix} x \\ y \\ z \end{pmatrix} &= R_z(\gamma)R_x(\alpha)R_y(\beta)S(s) \begin{pmatrix} x_o \\ y_o \\ z_o \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \\ &= M(s, \alpha, \beta, \gamma) \begin{pmatrix} x_o \\ y_o \\ z_o \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix} \end{aligned} \quad (5.7)$$

where

$$\begin{aligned} R_x(\alpha) &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix}; & R_y(\beta) &= \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix}; \\ R_z(\gamma) &= \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}; & S(s) &= \begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix} \end{aligned}$$

Therefore, $M(s, \alpha, \beta, \gamma)$ is represented as:

$$M(s, \alpha, \beta, \gamma) = s \begin{bmatrix} \cos \gamma & -\sin \gamma & 0 \\ \sin \gamma & \cos \gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix}$$

$$= \begin{bmatrix} s(\cos \beta \cos \gamma - \sin \alpha \sin \beta \sin \gamma) & -s(\cos \alpha \sin \gamma) & s(\sin \beta \cos \gamma + \sin \alpha \cos \beta \sin \gamma) \\ s(\cos \beta \sin \gamma + \sin \alpha \sin \beta \cos \gamma) & s(\cos \alpha \cos \gamma) & s(\sin \beta \sin \gamma - \sin \alpha \cos \beta \cos \gamma) \\ -s(\cos \alpha \sin \beta) & s(\sin \alpha) & s(\cos \alpha \cos \beta) \end{bmatrix} \quad (5.8)$$

5.4.2 Point Representation

As in 2D, the statistical point model is used as a 3D surface model. Given a set of m aligned examples and each example of a set of N aligned labeled points,

$$\mathbf{L}_i = (x_i(1), y_i(1), z_i(1), x_i(2), y_i(2), z_i(2), \dots, x_i(N), y_i(N), z_i(N))^T \\ (i = 1, 2, \dots, m),$$

we calculate the mean shape $\bar{\mathbf{L}}$, and the covariance about the mean, $C_{training}$. As before, it can be shown that by principal component analysis, the eigenvectors of the $3N \times 3N$ covariance matrix, $C_{training}$, corresponding to the largest eigenvalues describe the most significant modes of variation in the variables used to derive the covariance matrix, and that the proportion of the total variance explained by each eigenvector is equal to the corresponding eigenvalue λ_k [75, 76]. Typically, most of the variation can be explained by a small number of modes, t ($< 3N$). Thus, any shape in the training set can be approximated using the mean shape and a weighted sum of deviations obtained from the first t modes:

$$\mathbf{L} = \bar{\mathbf{L}} + Q\mathbf{a} \quad (5.9)$$

where $Q = [\mathbf{q}_1 \mid \mathbf{q}_2 \mid \cdots \mid \mathbf{q}_t]$ is the matrix of the first t eigenvectors, and $\mathbf{a} = (a_1 a_2 \cdots a_t)^T$ is a vector of weights, which is also the set of t shape parameters to be optimized later. This equation allows us to generate new examples of shapes by varying the parameter \mathbf{a} within suitable limits. Let $(x_{shape}(n), y_{shape}(n), z_{shape}(n))^T$ be the n th point of new generated surface with origin at the mean surface centroid $(C_x, C_y, C_z)^T$, then

$$\begin{cases} x_{shape}(n) = \bar{x}(n) + \sum_{k=1}^t Q_{3n,k} a_k - C_x \\ y_{shape}(n) = \bar{y}(n) + \sum_{k=1}^t Q_{3n+1,k} a_k - C_y \\ z_{shape}(n) = \bar{z}(n) + \sum_{k=1}^t Q_{3n+2,k} a_k - C_z \end{cases} \quad (5.10)$$

where $(\bar{x}(n), \bar{y}(n), \bar{z}(n))^T$ is the mean shape of the n th point, and C_x, C_y, C_z are calculated as:

$$\begin{cases} C_x = \frac{1}{mN} \sum_{n=1}^N \sum_{i=1}^m x_i(n) \\ C_y = \frac{1}{mN} \sum_{n=1}^N \sum_{i=1}^m y_i(n) \\ C_z = \frac{1}{mN} \sum_{n=1}^N \sum_{i=1}^m z_i(n) \end{cases} \quad (5.11)$$

The combined pose and shape parameter vector to be determined is

$$\mathbf{p} = (s, \alpha, \beta, \gamma, T_x, T_y, T_z, a_1, a_2, \cdots, a_t)^T \quad (5.12)$$

If the pose parameters are also incorporated into the point representation, the point formulation of the n th surface point ($n = 0, 1, \dots, N - 1$) is

$$\left\{ \begin{array}{l} x(\mathbf{p}, n) = s[M(1, 1)x_{shape}(n) + M(1, 2)y_{shape}(n) + M(1, 3)z_{shape}(n)] \\ \qquad \qquad \qquad + T_x + C_x \\ y(\mathbf{p}, n) = s[M(2, 1)x_{shape}(n) + M(2, 2)y_{shape}(n) + M(2, 3)z_{shape}(n)] \\ \qquad \qquad \qquad + T_y + C_y \\ z(\mathbf{p}, n) = s[M(3, 1)x_{shape}(n) + M(3, 2)y_{shape}(n) + M(3, 3)z_{shape}(n)] \\ \qquad \qquad \qquad + T_z + C_z \end{array} \right. \quad (5.13)$$

where $M(i, j)$ is the i th row and j th column entry of matrix $M(s, \alpha, \beta, \gamma)$ in Equation 5.8.

5.4.3 Bayesian Objective Function

As in 2D, by using Bayes rule, the *a posteriori* probability density of the deformed template given the input edge image can be expressed as:

$$\Pr(\mathbf{p} | E) = \frac{\Pr(E | \mathbf{p}) \Pr(\mathbf{p})}{\Pr(E)} \quad (5.14)$$

Our objective is to maximize the *a posteriori* density in Equation (5.14) with respect to \mathbf{p} . This can be simplified to maximize (see Chapter 3):

$$O(\mathbf{p}) = c \sum_{j=1}^{t+4} \left[-\frac{(p_j - m_j)^2}{2\sigma_j^2} \right] + \sum_{n=1}^N E(x(\mathbf{p}, n), y(\mathbf{p}, n), z(\mathbf{p}, n)) \quad (5.15)$$

with \mathbf{p} given by Equation (5.12) and c a constant coefficient. This equation is the maximum *a posteriori* objective incorporating a prior bias to likely shapes and poses (first term) and match to the edges in the image by maximizing the sum of the smoothed edge image intensity at the boundary points defined by vector \mathbf{p} (second term).

5.4.4 Gradient Formulation

We optimize the objective function $O(\mathbf{p})$ using the conjugate gradient method. We compute the gradient formulation as follows:

First, differentiate Equation (5.15) to get

$$\begin{aligned} \nabla O(\mathbf{p}) &= c \sum_{j=1}^{t+4} \left[-\frac{p_j - m_j}{\sigma_j^2} \right] \\ &+ \sum_{n=1}^N \left[\frac{\partial E(x(\mathbf{p}, n), y(\mathbf{p}, n), z(\mathbf{p}, n))}{\partial x} \frac{\partial x(\mathbf{p}, n)}{\partial \mathbf{p}} \right. \\ &+ \frac{\partial E(x(\mathbf{p}, n), y(\mathbf{p}, n), z(\mathbf{p}, n))}{\partial y} \frac{\partial y(\mathbf{p}, n)}{\partial \mathbf{p}} \\ &+ \left. \frac{\partial E(x(\mathbf{p}, n), y(\mathbf{p}, n), z(\mathbf{p}, n))}{\partial z} \frac{\partial z(\mathbf{p}, n)}{\partial \mathbf{p}} \right] \end{aligned} \quad (5.16)$$

The partials of E , the edge image, with respect to x , y and z can be calculated using a central discrete divided difference approximation. The partials of $x(\mathbf{p}, n)$, $y(\mathbf{p}, n)$ and $z(\mathbf{p}, n)$ with respect to \mathbf{p} can be calculated from Equations (5.13), (5.10) and (5.8) by:

$$\begin{cases} \frac{\partial x(\mathbf{p}, n)}{\partial s} = M(1, 1)x_{shape} + M(1, 2)y_{shape} + M(1, 3)z_{shape} \\ \frac{\partial y(\mathbf{p}, n)}{\partial s} = M(2, 1)x_{shape} + M(2, 2)y_{shape} + M(2, 3)z_{shape} \\ \frac{\partial z(\mathbf{p}, n)}{\partial s} = M(3, 1)x_{shape} + M(3, 2)y_{shape} + M(3, 3)z_{shape} \end{cases}$$

$$\left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial \alpha} = s[(-\cos \alpha \sin \beta \sin \gamma)x_{shape} + (\sin \alpha \sin \gamma)y_{shape} + (\cos \alpha \cos \beta \sin \gamma)z_{shape}] \\ \frac{\partial y(\mathbf{p},n)}{\partial \alpha} = s[(\cos \alpha \sin \beta \cos \gamma)x_{shape} - (\sin \alpha \cos \gamma)y_{shape} - (\cos \alpha \cos \beta \cos \gamma)z_{shape}] \\ \frac{\partial z(\mathbf{p},n)}{\partial \alpha} = s[(\sin \alpha \sin \beta)x_{shape} + (\cos \alpha)y_{shape} - (\sin \alpha \cos \beta)z_{shape}] \end{array} \right.$$

$$\left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial \beta} = s[-(\sin \beta \cos \gamma + \sin \alpha \cos \beta \sin \gamma)x_{shape} \\ \quad + (\cos \beta \cos \gamma - \sin \alpha \sin \beta \sin \gamma)z_{shape}] \\ \frac{\partial y(\mathbf{p},n)}{\partial \beta} = s[-\sin \beta \sin \gamma + \sin \alpha \cos \beta \cos \gamma)x_{shape} \\ \quad + (\cos \beta \sin \gamma + \sin \alpha \sin \beta \cos \gamma)z_{shape}] \\ \frac{\partial z(\mathbf{p},n)}{\partial \beta} = s[(-\cos \alpha \cos \beta)x_{shape} - (\cos \alpha \sin \beta)z_{shape}] \end{array} \right.$$

$$\left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial \gamma} = s[-(\cos \beta \sin \gamma + \sin \alpha \sin \beta \cos \gamma)x_{shape} \\ \quad - (\cos \alpha \cos \gamma)y_{shape} + (-\sin \beta \sin \gamma + \sin \alpha \cos \beta \cos \gamma)z_{shape}] \\ \frac{\partial y(\mathbf{p},n)}{\partial \gamma} = s[(\cos \beta \cos \gamma - \sin \alpha \sin \beta \sin \gamma)x_{shape} \\ \quad - (\cos \alpha \sin \beta)y_{shape} + (\sin \beta \cos \gamma + \sin \alpha \cos \beta \sin \gamma)z_{shape}] \\ \frac{\partial z(\mathbf{p},n)}{\partial \gamma} = 0 \end{array} \right.$$

$$\left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial T_x} = 1 \\ \frac{\partial y(\mathbf{p},n)}{\partial T_x} = 0 \\ \frac{\partial z(\mathbf{p},n)}{\partial T_x} = 0 \end{array} \right. \quad \left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial T_y} = 0 \\ \frac{\partial y(\mathbf{p},n)}{\partial T_y} = 1 \\ \frac{\partial z(\mathbf{p},n)}{\partial T_y} = 0 \end{array} \right. \quad \left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial T_z} = 0 \\ \frac{\partial y(\mathbf{p},n)}{\partial T_z} = 0 \\ \frac{\partial z(\mathbf{p},n)}{\partial T_z} = 1 \end{array} \right.$$

$$\left\{ \begin{array}{l} \frac{\partial x(\mathbf{p},n)}{\partial a_j} = s[M(1,1)Q_{3n,j} + M(1,2)Q_{3n+1,j} + M(1,3)Q_{3n+2,j}] \\ \frac{\partial y(\mathbf{p},n)}{\partial s} = s[M(2,1)Q_{3n,j} + M(2,2)Q_{3n+1,j} + M(2,3)Q_{3n+2,j}] \\ \frac{\partial z(\mathbf{p},n)}{\partial s} = s[M(3,1)Q_{3n,j} + M(3,2)Q_{3n+1,j} + M(3,3)Q_{3n+2,j}] \end{array} \right.$$

5.5 Initial Synthetic Results

Figure 5.5(a) is one example in a synthetic training set, showing a big sphere (equal x,y,z-axes ellipsoid) with 6 small half balls attached at each of the six directions: front, back, top, bottom, left and right. Figure 5.5(b) is another view of the surface.

First, the critical points on the surface are labeled by hand (38 points in total). An initial triangulation based on hand-labeled critical surface points is manually constructed as shown in Figure 5.5(c). Shown in Figures 5.5(d)(e)(f) are the results by our hierarchical approach after the 1st, 2nd and 3rd iterations. It can be seen that as the iterations continue, more and more dense surface points and triangulation are derived. At this point, except for the critical points labeling and initial triangle construction, the whole process of our proposed algorithm is completely automatic. Also, the identification of corresponding surface points and surface triangulation are computed at the same time.

For this model, we use a number of training examples changing the x,y,z-radii of the centered ellipsoid and holding the y-radius the same as the z-radius. We get the principal mode of variation a_1 of this synthetic model shown in Figure 5.6. The middle column corresponds to the mean shape, and the shape parameter for the left and right columns are respectively -2 and +2 standard deviations. The first two rows are viewed from the Z axis showing the changing x-y eccentricity. The bottom row is viewed from the X axis, showing the equal y and z radii. Both of them reflect the main variation in the training set.

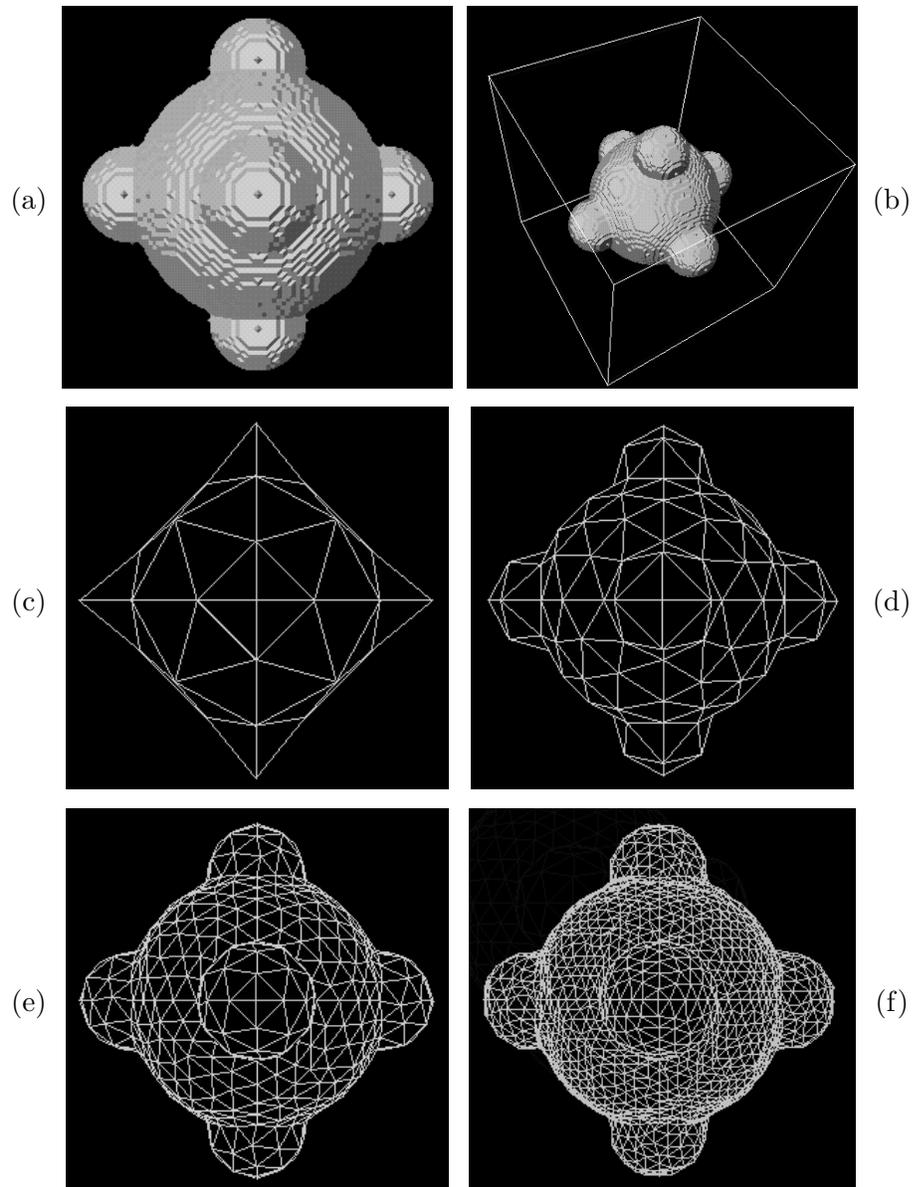


Figure 5.5: 3D Synthetic training set construction. (a): a synthetic example; (b): another view of (a); (c): initial triangulation; (d) to (f): labeled surface points and surface triangulation based on these points respectively after 1st, 2nd and 3rd iterations by our hierarchical approach. (Note: All figures are viewed from the Z axis except (b)).

The initial results I have shown here demonstrate the model construction part of the procedure, which is very different from that in 2D. The 3D image searching is just the direct generalization of that in 2D.

5.6 Conclusions

In this chapter, the 3D generalization to volumetric segmentation has been presented, which includes both the new techniques required for model construction and 3D extended formulation. Our model construction approach turns out to be a new efficient algorithm, which can identify the corresponding surface points and finish surface triangulation at the same time. Unlike Gonzalez-Ballester [60, 61] and Fleute's [51] methods, no extra step is needed for visualization in our approach since surface points are triangulated automatically. The validation of the proposed 3D segmentation is the subject of future work.

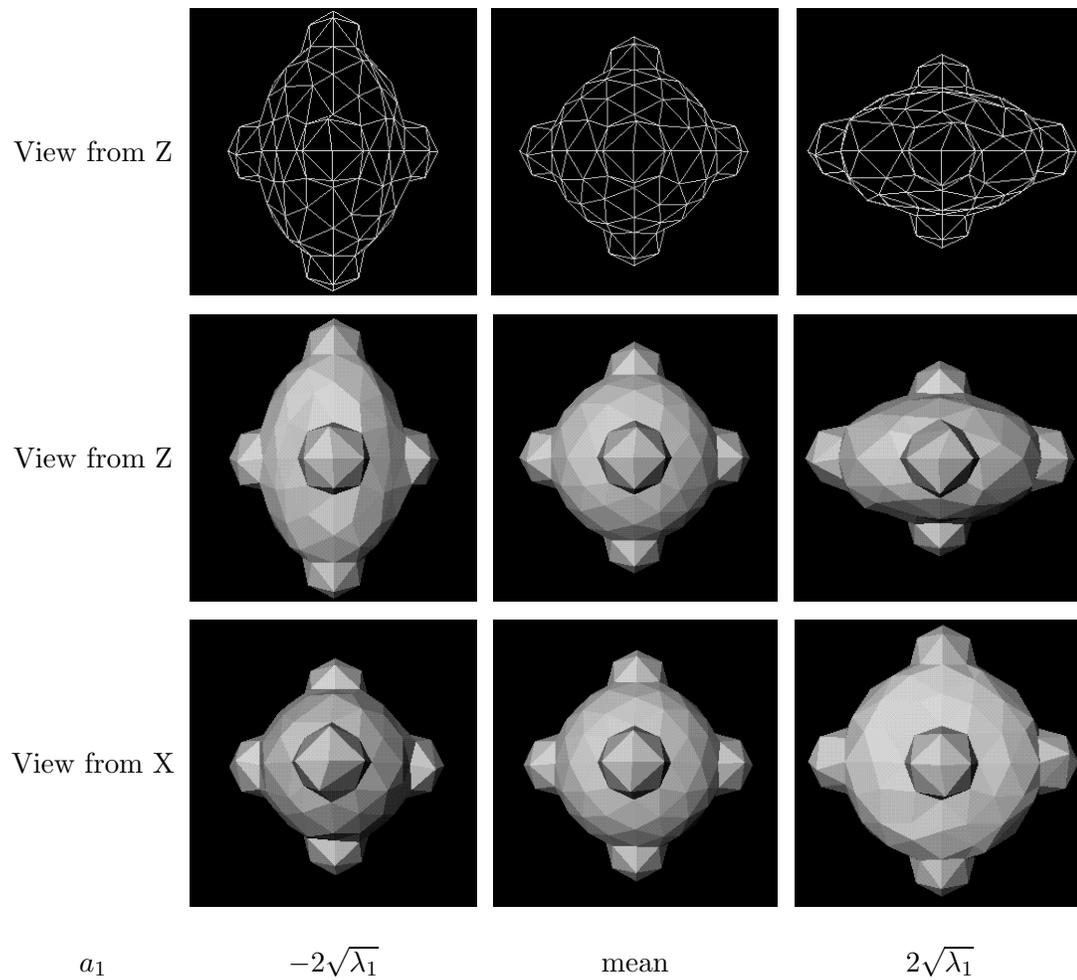


Figure 5.6: Principal mode a_1 of the 3D synthetic model. Top: eigen-wireframes viewed from the Z axis; Middle row: eigen-surfaces corresponding to the top wireframes; Bottom: eigen-surfaces viewed from the X axis.

Chapter 6

Summary

This dissertation presents statistical shape analysis methods for segmentation and physical model-based non-rigid registration (Figure 6.1). In 2D boundary finding with correspondence, the statistical point models ($C_{training}$) derived from a training set by principal component analysis are used as the prior probability in a Bayesian formulation. Moreover, in order to show the important role of the statistical prior shape model, we demonstrate the use of two other kinds of generic prior information: an independence model ($C_{identity}$) and a smoothness model (C_{smooth}). In addition, we consider prior shape models trained on small sets which may not allow enough variation to adequately span the space of plausible shapes (C_{mix}). The resulting boundary points derived from $C_{training}$ or C_{mix} which contain statistical shape information are then used as landmarks for our physical model-based non-rigid registration. The transformations are constrained to be consistent with the physical properties of deformable elastic solids in the first method and those of viscous fluids in the second, to maintain smoothness and continuity. A Bayesian formulation, based on each physical model, an intensity similarity measure, and statistical shape information embedded in corresponding boundary points, is employed to derive

more accurate and robust approaches to non-rigid registration. The new techniques required and the generalized formulation for 3D volumetric segmentation are also presented. Throughout all the work in this thesis, the key link is **statistical shape**.

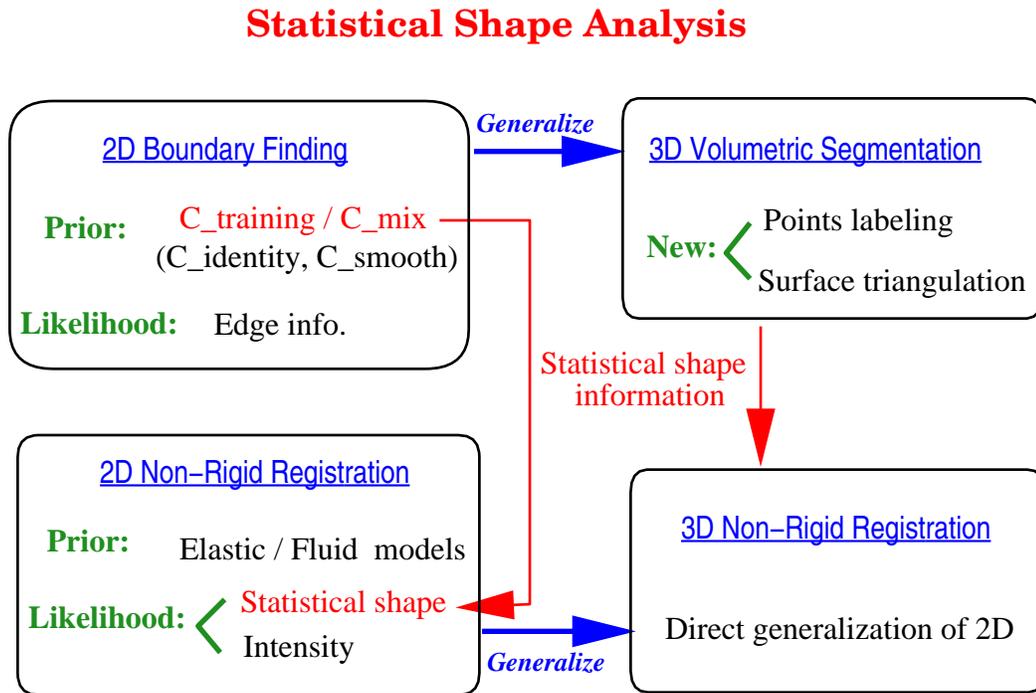


Figure 6.1: Statistical shape analysis framework for segmentation and registration.

Future directions include validation of the proposed 3D segmentation and generalization to 3D non-rigid registration. Although finding the corresponding surface points and statistical surface model construction in 3D require new strategies, generalizing the purely physical model deformation to 3D is straightforward and has been developed by Christensen *et al.* [29]. Therefore, the 3D non-rigid registration of our methods would be

the direct generalization of that in 2D. Of course, the computational cost increases with the number of voxels. The fast algorithm for non-rigid viscous fluid registration, which is based on a linear elastic deformation for the velocity field of the fluid and is derived through a convolution filter [16], could be applied in our fluid approach to improve the total computation time. Also, the coupling of boundary finding and registration can be increased by enabling registration to influence boundary finding, and thus both processes will enhance each other.

Bibliography

- [1] L. Alvarez, P. L. Lions and J. M. Morel, “Image selective smoothing and edge detection by nonlinear diffusion,” *SIAM Journal of Numerical Analysis*, Vol. 29, pp. 845-866, 1992.
- [2] A. A. Amini, T. E. Weymouth and R. C. Jain, “Using dynamic programming for solving variational problems in vision,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, no. 9, pp. 855-867, 1990.
- [3] Y. Amit, U. Grenander and M. Piccioni, “Structural image restoration through deformable templates,” *Journal of American Statistical Association*, Vol. 86, no. 414, pp. 376-387, June 1991.
- [4] K. S. Arun, T. S. Huang and S. D. Blostein, “Least-square fitting of two 3-D point sets,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, no. 5, pp. 698-700, 1987.
- [5] J. Ashburner and K. Friston, “Fully three-dimensional nonlinear spatial normalization: A new approach,” *NeuroImage*, Vol. 3, no. 3, pp. S111, June 1996 (2nd Intl. Conf. Mapping Human Brain).

- [6] J. Babaud, A. Witkin, M. Baudin and R. Duda, "Uniqueness of the gaussian kernel for scale space filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, no. 1, pp. 26-33, 1986.
- [7] R. Bajcsy, R. Lieberman and M. Reivich, "A computerized system for the elastic matching of deformed radiographic images to idealized atlas images," *Journal of Computer Assisted Tomography*, Vol. 7, no. 4, pp. 618-625, 1983.
- [8] R. Bajcsy and S. Kovacic, "Multiresolution elastic matching," *Computer Vision, Graphics and Image Processing*, Vol. 46, pp. 1-21, 1989.
- [9] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, Vol. 13, no. 2, pp. 111-122, 1981.
- [10] D. H. Ballard and C. M. Brown, *Computer Vision*, Prentice Hall, Englewood Cliffs, 1982.
- [11] A. Blake and A. Yuille, *Active Vision*, MIT Press, Cambridge, MA, 1992.
- [12] J. Boes, B. Kim, K. Frey and C. Meyer, "MRI-PET head registration via iterative minimization of mutual information," *NeuroImage*, Vol. 3, no. 3, S149, June 1996 (2nd Intl. Conf. Funct. Mapping Human Brain).
- [13] F. L. Bookstein and W. D. K. Green, "A feature space for derivatives of deformations," *Information Processing in Medical Imaging*, pp. 1-16, 1993.
- [14] F. L. Bookstein, "Shape and information in medical images: A decade of the morphometric synthesis," *Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pp. 2-12, June 1996.

- [15] F. L. Bookstein, "Landmark methods for forms without landmarks: morphometrics of group differences in outline shape," *Medical Image Analysis*, Vol. 1, no. 3, pp. 225-243, 1997.
- [16] M. Bro-Nielsen and C. Gramkow, "Fast Fluid Registration of Medical Images," *Proceedings of SPIE Visualization in Biomedical Computing*, pp. 267-276, 1996.
- [17] R. C. Buck, *Advanced Calculus*, McGraw-Hill Book Company, St. Louis, 3 edition, 1978.
- [18] R. L. Burden and J. D. Faires, *Numerical Analysis*, PWS Publishing Company, Boston, 1993.
- [19] D. J. Burr, "A dynamic model for image registration," *Computer Graphics Image Processing*, Vol. 15, pp. 102-112, 1981.
- [20] D. J. Burr, "Elastic matching of line drawings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 3, no. 6, pp. 708-713, Nov. 1981.
- [21] P. J. Burt, T. H. Hong and A. Rosenfeld, "Segmentation and estimation of region properties through co-operative hierarchical computation," *IEEE Transactions on System, Man and Cybernetics*, Vol. 11, pp. 802-809, 1981.
- [22] J. F. Canny, "A computational approach to edge-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, no. 6, pp. 679-698, 1986.
- [23] V. Caselles, R. Kimmel and G. Sapiro, "Geodesic active contours," *International Journal of Computer Vision*, Vol. 22, no. 1, pp. 61-79, 1997.

- [24] A. Chakraborty, L. H. Staib and J. S. Duncan, "An integrated approach to boundary finding in medical images," *IEEE Workshop on Biomedical Image Analysis*, pp. 13-22, 1994.
- [25] A. Chakraborty, L. H. Staib and J. S. Duncan, "Deformable boundary finding in medical images by integrating gradient and region information," *IEEE Transactions on Medical Imaging*, Vol. 15, no. 6, pp. 859-870, 1996.
- [26] Y. S. Chow, U. Grenander and D. M. Keenan, *HANDS. A Pattern-Theoretic Study of Biological Shapes*, New York: Springer-Verlag, 1991.
- [27] G. E. Christensen, *Deformable shape models for anatomy*, Doctoral dissertation in Electrical Engineering, Washington University, St. Louis, August 1994.
- [28] G. E. Christensen, R. Rabbitt, M. I. Miller, S. Joshi, U. Grenander, T. Coogan and D. van Essen, "Topological properties of smooth anatomic maps," *Information Processing in Medical Imaging*, pp. 101-112, Kluwer, Dordrecht, 1995.
- [29] G. E. Christensen, M. I. Miller and M. W. Vannier, "Individualizing neuroanatomical atlases using a massively parallel computer," *Computer*, pp. 32-38, January 1996.
- [30] G. E. Christensen, A. A. Kane, J. L. Marsh and M. W. Vannier, "Synthesis of an individualized cranial atlas with dysmorphic shape," *Proceedings of the IEEE Workshop on Mathematical Methods on Biomedical Image Analysis*, pp. 309-318, June 1996.
- [31] G. E. Christensen, R. D. Rabbitt and M. I. Miller, "Deformable templates using large deformation kinematics," *IEEE Transactions on Image Processing*, Vol. 5, no. 10, pp. 1435-1447, October 1996.

- [32] L. D. Cohen, "On active contour models and balloons," *Computer Vision, Graphics and Image Processing: Image Understanding*, Vol. 53, pp. 211-218, 1991.
- [33] D. L. Collins, A. C. Evans, C. Holmes and T. M. Peters, "Automatic 3D segmentation of neuro-anatomical structures from MRI," *Information Processing in Medical Imaging*, pp. 139-152, 1995.
- [34] D. L. Collins and A. C. Evans, "Automatic 3D estimation of gross morphometric variability in human brain," *NeuroImage*, Vol. 3, no. 3, pp. S129, June 1996 (2nd Intl. Conf. Funct. Mapping Human Brain).
- [35] T. F. Cootes, C. J. Taylor, A. Lanitis, D. H. Cooper and J. Graham, "Building and using flexible models incorporating grey-level information," *Proceedings of the Fourth International Conference on Computer Vision*, pp. 242-246, 1993.
- [36] T. F. Cootes, A. Hill, C. J. Taylor, and J. Haslam, "The use of active shape models for locating structure in medical images," *Information Processing in Medical Imaging*, pp. 33-47, 1993.
- [37] T. F. Cootes, C. J. Taylor and A. Lanitis, "Active shape models: Evaluation of a multi-resolution method for improving image search," *Proceedings of British Machine Vision Conf.*, Vol. 1, pp. 327-336, 1994.
- [38] T. F. Cootes, C. J. Taylor, D. H. Cooper and J. Graham, "Active shape models – their training and application," *Computer Vision, Graphics and Image Processing: Image Understanding*, Vol. 61, no. 1, pp. 38-59, 1995.

- [39] T. F. Cootes and C. J. Taylor, "Combining Point Distribution Models with Shape Models Based on Finite Element Analysis," *Image and Vision Computing*, Vol. 13, no. 5, pp. 403-410, 1995.
- [40] C. Davatzikos and J. Prince, "Brain image registration based on curve mapping," *IEEE Workshop on Biomedical Image Analysis*, pp. 245-254, 1994.
- [41] J. Declerck, G. Subsol, J. Thirion and N. Ayache, "Automatic retrieval of anatomical structures in 3D medical images," *Computer Vision, Virtual Reality and Robotics in Medicine (CVRMed '95, LNCS 905)*, pp. 153-162, Berlin, Springer-Verlag, 1995.
- [42] H. Delingette, "Simplex meshes: a general representation for 3D shape reconstruction," Technical Report 2214, INRIA, March 1994.
- [43] H. Derin, H. Elliott, R. Cristi and D. Geman, "Bayes smoothing algorithm for segmentation of binary images modeled by markov random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, pp. 707-720, 1984.
- [44] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using gibb's random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 9, pp. 39-55, 1987.
- [45] N. Duta and M. Sonka, "Segmentation and interpretation of MR brain images using an improved knowledge-based active shape models," *Information Processing in Medical Imaging*, pp. 375-380, Springer-Verlag, Berlin, 1997.
- [46] P. van den Elsen, J. Maintz, E. Pol and M. Viergever, "Image fusion using geometrical features," *Visualization Biomedical Computing 1992, Proceedings of SPIE 1808*, pp. 172-186, Editor: R. A. Robb, 1992.

- [47] P. van den Elsen, E. Pol and M. Viergever, "Medical image matching — A review with classification," *IEEE Engineering in Medicine and Biology*, pp. 26-39, March 1993.
- [48] A. C. Evans, S. Marret, L. Collins and T. M. Peters, "Anatomical-functional correlative analysis of the human brain using three dimensional imaging systems," *Proceedings of SPIE Vol 1092, Medical Imaging III: Image Processing*, SPIE Press, Bellingham, WA, pp. 264-274, 1989.
- [49] A. C. Evans, W. Dai, L. Collins, P. Neelin and S. Marret, "Warping of a computerized 3-D atlas to match brain image volumes for quantitative neuroanatomical and functional analysis," *Image Processing 1445*, pp. 236-246, 1991.
- [50] J. Feldmar, G. Malandain, J. Declerck and N. Ayache, "Extension of the ICP algorithm to non-rigid intensity-based registration of 3D volumes," *Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pp. 84-93, June 1996.
- [51] M. Fleute and S. Lavallée, "Building a complete surface model from sparse data using statistical shape models: application to computer assisted Knee surgery," *First International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 879-887, Cambridge, Massachusetts, October 1998.
- [52] K. S. Fu and J. K. Mui, "A survey of image segmentation," *Pattern Recognition*, Vol. 13, pp. 3-16, 1981.
- [53] J. Gee, L. L. Briquer, C. Barillot and D. Haynor, "Probabilistic matching of brain images," *Information Processing in Medical Imaging*, pp. 113-125, 1995.

- [54] D. Geman and S. Geman, "Stochastic relaxation, gibbs distribution and bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, pp. 721-741, 1984.
- [55] D. Geman, S. Geman C. Graffigne and P. Dong, "Boundary detection by constrained optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, pp. 609-628, 1990.
- [56] G. Gerig, O. Kubler, R. Kikinis and F. A. Jolesz, "Nonlinear anisotropic filtering of MRI data," *IEEE Transactions on Medical Imaging*, Vol. 11, pp. 221-232, 1992.
- [57] G. V. Glass and K. D. Hopkins, *Statistical Methods in Education and Psychology*, Second Edition, New Jersey: Prentice-Hall Inc., 1984.
- [58] D. B. Goldgof, H. Lee and T. S. Huang, "Motion analysis of nonrigid surfaces," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 375-380, June 1988.
- [59] G. H. Golub and C. F. Van Loan, *Matrix Computations*, John Hopkins University Press, Baltimore, 1983.
- [60] M. Á. González Ballester, A. Zisserman and M. Brady, "Combined statistical and geometrical 3D segmentation and measurement of brain structures," *IEEE Workshop on Biomedical Image Analysis*, pp. 14-23, Santa Barbara, California, June 1998.
- [61] M. Á. González Ballester, A. Zisserman and M. Brady, "Measurement of brain structure based on statistical and geometrical 3D segmentation," *First International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 499-508, Cambridge, Massachusetts, October 1998.

- [62] A. Goshtasby, "Picecwise linear mapping functions for image registration," *Pattern Recognition*, Vol. 19, pp. 459-466, 1986.
- [63] A. Goshtasby, "Picecwise cubic mapping functions for image registration," *Pattern Recognition*, Vol. 20, no. 5, pp. 525-533, 1987.
- [64] U. Grenander, "Pattern sythesis: Lectures in pattern theory," *Applied Mathematical Sciences*, Springer-Verlag, Vol. 18, 1976.
- [65] U. Grenander and M. I. Miller, "Representations of knowledge in complex systems," *Journal of the Royal Statistical Society*, Vol. 56, no. 3, pp. 549-603, 1994.
- [66] R. M. Haralick and L. G. Shapiro, "Survey, image segmentation techniques," *Computer Vision, Graphics and Image Processing*, Vol. 29, pp. 100-132, 1985.
- [67] J. Haslam, C. J. Taylor and T. F. Cootes, "A probabilistic fitness measure for deformable template models," *Proceedings of of the Fifth British Machine Vision Conf.*, pp. 33-42, September, 1994.
- [68] D. L. G. Hill, D. J. Hawkes, J. E. Crossman, M. J. Gleeson, T. C. S. Cox, E. C. M. L. Bracey, A. J. Strong and P. Graves, "Registration of MR and CT images for skull base surgery using point-like anatomical features," *British Journal of Radiology*, Vol. 64, no. 767, pp. 1030-1035, 1991.
- [69] D. L. G. Hill, D. J. Hawkes, N. Harrison and C. Ruff, "A strategy for automatic multimodality image registration incorporating anatomical knowledge and image characteristics," *Information Processing in Medical Imaging*, pp. 182-196, LNCS 687, editors: H. H. Barrett and A. F. Gmitro, Springer-Verlag, 1993.

- [70] P. V. C. Hough, "Method and means for recognizing complex patterns," *U.S. Patent 3069654*, 1962.
- [71] A. Hummel, B. Kimia and S. Zucker, "Deblurring gaussian blur," *Computer Vision, Graphics and Image Processing*, Vol. 38, pp. 66-80, 1987.
- [72] J. Illingworth and J. Kitter, "A survey of Hough transform," *Computer Vision, Graphics and Image Processing*, Vol. 44, pp. 87-116, 1988.
- [73] A. K. Jain, Y. Zhong and S. Lakshmanan, "Object matching using deformable templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, no. 3, pp. 267-278, 1996.
- [74] H. J. Jiang, R. A. Robb and K. S. Holton, "A new approach to 3-D registration of multimodality medical images by surface matching," *Visualization in Biomedical Computing 1992, Proceedings of SPIE 1808*, pp. 196-213, Editor: R. A. Robb, 1992.
- [75] R. A. Johnson and D. W. Wichern, *Multivariate statistics, a practical approach*, Chapman & Hall, 1988.
- [76] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag New York Inc., 1986.
- [77] S. Joshi, M. Miller, G. Christensen, A. Banerjee, T. Coogan and U. Grenander, "Hierarchical brain mapping via a generalized Dirichlet solution for mapping brain manifolds," *Vision Geometry IV*, Vol. 2573, pp. 278-289, SPIE, July 1995.
- [78] M. Kass, A. Witkin and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, Vol. 1, no. 4, pp. 312-331, 1988.

- [79] A. Kelemen, G. Székely and G. Gerig, "Three-dimensional model based segmentation of brain MRI," *IEEE Workshop on Biomedical Image Analysis*, pp. 4-13, Santa Barbara, California, June 1998.
- [80] R. Kimmel, A. Amir and A. M. Bruckstein, "Finding shortest paths on surfaces using level sets propagation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, no. 6, pp. 635-640, 1995.
- [81] R Kimmel and N Kiryati, "Finding shortest paths on surfaces by fast global approximation and precise local refinement," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 10, no. 6, pp. 643-656, 1996.
- [82] R. Kimmel and J. A. Sethian, "Computing Geodesic Paths on Manifolds," *Proceedings of National Academy of Sciences*, Vol. 95, no. 15, pp. 8431-8435, 1998.
- [83] J. Koenderink, "The structure of images," *Biological Cybernetics*, Vol. 50, pp. 363-370, 1984.
- [84] V. F. Leavers, "Survey: Which Hough transform?," *CVGIP: Image Understanding*, Vol. 58, no. 2, pp. 250-264, Sep. 1993.
- [85] W. E. Lorensen and H. E. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," *Computer Graphics*, Vol. 21, no. 4, pp. 163-169, 1987.
- [86] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal and P. Suetens, "Multimodality image registration by maximization of mutual information," *Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pp. 14-22, 1996.

- [87] H. Maître and Y. Wu, "A dynamic programming algorithm for elastic registration of distorted pictures based on autoregressive model," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 37, no. 3, pp. 288-297, February 1989.
- [88] R. Malladi, J. A. Sethian and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol: 17, pp. 158-175, 1995.
- [89] L. E. Malvern, *Introduction to the mechanics of a continuous medium*, Englewood Cliffs, NJ: Prentice-Hall, 1969.
- [90] D. Marr and E. Hildreth, "Theory of edge detection," *Proceedings of Royal Society, London*, Vol. 207, pp. 187-217, 1980.
- [91] S. Marrett, A. C. Evans, L. Collins and T. M. Peters, "A volume of interest (VOI) atlas for the analysis of neurophysiological image data," *Proceedings of SPIE Vol 1092, Medical Imaging III: Image Processing*, SPIE Press, Bellingham, WA, pp. 467-477, 1989.
- [92] T. McInerney and D. Terzopoulos, "Topologically adaptable snakes," *Proceedings of the International Conference on Computer Vision*, pp. 840-845, 1995.
- [93] T. McInerney and D. Terzopoulos, "Deformable models in medical image analysis," *Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, pp. 171-180, 1996.
- [94] J. Mceachen, A. Nehorai and J. Duncan, "A recursive filter for temporal analysis of cardiac motion," *IEEE Workshop on Biomedical Image Analysis*, pp. 124-133, 1994.

- [95] D. Metaxas and D. Terzopoulos, "Shape and nonrigid motion estimation through physics-based synthesis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, no. 6, pp. 580-591, 1993.
- [96] M. I. Miller, G. E. Christensen, Y. Amit and U. Grenander, "Mathematical textbook of deformable neuroanatomies," *Proceedings of National Academy of Science*, Vol. 90, no. 24, pp. 11944-11948, 1993.
- [97] M. Moshfeghi, S. Ranganth and K. Nawyn, "Three-dimensional elastic matching of volumes," *IEEE Transactions on Image Processing*, Vol. 3, no. 2, pp. 128-138, March 1994.
- [98] S. Osher and J. A. Sethian, "Fronts propagation with curvature dependent speed: Algorithms based on hamilton-jacobi formulations," *Journal of Computational Physics*, Vol. 79, pp. 12-49, 1988.
- [99] N. R. Pal and S. K. Pal, "A review of image segmentation techniques," *Pattern Recognition*, Vol. 26, pp. 1277-1294, 1993.
- [100] C. A. Pelizzari, G. T. Y. Chen, D. R. Spelbring, R. R. Weichselbaum and C. T. Chen, "Accurate three-dimensional registration of CT, PET and MR images of the brain," *Journal of Computer Assisted Tomography*, Vol. 13, no. 1, pp. 20-26, 1989.
- [101] A. P. Pentland and S. Sclaroff, "Closed-form solutions for physically based shape modeling and recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, no. 7, pp. 715-729, 1991.

- [102] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, pp. 629-639, 1990.
- [103] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes in C*, Cambridge University Press, 1988.
- [104] R. D. Rabbitt, G. E. Christensen and M. I. Miller, "Mapping of hyperelastic deformable templates using finite element method," *Vision Geometry IV*, Vol. 2573, pp. 252-265, July 1995.
- [105] P. Radeva, J. Serrat and E. Martí, "A snake for model-based segmentation," *Proceedings of the Fifth International Conference on Computer Vision*, pp. 816-821, 1995.
- [106] J. N. Reddy, *Energy and Variational Methods in Applied Mechanics*, New York: Wiley-Interscience, 1984.
- [107] R. A. Robb and D. P. Hanson, "ANALYZE: A software system for biomedical image analysis," *Proceedings of First Conf. Visualization Biomed. Computing*, pp. 507-518, IEEE Comp. Soc., Los Alamitos, CA, 1990.
- [108] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, Second Edition, San Diego: Academic Press Inc., 1982.
- [109] P. K. Sahoo, S. Soltani, A. K. C. Wong and Y. C. Chen, "A survey of thresholding techniques," *Computer Vision, Graphics and Image Processing*, Vol. 41, pp. 233-260, 1988.
- [110] F. Schmitt, X. Chen and W. H. Du, "Geometric modeling from range image data," *Proceedings of Eurographics'91*, pp. 317-328, Ed. Elsevier, 1991.

- [111] E. L. Schwartz, A. Shaw and E. Wolfson, "A numerical solution to the generalized mapmaker's problem: flattening nonconvex polyhedral surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, no. 9, pp. 1005-1008, 1989.
- [112] J. A. Sethian and J. Strain, "Crystal growth and dendritic solidification," *Journal of Differential Geometry*, Vol. 99, pp. 231-253, 1992.
- [113] J. A. Sethian, "A fast marching level set method for monotonically advancing fronts," *Proceedings of the National Academy of Sciences*, Vol. 93, Issue 4, pp. 1591-1595, 1996.
- [114] N. Sicotte, R. Woods and J. Mazziotta, "Automated image registration using a 105 parameter non-linear model," *NeuroImage*, Vol. 3, no. 3, pp. S123, June 1996 (2nd Intl. Conf. Funct. Mapping Human Brain).
- [115] G. W. Snedecor and W. G. Cochran, *Statistical Methods*, Seventh Edition, Iowa: The Iowa State University Press, 1980.
- [116] L. H. Staib, *Parametrically Deformable Contour Models for Image Analysis*, Ph.D thesis, Yale University, 1990.
- [117] L. H. Staib and J. S. Duncan, "Boundary finding with parametrically deformable models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, no. 11, pp. 1061-1075, 1992.
- [118] L. H. Staib and X. Lei, "Intermodality 3D medical image registration with global search," *Proceedings of the IEEE Workshop on Biomedical Image analysis*, pp. 225-234, 1994.

- [119] G. Székely, A. Keleman, C. Brechbüler and G. Gerig, “Segmentation of 2-D and 3-D objects from MRI volume data using constrained elastic deformations of flexible Fourier contours and surface models,” *Medical Image Analysis*, Vol. 1, no. 1, pp. 19-34, 1996.
- [120] R. Szeliski and S. Lavallée, “Matching 3-D anatomical surfaces with non-rigid deformations using octree-splines,” *Proceedings of the IEEE Workshop on Biomedical Image Analysis*, pp.144-153, 1994.
- [121] H. Tagare, D. O’Shea and A. Rangarajan, “A geometric criterion for shape-based non-rigid correspondence,” *Proceedings of the Fifth International Conference on Computer Vision*, pp. 434-439, 1995.
- [122] D. Terzopoulos and D. Metaxas, “Dynamic 3D models with local and global deformations: Deformable superquadrics” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vo. 13, no. 7, pp. 703-714, 1991.
- [123] J. P. Thirion, “Non-rigid matching using demons,” *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 245-251, June 1996.
- [124] B. C. Vemuri, A. Radisavljevic and C. Leonard, “Multiresolution 3D stochastic shape models for image segmentation,” *Information Processing in Medical Imaging*, pp. 62-76, 1993.
- [125] B. C. Vemuri and Y. Guo, “Snake pedals: geometric models with physics-based control,” *Proceedings of the Sixth International Conference on Computer Vision*, pp. 427-432, 1998.

- [126] Y. Wang and L. H. Staib, "Boundary finding with correspondence using statistical shape models," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 338-345, Santa Barbara, California, June 1998.
- [127] Y. Wang and L. H. Staib, "Elastic model based non-rigid registration incorporating statistical shape information," *First International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 1162-1173, Cambridge, Massachusetts, October 1998.
- [128] Y. Wang and L. H. Staib, "Integrated approaches to non-rigid registration in medical images," *Fourth IEEE Workshop on Applications of Computer Vision*, pp. 102-108, Princeton, New Jersey, October 1998.
- [129] Y. Wang and L. H. Staib, "A unified framework for boundary finding with different prior models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, submitted, 1999.
- [130] Y. Wang and L. H. Staib, "Physical model based non-rigid registration incorporating statistical shape information," *International Journal of Medical Image Analysis*, in press, 1999.
- [131] Y. Wang and L. H. Staib, "3D volumetric segmentation with statistically deformable models," In preparation.
- [132] W. Wells, P. Viola, H. Atsumi, S. Nakajima and R. Kikinis, "Multi-modal volume registration by maximization of mutual information," *Medical Image Analysis*, Vol. 1, no. 1, pp. 35-52, March 1996.

- [133] A. P. Witkin, "Scale-space filtering," *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 1019-1022, 1983.
- [134] A. P. Witkin, D. Terzopolos and M. Kass, "Signal matching through scale space," *International Journal of Computer Vision*, pp. 133-144, 1987.
- [135] G. Wolberg, *Digital Image Warping*, IEEE Comp. Soc., Los Alamitos, CA, 1990.
- [136] E. Wolfson and E. L. Schwartz, "Computing minimal distances on polyhedral surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, no. 9, pp. 1001-1005, 1989.
- [137] R. Woods, S. Cherry and J. Mazziotta, "Rapid automated algorithm for aligning and reslicing PET images," *Journal of Computer Assisted Tomography*, Vol. 16, pp. 620-633, 1992.
- [138] R. Woods, J. Mazziotta and S. Cherry, "MRI-PET registration with automated algorithm," *Journal of Computer Assisted Tomography*, Vol. 17, pp. 536-546, 1993.
- [139] M. Worring, A. W. M. Smeulders, L. H. Staib and J. S. Duncan, "Parameterized feasible boundaries in gradient vector fields," *Computer Vision and Image Understanding*, Vol. 63, no. 1, pp. 135-144, January 1996.
- [140] A. Yuille and T. Poggio, "Scaling theorems for zero crossings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, no. 1, pp. 15-25, 1986.
- [141] A. L. Yuille, P. W. Hallinan and D. S. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, Vol. 8, no. 2, pp. 133-144, 1992.

- [142] J. Zhu and J. A. Sethian, "Projection methods coupled to level set interface techniques," *Journal of Computational Physics*, Vol. 102, pp. 128-138, 1992.
- [143] S. C. Zhu and A. Yuille, "Region competition and its analysis: A unified theory for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, no. 9, pp. 884-900, 1996.