Statistical Models in Medical Image Analysis

by

Michael Emmanuel Leventon

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

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2000
[June 2000]

Author ....
Department of Electrical Engineering and Computer Science
May 4, 2000

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

Certified by ....
Olivier D. Faugeras
Adjunct Professor of Computer Science and Engineering
Thesis Supervisor

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

Statistical Models in Medical Image Analysis

by

Michael Emmanuel Leventon

Submitted to the Department of Electrical Engineering and Computer Science
on May 4, 2000, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

Computational tools for medical image analysis help clinicians diagnose, treat, monitor changes, and plan and execute procedures more safely and effectively. Two fundamental problems in analyzing medical imagery are registration, which brings two or more datasets into correspondence, and segmentation, which localizes the anatomical structures in an image. The noise and artifacts present in the scans, combined with the complexity and variability of patient anatomy, limit the effectiveness of simple image processing routines. Statistical models provide application-specific context to the problem by incorporating information derived from a training set consisting of instances of the problem along with the solution. In this thesis, we explore the benefits of statistical models for medical image registration and segmentation.

We present a technique for computing the rigid registration of pairs of medical images of the same patient. The method models the expected joint intensity distribution of two images when correctly aligned. The registration of a novel set of images is performed by maximizing the log likelihood of the transformation, given the joint intensity model. Results aligning SPGR and dual-echo magnetic resonance scans demonstrate sub-voxel accuracy and large region of convergence.

A novel segmentation method is presented that incorporates prior statistical models of intensity, local curvature, and global shape to direct the segmentation toward a likely outcome. Existing segmentation algorithms generally fit into one of the following three categories: boundary localization, voxel classification, and atlas matching, each with different strengths and weaknesses. Our algorithm unifies these approaches. A higher dimensional surface is evolved based on local and global priors such that the zero level set converges on the object boundary. Results segmenting images of the corpus callosum, knee, and spine illustrate the strength and diversity of this approach.

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

Thesis Supervisor: Olivier D. Faugeras
Title: Adjunct Professor of Computer Science and Engineering

Acknowledgments

I would like to thank my thesis advisors, Eric Grimson and Olivier Faugeras for all of their help and guidance in my research and overall graduate experience.

Five years ago, Eric introduced me to the field of medical image analysis, and through his excitement for research that can have such a positive impact, I found my interests immediately directed toward medical applications of computer vision. Over the years, Eric has simultaneously given me the freedom to work on the problems that I found most interesting and the direction to help me to stay aware of the big picture and to evaluate benefits and pitfalls of various projects I've considered. I'm grateful to Eric for his energy, which is always a great motivating factor, and for leading a terrific group that brings together the areas of computer vision and medical imaging.

I'm very thankful that Olivier joined our lab and that I've had the chance to work with him. Interactions with Olivier have greatly strengthened my mathematical knowledge, and I feel I've directly benefited from his energy and excitement for applications of mathematical theory. I thank Olivier for the many enjoyable, animated discussions with whiteboards full of sketches and derivations that have greatly helped me flush out and formalize many of the ideas in this thesis.

I'd like to thank the readers on my committee, Ron Kikinis, Sandy Wells, and Alan Willsky for their advice and direction. I thank Ron for always bridging the gap between interesting medical imaging algorithms and clinically practical systems. Ron has taught me a tremendous amount about assessing the needs of clinicians and building solutions that they can use effectively. I thank Sandy for always having the interest and the time to listen, digest, and suggest improvements for any idea that I would propose to him. I thank Alan for the enjoyable discussions shape theory and level sets and for his ideas on improvements and future areas to explore.

Over the last five years, I've had the opportunity to work at two amazing labs filled with great people. The MIT Artificial Intelligence Lab is a terrific place to be for graduate work. Discussions with people about work and about anything but work have greatly contributed to the overall enjoyment of my experience here. I want to thank all my colleagues and friends at the lab for their help and support. The Surgical Planning Lab at Brigham & Women's Hospital is a tremendous resource that brings together researchers and clinicians from so many different departments. Through the SPL, I was able to work with people from Radiology, Neurosurgery, Anesthesiology, and Psychiatry, to name a few. I was also given the unique opportunity to develop systems regularly used in clinical settings, while learning a tremendous amount by witnessing over thirty surgical procedures firsthand. I thank all those at the Brigham who helped make these experiences possible.

Most importantly, I would like to express my gratitude to my family, including my wife Larisa Matejic, my parents Joyce Arnowitz and Larry Leventon, my stepfather, Mickey Arnowitz, and my grandmother, Eunice Baradon for all their love, support, and encouragement.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology and at the Surgical Planning Lab of Brigham and Women's Hospital. This work was supported by NSF Contract IIS-9610249, NSF Contract DMS-9872228, and NSF ERC (Johns Hopkins University agreement) 8810-274. The MRI data was provided by Martha Shenton, Jane Anderson, and Robert W. McCarley, Department of Psychiatry and Surgical Planning Lab of Brigham & Women's Hospital, and Brockton VA Medical Center. The authors would like to acknowledge Dr. Shenton's NIMH grants, K02 M-01110 and R01 MH-50747, and Dr. McCarley's grant, R01-40799, and the Brockton Schizophrenia Center for the Department of Veterans Affairs.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The advances in internal imaging over the last thirty years have greatly improved the type of medical care that is available to patients. Clinicians now have the ability to non-invasively peer inside the human body to diagnose, treat, monitor changes, and plan and execute procedures more safely and effectively than before such medical imaging techniques existed. Imaging modalities such as magnetic resonance (MR), computed tomography (CT), ultrasound, and positron emission tomography (PET) all provide different measures of the structure and function of internal anatomy. However, no imaging method in existence today directly provides all the information that a doctor or surgeon might need to diagnose a condition or treat a patient.

While most imaging methods produce full three dimensional volumes of the anatomy, people cannot visualize the entire volume simultaneously. Traditionally, the medical scans are viewed as a series of consecutive two dimensional slices of the full 3D volume. This is often referred to as the "lightbox method" due to the medical films that are viewed on lightboxes. Figure 1-1 illustrates such 2D slices of an MR and CT scan. Many of the questions that doctors tend to ask rely on coherent, full 3D information, and are not easily answered when the scan is viewed slice by slice. Examples of such questions include:

- "How close is the brain tumor to the motor cortex?"
- "How has the volume of this lesion changed over time?"

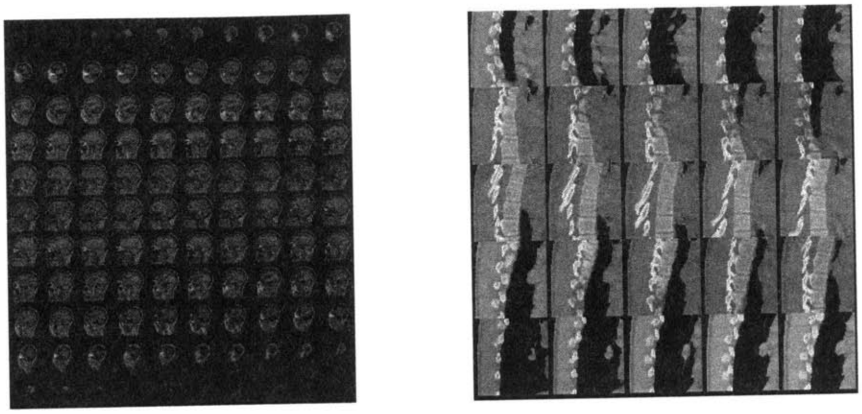


Figure 1-1: LEFT: Sagittal slices of an MR of the head. RIGHT: Sagittal slices of the CT of the spine.

- "What is the most direct path to the tumor that avoids all critical structures?"

In order to draw the necessary conclusions of diagnosis or treatment, the doctor must often mentally transform the 2D slices back into 3D, which can be difficult. In recent years, the development of computational tools for medical image analysis has made it easier for clinicians to visualize and interact with the internal structures and answer such questions important for better treatment for patients.

1.1 Medical Image Segmentation and Registration

One of the major classes of problems commonly addressed in computational processing of medical images is *segmentation*. Segmentation is the process of labeling each volume element or voxel in a scan based on the anatomical structure to which it corresponds. The label is assigned using properties of the observed intensities in the image as well as known anatomical information about normal subjects. Figure 1-2 shows an MR scan of the brain and the knee with the corresponding segmentations or *label maps*.

A fully segmented or labeled scan allows surgeons to both better qualitatively visualize the shapes and relative positions of internal structures and more accurately measure their volumes and distances quantitatively. Such labeling is critical to answering questions like those listed above that refer to the exact size or proximity of the structures of interest. One cannot exactly measure the distance between two structures without knowing the boundary of each. Segmentation also allows better visualization of the entire anatomy in an internal scan. Three dimensional surface models can be generated from the label maps using an algorithm such as Marching Cubes [66] or SurfaceNets [32]. Figure 1-2 shows 3D surface models of the brain and knee. Segmentation has uses both for patients with pathology and for normal volunteers. Scans of people without pathology can be used as a method of comparison to define abnormality. Detailed segmentations and subsequent 3D models can be used to generate an *anatomical atlas* for visualization, teaching, and as training data for other algorithms [35].

Due to the noise typically present in medical imagery, the similar appearance of different tissues in many modalities, and the complexity of anatomical structures, accurate segmentation is difficult. While segmentation is performed automatically whenever possible, most applications still require at least some amount of manual intervention, and some are still performed completely manually. Full manual segmentation typically consists of the user outlining each anatomical structure in 2D, one slice at a time, through the entire volume. Such an approach to segmentation is extremely tedious and time consuming, and still suffers from errors, due to the difficulty in maintaining consistency across slices. Studies of the consistency of segmentations across experts and the reproducibility of the a segmentation by the same expert over time shows a high degree ($\sim 15\%$) of variance [50]. Yet due to the lack of satisfactory automatic segmentation algorithms, for many applications manual outlining is still common.

In addition to segmentation, registration is another important problem frequently addressed in medical image analysis. Registration is the process of aligning data that arise from different sources into one consistent coordinate frame. For example, various anatomical structures appear more clearly in different types of internal scans. Soft tissue, for example, is imaged well in MR scans, while bone is more easily discernible in CT scans. Blood vessels are often highlighted better in an MR angiogram than in a standard MR scan. Multiple scans of the same patient will generally be unregistered when acquired, as the patient will may be in different positions in each scanner, and

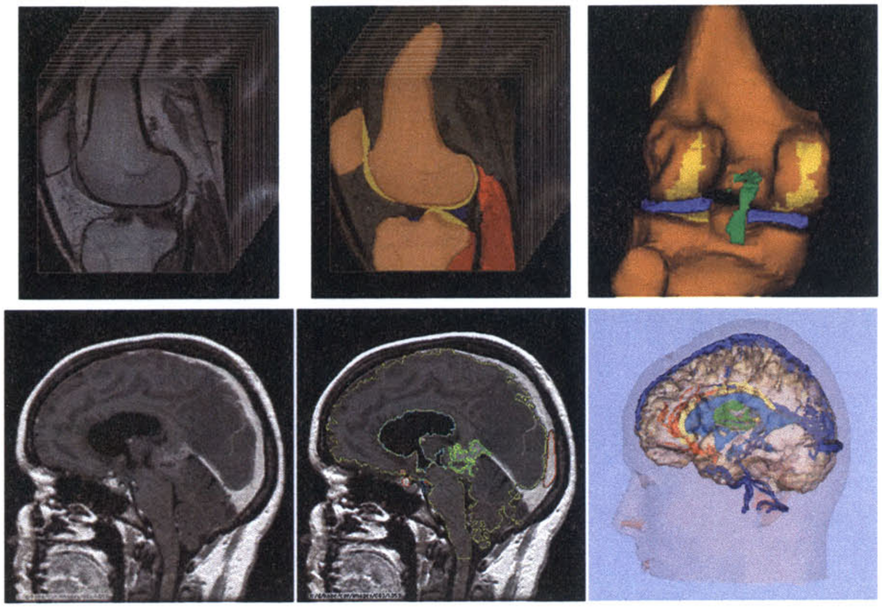


Figure 1-2: The process of segmentation and model generation. LEFT: The raw MR scans of the knee and the head. MIDDLE: The label-map overlayed on the MR. RIGHT: Three dimensional surface models generated from the label-map using Marching Cubes[66].

each scanner has its own coordinate system. In order to fuse the information from all scans into one coherent frame, the scans must be registered. The very reason why multiple scans are useful is what makes the registration process difficult. As each modality images tissue differently and has its own artifacts and noise characteristics, accurately modeling the intensity relationship between the scans, and subsequently aligning them, is difficult.

The class of transformations considered by the registration algorithm depends on the application. Rigid registration corresponds to finding the six degree of freedom pose that best aligns the two datasets. This type of alignment is used when there is no significant tissue shift between acquisitions. Elastic registration is required when dealing with soft tissue that deforms between scans. Many more degrees are freedom are required to model these non-rigid transformations, which makes the registration problem much more challenging.

The segmentation and registration problems are closely related, in that the solution of one greatly assists in the computation the other. If one could bring two

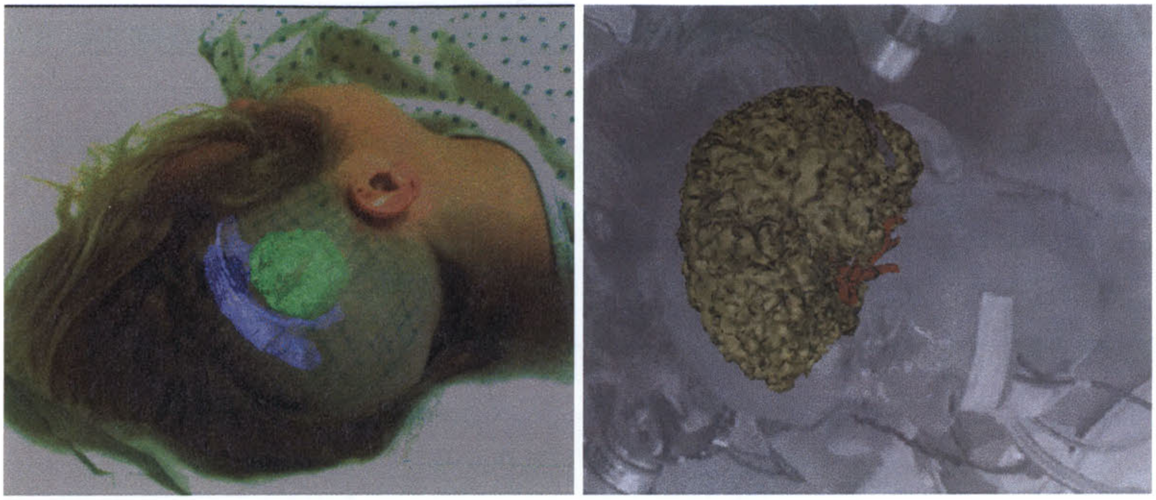


Figure 1-3: Augmented Reality Visualization where internal structures are overlaid on a video image of the patient. LEFT: The ventricles appear in blue and the tumor appears in green. RIGHT: An aneurysm appears off the blood vessels shown in red.

different images perfectly into registration, then an anatomical atlas could be registered to a novel scan, producing a full segmentation. If, on the other hand, one could reliably segment all the anatomical structures in an image, then registration becomes straightforward, as the alignment of two label maps is a simpler task.

1.2 Applications of Computational Medical Image Analysis

The registration and segmentation tools for described above have many applications that can provide better treatment for patients. In this section, we describe some applications where medical image analysis is currently being used, such as surgical navigation, functional brain mapping, surgical simulation, and therapy evaluation.

1.2.1 Surgical Navigation

Many surgical procedures require highly precise localization, often of deeply buried structures, in order for the surgeon to extract or repair targeted tissue with minimal damage to nearby structures. Three dimensional, registered visualization and tracking techniques can help surgeons to safely navigate through the surgical field by trans-

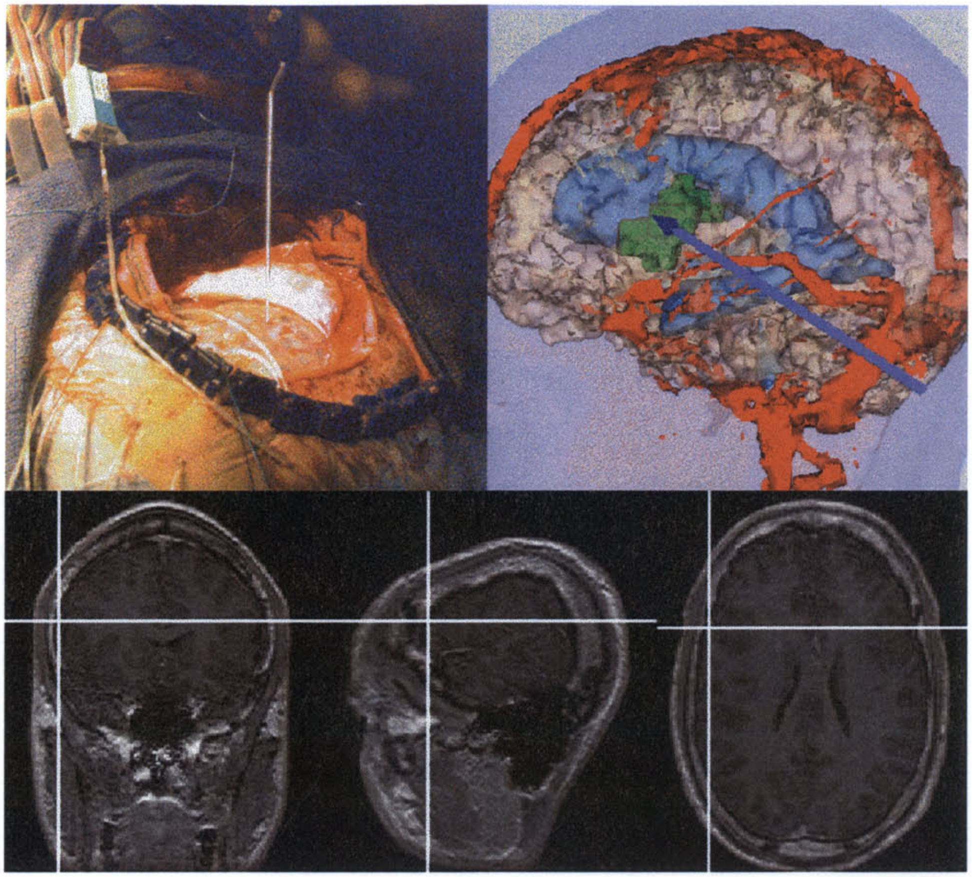


Figure 1-4: As the surgeon moves the trackable probe around the patient's head (upper left), an arrow shows the corresponding position in the rendered image of 3D models (upper right) and the cross-hairs highlight the probe position in the orthogonal slices of the grayscale MR scan (bottom three images).

ferring reconstructions of internal anatomy into the frame of the patient. Creating 3D visualizations of anatomical structures requires methods of accurately segmenting those structures from the internal scans. Combining 3D models of soft tissue (acquired from MR) with bone (from CT) and function data (fMRI) into one coherent coordinate frame requires reliable multi-modal registration algorithms. Alignment techniques are also required to precisely register the 3D anatomical models with the patient in operating room.

Such image-guided surgical tools allow the surgeon to directly visualize important structures, and plan and act accordingly, using methods of visualization such as: (1) "enhanced reality visualization" [39], in which rendered internal anatomical structures are overlaid on the surgeon's field-of-view (Figure 1-3), and (2) instrument tracking

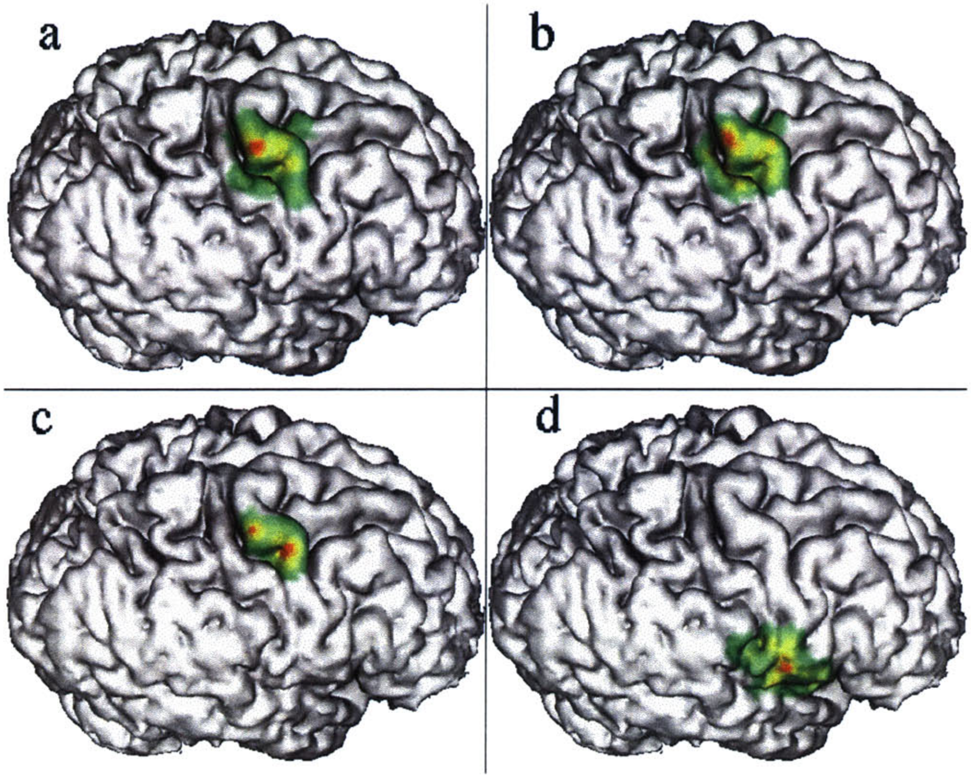


Figure 1-5: Non-invasive functional maps produced from trans-cranial magnetic stimulation. Muscle responses are shown blended on the cortex model. Red is large response; yellow is medium response, and green is small response. The muscles mapped are (a) index finger, (b) forearm, (c) biceps, and (d) jaw.

[60], in which medical instruments acting on the patient are localized in the 3D coordinate frame of the MR or CT imagery (Figure 1-4). The resultant benefits include: accelerated migration to minimally-invasive surgeries via improved hand-eye coordination and better transfer of a priori plans to the patient; shorter procedures through increased visualization of the surgical field; reduced risk of sensitive tissue damage; and more accurate and complete tissue resection, ablation, or biopsy.

1.2.2 Functional Brain Mapping

Segmentation of anatomical structures from an MR scan and subsequently generating models from those labelings can provide useful information to the surgeon about the relative positions of internal structures. However, functional areas of the brain, such as motor cortex or language area, cannot be segmented from a standard MR scan because they often appear the same as the surrounding tissue. Yet knowing the positions of functional areas can be very important to a surgeon during the planning of a surgery, and during a surgery itself. For example, brain tumors near functional areas can invade the surrounding tissue or can push it out of the way. A tumor that has invaded the motor cortex may be inoperable due to the risk of paralysis while a tumor that has displaced the motor cortex may still be operable.

The most common techniques for functional brain mapping utilize 3D medical scanners to image the brain while the subject undergoes an activity aimed at activating the functional area of interest. Scanners currently used for this purpose are single photon emission computed tomography (SPECT), positron emission tomography (PET), and functional magnetic resonance imaging (fMRI). Multi-modal registration of these scans to a structural image is necessary to provide context to ground the functional information with respect to the rest of the anatomy. Transcranial Magnetic Stimulation (TMS) is a functional mapping method that actively maps out regions of the brain by registering and tracking the position of a magnetic coil placed on the scalp with an MR image. When the coil is fired, the position with respect to the image is correlated with a measured functional response (such as a muscle twitch or momentary visual suppression). Registered functional maps are produced that combine the function data with the 3D models of the anatomy (see Figure 1-5).

1.2.3 Surgical Simulation

There are many challenges faced in the process of training surgeons in a new procedure. The feasibility of surgeons practicing on cadavers is limited due to the expense and lack of availability. For many types of procedures, the primary method of learning is first by observation and then by actually performing the procedure, only sometimes under supervision. Thus, the phrase "watch one, do one, teach one" quite often describes the very intense learning environment many surgeons find themselves facing.

Another approach to the problem of training surgeons is the development of surgical simulation systems. Simulation systems generally combine 3D visualization with

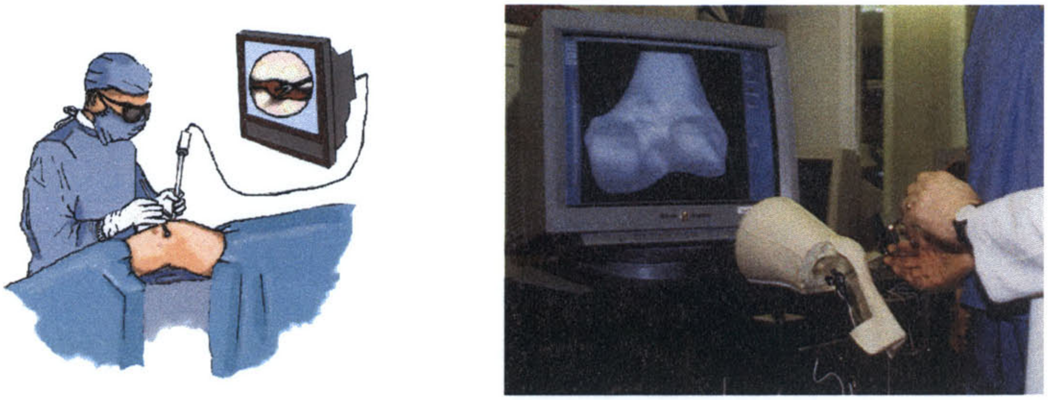


Figure 1-6: LEFT: Diagram of an arthroscopic knee surgery. The surgeon holds a scope with one hand and an instrument with the other. The view from the scope is projected on a monitor. RIGHT: A surgeon using the knee simulator. A force-feedback device lets the surgeon feel a 3D model of the femur, shown on the monitor. (Images courtesy MERL: http://www.merl.com)

haptic feedback to provide the user with as realistic an experience as possible [33]. The development of such a system relies heavily on the ability to generate accurate models of anatomical structures and their types of deformations. Realistic geometric models require accurate segmentation of the medical imagery and subsequent model generation that stays faithful to the original data [62]. Figure 1-6 illustrates a prototype simulation system for arthroscopic knee surgery. In such a procedure, the surgeon holds a scope in one hand and an instrument in the other, and performs the entire surgery percutaneously. Due to the limited field of view, there is potential that an inexperienced surgeon may damage critical structures in the knee. The simulator consists of a Phantom force-feedback device [91] allows the user to feel the virtual structures while at the same time visual feedback presented on a monitor shows the instrument being moved with respect to the knee [33].

1.3 Change Detection

When treating a patient, the physician must not only consider the health of the patient at that one moment in time, but also the change in that patient's condition over time. In a sense, patients are moving targets that the physician must track to determine the path of the disease and the effectiveness of the treatment. Medical

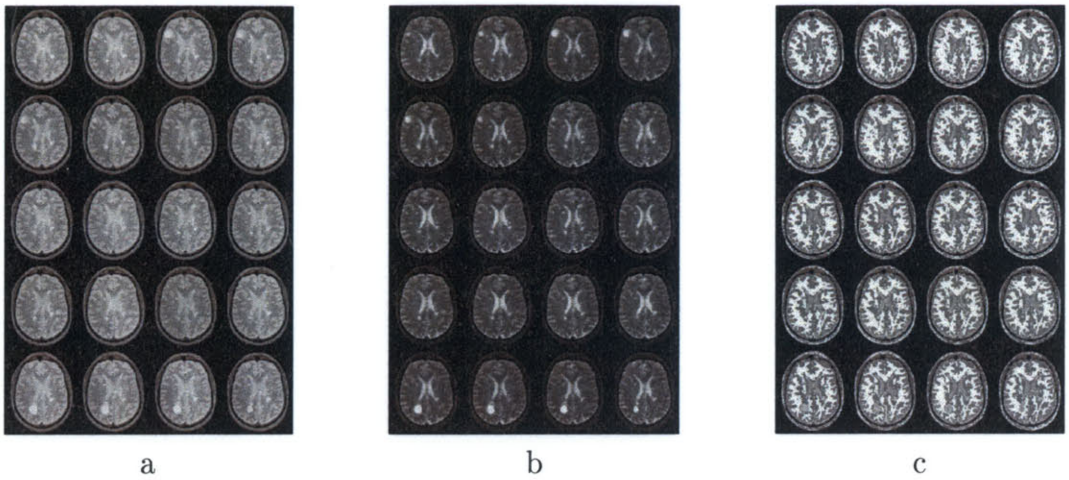


Figure 1-7: An example of the importance of registration and segmentation for change detection studies. (a) The same slice through 20 proton-density images of the same patient over time. The images were all registered into one coordinate frame. (b) The same 20 slices as T2-weighted images. (c) Adaptive segmentation of the gray matter and white matter [41, 108].

image analysis tools can provide accurate, quantitative measures of progress that can be used to adjust the therapy.

One example of a change detection system is the multiple sclerosis study at Brigham and Women's Hospital [26, 41]. Indications of multiple sclerosis include lesions that appear, change size, and disappear in the white matter of the brain. The change detection protocol takes a series of MR scans of a patient, registers them together and segments out the lesions. Figure 1-7 shows corresponding slices of 20 MR scans of a patient over time, along with the gray matter / white matter segmentations [41, 108]. The change in number and size of the lesions over time can be used to better understand the disease and also monitor the treatment.

1.4 Statistical Models

Segmentation and registration are generally quite challenging due to the complexity both of the human anatomy and of the imaging techniques used to visualize the internal structures. The noise and artifacts present in the images combined with the variability that exists across patient anatomy limit the effectiveness of simple image processing routines. To perform a satisfactory analysis of the images at hand, some amount of understanding or interpretation is required. This information is typically very general (e.g. not specific to the inputs) and is hard-coded into the algorithm. Such heuristics include:

- "A strong edge often signifies an object boundary."
- "Object boundaries are generally smooth."
- "Images, when aligned, explain each other well."

In semi-automatic approaches to the problems of medical image analysis, the important contextual knowledge is transferred to the algorithm by means of manual interaction. Example interactions include:

- "The object of interest is in this region."
- "Bone, in this CT image, has intensity above this threshold."
- "These three landmarks correspond in both images."

Such assertions are designed by the developer or provided by the operator based on their prior knowledge or experience in working with the medical imagery. It is difficult to automatically encode richer, more complex information using these methods.

Statistical models provide understanding or context to the problem by incorporating information derived from a *training set*, usually consisting of instances of the problem along with the solution. While the model framework is general and devised by the developer, the model itself can be quite rich and is specific to the problem being addressed. Example models presented in thesis include:

- "Here is an estimation of the probability distribution over the shape of all normal femurs."
- "A normal corpus callosum has curvature roughly according to this distribution."

- "In SPGR-MR, the intensity of the femur has the following distribution as a function of the distance to its boundary."
- "Images such as these, when aligned, have intensity co-occurrence similar to this joint intensity profile."

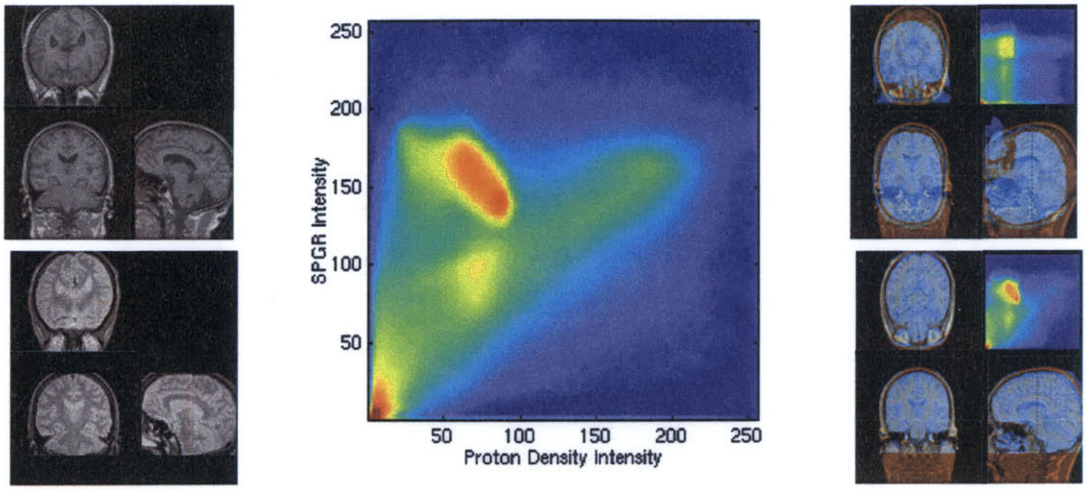
This type of knowledge can be used to fill in the gaps due to incomplete or noisy information, as typically found when analyzing medical images.

1.5 Contributions

This thesis addresses the issues of modeling prior knowledge of medical image analysis problems, and incorporating those derived statistical models into the solution of the problems themselves. An approach is presented to the multi-modal registration problem that analyzes the co-occurrence of the intensity values at corresponding points in registered images of different modalities, and uses that information to direct the alignment of novel image pairs. Various models are explored to assist in the segmentation or boundary detection problem. Models of the shape of anatomical structures across a population assist in the localization of those structures in novel images. The intensity distribution of a structure is modeled as a function of the distance to its boundary, providing richer image constraints than the commonly used large-gradient boundary detector. While most segmentation methods include a user-adjusted regularization term to keep the boundary smooth, the approach presented here builds a curvature model of the object to determine the smoothness constraints. Examples of the statistical models and their uses are highlighted here, and are fully explored in the upcoming chapters.

1.5.1 Joint Intensity for Multi-Modal Registration

The registration of multi-modal medical images is an important tool for fusing many data sources into one coherent frame, since various scan modalities highlight complementary anatomical structures (see Figure 1-8a). An approach is presented to the



a b c

Figure 1-8: (a) An SPGR and a Proton Density image of the same patient, in alignment. (b) The joint intensity histogram for the image pair. (c) A novel image being registered using the prior model, with intensity histogram shown. The top image shows the initial alignment, and the bottom shows the final pose. Each image shows three orthogonal slices of the two scans blended with two different colors. The joint intensity histogram under the pose in shown in the upper right quadrant.

multi-modal registration problem that incorporates prior information from a pair of pre-registered images to use in the alignment of a novel pair of such images. The joint intensity distribution is estimated based on the co-occurrence of intensity pairs at corresponding points in the registered training images. Figure 1-8b illustrates a joint intensity or co-occurrence map for a Spoiled Gradient-Recalled (SPGR) and a Proton Density (PD) MR image. Given a novel set of unregistered images, the algorithm computes the best registration by maximizing the log likelihood of the two images, given the transformation and the prior joint intensity model. Figure 1-8c shows the initial pose and computed registration for a novel image pair. Notice that the sampled joint intensity under the final pose closely resembles the co-occurrence model. This algorithm is tested on 36 pairs of SPGR / PD images and 36 pairs of PD / T2-weighted images, and typically results in sub-voxel registration accuracy.

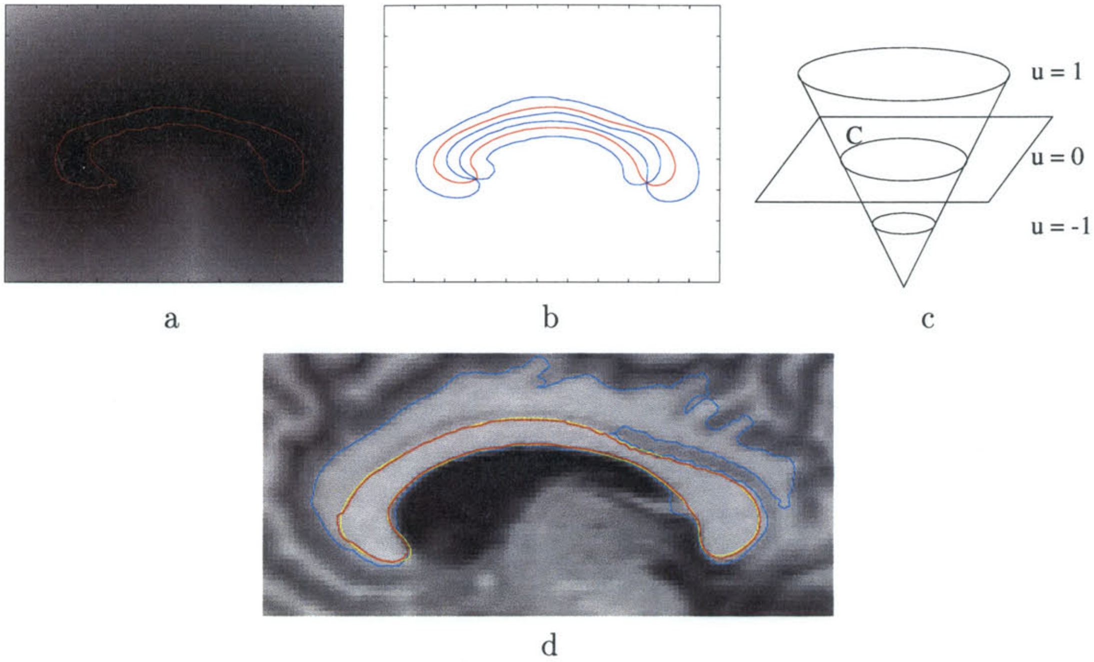


Figure 1-9: (a) The boundary of a corpus callosum overlaid on the signed distance map to the shape. (b) The primary mode of shape variance captured from the training set of 50 corpora callosa. (c) An illustration of the level set representation with a signed distance hyper-surface. (d) The automatic segmentation of a novel image using the shape prior is shown in red. The cyan contour is the result of the standard evolution *without* the shape influence.

1.5.2 Statistical Anatomical Shape Models for Segmentation

Having an expectation of the shape of an anatomical structure can greatly assist in the segmentation of that structure from an image. Shape models are derived to express the probability that a certain shape will occur, and these priors are then used to help localize the boundary of an object in the image. Building a probability distribution over shape is a challenging, possibly ill-posed problem. The effectiveness of the shape model is closely related to the representation of shape that is chosen. Many different shape representations have been proposed that all handle the issues of alignment, feasible transformations, choice of features, and correspondence differently. An overview of existing approaches is presented in the next chapter.

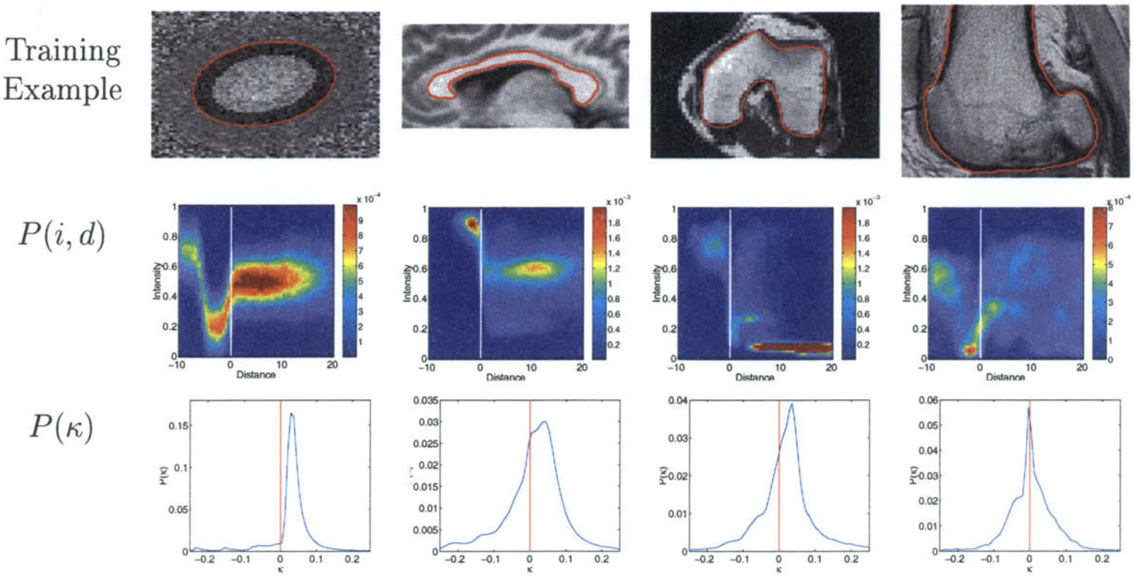
The shape representation chosen here is the signed distance map, where each point on a grid stores the distance to the boundary with negative values when inside the object (Figure 1-9a). Each such distance map is considered to be a point in a high dimensional space. A population of training shapes is then a cloud in that space, and a Gaussian model is fit to represent shape variance. Figure 1-9b illustrates the primary mode of shape variance captured by the shape model.

The shape representation and prior model are then used to direct a *level set* based segmentation [9]. Level set approaches to image segmentation involve solving the energy-based active contours minimization problem by the computation of geodesics or minimal distance curves [93]. In this approach, a curve is embedded as a zero level set of a higher dimensional surface (Figure 1-9c). The entire surface is evolved to minimize a metric defined by the curvature and image gradient. In addition to the curvature and image gradient terms, a global shape force is added to the evolution that pulls the surface towards more likely shapes, based on the prior model. Figure 1-9d shows the segmentation of a corpus callosum with and without shape influence.

1.5.3 Intensity-Locality Priors for Segmentation

Voxel classification approaches to image segmentation often utilize prior intensity models of tissues [12]. Local spatial priors and models of distances between tissue types have also been added to the tissue classification framework [51]. Extensions to level sets have been proposed that incorporate global intensity information instead of only local gradient forces. In [113], segmentation is performed by evolving a curve to maximally separate predetermined statistics inside and outside the curve. The combination of global and local information adds robustness to noise and weak boundaries. Such approaches, however, do not incorporate a prior intensity model based on training data.

This thesis presents a method of modeling the distribution of intensity over the entire image as a function of the signed distance from the boundary of the structure. This provides a means of representing both changes of the intensity of the structure relative to its boundary and the intensity profiles of neighboring structures as a function of their distances to the object of interest. A training example consists of an image and an object boundary. A distance map is computed from the object



(a) (b) (c) (d)

Figure 1-10: Top: One example of each of four training sets of objects. Middle: The joint intensity/distance-to-boundary PDF derived from the training set. Bottom: The curvature profile of each object class. Notice that the ellipse class has only positive curvature.

boundary, and a distribution of intensity vs. distance is derived. Measuring the co-occurrence of distance and intensity captures the intensity profile inside the object, at and around the boundary, and outside the object. Figure 1-10 shows examples of the joint distributions for a few images. During segmentation, the evolving surface is pulled towards the boundary of the object based on the intensity profile. Incorporating distance into the model allows convergence from far initialization positions, as opposed to a local attraction only at the boundary.

1.5.4 Curvature Models for Segmentation

Boundary estimation methods of segmentation generally consist of influence from the image along with a regularization effect to keep the boundary smooth (having small curvature) in the face of noise. Without such a smoothing term, the algorithm is more likely to get caught in a local minimum or find a highly erratic boundary not usually representative of the structure being segmented. Determining the amount of

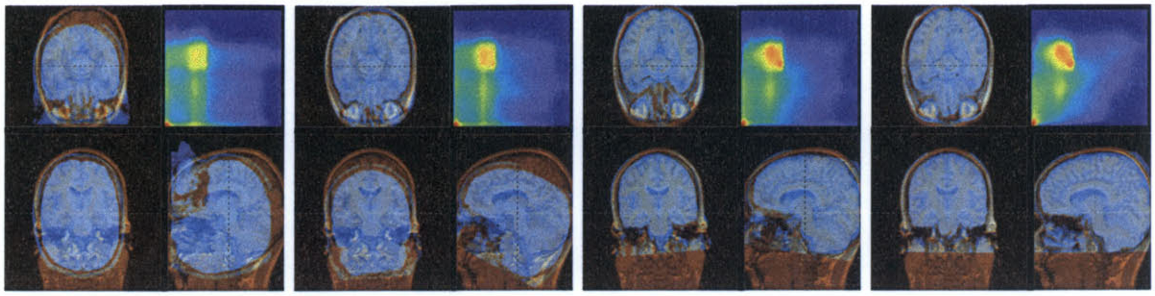


Figure 1-11: Starting, middle, and the final alignments computed by the registration algorithm. Each image shows the colorized SPGR and PD images overlaid in the three orthogonal slices. The images in the upper right depict the histogram of the intensity pairs at that alignment.

regularization that should be used for a particular application is difficult. Generally, a user-adjustable empirically-set smoothing parameter is used to trade off the weight of the image term with the simple smoothing model.

Instead of modeling smoothness as a manually-adjusted one degree-of-freedom parameter, a training set of examples of the same anatomical structure can be used to derive a richer model of curvature. A distribution of the curvature over the structure is derived from training examples, and subsequently used in the segmentation of a novel image. For example, when segmenting a structure that only consists of positive curvature (e.g. a convex shape), the evolving boundary will be influenced by the curvature prior to maintain convexity, making a non-convex solution very unlikely. Figure 1-10 illustrates curvature profiles derived from various structures.

1.6 Summary of Results

The algorithms presented in this thesis have been tested on a variety of medical image data. Multi-modal registration experiments were performed on T2-weighted and PD image pairs, as well as SPGR and PD images. Figure 1-11 shows an example of the progress of the registration algorithm on a SPGR/PD pair, starting from about 90° away, and registering within one voxel of the ground truth alignment. Figure 1-12 tabulates the results of the registration validation experiments, where many image pairs were aligned within a millimeter, and most were aligned within a voxel (which

PD / T2 Registration

#	Error	#	Error	#	Error	#	Error
1	0.74	10	0.79	19	1.42	28	1.26
2	0.89	11	0.73	20	0.75	29	0.76
3	0.86	12	0.68	21	0.69	30	2.24
4	0.79	13	1.52	22	1.33	31	0.90
5	0.90	14	0.63	23	0.66	32	0.78
6	0.80	15	0.80	24	1.01	33	1.28
7	0.81	16	0.89	25	1.07	34	0.74
8	0.80	17	0.82	26	0.70	35	1.08
9	0.82	18	1.36	27	0.81	36	0.89

SPGR / PD Registration

#	Error	#	Error	#	Error	#	Error
1	1.45	10	1.47	19	2.57	28	1.54
2	2.69	11	3.17	20	2.67	29	2.15
3	0.16	12	1.32	21	2.77	30	1.50
4	0.78	13	1.18	22	3.47	31	1.43
5	1.69	14	1.59	23	3.87	32	1.83
6	1.53	15	1.25	24	2.48	33	1.64
7	0.85	16	1.26	25	1.60	34	3.18
8	1.36	17	1.19	26	2.03	35	2.01
9	0.73	18	1.85	27	1.31	36	2.46

Figure 1-12: The results of registering 36 test images. The registration error (in mm) was computed by taking the average error of the eight vertices of the imaging volume.

has dimensions 0.9753 × 0.9375 × 3.0 mm). The registration results are discussed further in Section 4.5.

The segmentation algorithm was tested on MR images of the corpus callosum and knee, as well as CT images of the spine. The results of a few of the segmentations are shown in Figure 1-13. For controlled testing, experiments on synthetic imagery were also performed, and are described further in Chapter 5. Validation experiments were performed on a dataset of 50 corpora callosa, where each corpus was segmented automatically using our algorithm and also manually outlined by two operators, with one manual segmentation chosen as ground truth. Figure 1-14 compares the ground truth segmentation to both the automatic and other manual segmentation. The automatic segmentation compares favorably to the variance found between the two sets of manual outlines.

1.7 Overview of Validation Techniques

Validation of any medical image analysis technique is an important part of the algorithmic development process, especially when the results will be used in image-guided therapy. One technique for validating a registration or segmentation algorithm involves comparison of the output of the method to some form of ground truth or gold

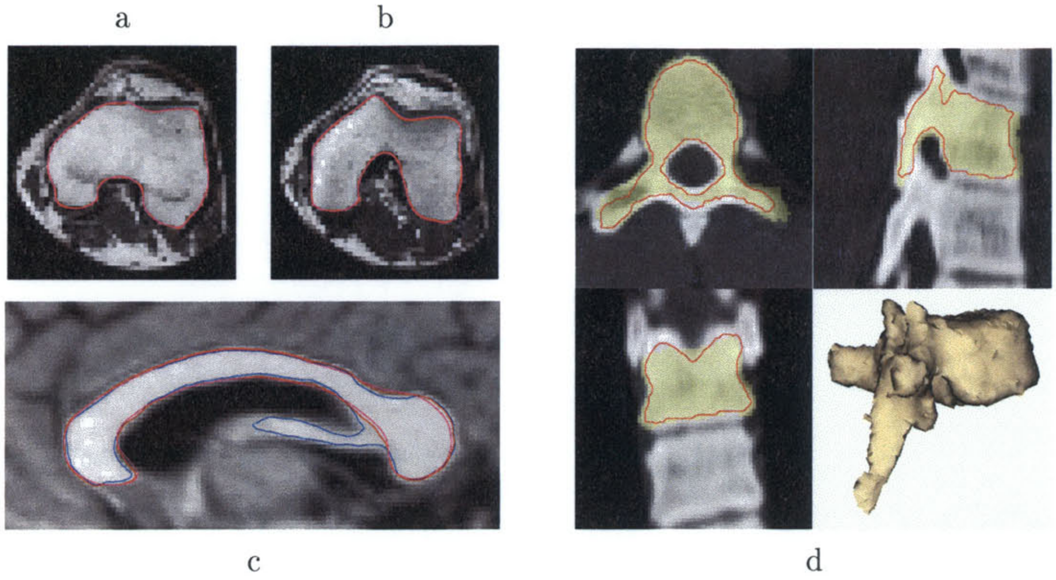


Figure 1-13: Segmentation results of various anatomical structures using statistical priors. (a,b) Segmentation of two 2D slices of the femur. The training data consisted of neighboring slices. (c) Segmentation of a corpus callosum, with training on 49 other corpora callosa. The blue outline is the segmentation without shape information. (d) Segmentation of the T6 vertebra of the spine, by training on T3-T5 and T7-T9.

standard. For segmentation, one can compare the percentage of volume overlap or maximum nearest neighbor distance between the gold standard and the label map produced by the algorithm. Error measures that compare the ground-truth transformation with the computed one can be used to evaluate registration methods.

While such evaluation may seem quite straightforward, in many applications, acquiring satisfactory ground truth is nearly as difficult as solving the problem itself. For segmentation of very complicated structures, such as the gray matter / white matter boundary in the brain, manual segmentations by experts can vary in voxel overlap by as much as 15% [50]. Consistent, accurate manual segmentation of tube-like structures such as blood vessels or nerve fibers is very difficult due to the way they wind in and out of the 2D slices generally used for outlining. Ground truth for rigid registration techniques can be computed using reliable fiducials that stay fixed in the same position on the patient throughout various scans [110]. Computing a gold standard for most types of non-rigid deformation of tissue that occurs between acquisitions continues to be an open research problem, making validation of non-rigid

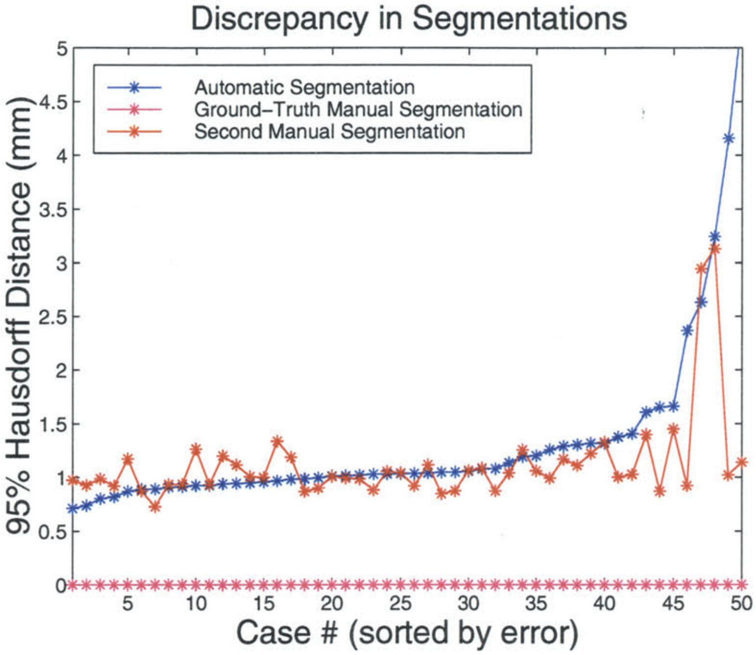


Figure 1-14: A plot comparing the automatic segmentation of 50 corpora callosa to the variability found between two manual segmentations. One manual segmentation (in magenta) was chosen as the ground truth. The cases were sorted by the automatic segmentation, shown in blue, with corresponding second manual segmentation shown in red.

registration algorithms quite challenging.

The framework of statistical models that we explore requires training data from which to derive the prior knowledge. The training data are assumed to be accurate solutions to the segmentation or registration problem being explored. Correctly segmented label maps are assumed to be available for some number of images of the same modality and anatomical region to be used as test data by the algorithm. At least one registered pair of scans is required to build the statistical model used in the multi-modal alignment process. Again, it is assumed that the modalities of scan and the structures being imaged are consistent from training data to test data.

The need for accurate training data to derive the statistical models may limit the practical use of such models in applications where the ground-truth is difficult to come by. However, in the many applications for which some amount of ground-truth is available, statistical models are applicable and the validation process is less of a challenge. To evaluate the performance of the algorithms presented in this thesis, we generally adhere to a “leave-some-out cross-validation” approach. Of all the datasets available (with ground truth), at least one set is left out of the training process used to derive the prior model. Each scan left out is then used as input to the algorithm in turn. The output or result is then compared to the gold standard for each corresponding test set.

1.8 Roadmap

In this dissertation, Chapter 2 highlights related work and additional approaches to image segmentation and the multi-modal registration problem, both with and without the use of statistical models. Chapter 3 describes the framework of using statistical models and the tools to derive them. Issues of representation and choice of density estimator are also discussed. The application of joint intensity models to the multi-modal registration problem is presented in Chapter 4, along with experiments and results. Chapter 5 describes our approach to model-based segmentation that incorporates prior shape, spatial intensity, and curvature information into the level set method of boundary detection. Examples on a variety of medical data are shown and used for a basis of evaluation. Chapter 6 ties together the use of statistical models for registration and segmentation, and concludes with a discussion of promising areas of future work.

Chapter 2

Background

This chapter discusses previous work in multi-modal volume registration and segmentation, and relates these methods to approaches presented here. We further discuss the use of statistical models in the solution to these medical imaging problems.

2.1 Multi-Modal Volume Registration

The registration of two images consists of finding the transformation that best maps one image into the other. If I_1 and I_2 are two images of the same patient and T is the correct transformation, then the voxel $I_1(x)$ corresponds to the same position in the patient as the voxel $I_2(T(x))$. In the simplest case, T is a rigid transformation, consisting of three degrees of freedom of rotation and three degrees of freedom of translation. The need for rigid registration arises primarily from the patient being in different positions in the scanning devices used to image the anatomy. The information from all the images is best used when presented in one unified coordinate system. Without such image fusion, the clinician must mentally relate the information from the disparate coordinate frames.

2.1.1 Fiducial / Feature Alignment

One method of aligning the two images is to define an intermediate, patient-centered coordinate system, instead of trying to directly register the images to one another. An

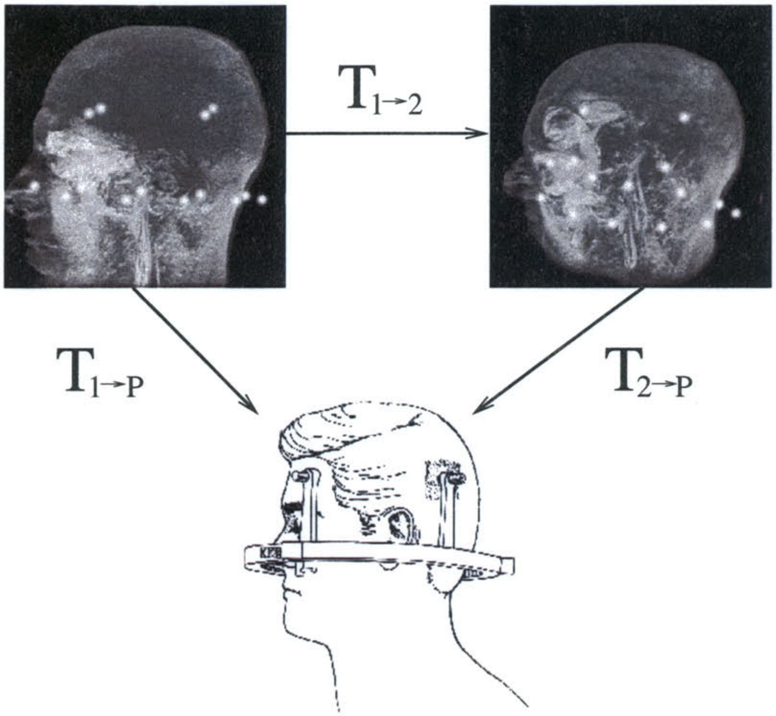


Figure 2-1: Maximum intensity projections of two 3D scans, with fiducials placed on the patient. The fiducials define a patient-centered coordinate system that can be used to align the two scans.

example of a patient-centered reference frame is the use of fiducial markers attached to a patient throughout the various image acquisitions. The fiducial markers define a coordinate system specific to the patient, independent of the scanner or choice of imaging modality. If the markers remain fixed and can be accurately localized in all the images, then the volumes can be registered by computing the best alignment of the corresponding fiducials [45, 76]. The main drawback of this method is that the markers must remain attached to the patient at the same positions throughout all image acquisitions. For applications such as change detection over months or years, this registration method is not suitable. Fiducial registration is typically used as ground-truth to evaluate the accuracy of other methods, as careful placement and localization of the markers can provide very accurate alignment [110].

When fiducial markers are not available to define the patient coordinate frame, corresponding anatomical feature points can be extracted from the images and used to compute the best alignment [74, 73]. This approach depends greatly on the ability

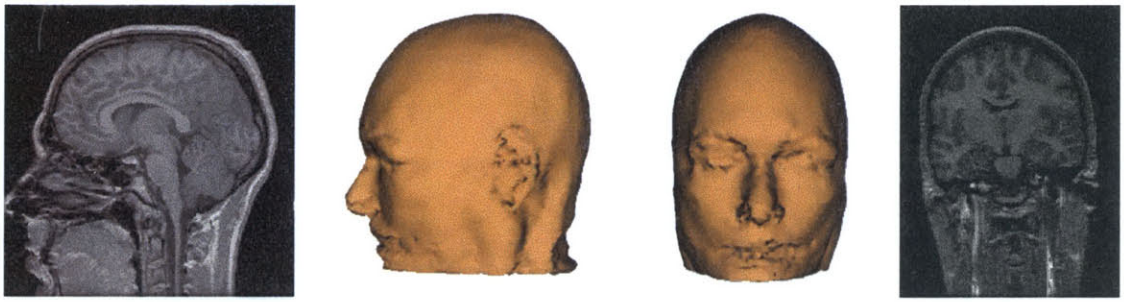
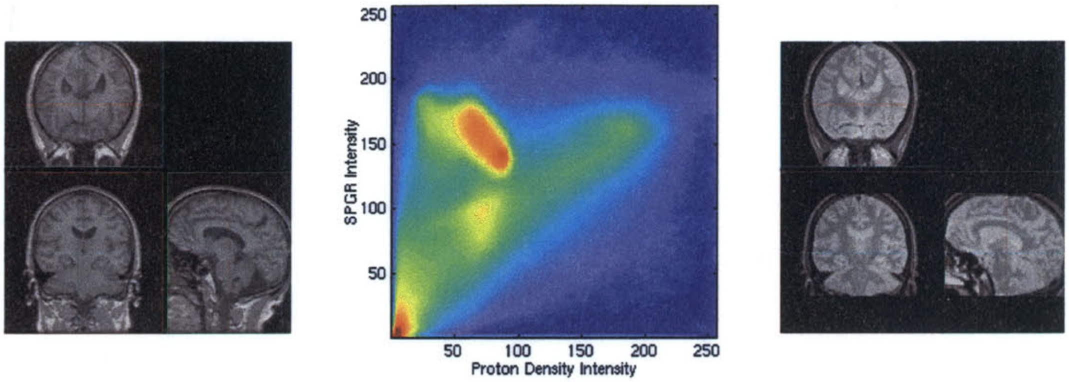


Figure 2-2: An illustration of surface-based registration, where the same surface (e.g. skin) is extracted from two different scans and then aligned, which implies the alignment of the original images.

to automatically and accurately extract reliable image features. In general, methods of feature extraction such as intensity thresholding or edge detection do not work well on medical scans, due to non-linear gain fields and highly textured structures. Even manual identification of corresponding 3D anatomical points can be unreliable. Without the ability to accurately localize corresponding features in the images, alignment in this manner is difficult.

2.1.2 Surface-Based Techniques

Instead of localizing feature points in the images, richer structures such as object surfaces can be extracted and used as a basis of registration. A common method of registering MR and CT of the head involves extracting the skin (or skull) surfaces from both images, and aligning the 3D head models [48, 57]. For PET/MR registration, the brain surface is typically used since the skull is not clearly visible in PET [82]. The 3D models are then rigidly registered using surface-based registration techniques [26]. The success of such methods relies on the structures being accurately and consistently segmented across modalities and the surfaces having rich enough structure to be unambiguously registered.



a b c

Figure 2-3: (a,c) An SPGR and a Proton Density image of the same patient, in alignment. (b) The joint intensity histogram for the image pair.

2.1.3 Correlation

Voxel-based approaches to registration do not extract any features from the images, but use the intensities themselves to register the two images. Such approaches model the relationships between intensities of the two images when they are registered, and then search through the transformation space to find an alignment that best agrees with the model. Various intensity models are discussed, including correlation, mutual information, and joint intensity priors.

Correlation is a measure commonly used to compare two images or regions of images for computer vision problems such as alignment or matching. Given the intensity values of two image patches stacked in the vectors **u** and **v**, the normalized correlation measure is the dot product of unit vectors in the directions of **u** and **v**:

$$\frac{\mathbf{u} \cdot \mathbf{v}}{||\mathbf{u}||\,||\mathbf{v}||} \tag{2.1}$$

An advantage of correlation-based methods is that they can be computed quite efficiently using convolution operators. Correlation is applicable when one expects a linear relationship between the intensities in the two images. In computer vision problems, normalized correlation provides some amount of robustness to lighting variation over a measure such as sum of square differences (SSD), $||\mathbf{u} - \mathbf{v}||^2$. The primary reason for acquiring more than one medical scan of a patient stems from the fact that each scan provides different information to the clinician. Therefore, two images that have a simple linear intensity relationship may be straightforward to register, but do not provide any additional information than one scan by itself. On the other hand, if the images are completely independent, (e.g. no intensity relationship exists between them), then they cannot be registered using voxel-based methods. In general, there is some dependence between images of different modalities, and each modality does provide additional information.

One simplified model of the medical imaging process is that an internal image is a rendering function R of underlying tissue properties, $\mathcal{P}(\mathbf{x})$, over positions $\mathbf{x}$. An MR image could be represented as a function $R_{\text{MR}}(\mathcal{P})$ and a registered CT image of the same patient would be another function, say $R_{\text{CT}}(\mathcal{P})$. Suppose a function $F(\cdot)$ could be computed relating the two rendering functions such that the following is true (with the possible addition of some Gaussian noise, $\mathcal{N}$):

$$F(R_{\text{CT}}(\mathcal{P})) = R_{\text{MR}}(\mathcal{P}) + \mathcal{N} \tag{2.2}$$

The function F would predict the intensity at a point in the MR image given the intensity at the corresponding point in the CT image. Such a function could be used to align an image pair that are initially in different coordinate systems using SSD:

$$T^* = \underset{T}{\operatorname{argmin}} \sum_{\mathbf{x}} \left(F(R_{\text{CT}}(\mathcal{P}(\mathbf{x}))) - R_{\text{MR}}(\mathcal{P}(T(\mathbf{x})))\right)^2 \tag{2.3}$$

where T is the transformation between MR and CT coordinates. In [102], van del Elsen, *et al.*, compute such a mapping that makes a CT image appear more like an MR, and then register the images using correlation. In general, explicitly computing the function F that relates two imaging modalities is difficult and under-constrained.

2.1.4 Mutual Information

Maximization of mutual information (MI) is a general approach applicable to a wide range of multi-modality registration applications [3, 15, 72, 109]. One of the strengths of using mutual information (and perhaps in some special cases, one of the weaknesses) is that MI does not use any prior information about the relationship between joint intensity distributions. While mutual information does not explicitly model the function F that relates the two imaging modalities, it assumes that when the images are aligned, each image should explain the other better than when the images are not aligned.

Given two random variables U and V, mutual information is defined as [3]:

$$MI(U,V) = H(U) + H(V) - H(U,V) \tag{2.4}$$

where H(U) and H(V) are the entropies of the two variables, and H(U,V) is the joint entropy. The entropy of a random variable is defined as:

$$H(U) = -\int p_U(u) \log p_U(u) du \tag{2.5}$$

where $p_U(u)$ is the PDF associated with U. Similarly, the expression for joint entropy entropy operates over the joint PDF:

$$H(U,V) = -\int\int p_{U,V}(u,v) \log p_{U,V}(u,v)\, du\, dv \tag{2.6}$$

When U and V are independent, $H(U,V) = H(U) + H(V)$, which implies the mutual information is zero. When there is a one-to-one functional relationship between U and V, (*i.e.* they are completely dependent), the mutual information is maximized as $MI(U,V) = H(U) = H(V) = H(U,V)$.

To operate on images over a transformation, we consider the two images, $I_1(\mathbf{x})$ and $I_2(\mathbf{x})$ to be random variables under a spatial parameterization, $\mathbf{x}$. We seek to

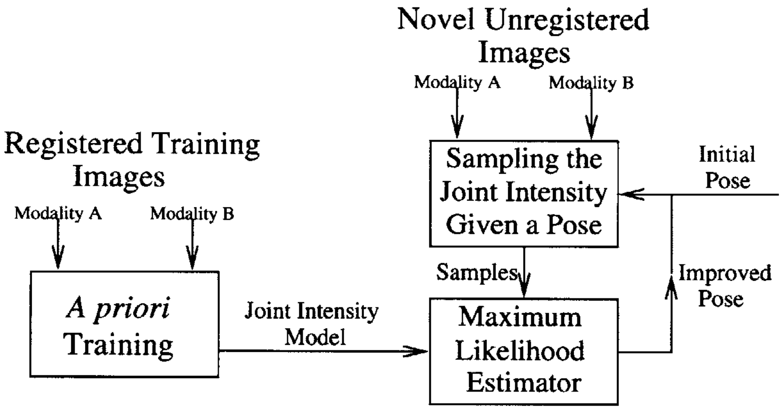


Figure 2-4: Flowchart of the *a priori* training and the online registration.

find the value of the transformation T that maximizes the mutual information [109]:

$$\begin{aligned} T^* &= \underset{T}{\operatorname{argmax}} \, MI\left(I_1(\mathbf{x}),\ I_2(T(\mathbf{x}))\right) && (2.7) \\ &= \underset{T}{\operatorname{argmax}} \, H(I_1(\mathbf{x})) + H(I_2(T(\mathbf{x}))) - H(I_1(\mathbf{x}), I_2(T(\mathbf{x}))) && (2.8) \end{aligned}$$

The entropies of the two images encourage transformations that project I_1 onto complex parts of I_2. The third term, the (negative) joint entropy of I_1 and I_2, takes on large values when X explains Y well. The derivatives of the entropies with respect to the pose parameters can be calculated and used to perform stochastic gradient ascent [109]. West *et al.* compare many multi-modal registration techniques and find mutual information to be one of the most accurate across all pairs of modalities [110].

2.1.5 Prior Intensity Model

Our approach to multi-modal registration, presented in Chapter 4, involves building a prior model of the intensity relationship between the two scans being registered. The framework for our registration process is illustrated in Figure 2-4. The method requires a pair of registered training images of the same modalities as those we wish to register in order to build the joint intensity model. To align a novel pair of images, we compute the likelihood of the two images given a certain pose based on our model by sampling the intensities at corresponding points. We improve the current hypothesized pose by ascending the log likelihood function. In essence, we compute

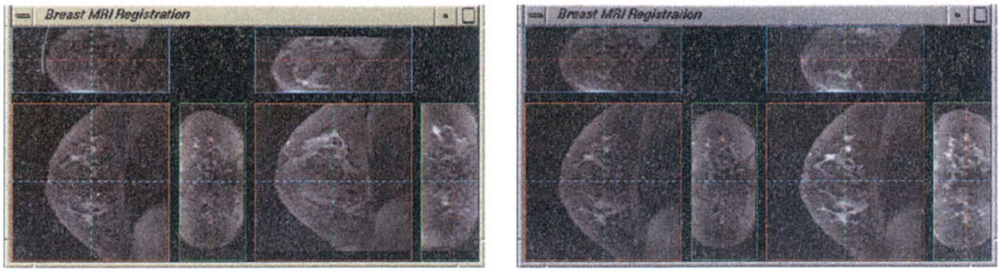


Figure 2-5: LEFT: Breast images of a patient taken at two different time points (pre- and post-gadolinium injection). RIGHT: The second image has been aligned to the first using non-rigid registration to determine areas that could contain lesions [51].

a probabilistic estimate of the function F (that relates the two imaging modalities) based on intensity co-occurrence. To align the novel images, the pose is found that maximizes the likelihood that those images arose from the same relation F.

Building a joint-intensity model does require having access to a registered pair of images of the same modality and approximately the same coverage as the novel pair to be registered. Mutual information approaches do not need to draw upon previously registered scans. However, when this information is available, the prior joint intensity model provides the registration algorithm with additional guidance which results in convergence on the correct alignment more quickly, more reliably and from more remote initial starting points.

2.1.6 Flexible Registration

The registration methods described thus far assume that a rigid transformation can be found to correctly align the images. When the underlying anatomy shifts or deforms between the acquisitions, a non-rigid transformation is required to relate the two images. Applications such as therapy evaluation and, more generally, change detection require the alignment of flexible anatomical structures over time. Deformable registration is also used to align an anatomical atlas with a patient scan for atlas-based segmentation. Non-rigid registration is quite challenging, even when the images are of the same modality, as the space of feasible deformations is high dimensional, complex, and depends on properties of the tissue. Methods such as splines and templates are used to characterize the deformation space [2, 10, 98]. Such approaches

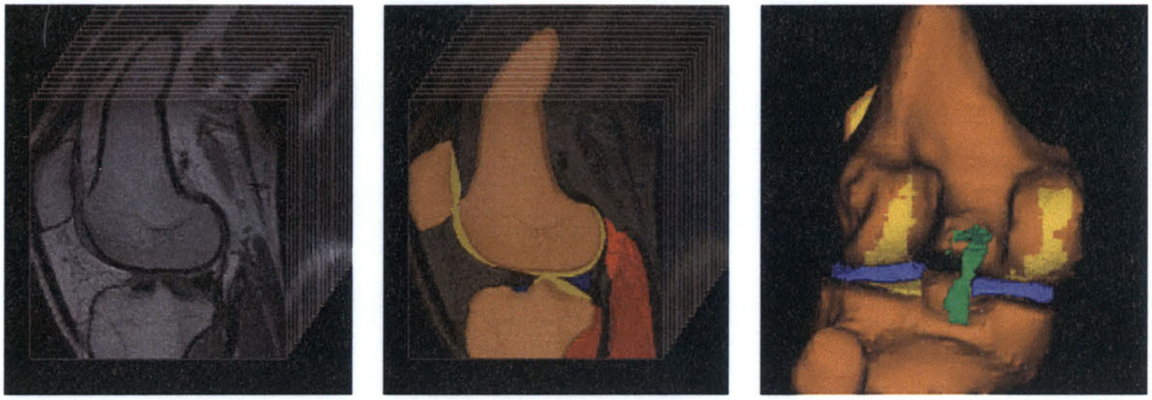


Figure 2-6: The process of segmentation and model generation. LEFT: The raw MR scans of the knee. MIDDLE: The label-map overlayed on the MR. RIGHT: Three dimensional surface models generated from the label-map using Marching Cubes[66].

have been used with mutual information as the measure of voxel similarity. Since the search space is so vast with non-rigid matching, mutual information may not be a strong enough intensity model since the intensity relationship is not actually known. Having a prior intensity model at hand can help to constrain and direct the search more effectively.

2.2 Segmentation

The segmentation problem can be formulated in a variety of ways, giving rise to algorithms with fairly different approaches to the problem. Three types of segmentation algorithms discussed here are boundary localization, voxel classification, and deformable atlases.

Segmentation via boundary detection consists of computing the closed curves, $\{\mathcal{C}_1, \ldots \mathcal{C}_k\}$ (or in 3D closed surfaces $\{\mathcal{S}_i\}$) such that all points inside the curve $\mathcal{C}_i$ correspond to tissue class ω_i, and conversely, all points in the image that represent the anatomical structure ω_i are contained by $\mathcal{C}_i$. Boundary localization techniques typically model some property of the border of the object of interest with neighboring objects. The assumption generally made is that high-gradients are indicative of boundaries, although other measures such as texture discontinuities are also used.

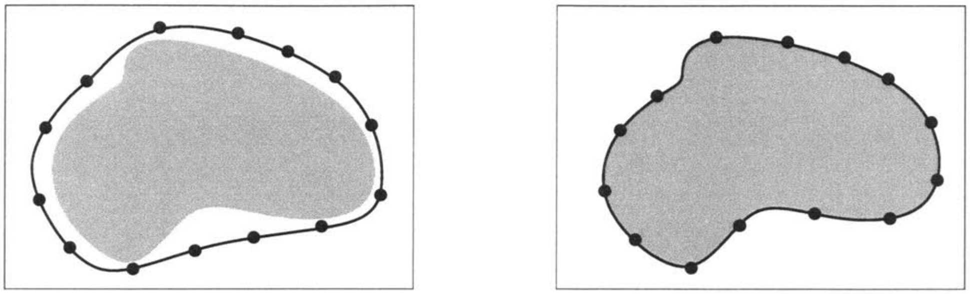
The strength of an "edge" at the boundary of the structure may vary or be weak relative to the texture inside the object, creating difficulties for such gradient-based boundary detection methods.

Voxel classification methods consist of computing a label map $L(\mathbf{x})$ over the region covered by image $I(\mathbf{x})$ such that $L(\mathbf{x}) = \omega_i$ if and only if $I(\mathbf{x})$ images the anatomical structure ω_i. These algorithms make a decision on the tissue class associated with each voxel in the image based on factors such as the intensity value, decisions of neighboring voxels, and properties of the imaging modality. However, the distribution of intensity values corresponding to one structure may vary throughout the structure and also overlap those of another structure, defeating intensity-based segmentation techniques.

The third approach seeks to deform a given labeled atlas to the novel image to be segmented [11, 78, 107]. The atlas generally consists of at least a scan $I^*(\mathbf{x})$ and its segmentation $L^*(\mathbf{x})$. Given a new image $I(\mathbf{x})$, the algorithm computes a non-rigid transformation or warp T such that $I^*(T(\mathbf{x}))$ is in correspondence with $I(\mathbf{x})$. If the correspondences are computed correctly, then the deformed atlas determines the labeling or segmentation of the new scan: $L(\mathbf{x}) = L^*(T(\mathbf{x}))$. Modeling the class of deformations that correctly warp one person's anatomy into another's can be quite challenging, and can result in correspondence mismatches, or errors in the segmentation.

2.2.1 Boundary Localization Methods

Active contour methods are commonly used to localize the boundary of an object. The contour can be represented in a variety of ways, and is considered "active" as it generally evolves over time with the goal of converging on the object boundary. The classical active contour, a snake, is an energy-minimizing spline parameterized by a set of node points that is commonly used to extract the boundary of an object. The snake methodology defines an energy function $E(\mathcal{C})$ over a curve $\mathcal{C}$ as the sum of an internal and external energy of the curve. The curve is evolved to minimize the



C(q,t=0) C(q,t=1)

Figure 2-7: An illustration of a snake evolving and converging on the boundary of an object in the image. The curve is represented as a spline of node points.

energy [52].

$$E(\mathcal{C}) = \beta \int |\mathcal{C}'(q)|^2 dq - \lambda \int |\nabla I(\mathcal{C}(q))| dq \tag{2.9}$$

The internal energy is a regularization force keeping the curve smooth and is added to give the snake robustness in the presence of image noise. The external energy term is a function of the image, I, designed to exhibit an attraction force towards the feature being localized (such as an edge). The tradeoff between the image term and the regularization term is generally adjusted empirically depending on the shape being segmented and the properties of the image.

While regularization is important for insensitivity to noise, the curvature penalization term in the update equation causes an asymmetry in the snake. In the absence of image information, a curve evolving to minimize the energy will shrink smoothly to a point. Therefore, if the snake is initialized inside the object too far from an edge, the snake will disappear and not localize the boundary. On the other hand, if the snake is initialized outside the object, but still far from any edges, the snake will shrink based on the regularization term towards the boundary, and then will be attracted to the edge. In many applications, including medical image segmentation, the intensity inside the object to be segmented is relatively uniform as compared to the region outside the object, where other structures may have various intensity distributions. Therefore, boundary localization via snake may be more successful when the snake segments the object from the inside.

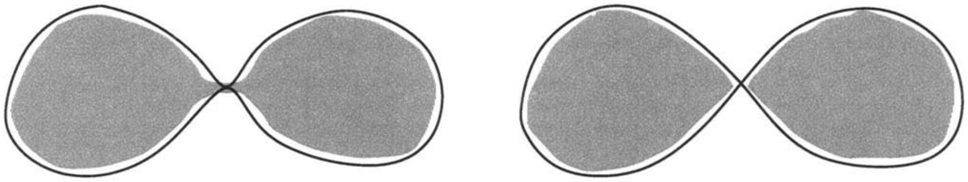


Figure 2-8: An illustration of a topological singularity for a snake. At the moment that the snake meets itself, it is ambiguous whether there are two objects touching or one pinched object. Level set techniques handle topology changes naturally.

An extension to snakes called balloons addresses the shrinking problem, along with some other drawbacks inherent to classical snakes [13]. Balloons include an outward pointing force at each position on the contour that is strong enough to counter the shrinking effect of the regularization, but weak enough to ensure that the contour will stop at a sufficiently strong edge. Furthermore, stability issues that plague snakes are addressed, such as the normalization of the image force so as to not move more than a pixel, and interpolation to handle discretization errors.

One of the challenges with classical snakes and balloons involves adjusting the spacing between the nodes as the curve evolves. As a snake expands, the resolution of the spline is reduced if the number of node points stays constant. During contraction, the nodes move much closer together, resulting in unnecessary computational expense. Therefore, one must dynamically adjust the positions of the node points on the spline when it undergoes large changes. Topology changes pose another difficulty for classical snakes. Consider the snake in Figure 2-8 that is an oval pinching into a figure-8. A singularity exists at the time in which the snake meets itself, as it is not clear whether there are two curves in contact, about to break apart, or one pinched curve. In order to handle these factors with traditional snakes, one must reparameterize the nodes to reflect topology changes, which is difficult, given the inherent ambiguity.

The ambiguity of topology changes can be addressed by making the "grass burning" assumption [71, 93]. Initially, the entire region inside (or equivalently outside) the curve is assumed to be "burned", and as the front moves, it causes further burning of the space. If one assumes that a point, once burned will never regenerate, then the topological ambiguity is resolved. Topologically adaptable snakes (or T-Snakes) address topology changes in this way and also provide an efficient and effective method

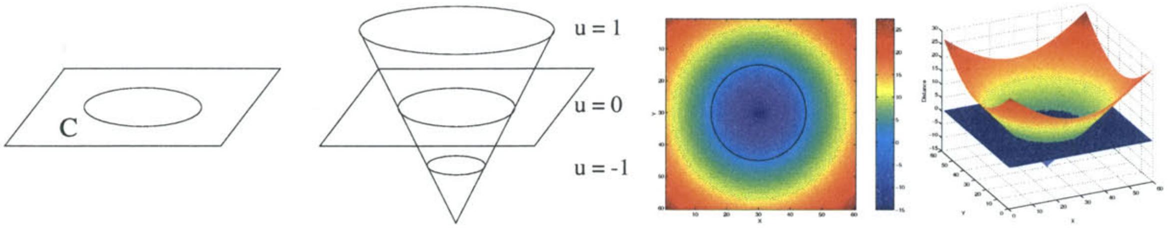


Figure 2-9: Level sets of an embedding function u for a closed curve $\mathcal{C}$ in $\Re^2$.

of reparameterizing the nodes to handle sampling issues [71]. At a given point in time, the evolving curve is represented as a spline of node points on a grid, with all interior grid points labeled as burned. Each of the node points are adjusted as commonly done to minimize an energy function consisting of internal and external terms. Any points on the grid that are crossed by the contour are also labeled as burned. The curve is then reparameterized by intersecting the boundary with a uniform triangularization of the embedding space, and using the intersection points as the new nodes of the spline. Thus, the nodes are guaranteed to have a bounded proximity as a function of the resolution of the underlying grid.

Issues of sampling and topology can also be addressed by reformulating the problem using level set methods [93]. Segmentation in this way, introduced in both [9] and [54], involves solving the energy-based active contours minimization problem by the computation of geodesics or minimal distance curves [93]. In this approach, a curve is embedded as a zero level set of a higher dimensional surface (Figure 2-9). The entire surface is evolved to minimize a metric defined by the curvature and image gradient.

The equivalence of geodesic active contours to the traditional energy-based active contours (snakes) framework is derived by first reducing the minimization problem to the following form [9]:

$$\min_{\mathcal{C}(q)} \int g(|\nabla I(\mathcal{C}(q))|)\ |\mathcal{C}'(q)|\ dq \tag{2.10}$$

where g is a function of the image gradient (usually of the form $\frac{1}{1+|\nabla I|^2}$). Using

Euler-Lagrange, the following curve evolution equation is derived [9]

$$\frac{\partial \mathcal{C}(t)}{\partial t} = g\kappa\mathcal{N} - (\nabla g \cdot \mathcal{N})\mathcal{N} \tag{2.11}$$

where κ is the curvature and $\mathcal{N}$ is the unit normal. By defining an embedding function u of the curve $\mathcal{C}$, the update equation for a higher dimensional surface is computed [9]

$$\frac{\partial u}{\partial t} = g\ \kappa\ |\nabla u| + \nabla u \cdot \nabla g \tag{2.12}$$

In this level set framework, the surface, u, evolves at every point perpendicular to the level sets as a function of the curvature at that point and the image gradient. The curvature term is used to keep the level set smooth as it evolves and the image term is effectively an attractive force towards high gradients. Further details on the derivation and implementation of level sets is presented in Chapter 5.

Recent work based on level sets include extensions such as codimension-2 regularization [68], texture models [80], global intensity statistics [113], and pedal curve evolution [40].

The snake and level set framework is applicable both in segmenting 1D curves in 2D space and 2D surfaces in 3D. Two-dimensional surfaces have two principal curvatures at every point (κ_1 and κ_2), and generally the regularization method chosen penalizes high mean curvature ($\frac{\kappa_1+\kappa_2}{2}$) or high Gaussian curvature ($\kappa_1\kappa_2$). In some applications, such as the segmentation of tubular structures, penalizing a combination of both principal curvatures does not make sense, as the maximal curvature (κ_1) *should* be large given the tube-like structure. In [68], Lorigo, *et al.* develop a system of segmenting blood vessels and other tubular structures by changing the curvature term to only penalize the minimal curvature, κ_2. Their curvature model accurately reflects the properties of the shapes they are segmenting, which results in more accurate vessel extraction than when using the traditional curvature terms.

The boundary segmentation algorithms described above all assume that there is a strong edge at the boundary of the object as compared to the gradients inside (or outside) the object. In some cases, the regions do not have separable uniform intensities, but instead differ in their texture properties. Geodesic active contours can be extended to incorporate vector values, such as intensity variances over a window or responses from oriented filters [87, 67].

The active contour methods described thus far include only local information in the evolution process. However, in many cases, the segmentation process can be effectively directed by the global intensity statistics over the image. In [113], segmentation is performed by evolving a curve to maximally separate predetermined statistics inside and outside the curve. The statistics chosen can be mean intensity or richer, vector-valued texture. This method includes both global and local information, adding robustness to noise and weak boundaries. The algorithm is formulated using level set techniques providing the advantages of numerical stability and topological flexibility.

Paragios and Deriche in [80], also perform texture segmentation by combining local (boundary) and global (region) information, but in addition, they include a prior texture model. The supervised texture model is constructed by extracting M features (based on the response of M filters) from each of N example texture classes. The bank of filters included are the Gaussian, Laplacian of Gaussian, and 2D Gabor operators. A probabilistic model for each texture class is computed using a mixture of Gaussian model. The energy function is defined as a convex combination of a boundary term and a region term that both incorporate the probabilities of the observed textures given the prior statistical model. The approach is also implemented using a level set framework.

When segmenting or localizing an anatomical structure, having prior information about the expected shape of that structure can significantly aid in the segmentation process. In [40], Guo and Vemuri evolve pedal generating curves to perform segmentation, instead of directly evolving the boundary surface itself. The choice of the generator curve (such as an ellipse or rhombus) allows the addition of rough shape constraints. However, the shape model derived from the pedal curves is relatively weak, due to the diverse set of pedal curves that can be derived from a given generator.

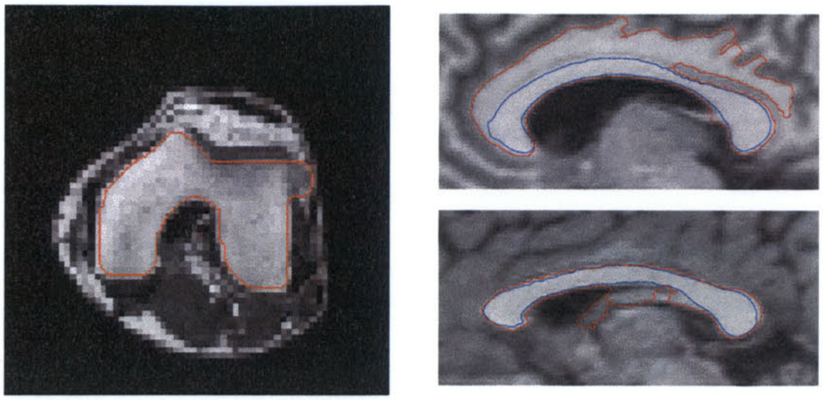


Figure 2-10: An example of how local boundary detectors can leak through regions where the gradient is not strong enough. Global information, such as shape priors, can prevent this type of error.

As snakes and level sets are traditionally local algorithms, they may "leak" through the boundary of the object if the edge feature is not salient enough in a certain region in the image. Figure 2-10 illustrates examples of the segmentation of a few anatomical structures missing part of the boundaries. Furthermore, boundary finding methods can get caught in a local minimum, and are thus sensitive to the starting position.

2.2.2 Voxel Classification

Voxel classification approaches to segmentation differ from curve evolution methods in that the boundary is not explicitly represented during the segmentation process. Instead, each voxel is associated with a tissue class, based on information such as the intensity of that voxel, imaging artifacts, the decisions of neighboring voxels, and relative position of neighboring structures. The object boundaries are implied from the label maps once the segmentation process converges on a solution.

In cases when the intensity ranges of the different tissue classes are disjoint, segmentation can be achieved by simply thresholding the image between the intensity distributions of the different structures. Figure 2-11 show an example of thresholding CT data. In this instance, thresholding performs reasonably well picking out the vertebrae of the spine, but incorrectly includes the cartilage between vertebrae, due to the intensity overlap. There are also "holes" inside the vertebrae model due to

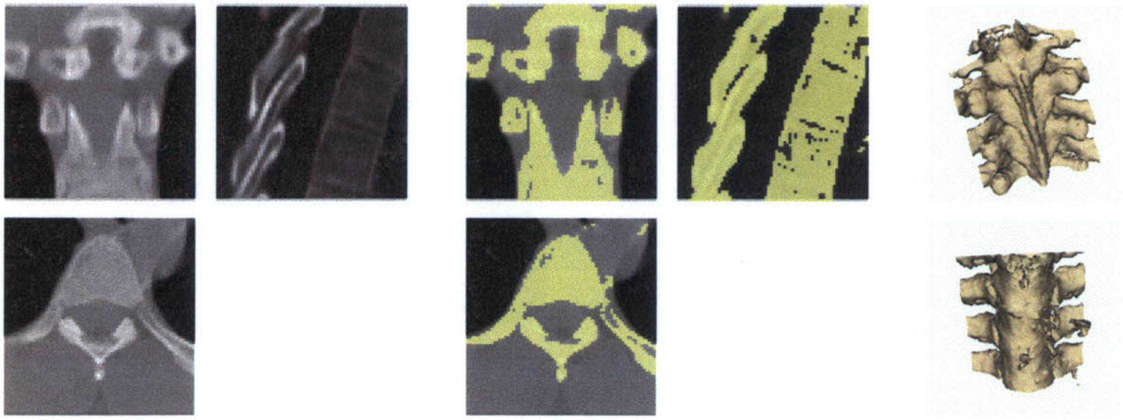


Figure 2-11: An illustration of thresholding, the simplest segmentation technique, that is applicable when the intensity ranges of the object and the rest of the image are disjoint. LEFT: A CT scan of the spine. MIDDLE: Colorized overlay of the thresholded region. RIGHT: Three dimensional model of the spine created from the thresholding. Notice that there are holes in the vertebrae models and the cartilage is sometime classified as bone.

regions that were imaged darker than the surrounding areas of bone. While simple intensity information may not be sufficient for accurate labeling, it can still provide constraints in a more sophisticated segmentation framework.

In the spine example above, the binary intensity threshold was chosen manually. A tissue intensity model can be used instead of a manual threshold to compute a soft assignment of tissue classes. In this approach, the probability of an intensity value x is estimated for each tissue class ω_i based on a set of training data. The tissue intensity model can be used alone, by just assigning a voxel with intensity x to the class ω_i that maximizes $P(x|\omega_i)$, or a term can be used in conjunction with other factors to provide a richer segmentation model.

In practice, observed intensity classes of different anatomical structures are rarely separable. Imaging noise and artifacts can cause intensity overlap, defeating methods based solely on intensity. Thus, imaging models are developed to account for the effect seen in the scan. For example, magnetic resonance images can contain spatially varying, non-linear gain artifacts due to inhomogeneities in the RF coil. Wells *et al.* introduced a means of simultaneously estimating the gain field and the tissue classes

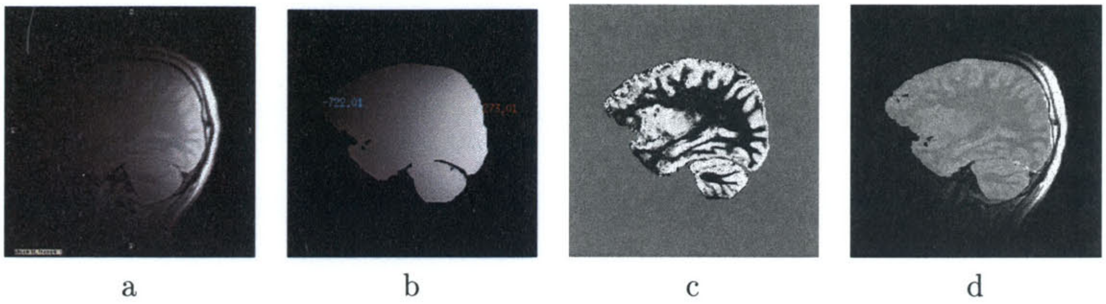


Figure 2-12: (a) Original MR scan with non-linear gain artifacts. (b) Estimation of the gain field. (c) Image of gray matter probability. (d) Corrected MR image (with gain field removed).

by means of the Expectation-Maximization algorithm [108]. Figure 2-12 illustrates the results of gray matter / white matter segmentation with gain field correction.

In the original EM segmentation approach, the estimation of the smoothly-varying gain field incorporates spatial dependence, but the classification of each voxel is performed independent from all other voxels. Other types of imaging artifacts, such as speckle due to thermal noise, can cause local mis-classifications. In addition to non-linear smoothing operators, the effects of noise can be reduced with the addition of spatial priors on tissue classes, such as the Gibbs prior in [51] which models the piecewise homogeneity of tissues. Such a prior is similar to the regularization term found in boundary detection methods.

A second type of spatial prior described in [51] involves incorporating a model of relative distance between anatomical structures, encoding assertions such as "white matter should be a certain distance away from the scalp." Without a model-dependent, locally defined reference frame, the richness of such spatial priors is limited to measures such as relative distance that are invariant to the choice of coordinate system. Approaches that register the model to the image data (such as deformable atlases or our shape-based segmentation algorithm) can incorporate richer information about the relative positions and poses of anatomical structures to aid in multi-structure segmentation.

2.2.3 Deformable Atlases

A medical image atlas consists of at least one internal scan that has been carefully annotated, such as the fully segmented 3D anatomical atlases described in [35, 55]. A deformable atlas generally includes of a model of the feasible transformations from the "standard" atlas, perhaps along with likelihoods, used to describe the variance seen across a population. If the atlas can be successfully matched to a novel scan, then all information present in the atlas (such as the tissue labels) is then known with respect to the image, achieving segmentation of the new scan. Some deformable atlases consist of sparse corresponding points, finite elements, or features to represent the shape [16, 17, 19, 18, 83, 89, 105], while others represent deformations as dense warps, a cascade of similarity groups, or flow fields [10, 11, 25, 37, 65, 78, 106, 107].

Cootes, Taylor, and others have developed *Active Shape Models*, which is a framework for building a deformable atlas and subsequently matching the atlas to a novel image [16, 17, 19]. In this approach, a set of corresponding points are identified across a set of training images. The principal modes (eigenvectors) of the points are computed and used as a probability model over shape variation. The best match of the model to the image is found by searching over the model parameters so that the points on the spline move along the normal towards the nearest edges in the image. A similar approach described in [105] allows for the points to move tangentially as well, which simultaneously solves for the point correspondence and provides robustness to initial misalignment.

Cootes, *et al.* [18] have extended their shape work to include grayscale information in their Active Appearance Models. Along with the variance of the point positions, intensity distributions over the object are included in the model. The model fitting stage again consists of searching over the eigen-coefficients, but now since the model includes intensity, the goodness of fit is determined by matching the grayscale values of the model and the image. Once the algorithm has converged, the correspondence to the novel image (*i.e.* the segmentation) is given by the model parameters.

The warping based approaches to atlas-based segmentation need to find the best non-rigid transformation to match the atlas to the image. Warfield, *et al.* [107] use a combination of feature identification, object classification, and model alignment to perform template driven segmentation. The deformations are generally sufficiently high-dimensional that a hierarchical registration approach is necessary to avoid local minima. Christensen *et al.* [10, 11] compute the deformation field as a sum of basis vectors, allowing for a coarse to fine solution. Other approaches use a cascade of similarity groups beginning with rigid transformations, and subsequently allowing more flexible warps, such as piece-wise affine [37, 77, 78]. Tissue deformation models can also assist in computing the deformation field. One approach is to model structures as either rigid, fluid, or flexible, which is successful when dealing with anatomy containing very different structures such as bone, muscle, and cerebrospinal-fluid (CSF) [25, 65].

2.2.4 Statistical Models in Level Set Segmentation

The segmentation framework presented here is a hybrid of boundary localization, voxel classification, and atlas matching. The following list summarizes key features of the three general classes of segmentation algorithms.

- **Boundary Detection:** A curve or surface evolves towards the object boundary based on an image term and a regularization term.
- **Voxel Classification:** Each voxel is classified given a prior intensity model, along with homogeneity and relative geometry constraints.
- **Deformable Atlas:** Shape models are derived from training data and aligned with a novel image to provide the segmentation.

Our approach to object segmentation extends geodesic active contours [9, 54] by incorporating prior intensity, curvature, and shape information into the evolution process. Specifically, we use the idea of evolving a higher dimensional signed distance surface whose zero level set is the implicit boundary of the object. Unlike traditional level set techniques that maintain equivalence to minimizing an energy function over a curve, our method uses information over the entire image and higher dimensional surface.

We model the distribution of intensity over the image as a function of the signed distance from the boundary of the structure. This provides a means of representing both changes of the intensity of the structure relative to its boundary and the intensity profiles of neighboring structures as a function of their distances to the object of interest. At every iteration of the segmentation, each position on the surface (representing a distance to the boundary) is adjusted towards a maximum *a posteriori* distance, based on the image information at that position and the values of neighboring points on the surface. A distribution of the curvature of the structure is also modeled from the training data to determine the degree of regularization of the underlying level sets. The intensity-distance model directs the height of the surface towards a likely distance and the curvature model keeps the surface regular. Tissue homogeneity is achieved based on a two-part regularization term that keeps the evolving surface regular and the zero level set smooth, as determined by the prior curvature model.

Deformable shape models are derived from a training set using Principal Component Analysis on the distance surface to the object. This process captures the primary modes of shape variation and induces a Gaussian probably model over shapes. At each step of the surface evolution, we estimate the maximum *a posteriori* (MAP) position and shape of the object in the image, based on the prior shape information and the image information. The estimated MAP model is then used as a global force to direct the segmentation.

Chapter 3

The Construction of Statistical Models

Any algorithm that analyzes medical scans must include, at least implicitly, some model of the imaging modality and/or the underlying anatomy. Even a simple segmentation scheme such as thresholding has the implied model that the intensity distribution of the object to be extracted must be disjoint from the intensity of surrounding objects in the image. Classical snake-based approaches model the boundary of objects as having high gradients or edges. An elementary correlation-based method to multi-modal image registration assumes that the joint intensity model between the images has a linear relationship when registered. The image analysis algorithm is typically chosen based on the user's prior knowledge of the problem. For example, based on the knowledge of various imaging processes or having seen different scans, one may know that thresholding for segmentation can produce reasonable results in CT images, but is likely to fail miserably with ultrasound.

In this framework, all the knowledge imparted to the algorithm is provided either by the developer (in design) or by the user (through interaction). Due to the difficulties of manually encoding accurate, application specific information, the richness of such models is limited. For example, the same segmentation algorithm (such as snakes) may be applied to very different anatomical structures (say femur and heart) with varying degrees of success. The user may be able to adjust some rough parameters of the algorithm for each application, but there may not be way of encoding specific information important to the problem, such as the shape of the object, intensity profile, imaging modality, etc. While the ideal solution to segmentation may be a general, unsupervised method that solves a wide variety of problems, an alternative approach is to devise a framework in which specific knowledge inherent to the problem can be easily incorporated. Statistical models provide a means of automatically deriving more complex information directly from a training set, once the important relationships have been identified.

Statistical models developed in this thesis consist of a representation, a training set, and a probability density estimator. The *representation* defines the way a certain characteristic or relationship present in the data is described as some d-dimensional vector $\mathbf{z}$. For simple measures such as intensity, the representation is just the scalar measure itself. Modeling more complex properties such as pose, shape, curvature, texture, or biomechanics requires choosing a means of representing the features as a vector of coefficients. The co-occurrence of two or more values, such as intensity and distance to boundary, can also be included in one representation vector. Generally in medical image applications, the information encoded consists of either imaging models (such as intensity relationships between two modalities) or anatomical models (such as the shape or relative position of structures). The choice of what to include in the model depends on application and dependencies present in the data. The goal is to choose measures that are predictive and discriminating that can be used to direct the algorithm when processing a novel image.

The *training set* consists of n examples of the properties across a population, $\mathcal{T} = \{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$. This set can be considered a cloud of n points embedded in a d-dimensional space. From this training set, we wish to define a *probability density estimator*, which is a function $P(\mathbf{x}|\mathcal{T})$ that provides a reasonable estimate of the probability of a novel element $\mathbf{x}$, given the prior model derived from the training data.

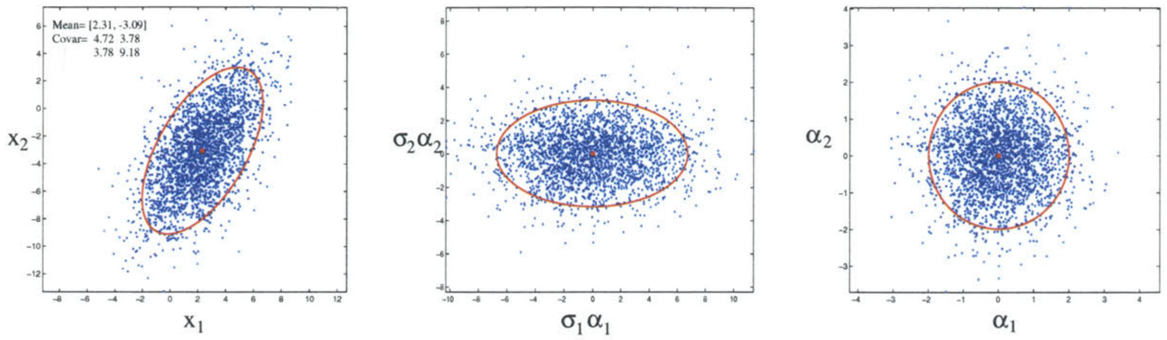


Figure 3-1: LEFT: An illustration of fitting the Gaussian model of an Gaussian with an arbitrary mean and covariance matrix. MIDDLE: Principal Component Analysis aligns the Gaussian with the standard axis, making the variables statistically independent. RIGHT: PCA also captures the principal variances, which can be factored out, leaving an independent, variance-one Gaussian.

3.1 Gaussian Model

The Gaussian distribution is one of the most commonly used density estimators, primarily due to the existence of simple, analytical methods of estimating the model parameters, even with very high dimensional data. The model is determined by two parameters: the mean, a d-dimensional vector $\boldsymbol{\mu}$, and the covariance, a $d \times d$ symmetric, positive semi-definite matrix Σ. The Gaussian is the function with the highest entropy of all distributions with the same variance, which, in terms of information theory, indicates that it is the best model when only the mean and variance is known.

The *central limit theorem* often further justifies the use of the normal distribution. The central limit theorem states that, in fairly general terms, the mean of a set of random variables approaches a Gaussian distribution as the number of random variables increases [5]. Therefore, an observed process will resemble a Gaussian in nature when it is the result of a very large linear combination of possibly unknown, unrelated factors, as is often assumed with naturally occurring quantities. For example, the height of an adult depends on many factors such as genetics, race, development, nutrition, exercise, heath, etc. Therefore, one may expect that the distribution of heights of randomly chosen adults would approximate a Gaussian.

Given n, d-dimensional training elements, $\{\mathbf{z}_1, \ldots, \mathbf{z}_n\}$, we seek to estimate the model parameters (mean and variance) of the best fit Gaussian, using a maximum likelihood approach. The mean can be estimated using the following:

$$\boldsymbol{\mu} \;=\; \frac{1}{n}\sum_{i=1}^{n}\mathbf{z}_i \tag{3.1}$$

The mean is then subtracted from each vector to align the centroid of the distribution with the origin. Define $\hat{\mathbf{z}}_i = \mathbf{z}_i - \boldsymbol{\mu}$ as the centroid-aligned points. The training matrix M, is defined as follows:

$$M = \begin{bmatrix} | & | & & | \\ \bar{\mathbf{z}}_1 & \bar{\mathbf{z}}_2 & \dots & \bar{\mathbf{z}}_n \\ | & | & & | \end{bmatrix}_{d\times n} \tag{3.2}$$

The remaining parameter of the Gaussian model to estimate is the covariance matrix Σ:

$$\begin{aligned} \Sigma &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{z}_i - \boldsymbol{\mu})(\mathbf{z}_i - \boldsymbol{\mu})^{\mathsf{T}} \quad (3.3) \\ &= \frac{1}{n}MM^{\mathsf{T}} \quad (3.4) \end{aligned}$$

3.1.1 Principal Component Analysis

In general, the principal axes of the Gaussian will not lie along the standard basis axes of the training vectors, but will be rotated arbitrarily in the d dimensional space. A nice property of the Gaussian stems from the fact that the co-variance matrix is symmetric and positive semi-definite: There exists a $d \times d$ dimensional rotation matrix U that diagonalizes the co-variance matrix Σ such that the principal axes lie along the standard basis, and the variables in that basis are statistically independent. Using Singular Value Decomposition, the covariance matrix Σ is decomposed as:

$$U\Lambda U^{\mathsf{T}} = \Sigma \tag{3.5}$$

where Λ is a diagonal matrix of eigenvalues of Σ and the columns of $U = [\boldsymbol{u}_i]$, are the corresponding eigenvectors (as $U\Lambda = \Sigma U$).

When $d \gg n$, or the dimensionality exceeds the number of training elements, the system is underconstrained, and a number of the eigenvalues will be zero, indicating the Gaussian has collapsed into the number of dimensions spanned by the training set. In these cases, the covariance matrix could be very large $d \times d$, especially when the representation has one component for each voxel, as used in this work. Computing eigenvectors of such a large matrix is inefficient, especially when the number of non-zero eigenvalues is relatively small. Instead, the eigenvalues and eigenvectors of Σ can be easily computed from those of a smaller matrix, T:

$$\begin{aligned} T &= \frac{1}{n}\sum_{i=1}^{n}(\mathbf{z}_i - \boldsymbol{\mu})^{\mathsf{T}}(\mathbf{z}_i - \boldsymbol{\mu}) && (3.6) \\ &= \frac{1}{n}M^{\mathsf{T}}M && (3.7) \end{aligned}$$

Let $\mathbf{v}$ be an eigenvector of T with corresponding non-zero eigenvalue λ. The following proves that $M\mathbf{v}$ is an eigenvector of Σ with eigenvalue λ.

$$\begin{aligned} \Sigma(M\mathbf{v}) &= \frac{1}{n}MM^{\mathsf{T}}(M\mathbf{v}) && (3.8) \\ &= M(\frac{1}{n}M^{\mathsf{T}}M)\mathbf{v} && (3.9) \\ &= M(T\mathbf{v}) && (3.10) \\ &= M\lambda\mathbf{v} && (3.11) \\ &= \lambda(M\mathbf{v}) && (3.12) \end{aligned}$$

Thus for every eigenvector $\mathbf{v}$ of T, $M\mathbf{v}$ is an eigenvector of Σ. Since Σ and T have the same rank, all remaining eigenvalues of Σ are zero. Computing the eigenvectors of T is of complexity $O(n^3)$, which can be a considerable savings over directly computing those of Σ, which would take $O(d^3)$ time.

Under the Gaussian model, the eigenvectors represent the principal axes of variation in the training set, with the corresponding eigenvalues being the degree of variation along that axis. For many representations of naturally occurring data sources, the degree of variation of the principal modes drops off very quickly, indicating that a high percentage of the variation present in the dataset can be accounted for using a relatively small number of modes. The matrix of eigenvectors, U, can be used to rotate (and project) an example into the eigenspace. Let k be the number of modes to consider, or the dimension of the reduced eigenspace. An example element $\mathbf{x}$ can be represented by k principal components as follows:

$$\boldsymbol{\alpha} = U_k^{\mathsf{T}}(\mathbf{x} - \boldsymbol{\mu}) \tag{3.13}$$

where U_k is a matrix consisting of the first k columns of U, and $\boldsymbol{\alpha}$ is a vector of eigen-coefficients. Given a vector $\boldsymbol{\alpha}$ in the eigen-space, an estimate of the example can be reconstructed:

$$\tilde{\mathbf{x}} = U_k \boldsymbol{\alpha} + \boldsymbol{\mu} \tag{3.14}$$

As the dimension of the eigen-space, k, is generally much smaller than the embedding dimension of the examples, d, tasks such as optimization can be performed more efficiently in the eigenspace.

3.1.2 Derived Probability Density

The primary goal of modeling a set of examples with a Gaussian distribution is to be able to estimate the probability of a certain example. Once the parameters of the model have been estimated, the probability distribution over examples is as follows:

$$P(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^{\mathsf{T}}\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right) \tag{3.15}$$

It is also convenient to express the probability density as a function of the coefficients of the eigenvectors, γ. In full rank cases when $k = d$, the above expression is equivalent to the following:

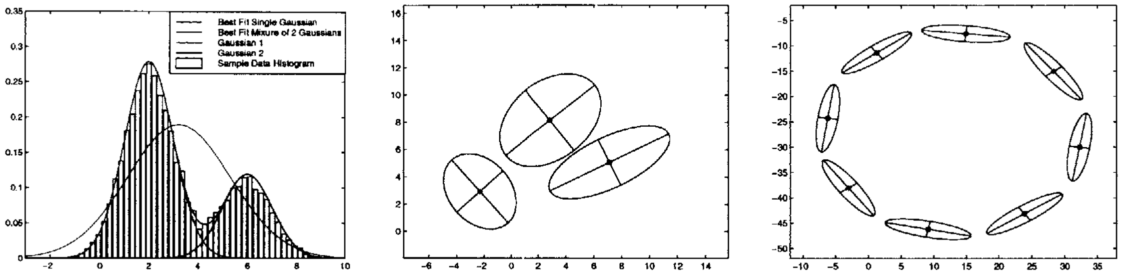
$$P(\gamma) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Lambda|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\gamma^{\mathrm{T}}\Lambda^{-1}\gamma\right) \tag{3.16}$$

In the more general case, when $k < d$ or when the system is rank deficient, the following expression is used over coefficients $\boldsymbol{\alpha}$ for the probability subspace defined by the k-dimensional Gaussian:

$$P(\boldsymbol{\alpha}) = \frac{1}{(2\pi)^{\frac{k}{2}}|\Lambda_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\Lambda_k^{-1}\boldsymbol{\alpha}\right) \tag{3.17}$$

where Λ_k is the matrix consisting of the first k rows and columns of Λ. Depending on the dimensionality of the variation in the training set and the desired fidelity of the reconstruction, any value $0 < k \leq d$ can be used. Many methods use a multi-resolution scheme, beginning with a small k, and increasing k as the algorithm approaches the solution. Care must be taken not to directly compare the value of the PDFs over different choices of k, as each density function is different, and depends on the values of the eigenvectors included for that k.

In summary, the Gaussian model is quite useful for wide ranges of applications, from when there are 10^6 one-dimensional samples to when there are twenty 256^3-dimensional samples. (Although clearly in the latter case, one expects a terrific amount of redundancy in the system to be able to model such a large space with so few examples.) The analytic solution of the parameters, along with the ability to compute the principal components efficiently make the computations tractable when more complex models are not. However, the Gaussian assumption is commonly overused for such reasons, when the underlying distribution may not be Gaussian at all. In such cases, other models should be used, such as mixtures of Gaussians or non-parametric estimators as described below. In cases where the dimensionality is too high, principal component analysis can be used to compute the subspace of maximal variance, and then a more sophisticated model can be used to estimate the distribution of the eigen-coefficients.



a b c

Figure 3-2: Illustrations of fitting mixtures of Gaussians. (a) Histogram of 1D data, poorly fit by a single Gaussian, and better fit by a mixture of two Gaussians. (b) Scatter plot of 2D Data fit with 3 Gaussians. (c) Eight Gaussians are used to fit very non-Gaussian data.

3.2 Mixtures of Gaussians

While single Gaussian models can be estimated quite easily, they are limited in their ability to effectively explain more complex distributions. Non-Gaussian distributions often arise that may be modeled very poorly with a single Gaussian. Simple examples of such distributions are shown in Figure 3-2. Mixtures of Gaussians are richer and can account for a wide variety of distributions, at the price of added difficulty in reliably estimating the model parameters. The Gaussian mixture derives a probability density function that is the convex combination of K Gaussians. Equivalently, the model breaks up the space into K classes, $\{\omega_1, \ldots, \omega_K\}$, with prior probabilities $\pi_k = P(\omega_k)$ that a random point was generated from that class. Since each class represents a Gaussian distribution, the probability of a point $\mathbf{x}$ under this model is given as

$$P(\mathbf{x}) = \sum_{k=1}^{K} P(\mathbf{x}|\omega_k)P(\omega_k) \tag{3.18}$$

$$P(\mathbf{x}) = \sum_{k=1}^{K} \frac{\pi_k}{(2\pi)^{\frac{d}{2}}|\Sigma_k|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^{\mathsf{T}}\Sigma_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k)\right) \tag{3.19}$$

where $\boldsymbol{\mu}_k$, Σ_k, π_k are the mean, covariance, and prior probability of the kth Gaussian and together fully determine the mixture model. For the entire PDF to sum to one, the sum of the π_k's must be one.

3.2.1 Parameter Estimation using K-Means

A simple method of estimating the parameters of the mixture model is to cluster the training points via the K-means algorithm, and then perform the single Gaussian parameter estimation based on the clusters. The cluster algorithm begins by choosing random values for the means, $\boldsymbol{\mu}_k$. We define the assignment $h_k^{(i)}$ of example i into class k as follows:

$$\begin{aligned} h_k^{(i)} &= 1 \quad \text{if } \forall_{j \neq k} \ ||\mathbf{z}_i - \boldsymbol{\mu}_k|| < ||\mathbf{z}_i - \boldsymbol{\mu}_j|| && (3.20) \\ h_k^{(i)} &= 0 \quad \text{otherwise} && (3.21) \end{aligned}$$

where each point is assigned to the class with the nearest mean. Once the assignments have been made, the means are recalculated based on the assignments:

$$\boldsymbol{\mu}_k = \frac{\sum h_k^{(i)} \mathbf{z}_i}{\sum h_k^{(i)}} \tag{3.22}$$

The algorithm iterates between these two steps until the portions remain the same, indicating convergence.

While K-means clustering is simple and computationally efficient, which is advantageous in high dimensional spaces, the algorithm can perform poorly when the clusters overlap, due to the hard class assignments made. Furthermore, K-means is not very robust to outliers.

3.2.2 Parameter Estimation using Expectation Maximization

A more robust method of parameter estimation for the mixture model is the Expectation Maximization (EM) algorithm. In the EM approach, a set of hidden variables are identified, such that if the values of those variables were known, then the maximum likelihood estimates of the model parameters could be determined. Furthermore, if the model parameters were known, then the hidden variables could be computed. The estimation step (E-step) of the algorithm assigns the hidden variables to their expected values, based on the current model parameters. The maximization step

(M-step) subsequently maximizes the likelihood of the model given the training data and the hidden variables. Nice properties of the EM algorithm include assured increase (or no change) in the likelihood with each iteration which implies guaranteed convergence to a local maximum [22, 5].

The E-step of the algorithm is to compute the probability of each class, given a certain data point, using the current estimate of the model parameters. This quantity, h_k, is defined as

$$\begin{aligned} h_k^{(i)} &= P(\omega_k \,|\, \mathbf{z}_i) && (3.23) \\ &= \frac{P(\mathbf{z}_i \,|\, \omega_k) P(\omega_k)}{\sum_j P(\mathbf{z}_i \,|\, \omega_j) P(\omega_j)} && (3.24) \\ &= \frac{\pi_k}{Z} \left(\frac{1}{2\pi |\Sigma_k|} \exp\left(-\frac{1}{2} (\mathbf{z}_i - \boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{z}_i - \boldsymbol{\mu}_k) \right) \right) && (3.25) \end{aligned}$$

where the normalization factor Z is the denominator of equation 3.24.

Given the posterior probabilities computed in the E-step, in the M-step we seek to maximize the log likelihood with respect to each of the model parameters. The likelihood of a training element $\mathbf{z}_i$ is given as

$$P(\mathbf{z}_i) = \sum_{k=1}^{K} \pi_k P(\mathbf{z}_i | \omega_k) \tag{3.26}$$

With the independence assumption, the likelihood of the entire training set is just the product

$$P(\{\mathbf{z}_1, \ldots, \mathbf{z}_n\}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k P(\mathbf{z}_i | \omega_k) \tag{3.27}$$

and the associated log likelihood is

$$l = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k P(\mathbf{z}_i | \omega_k) \tag{3.28}$$

To compute the update for each of the model parameters, we differentiate the log likelihood function with respect to each parameter in turn.

$$\frac{\partial l}{\partial \boldsymbol{\mu}_k} = \sum_{i=1}^{n} h_k^{(i)} \Sigma^{-1} (\mathbf{z}_i - \boldsymbol{\mu}_i) = 0 \tag{3.29}$$

$$\sum_{i=1}^{n} h_k^{(i)} \Sigma^{-1} \boldsymbol{\mu}_i = \sum_{i=1}^{n} h_k^{(i)} \Sigma^{-1} \mathbf{z}_i \tag{3.30}$$

$$\sum_{i=1}^{n} h_k^{(i)} (\Sigma\Sigma^{-1}) \boldsymbol{\mu}_i = \sum_{i=1}^{n} h_k^{(i)} (\Sigma\Sigma^{-1}) \mathbf{z}_i \tag{3.31}$$

$$\sum_{i=1}^{n} h_k^{(i)} \boldsymbol{\mu}_i = \sum_{i=1}^{n} h_k^{(i)} \mathbf{z}_i \tag{3.32}$$

$$\boldsymbol{\mu}_i = \frac{\sum_{i=1}^{n} h_k^{(i)} \mathbf{z}_i}{\sum_{i=1}^{n} h_k^{(i)}} \tag{3.33}$$

$$\tag{3.34}$$

Similarly, the log likelihood is differentiated with respect to the remaining parameters (taking care to normalize the prior probabilities), yielding the following M-step updates [5]:

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{n} h_k^{(i)} \mathbf{x}^{(i)}}{\sum_{i=1}^{n} h_k^{(i)}} \tag{3.35}$$

$$\pi_k = \frac{\sum_{i=1}^{n} h_k^{(i)}}{n} \tag{3.36}$$

$$\Sigma_k = \frac{\sum_{n=1}^{n} h_k^{(i)} (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k)^T (\mathbf{x}^{(i)} - \boldsymbol{\mu}_k)}{\sum_{i=1}^{n} h_k^{(i)}} \tag{3.37}$$

When Σ_k is a diagonal matrix of constant variance σ^2 for every class, the Expectation-Maximization algorithm actually reduces to K-means as $\sigma \rightarrow 0$. The EM approach for mixture model parameter estimation generally works well for relatively low dimensional data without significant overlap. However, the algorithm can get trapped in a local minimum due to poor initial conditions.

3.3 Parzen Windowing

The single Gaussian and mixture model are density estimators that require the estimation of parameters based on the training set. Non-parametric models such as

Parzen Windowing are derived by directly sampling from the training data, and do not require a model-fitting stage [81]. A Parzen density estimator consists of a *kernel function* $f(\mathbf{x})$ that is centered at each training element. The sum of the kernel functions represents the probability distribution. Typical kernels include a symmetric Gaussian (Eq. 3.15) or the uniform distribution:

$$f(\mathbf{x}) = \frac{1}{w^d} \quad |\mathbf{x}_i| < \frac{w}{2} \quad i = 1, \ldots, d \tag{3.38}$$

Given the windowing function, the probability density function is derived as

$$P(\mathbf{x}) = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x} - \mathbf{z}_i) \tag{3.39}$$

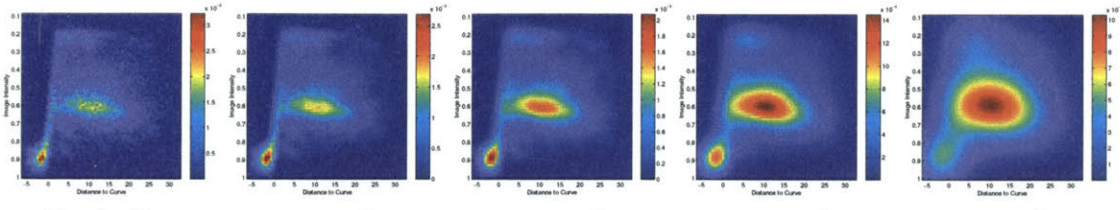
The windowing function f must satisfy the following for $P(\mathbf{x})$ to be a probability:

$$\int_{-\infty}^{+\infty} f(\mathbf{x})\, d\mathbf{x} = 1 \tag{3.40}$$

$$\forall \mathbf{x}\; f(\mathbf{x}) \geq 0 \tag{3.41}$$

The choice of the windowing function is an important factor, dependent on the application. The uniform model is much more efficient, and may be suitable in situations where there is plenty of training data. Since the uniform model has a hard cutoff, there will be regions of the space where the probability is exactly zero, which could cause instabilities in certain types of algorithms (such as gradient ascent on the log likelihood). The Gaussian windowing function is more computationally expensive, but offers a smoother decay.

No matter which form of function is selected, a scale parameter must be chosen. In the uniform case, the scale is the extent of the distribution, w, and in the normal distribution, the smoothness is the variance, σ. Smoothing the distribution in this way provides robustness to noise in the training data and to sampling errors. Smoothing too much weakens the model (in the extreme, infinite smoothing creates a uniform distribution). Smoothing too little causes holes in the probability distribution, yielding incorrect probability (using a delta function for each training element yields zero



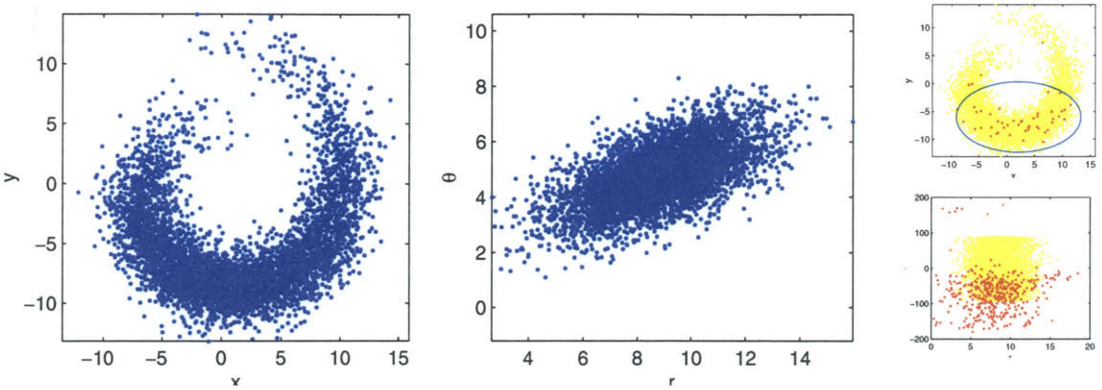
Bucketing $\sigma = 1$ $\sigma = 2$ $\sigma = 4$ $\sigma = 8$

Figure 3-3: Probability density functions estimated via Parzen Windowing with increasing smoothing parameters. Smoothing helps generalize a weak model, but too much smoothing results in loss of fidelity.

probability for any novel point). Figure 3-3 shows various Parzen windowing with various smoothing parameters. When one has a very large amount of sample training data relative to the space to fill, a low scale parameter can be used. High smoothing is generally required in cases of sparse data, where lack of smoothing causes sampling errors.

One approach to computing the scale parameter and assessing the model fit is *cross-validation* [95, 5]. In this scheme, the training set, $\mathcal{T}$, is partitioned into s partitions, $\{\mathcal{T}_1, \ldots, \mathcal{T}_s\}$. The model is then fit using the set $\{\mathcal{T}_1, \ldots, \mathcal{T}_{i-1}, \mathcal{T}_{i+1}, \ldots, \mathcal{T}_s\}$ and then tested by computing the probability of $\mathcal{T}_i$, for every i. When $s = n$, this process is referred to as *leave-one-out cross validation.* While such a method of selecting the smoothing parameter is more well founded than the often-used empirical approach, cross-validation can greatly increase the time it takes to determine the model, as the density function must be recomputed many times for each partition.

When the number of training elements is relatively large, it is inefficient to compute the sum in Equation 3.39 to compute the probability of every novel point $\mathbf{x}$. Instead, a table can be generated that *a priori* samples the probability density function over the space of $\mathbf{x}$. Then, computing a new probability only consists of a table lookup, and perhaps an interpolation. For high dimensional data, the table itself can be quite large, and infeasible to store, so one must balance the time/space tradeoffs.



a b c

Figure 3-4: A random set of points under some distribution. A sophisticated model is required when using a Cartesian representation (a), while the points in polar coordinates are very close to a Gaussian (b). A sample of 50 red points from the unknown distribution is chosen in (c, top), and appear roughly Gaussian. A Gaussian model is fit and then samples from the model are shown back in polar coordinates in red (c, bottom) with samples (in yellow) from the true distribution.

3.4 Discussion

When using any of the above density estimators to build a statistical model, important issues arises that can greatly affect the accuracy and reliability of the model. The choice of data representation greatly influences the ability of the model to generalize. Most models generalize by considering regions "near" elements of the training set to be more likely than those far away. While all the feasible instances may actually be near each other in some ideal space, in an inconsistent or unnatural choice of embedding space, the likely elements but could be spread out with unlikely regions in between.

3.4.1 Choice of Representation

The representation chosen for the training data has significant impact on the ability to derive an accurate, concise model. Consider the example distribution of points illustrated in Figure 3-4a. If we represent an example by the obvious choice, say, $\mathbf{z} = [x\ y]^{\mathsf{T}}$, then this distribution is clearly non-Gaussian. If this representation is to be used, then a more sophisticated model must be fit, such as the mixture of Gaussians or a Parzen estimation. However, consider instead representing the example as $\mathbf{z} = [r \;\; \theta]^{\mathsf{T}}$. Figure 3-4b shows a scatter plot of the same points in this basis. In polar coordinates, the distribution can be accurately encoded by 5 degrees of freedom (2 for $\boldsymbol{\mu}$ and 3 for the symmetrical Σ), whereas under the Cartesian representation, many more parameters would be required to accurately estimate the distribution.

While this Cartesian/polar example is trivial, issues of the choice of representation arise often and are crucial to the degree to which the model captures the true underlying distribution. An incorrect model will cause likely instances to be discounted or unlikely instances to be considered plausible, resulting in the model influencing the algorithm in a detrimental manner.

The choice of representation is very important when modeling variances in shape across a population. Generally we are given a relatively small number of example shapes (as compared to the large number of degrees of freedom of a shape) from which we wish to generalize the plausible variances. Consider if we only had access to around 50 random points in the previous example (Figure 3-4c). If they happen to fall mostly at the bottom of the spiral (as is most likely), a Gaussian model would seem to represent them reasonably well, but would actually be a very poor model of the true underlying distribution. The wrong choice of shape representation (or a misalignment of the features of the training objects) results is this type of error, as discussed further in Chapter 5.

Chapter 4

Intensity Priors in Registration

Various modalities of medical imagery are commonly used together in diagnosis or planning, as each modality captures different types of information about the anatomy. Soft tissue, for example, is imaged well in MR scans, while bone is more easily discernible in CT scans. Blood vessels are often highlighted better in an MR angiogram than in a standard MR scan. Figure 4-1 shows three different acquisitions of MR scans. Notice that some anatomical structures appear with more contrast in one image than in the others. The union of all the imagery is most effective when all are presented in the same coordinate system. Since the scans are acquired in different machinery and the patient is in different positions, the transformations between the various acquisitions is generally not readily available. Chapter 2 described various methods of fusing medical images, including alignment using fiducials, landmarks, surfaces, and voxels.

This chapter describes a voxel-based multi-modal registration algorithm that incorporates prior knowledge on the expected joint intensity distribution of the two images when correctly aligned. The goal of the joint intensity model is to estimate the relationship between the intensity values of two imaging modalities. The framework for our registration process is illustrated in Figure 4-2. The training step requires one pair of registered scans of the modalities and anatomical structures to be captured in the model. The intensity pair at a certain point is a sample of a joint distribution of intensities of corresponding points in the two images. This joint distribution is

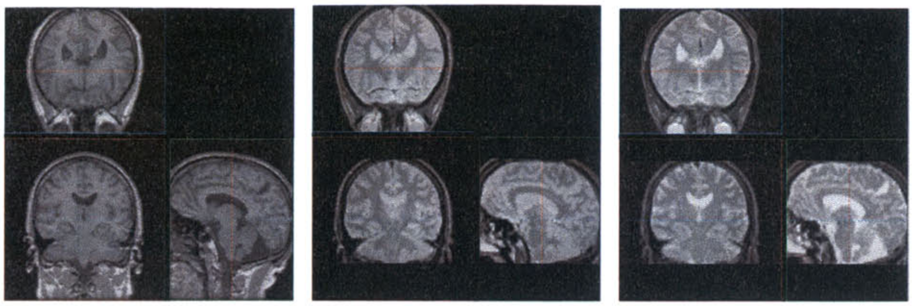


Figure 4-1: SPGR, Proton Density, and T2 Weighted Magnetic Resonance images of a brain. Notice that some anatomical structures appear with more contrast in one image than the others.

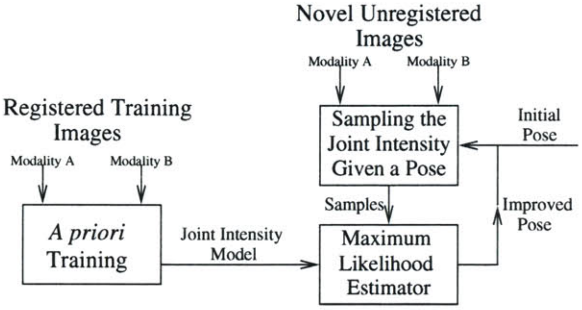


Figure 4-2: Flowchart of the *a priori* training and the online registration.

what we seek to estimate. The model can then be used to register novel scans of the same two modalities and same anatomical structures as in the training set. To align a novel pair of images, we compute the likelihood of the two images given a certain pose based on our model by sampling the intensities at corresponding points. We improve the current hypothesized pose by ascending the log likelihood function.

We consider two models of joint intensity: mixture of Gaussians and Parzen Window Density. In both methods, we seek to estimate the probability of observing a given intensity pair at the corresponding point in the two images.

4.1 Modeling by Mixture of Gaussians

Given a pair of registered images from two different medical image acquisitions, we can assume that each voxel with coordinate $\mathbf{x} = [x_1, x_2, x_3]^T$ in one image, I_1, corresponds to the same position in the patient's anatomy as the voxel with coordinate $\mathbf{x}$ in the other image, I_2. Further, consider that the anatomical structure S_k at some position

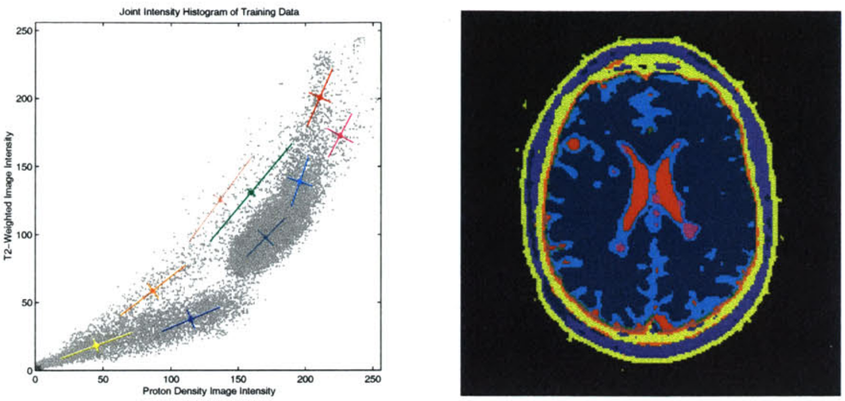


Figure 4-3: LEFT: Joint intensity histogram of the registered MR PD/T2 training images used to fit a mixture of Gaussians. RIGHT: Rough segmentation of a registered image pair. Each voxel is classified based on which Gaussian it most likely belongs to, based on the mixture of Gaussian model.

in the patient will appear with some intensity value i_1 in the first image and i_2 in the second image with joint probability $P(i_1, i_2 \mid S_k)$. We also define $P(S_k) = \pi_k$ to be the prior probability that a random point in the medical scan corresponds to structure S_k.

By making the assumption that voxels are independent samples from this distribution (and ignoring relative positions of voxels), we have

$$\begin{aligned} P(I_1, I_2) &= \prod_{x \in I} P(I_1(x), I_2(x)) && (4.1) \\ &= \prod_{x \in I} \sum_k \pi_k P(I_1(x), I_2(x) \mid S_k) && (4.2) \end{aligned}$$

We model the joint intensity of a particular internal structure S_k to be a two dimensional (dependent) Gaussian with mean $\boldsymbol{\mu}_k$ and full covariance matrix Σ_k. Letting $\mathbf{i}$ be intensity pair $[i_1, i_2]^T$,

$$P(\mathbf{i} \mid S_k) = \left(\frac{1}{2\pi |\Sigma_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{i}-\boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{i}-\boldsymbol{\mu}_k)} \right) \tag{4.3}$$

This model of the intensities corresponds to a mixture of Gaussians distribution, where each 2D Gaussian G_k corresponds to the joint intensity distribution of an internal anatomical structure S_k. Thus, the probability of a certain intensity pair (independent of the anatomical structure) given the model, M is

$$P(\mathbf{i}\,|\,M) = \sum_k \left(\frac{\pi_k}{2\pi |\Sigma_k|^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{i}-\boldsymbol{\mu}_k)^T \Sigma_k^{-1} (\mathbf{i}-\boldsymbol{\mu}_k)} \right). \tag{4.4}$$

To learn the joint intensity distribution under this model, we estimate the parameters π_k, $\boldsymbol{\mu}_k$, and Σ_k using the Expectation-Maximization (EM) method [23].

For the MR image pairs of the brain, a mixture of $K = 10$ Gaussians was used to fit the data. This perhaps is a slight over-fitting of the data, but in this case, it seems this only results in one anatomical structure being explained by more than one class. Figure 4-3a shows the joint intensity histogram of the training image pair with the converged mixture of Gaussian fit superimposed.

The mixture of Gaussians model was chosen to represent the joint intensity distribution because we are imaging a volume with various anatomical structures that respond with different ranges of intensity values in the two acquisitions. We assume that those ranges of responses are approximately Gaussian in nature. Therefore, one might expect that each Gaussian in the mixture may correspond roughly to one type of anatomical structure. In other words, the model produces an approximate segmentation of the structures in the images. Figure 4-3b shows the segmentation of a registered pair of MR images using the Gaussian mixture model prior. Gerig, *et al.* [31] used similar methods of statistical classification to produce accurate unsupervised segmentation of 3D dual-echo MR data.

Segmentation of medical images based solely on intensity classification (without using position or shape information) is, in general, very difficult. Often different tissue types may produce a similar or overlapping range of intensity responses in a given medical scan, making classification by intensity alone quite challenging. MR images include nonlinear gain artifacts due to inhomogeneities in the receiver or transmit coils [51]. Furthermore, the signal can also be degraded by motion artifacts from movement of the patient during the scan.

The segmentation produced by this method shown in Figure 4-3b suffers from the difficulties described above. For example white matter and gray matter have

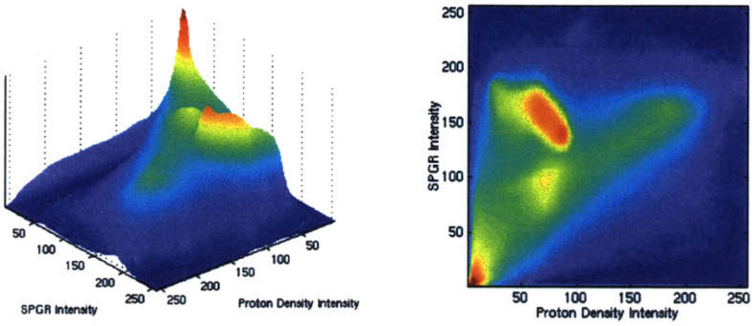


Figure 4-4: Two views of the joint intensity distribution function computed using Parzen estimation with a Gaussian windowing function.

overlapping ranges of intensities in both image acquisitions. Furthermore, note that the distinction between gray and white matter on the right hand side is not segmented clearly. This is most likely due to the bias field present in the image.

Despite these difficulties, the segmentation does a reasonable job of picking out the major structures, although it is inaccurate at region boundaries. Therefore, we do not intend to use this method alone to compute an accurate segmentation of the underlying structures. Instead, we could use the mixture model in combination with a more sophisticated algorithm to solve for segmentation, or for registration purposes, as described in section 4.3.

4.2 Modeling by Parzen Windowing

The prior joint intensity distribution can also be modeled using Parzen window density estimation. A mixture of Gaussians model follows from the idea that the different classes should roughly correspond to different anatomical structures and thus provides an approximate segmentation into tissue classes. However, the EM algorithm for estimating the parameters of a mixture of Gaussians is sensitive to the initialization of the parameters and in some cases can result in an inaccurate prior model of the joint intensities.

We therefore also consider modeling the joint intensity distribution based on the Parzen window density estimation using Gaussians as the windowing function. In practice, this model defined by directly sampling the training data provides a better

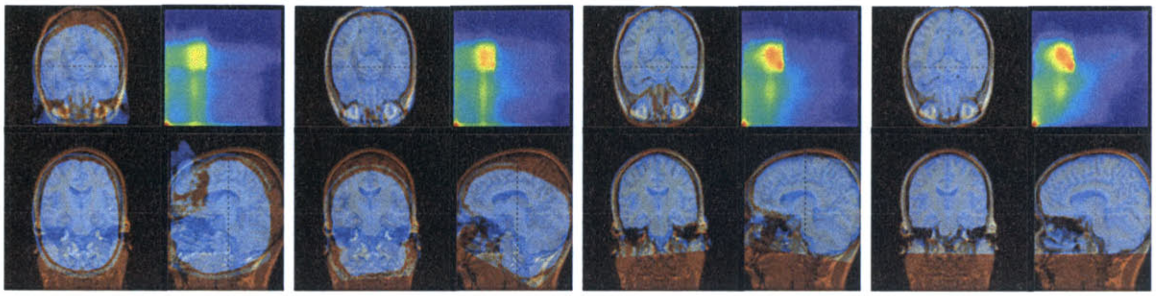


Figure 4-5: Starting, middle, and the final alignments computed by the registration gradient ascent algorithm. Each image shows the colorized SPGR and PD images overlaid in the three orthogonal slices. The images in the upper right depict the histogram of the intensity pairs at that alignment. When the images are aligned, the histogram should resemble the distribution in Figure 4-4.

explanation of the intensity relationship than the Gaussian mixtures that require the estimation of various parameters.

Consider our registered training image pair $\langle I_1, I_2 \rangle$. We estimate the joint intensity distribution of an intensity pair $\mathbf{i} = [i_1, i_2]^T$ given the prior model, M:

$$P(\mathbf{i} \,|\, M) = \frac{1}{N} \sum_{\mu \in \langle I_1, I_2 \rangle} \left(\frac{1}{2\pi\sigma^2} e^{-\frac{1}{2\sigma^2}(\mathbf{i}-\boldsymbol{\mu})^T(\mathbf{i}-\boldsymbol{\mu})} \right) , \tag{4.5}$$

where the $\boldsymbol{\mu}$'s are N samples of corresponding intensity pairs from the training images. Figure 4-4 illustrates this estimated joint intensity distribution.

4.3 Maximum Likelihood Registration

Given a novel pair of unregistered images of the same modalities as our training images, we assume that when registered, the joint intensity distribution of the novel images should be similar to that of the training data. When mis-registered, one structure in the first image will overlap a different structure in the second image, and the joint intensity distribution will most likely look quite different from the learned model. Given a hypothesis of registration transformation, T, and the Gaussian mixture model, M, we can compute the likelihood of the two images using Equation

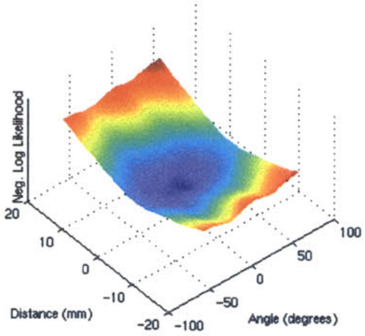


Figure 4-6: Samples from the negative log likelihood function over various angles and x-shifts. Note that over this range, the function is very smooth and has one distinct minimum, which in this case occurs 0.86 mm away from the correct alignment.

4.1:

$$P(I_1, I_2 \mid T, M) = \prod_{\mathbf{x}} P(I_1(\mathbf{x}), I_2(T(\mathbf{x})) \mid T, M). \tag{4.6}$$

We register the images by maximizing the log likelihood of the images, given the transformation and the model, and define the maximum likelihood transformation, T_{ML}, as follows:

$$\hat{T}_{ML} = \underset{T}{\operatorname{argmax}} \sum_{\mathbf{x}} \log(P(I_1(\mathbf{x}), I_2(T(\mathbf{x})) \mid T, M)) \tag{4.7}$$

The likelihood term in this equation can be substituted with either Equation 4.4 or 4.5, depending on which joint intensity model is chosen. For the results presented here, the Parzen model is used, as it better explains the intensity relationship between the two modalities. However, the mixture of Gaussians model encodes coarse tissue type classes and thus provides a framework for later incorporating into the registration process prior knowledge of the relative positions and shapes of the various internal structures.

To find the maximum likelihood transformation, T_{ML}, we use Powell maximization [84] to ascend the log likelihood function defined in Equation 4.7, finding the best rigid transformation. In both the mixture of Gaussian and Parzen window models of the distribution, the log likelihood objective function is quite smooth. Figure 4-6 illustrates samples from the negated objective function for various rotation angles

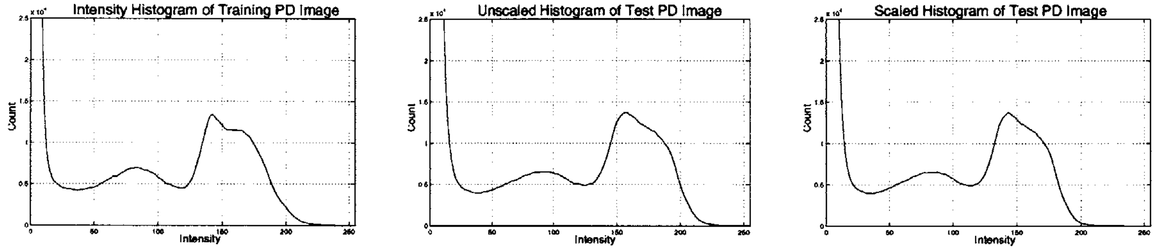


Figure 4-7: One dimensional histograms of two PD weighted images. Notice that the training image histogram (left) is not aligned with the test image histogram (center). We therefore scale the intensity values by the appropriate factor to best align the histograms. The scaled test histogram appears on the right.

(along one dimension) and x position shifts of the transformation. Over this sampled range of ±60 degrees and ±20 mm, the function is always concave and has one minimum which occurs within a millimeter of the correct transformation. Computing the registration by maximizing the likelihood of the image pair given the transformation and the model seems to be an efficient, accurate method of registration.

4.4 Intensity Histogram Normalization

The registration method presented here relies on the joint intensity distribution of the registered test images being similar to that of the training data. In some cases, due to an intensity artifacts present in the image or other related factors, the registered image pair will not have a similar distribution to that of the training data. Consider the 1D histograms in Figure 4-7. Notice that the unscaled test image intensity values (center) are not aligned with that of the training image (left). To resolve this problem, we adjust the intensities of each novel image to make the test image intensity distribution more closely match that of the training data. We find the factor by which to scale the image intensities so that the intensity histogram best matches that of the corresponding modality used in training.

PD / T2 Registration

#	Error	#	Error	#	Error	#	Error
1	0.74	10	0.79	19	1.42	28	1.26
2	0.89	11	0.73	20	0.75	29	0.76
3	0.86	12	0.68	21	0.69	30	2.24
4	0.79	13	1.52	22	1.33	31	0.90
5	0.90	14	0.63	23	0.66	32	0.78
6	0.80	15	0.80	24	1.01	33	1.28
7	0.81	16	0.89	25	1.07	34	0.74
8	0.80	17	0.82	26	0.70	35	1.08
9	0.82	18	1.36	27	0.81	36	0.89

Table 4.1: The results of registering 36 test images. The registration error (in mm) was computed by taking the average error of the eight vertices of the imaging volume.

4.5 Results

The training and registration algorithms described above were tested using MR datasets from 37 patients[1]. Each patient was scanned using two protocols, a coronal SPGR scan of $256 \times 256 \times 124$ voxels with a voxel size of $0.9375 \times 0.9375 \times 1.5$ mm and a dual echo (proton density and T2-weighted) axial scan of $256 \times 256 \times 52$ voxels with a voxel size of $0.9753 \times 0.9375 \times 3.0$ mm. One of the patients' datasets was used for training purposes and the remaining 36 were used for testing. The two types of registration performed in these experiments were PD with T2 and SPGR with PD. The joint intensity distribution for each modality pair was modeled using a Parzen estimation. The initial pose in all cases was about 90° and a few centimeters away. Each registration converged in $1 - 2$ minutes on a Pentium Pro 200. Tables 4.1 and 4.2 show the results of the registrations.

4.5.1 Error Metric

Given a transformation T produced by registration algorithm and the ground truth alignment, T^*, we consider various error measures that accurately represent the misalignment. One commonly used method of evaluating error is to compare the trans-

---

[1] The MR datasets were provided by M. Shenton and others, see acknowledgments.

SPGR / PD Registration

#	Error	#	Error	#	Error	#	Error
1	1.45*	10	1.47	19	2.57	28	1.54
2	2.69	11	3.17	20	2.67	29	2.15
3	0.16	12	1.32	21	2.77	30	1.50
4	0.78	13	1.18	22	3.47	31	1.43
5	1.69	14	1.59	23	3.87	32	1.83
6	1.53*	15	1.25	24	2.48*	33	1.64
7	0.85*	16	1.26	25	1.60*	34	3.18
8	1.36	17	1.19	26	2.03	35	2.01
9	0.73	18	1.85	27	1.31*	36	2.46

Table 4.2: The results of registering 36 test images. The registration error (in mm) was computed by taking the average error of the eight vertices of the imaging volume. *The patient moved between scans and thus scanner coordinates cannot be used as ground truth. The ground truth registration for these scans was determined by aligning manually-defined corresponding landmarks.

formation parameters themselves. For example, T could be presented as Euler angles and a translation vector: $T = \langle \theta,\ \phi,\ \psi,\ \mathbf{t}_x,\ \mathbf{t}_y,\ \mathbf{t}_z \rangle$, and the error for each parameter is computed, such as:

$$\begin{array}{lll} \mathcal{E}_\theta = |\theta - \theta^*| & \mathcal{E}_\phi = |\phi - \phi^*| & \mathcal{E}_\psi = |\psi - \psi^*| \\ \mathcal{E}_{\mathbf{t}_x} = |\mathbf{t}_x - \mathbf{t}_x^*| & \mathcal{E}_{\mathbf{t}_y} = |\mathbf{t}_y - \mathbf{t}_y^*| & \mathcal{E}_{\mathbf{t}_z} = |\mathbf{t}_z - \mathbf{t}_z^*| \end{array} \tag{4.8}$$

Representing alignment error as a function of Euler angles can be misleading due to the non-linearities of 3D rotation and the non-uniformity of the space spanned by the three angles. If, for example, the Euler rotations were performed in a different order, the magnitudes of the angles could be very different even for the same transformation. Ideally error should be independent of the choice of parameterization of the transformation.

Another approach to measuring error is to report two values, translation and rotation offsets, which is less dependent on the parameterization. To compute these values, consider representing T in a quaternion-like form such as $T = \langle \omega,\ \mathbf{v}_x,\ \mathbf{v}_y,\ \mathbf{v}_z,\ \mathbf{t}_x,\ \mathbf{t}_y,\ \mathbf{t}_z \rangle$, where $\mathbf{v}$ is a unit vector axis of rotation and ω is the rotation angle. In this framework, rotation and translation angle is computed by:

$$\mathcal{E}_\omega = |\omega - \omega^*| \tag{4.9}$$

$$\mathcal{E}_\mathbf{t} = ||\mathbf{t} - \mathbf{t}^*|| \tag{4.10}$$

In this approach, the rotation error can still be misleading, as it still depends on the point about which the rotation is performed.

The error metric we use in this work computes the average error of transformed points in a region of interest specified by the user, and is completely independent of the way in which the transformations are parameterized. Consider a point $\mathbf{v}$ in the image I_2. Based on the ground truth assumption, $T^*(\mathbf{v})$ is the corresponding point in I_1. The point $T(\mathbf{v})$ is the *computed* corresponding point, and the distance between the two transformed points is the transformation error *for that particular point.* If all points of interest in the images were known, then the average or RMS of the distances between the points under two transformations could be used as a metric of the expected error of a random point in the region of interest, $\mathcal{V}$:

$$E_\mathcal{V} = \frac{1}{|\mathcal{V}|} \sum_{\mathbf{v} \in \mathcal{V}} |T(\mathbf{v}) - T^*(\mathbf{v})| \tag{4.11}$$

For the brain images used here, we did not select out a particular region of interest $\mathcal{V}$ per se, but instead choose $\mathcal{V}$ to consist of the eight vertices of the imaging cube. Note that this bounding box is an *overestimate* of the region of interest, as it contains a few centimeters of empty space around the patient's head, where the voxel information does not contribute to the registration.

4.5.2 Proton Density and T2-Weighted Images

We first consider the problem of registering PD images to transformed T2 as a means of testing the registration algorithm. Each T2-weighted scan was rotated by 90° and translated 5 cm and then registered with the corresponding PD scan. Since these T2 and PD images were acquired at the same time and are originally in the same coordinate system, by perturbing one scan by a known amount, we have accurate ground-truth upon which to validate the registration process. All 36 test cases registered with sub-voxel, and in most cases with sub-millimeter accuracy. Note that the method has a large region of convergence and thus does not require the starting pose to be very near the correct solution.

4.5.3 SPGR and Proton Density Images

A more interesting and more challenging alignment problem consists of registering each patient's coronal SPGR scan with the same patient's PD image. This registration problem is more challenging given that the two modalities were not acquired simultaneously and also do not contain the same region of the head: the PD/T2 images are cropped at the chin (see Figure 4-1).

Since each patient had the scans performed during the same sitting in the scanner, the headers of the scans provide the ground-truth alignment between the various acquisitions, assuming the patient did not move between acquisitions. However, since the patients' heads were not fixed in the scanner, patients could move between acquisitions. Despite this issue, we use the scanner poses as ground truth, since in most cases it seems that the patient did not move significantly. In six cases (marked with an $*$ in Table 4.2), the patients did move significantly (by a few centimeters) in the scanner between the SPGR and PD acquisitions. Since scanner coordinates do not provide meaningful ground truth for these cases, the ground truth alignment was determined by manually selecting corresponding landmarks in each scan and computing the best rigid alignment given the corresponding points [45]. For all cases, the initial pose was set to the identity, which, due to the directionality of the acquisitions corresponded to a starting position of about 90° and a few centimeters away from the correct poses. Of the 36 cases we have used in our initial tests of this method, almost all of the cases automatically registered to within one voxel.

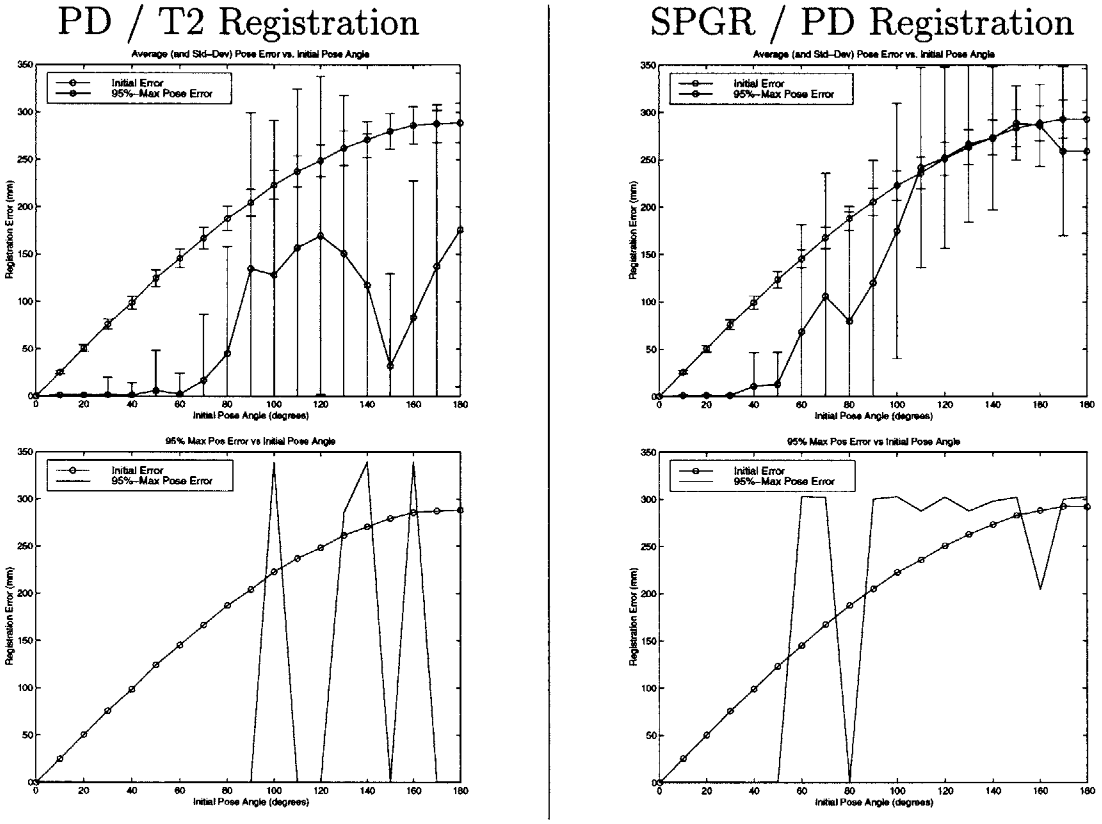


Figure 4-8: Analysis of the convergence of the algorithm from various initial angle offsets. The left-hand plots illustrate results for PD / T2 registration and right-hand plots illustrate results for the SPGR / PD registration. The top graphs show the average and standard deviation of final pose error. The bottom row shows the 95% of maximum final error.

4.5.4 Region of Convergence

To evaluate the robustness of the algorithm to initial alignment, the starting pose was varied and the registration was performed repeatedly at initial angle offsets of zero to 180 degrees, at ten degree increments. For each trial, a random rotation axis was selected, and the initial pose was perturbed by the given angle about the random axis. Three hundred trials were run at each angle offset. The results of this experiment are shown in Figure 4-8. For the PD / T2 experiments, the algorithm correctly registered the scans to within a voxel 95% of the time from as far as 90° away, and often even registered correctly when a full 180° away. For the SPGR / PD experiments, the algorithm successfully registered the scans 95% of the time for up to about 60° initial

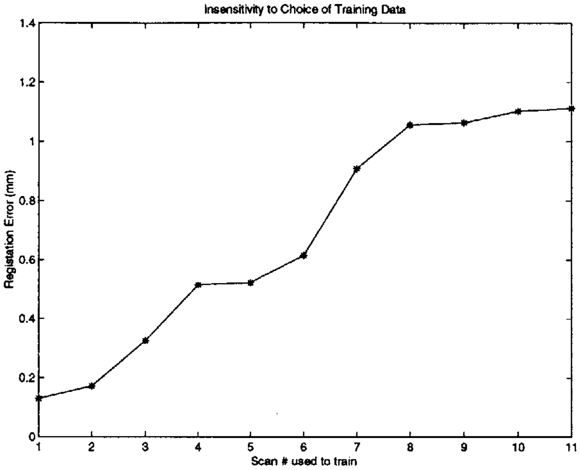


Figure 4-9: The graph indicates the robustness of the algorithm over choice of training data. For each trial, one of eleven different scans was used as training data and one other scan was used as the test. For clarity, the scan numbers on x-axis are sorted based on error. All trials resulted in an alignment within one voxel of the ground truth and within 1 mm of all other trials.

offset, still showing a relatively large region of convergence.

4.5.5 Insensitivity to Choice of Training Data

One patient's scans were selected to be used as the training data for all of the registration experiments described earlier in this section. To assess the dependence on the particular scans chosen as training data, an experiment was run where training was performed once on each of eleven different scans, and a novel scan was registered once for each of the eleven training sets. The results of the registrations are shown in Figure 4-9. Each time, the novel scan was registered to within a voxel of the ground truth, with a variance of 0.13 mm.

4.6 Discussion

The multi-modal registration method described here incorporates prior knowledge of the intensity relationship between the two imaging modalities based on already registered training images. A joint intensity distribution, $P(i_1, i_2)$, is derived and then used to compute the maximum likelihood pose of two novel images as in Equation 4.7. Note that each voxel is assumed to be completely independent from all other voxels, which is a not a very realistic assumption. While the algorithm performs well despite the independence assumption, the a stronger, spatially dependent model may converge more quickly and from further starting points given the additional information.

One approach to strengthen the model is to derive a distribution over small neighborhoods of both images. This approach could certainly assist in registering highly textured regions. However, more training data may been required to fill the larger probability space. Another approach to relaxing the independence assumption is to compare the distributions as a whole. Instead of finding a maximum likelihood set of independent intensity pairs, one could use the Kullback Leibler distance to minimize the divergence between the training and sampled distribution, effectively finding a typical set of intensity pairs.

The use of statistical models for registration may be even more beneficial when applied to elastic registration problems. In the non-rigid case, there are significantly more alignment parameters to estimate, and a strong joint intensity model can potentially direct the parameter search more effectively than without any prior knowledge.

Chapter 5

Statistical Priors for Segmentation

Statistical models have the potential to add useful information to the segmentation process based on example training data. Instead of simply looking for high gradients or requiring the user to interactively cue the algorithm, we seek to encode into the algorithm answers to questions such as these:

- "What shape am I looking for?"
- "What intensity profile should I expect?"
- "How should the local curvature behave?"
- "What are the relative positions of the objects?"

We begin by discussing means of representing shape and building distributions over shape variances present across a population. The shape model used in this work is defined and compared to other shape representations. We then consider methods of adding statistical intensity information into the model to relax the common assumption that high gradients appear at the boundary. While the shape model capture global shape information, a curvature prior is used to encode the type of local curve regularization required. Segmentation experiments and results are discussed along the way on both synthetic images and 2D and 3D medical scans.

5.1 Statistical Distribution over Shape

To incorporate shape information into the process of segmenting an object in an image, we consider a probabilistic approach, and compute a prior on shape variation given a set of training instances. To build the shape model, we choose a representation of curves, and then define a probability density function over the parameters of the representation.

5.1.1 Curve Representations

Accurately modeling the shape variance across a distribution of examples is difficult, and requires careful choice of the shape representation. The ability of the model to capture the true deformations across a population is closely tied to issues of invariance and correspondence, which we discuss here. Invariance relates to the types of transformations that affect the representation. For example, the measure of an object's volume is invariant to pose, but not to scale. When comparing two objects, correspondence involves the matching of similar parts of an object, which is crucial when the objects differ in a way to which their representation is sensitive. Comparing two identical squares that are misaligned by 45° will result in a mismatch or poor model of population variance if the representation is not invariant to rotation.

When defining the shape representation, one must ask, "What types of differences do I care about, and what are nuisance parameters?" For example, since each object or image is generally given in its own coordinate system, the pose of the object is not an important factor when trying to capture object shape variance. Before two objects can be compared, they must be aligned or, equivalently, transferred into a representation that is pose invariant. Therefore, pose is a nuisance parameter in this scenario.

Scale, however, may or may not be important depending on the application. In computer vision, scale invariance is typically desired, since the size of the projection in an image depends on distance from the object to the camera. If we care about the object itself, and not the distance from the camera, scale becomes a nuisance.

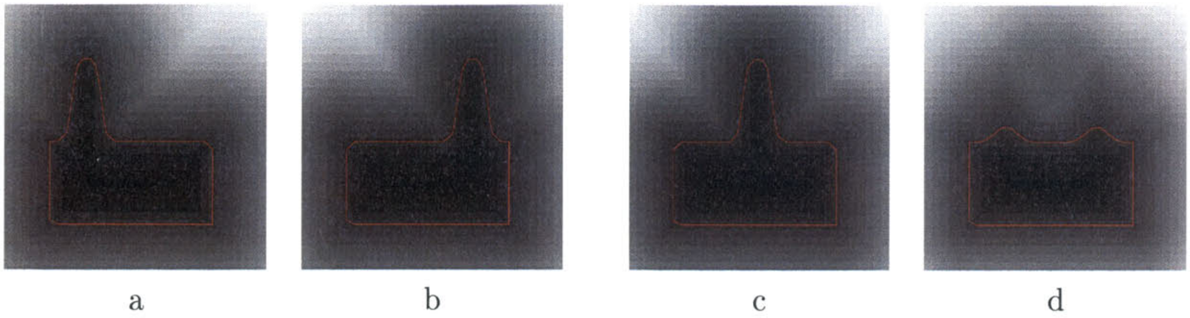

a b c d

Figure 5-1: An illustration of shape correspondence errors. (a, b) Two curves represented as zero level sets of the signed distance function. (c) A "reasonable" mean or half-way morph between the two shapes, that resembles each of the originals. (d) The result of averaging the distance map, which is the result of comparing un-like parts of the object. Any shape representation that does not solve the correspondence problem at some level will be susceptible to these issues.

In medical applications, a 3D scan accurately portrays the size of an object, so scale changes reflect the actual characteristic of the object, which should be captured by the model (e.g. clinically, an enlarged kidney is an important characteristic).

Even with a representation that factors out all of the nuisance parameters, the issue of correspondence still arises due to the importance of comparing only like regions of the objects. A shape model can capture the variation of a certain feature of an object, only if it is compared to the same feature in other objects. For example, when looking at variances in the shape of the vertebrae, if two training examples are misaligned and a process of one is overlapping a notch of the other, then the model will not be capturing the appropriate anatomical shape variance seen across vertebrae. Figure 5-1 shows an example of correspondence mismatches. The bump in this object shifts from left to right, and most shape representations will cause misalignment of this feature. One method of inspecting correspondence between two objects is to morph between them. In general, if the intermediate stages of the morph appear to be from the same class of object, then the corresponding regions of the objects are reasonably matched.

A common means of representing a 1D curve in 2D space is to parameterize it by some measure such as arc length. Given two curves $\mathcal{C}_1(p)$ and $\mathcal{C}_2(q)$, the average

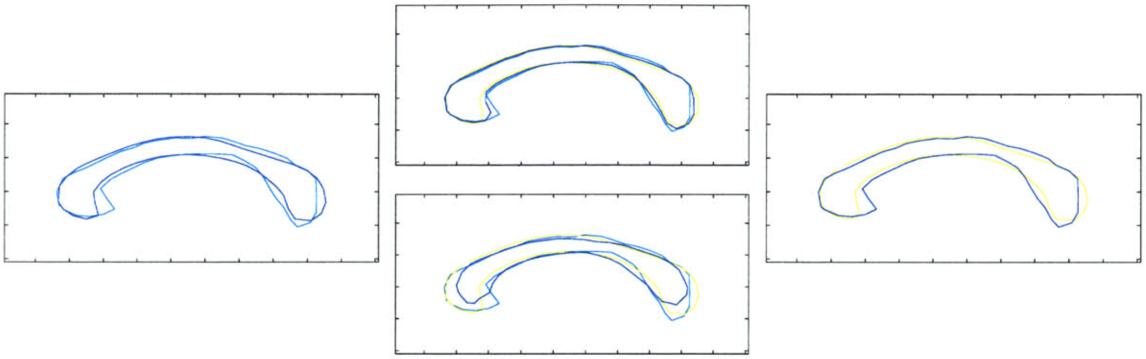


Figure 5-2: Correspondence errors when combining parameterized curves. LEFT and RIGHT: Boundary curves of two different corpora callosa. TOP: The result of averaging the curves when their parameter is in correspondence. The mean curve looks like a corpus callosum. BOTTOM: A small mismatch in correspondence causes the average to look unlike the original shapes.

curve can be computed as:

$$\mathcal{C}(p) = \frac{\mathcal{C}_1(p) + \mathcal{C}_2(T(q))}{2} \tag{5.1}$$

where T determines the correspondence between $\mathcal{C}_1(p)$ and $\mathcal{C}_2(q)$. The appearance of the average depends greatly on the correctness of the transform T (see Figure 5-2). Similarly, when relating or combining multiple curves in order to derive a shape distribution, the consistency of parameterizations of the training curves is important to capture the actual shape variances present in the population.

One solution to the correspondence problem is to explicitly generate all point-wise correspondences to ensure that comparisons are done consistently. In general, localizing corresponding features automatically is a very difficult problem. Low level vision algorithms offer detectors for features such as edges, ridges, lines, and corners, but then robust correspondence of these features is still necessary. Higher level features such as the base of the femur condile or the valley of the right central sulcus are more difficult to detect reliably without a template and are more limited in number. In reality, we are back to the problem of non-rigid registration just to create the shape model, to later use for segmentation.

Thus, algorithms that require point-wise correspondence for their training set generally rely on significant human interaction. Manually localizing corresponding landmarks is tedious and also prone to errors, especially when dealing with 3D objects. There are very few naturally occurring feature *points* (e.g. zero-dimensional) in 3D anatomical structures. Instead, features are typically one-dimensional curves such as ridges of sulci and gyri, or even two-dimensional surfaces such as the falx. This, coupled with the fact that 3D imagery is often viewed a only one slice at a time, makes the consistent selection of points difficult. In [18], Cootes *et al.* require the labeling of 123 corresponding landmarks on each of 72 training instances when building a 2D model of the region of the brain around the ventricles. Almost 9000 mouse clicks were used to define only a 2D model. In many applications, especially in 3D, careful labeling of such a large training set is infeasible.

If the point correspondence are labeled correctly, the Active Shape Models of Cootes, Taylor, *et al.* do an excellent job of representing the shape deformation and performing the model-based matching. A vector $\mathbf{v}_i$ representing the ith training shape is defined from the m corresponding feature points:

$$\mathbf{v}_i = [x_1 \;\; y_1 \;\; z_1 \;\; x_2 \;\; y_2 \;\; z_2 \;\; \ldots \;\; x_m \;\; y_m \;\; z_m]^{\mathsf{T}} \tag{5.2}$$

A Gaussian model is then fit to the training set, $\mathcal{T} = \{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ as described in Chapter 3. Mixture models are also used to fit to the data, especially when the distributions are clearly non-Gaussian, due to the type of data in a specific application [17]. A small number of principal modes are determined and each instance of the object can be represented by a vector of coefficients. The model is then matched to a novel image, by searching through the space of coefficients for a shape whose boundary falls on edges in the image. A detailed report on Active Shape and Appearance Models, still under preparation, can be found on their web site [19].

Instead of representing a shape as a spline of node points along the boundary of the object, the skeleton or medial axis can be used. Node points (also called core atoms) can be placed along the skeleton of the object, equidistant from two

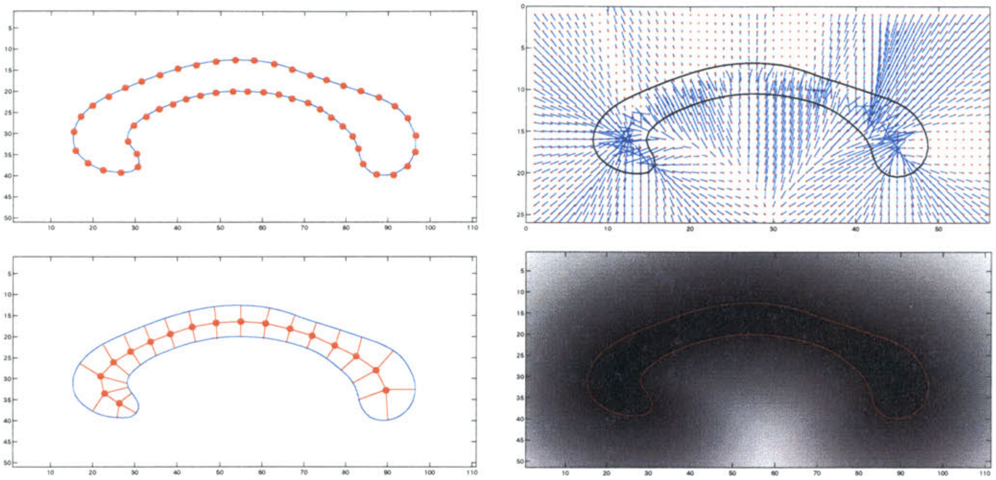


Figure 5-3: LEFT: Two examples of Lagrangian representations of a curve, a spline parameterized by node points (top), and the medial axis (bottom). RIGHT: Eulerian representations, namely a dense flow field from a known shape (top) and the signed distance map (bottom).

different points on the boundary. Each core atom encodes a local coordinate system as it stores the distance to the curve and the relative angle of the two connecting segments. Figure 5-3 includes an image of a medial axis representation. Medial-node models are generally constructed by hand and then fit to a set of training data in order to derive a distribution of local shape properties [101]. Given a novel image, the model is fit to the image data while each node includes a local shape prior. In this approach, the correspondence is solved for the training data using supervised gradient-based model-fitting. For a novel image, the same image force is used, with the local shape priors included.

A shape model has potential to be much stronger when the model is derived from a set of training examples of the same class of object to be segmented. Staib and Duncan [94] incorporate global shape information into the segmentation task by using an elliptic Fourier decomposition of the boundary. Figure 5-4 shows an example of a hierarchical set of coupled ellipsis such that each ellipse moves along the path of the parent one at a certain rate, resulting in the desired curve being drawn. The parameters of this representation consist of the Fourier descriptors of the ellipses. Based on a training set of curves, a Gaussian prior is placed on all the

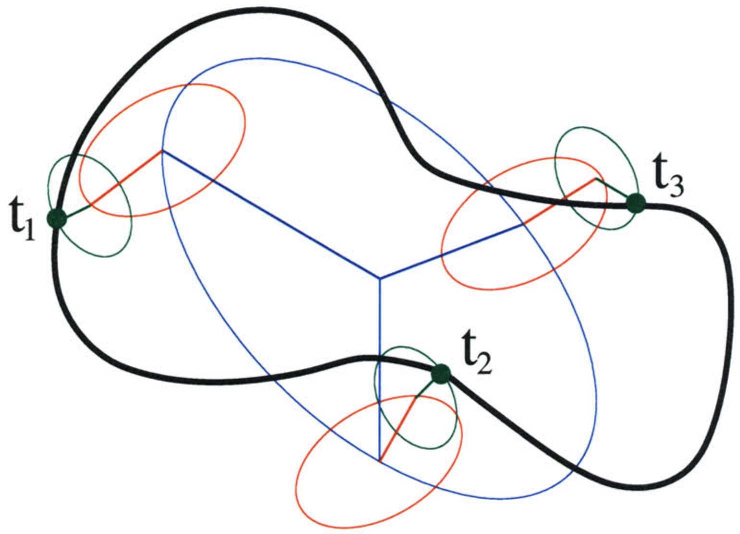


Figure 5-4: Fourier decomposition of a curve. As each ellipse moves along the path of the preceding one at a predetermined rate, the black curve is drawn. Three time steps are shown. (Figure reproduced from [94])

Fourier coefficients. An object in a new image is localized by adjusting the parameters using a maximum *a posteriori* approach, combining the image term, that places the computed boundary along an edge, and a prior term, which seeks probable coefficients given the model.

The representations discussed thus far have been Lagrangian in nature, meaning that coordinates representing the shape move as the shape changes. In contrast, and Eulerian representation is one where the underlying coordinate system remains fixed as the shapes change. Figure 5-3 shows an example of both Eulerian and Lagrangian representations.

5.1.2 Distance Maps

We propose the *signed distance map* as a representation to model shape variance. A curve $\mathcal{C}$ to be represented is embedded as the zero level set of a higher dimensional surface, u, whose height is sampled at regular intervals (say N^d samples, where d is the number of dimensions). Each sample encodes the distance to the nearest point on the curve, with negative values inside the curve. (see Figure 5-5). Formally, the

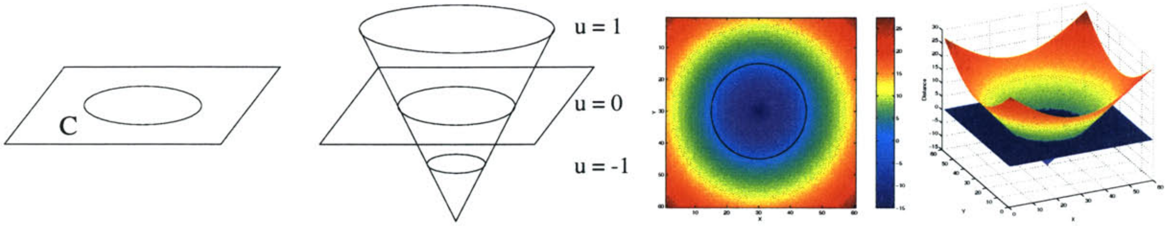


Figure 5-5: Level sets of an embedding function u for a closed curve $\mathcal{C}$ in $\Re^2$.

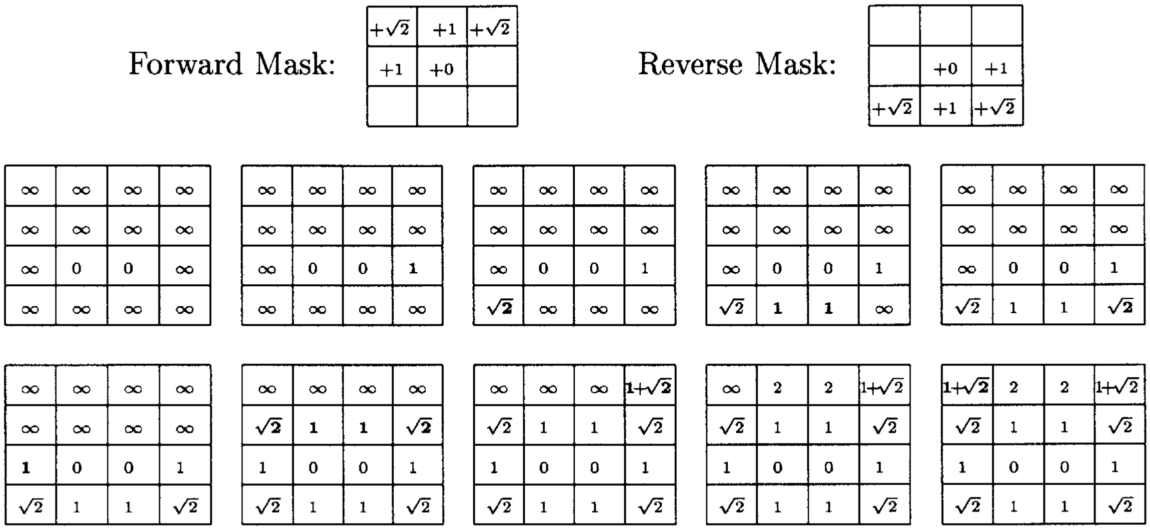
unsigned surface u is defined as:

$$|u(\mathbf{x})| = \min_{q} ||\mathcal{C}(q) - \mathbf{x}|| \tag{5.3}$$

There are many nice properties of the distance map worth noting. First, the gradient magnitude of the surface is constant across the image and equal to one ($|\nabla u| = 1$). The direction of the gradient is equal to the outward normal of the nearest point on the curve C. Given the surface u, the nearest point on the curve from any point $\mathbf{x}$ in space can be computed by $\mathbf{x} - u(\mathbf{x})\,\nabla u(\mathbf{x})$. The distance map is smooth and provides the propagation of the boundary information without loss of fidelity. The redundancy of information over a region in space also offers stability in many types of computation. The only region of the distance map that can be unstable is on the *skeleton*. The skeleton is a set of points in space that are equidistant from two or more different closest points on $\mathcal{C}$. On the skeleton, the gradient is undefined.

Exactly computing the distance map can be quite an expensive operation. For a curve in 2D represented by a spline with m nodes, and $N \times N$ pixel image, the naïve approach is to compute the distance from each pixel to all $m + 1$ spline segments, and take the minimum, which is $O(mN^2)$ time. There are various approximation algorithms that are typically used to speed up the computation.

An estimation of the distance map can be computed using bi-directional propagation. The image is initialized with 0 for pixels on the boundary and ∞ everywhere else. A mask (such as the one in Figure 5-6) is swept over the image in the forward direction. The value of that pixel is replaced by the minimum of its value and the



Forward Mask:

$+\sqrt{2}$	+1	$+\sqrt{2}$
+1	+0	

Reverse Mask:

	+0	+1
$+\sqrt{2}$	+1	$+\sqrt{2}$

∞	∞	∞	∞
∞	∞	∞	∞
∞	0	0	∞
∞	∞	∞	∞

∞	∞	∞	∞
∞	∞	∞	∞
∞	0	0	1
∞	∞	∞	∞

∞	∞	∞	∞
∞	∞	∞	∞
∞	0	0	1
$\sqrt{2}$	∞	∞	∞

∞	∞	∞	∞
∞	∞	∞	∞
∞	0	0	1
$\sqrt{2}$	1	1	∞

∞	∞	∞	∞
∞	∞	∞	∞
∞	0	0	1
$\sqrt{2}$	1	1	$\sqrt{2}$

∞	∞	∞	∞
∞	∞	∞	∞
1	0	0	1
$\sqrt{2}$	1	1	$\sqrt{2}$

∞	∞	∞	∞
$\sqrt{2}$	1	1	$\sqrt{2}$
1	0	0	1
$\sqrt{2}$	1	1	$\sqrt{2}$

∞	∞	∞	$1+\sqrt{2}$
$\sqrt{2}$	1	1	$\sqrt{2}$
1	0	0	1
$\sqrt{2}$	1	1	$\sqrt{2}$

∞	2	2	$1+\sqrt{2}$
$\sqrt{2}$	1	1	$\sqrt{2}$
1	0	0	1
$\sqrt{2}$	1	1	$\sqrt{2}$

$1+\sqrt{2}$	2	2	$1+\sqrt{2}$
$\sqrt{2}$	1	1	$\sqrt{2}$
1	0	0	1
$\sqrt{2}$	1	1	$\sqrt{2}$

Figure 5-6: Bi-directional Chamfer algorithm for computing distance maps. The forward and reverse masks are shown at the top. The first block in the first row shows the initialization. The remaining blocks show the result of the forward pass. The bottom row shows the result of the reverse pass.

sums of the mask values with the neighboring pixels:

$$u(x,y) \leftarrow \min_{-1 \leq i,j \leq 1} m(i,j) + u(x+i, y+j) \tag{5.4}$$

where u is the distance map and m is the mask. In this pass, all the information is propagated from the upper-left corner of the image to the lower-right. In the second pass, the mask is reversed, and so is the direction of the sweep (lower-left to upper-right). Various masks can be used, such as the Chamfer distance (used in Figure 5-6) or the Manhattan distance (without the diagonal elements).

While this method is efficient and independent of the complexity of the object, it is prone to inaccuracies. For example, note that the value in the upper-left and upper-right corners of the final frame in Figure 5-6 is $1 + \sqrt{2} \approx 2.414$, while the true distance is $\sqrt{3} \approx 1.732$, which is a relatively large error. The algorithm can be made more accurate by expanding the mask to 5×5 or larger, which, of course increases the computation time. Another trick to increase the accuracy is to store at each pixel both the distance to the nearest point and the coordinates of the nearest point itself (*i.e.* both the min and the argmin). The distance between the coordinates of the current point and the (estimated) nearest point is computed instead of just adding the value in the mask to the neighbor's distance.

Often times the entire distance map need not be computed. Using narrow band level set techniques [93], only the distance values within a distance of around 5 pixels are necessary, but need to be very accurate. The bidirectional propagation approach fills the entire space with an approximation, so other techniques are necessary. One heuristic involves propagation of the distance information via a heap. All the pixels with zero distance are set to zero, and all their neighbors are placed into a heap, noting the closest point. The top (closest) element is taken from the heap and its distance from the neighbor's closest point is computed. Its neighbors are then placed into the heap. There are some subtleties with this approach such as when to stop propagating (when the front meets itself at the skeleton).

The distance map of a curve can also be computed using level set techniques. Consider the situation where the original curve $\mathcal{C}$ is evolved with unit speed in the normal direction $\mathcal{N}$:

$$\mathcal{C}_t = \mathcal{N} \tag{5.5}$$

In this scenario, the time at which the front crosses each point is the distance of that point to the original front. This computation must be done both in the positive and negative directions to compute the distance both inside and outside the curve. This process can also be implemented in the level set framework, as described in the next section. We now use the distance map as the representation upon which we build the shape model.

5.1.3 Building the Training Set

All the curves in the training dataset are rigidly aligned to one coherent coordinate frame to account for any gross offsets in translation or rotation. Note that since the structures in the training set can be quite different, rigid registration is not sufficient

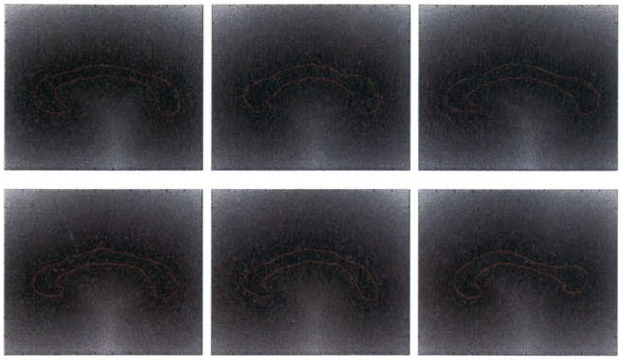


Figure 5-7: Outlines of the corpus callosum of 6 out of 51 patients in the training set embedded as the zero level set of a higher dimensional surface, the signed distance function.

to achieve perfect correspondence. However, the goal is to develop a representation robust to slight misalignments, so the registrations need not be perfect. We have used alignment methods such as the Hausdorff distance [47], iterative closest point (ICP) [4], and first and second moments [44]. For the structures we have modeled, all of these methods perform well for this rough alignment of the curves.

Each aligned curve is then embedded as the zero level set of a higher dimensional surface, u, (the signed distance map) whose height is sampled at regular intervals (say N^d samples, where d is the number of dimensions). Each such sampled surface (distance map) can be considered a point in a high dimensional space ($u \in \Re^{N^d}$). The training set, $\mathcal{T}$, consists of a set of surfaces $\mathcal{T} = \{u_1, u_2, \ldots, u_n\}$. Our goal is to build a shape model over this distribution of surfaces. Since a signed distance map is uniquely determined from the zero level set, each distance map has a large amount of redundancy. Furthermore, the collection of curves in the training set presumably has some dependence, as they are shapes of the same class of object, introducing more redundancy in the training set. The cloud of points corresponding to the training set is approximated to have a Gaussian distribution, where most of the dimensions of the Gaussian collapse, leaving the principal modes of shape variation.

As described in Chapter 3, we use Principal Component Analysis to derive the shape model. The mean surface, μ, is computed by taking the mean of the signed distance functions, $\mu = \frac{1}{n} \sum u_i$. The matrix of eigen-vectors U along with the diagonal

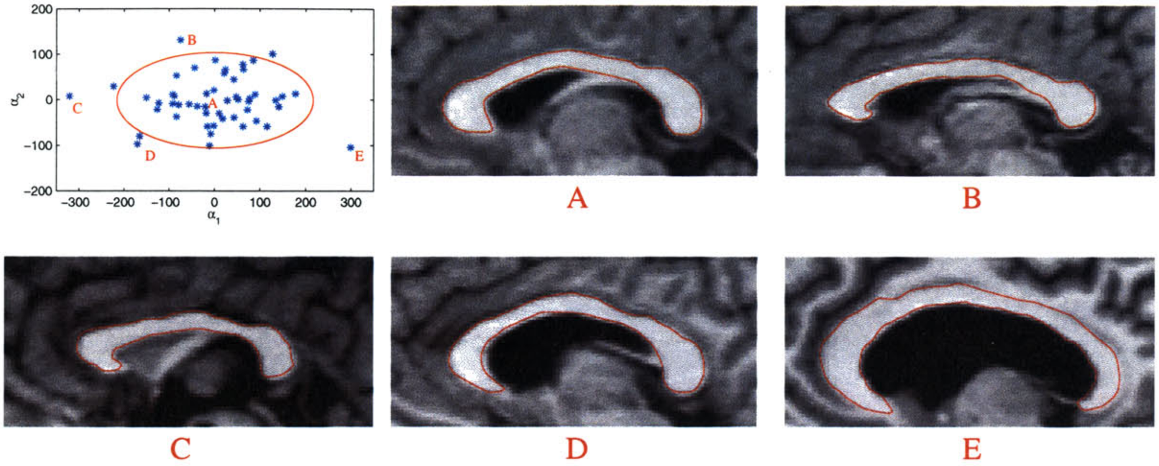


Figure 5-8: The plot in the upper left shows the 51 corpora callosa training shapes projected into a 2D eigenspace. The red contour is the 2σ variance ellipse. The α_1 axis roughly corresponds to size, and α_2 to concavity. The five corpora shown are labeled on the plot. Shape (A) is very close to the mean. Notice that (C) is an outlier in size, and (E) is an outlier in both size and concavity.

matrix of corresponding eigen-values Σ is computed using from the co-variance matrix using SVD.

An estimate of a novel shape, u, of the same class of object can be represented by k principal components in a k-dimensional vector of coefficients, α.

$$\alpha = U_k^{\mathsf{T}}(u - \mu) \tag{5.6}$$

where U_k is a matrix consisting of the first k columns of U that is used to project a surface into the eigen-space. Given the coefficients α, an estimate of the shape u, namely $\tilde{u}$, is reconstructed from U_k and μ.

$$\tilde{u} = U_k\,\alpha + \mu \tag{5.7}$$

Note that in general $\tilde{u}$ will not be a true distance function, since convex linear combinations of distance maps do not produce distance maps. However, the surfaces generally still have advantageous properties of smoothness, local dependence, and zero level sets consistent with the combination of original curves.

Under the assumption of a Gaussian distribution of shape represented by α, we

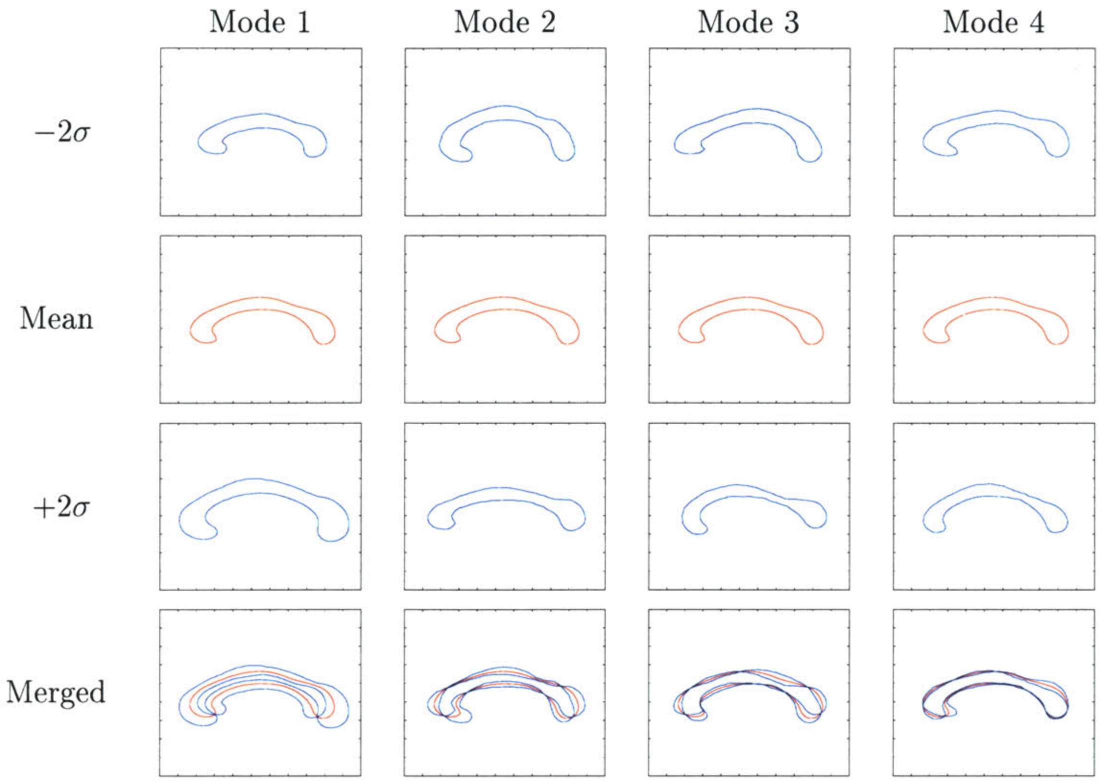


Figure 5-9: The four primary modes of variance of the corpus callosum training dataset.

can compute the probability of a certain curve as:

$$P(\alpha) = \frac{1}{\sqrt{(2\pi)^k|\Sigma_k|}} \exp\left(-\frac{1}{2}\alpha^\top \Sigma_k^{-1}\alpha\right) \tag{5.8}$$

where Σ_k contains the first k rows and columns of Σ.

Figure 5-7 shows a few of the 51 training curves used to define the shape models of the corpus callosum. The original segmentations of the images are shown as red curves. The curves are overlaid on the signed-distance map. Figure 5-9 illustrates zero level sets corresponding to the means and three primary modes of variance of the shape distribution of the corpus callosum. Figure 5-8 shows the projection of the training elements onto the two principal directions, and illustrates some of the outliers. Figure 5-13 shows the zero level set (as a triangle surface model) of seven rigidly aligned vertebrae of one patient used as training data. The zero level sets of

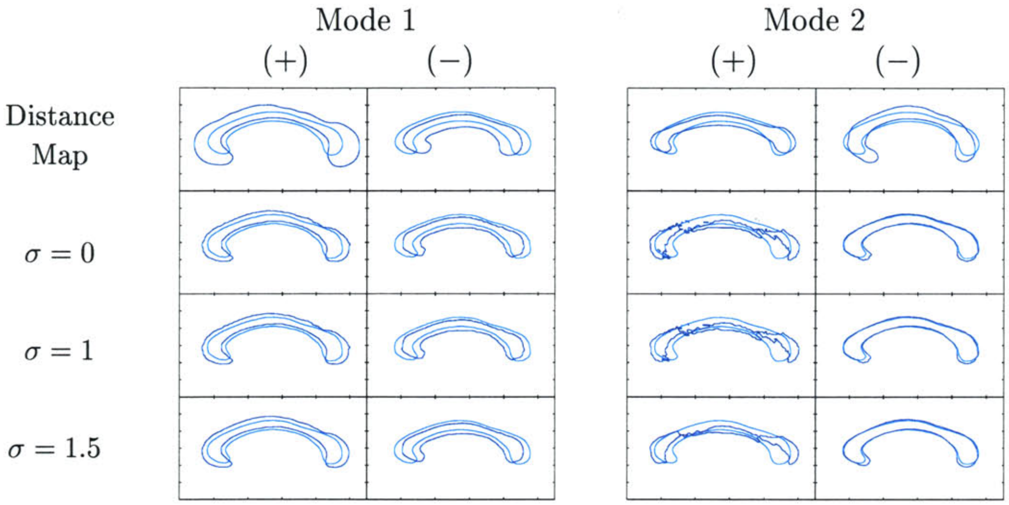


Figure 5-10: The first and second mode of shape variance are shown for the distance map, as well as for a binary image, smoothed with Gaussian masks of different σ. Notice that the distance map representation captures shape variance much better than the others, due to the robustness to mismatch of correspondence.

the two primary modes are shown as well. Note that for both the corpus and the vertebrae, the mean shapes and primary modes appear to be reasonable representative shapes of the classes of objects being learned. In the case of the corpus callosum, the first mode seems to capture size, while the second mode roughly captures the degree of curvature of the corpus. The third mode appears to represent the shifting of the bulk of the corpus from front to back.

Another approach to correspondence is to roughly align the training data before performing the comparison and variance calculation. A rough alignment will not match every part of each training instance perfectly, so one must consider the robustness of the representation to misalignment. Turk and Pentland [99] introduced Eigenfaces as a method of building models for recognition. Each image in a set of face images ($N \times N$ array of intensities) is considered as a point in an N^2-dimensional space, from which the principal components are computed. The Eigenface method is similar to our method of combining signed distance maps of binary images, with an important distinction. Any slight misalignment in the faces compares the intensity variance between independent objects, while slightly misaligned pixels in a distance

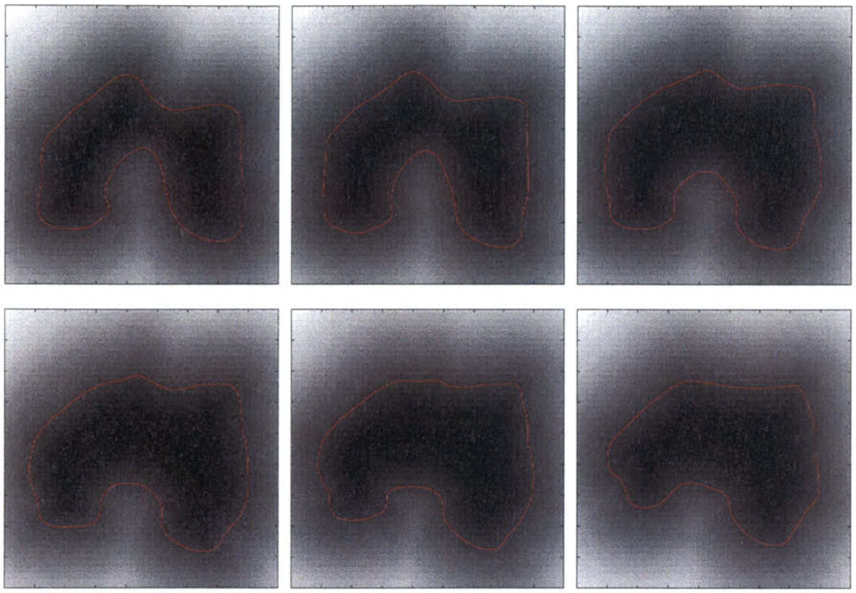


Figure 5-11: Curves segmented from six slices of a patient's femur. The curves are embedded as level-sets of a higher dimensional geodesic surface.

map are generally very highly correlated. Smoothing a grayscale or binary image propagates information spatially as well, increasing the correlation between neighboring pixels, but results in loss of information, whereas no information about the binary image is lost by computing its signed distance function.

Using the signed distance map as the representation of shape provides tolerance to slight misalignment of object features, in the attempt to avoid having to solve the general correspondence problem. In the examples presented here, the rough rigid alignment of the training instances resulted in the model capturing the shape variances inherent in the population due to the dependence of nearby pixels in the shape representation. We are exploring the tradeoffs between leaving certain transformation parameters intrinsic in the representation, or extrinsic by aligning the training set under those classes of transformation. Currently, rigid pose is extrinsic, but scale and affine warp are left as intrinsic. We are also extending this method to use landmark correspondences (when they are available) to ensure the comparison of "like" pixels instead of "near" pixels using flow fields as in [49].

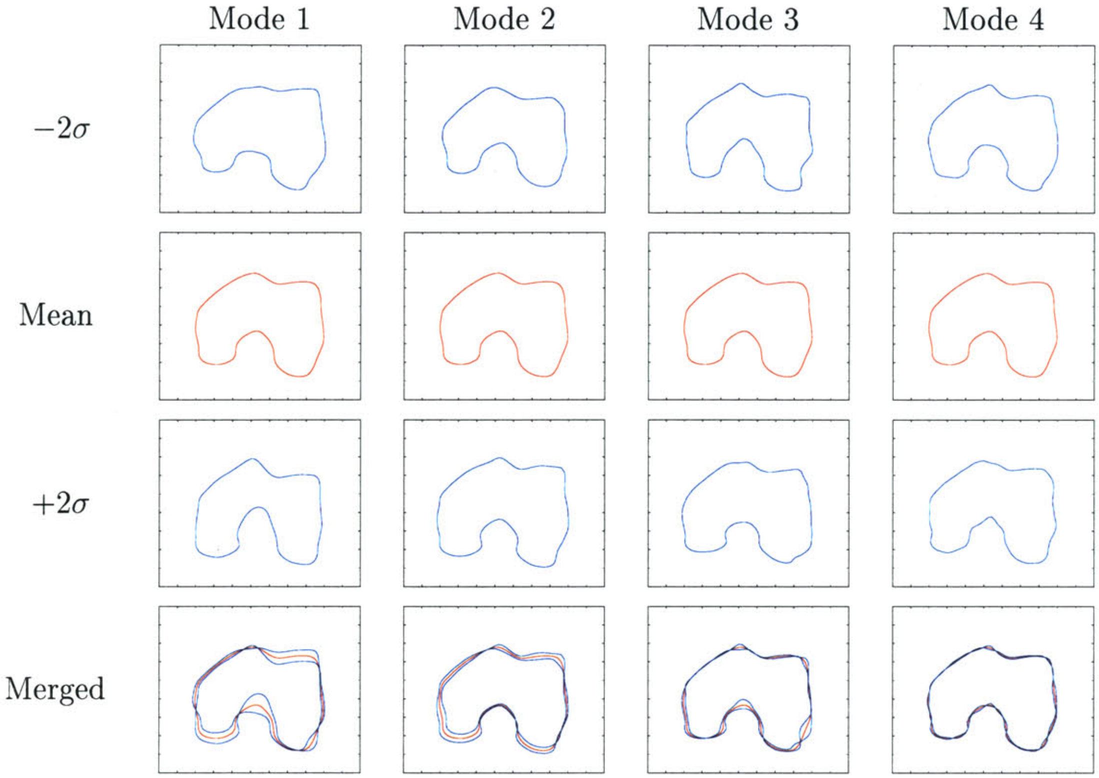


Figure 5-12: The four primary modes of variance of the slices of the femur.

5.2 Geodesic Active Contours

Once the shape model has been derived, a method is required to actually perform the segmentation. In this section, we discuss the level set or geodesic active contour boundary localization method, which will be adapted in the next section to incorporate the shape prior.

In order to derive the level set method of segmentation [9, 54, 75] we first explore examples of curvature flow. Let $\mathcal{C}(q)$ be a simple, closed curve in the plane parameterized by arclength. The total arclength about $\mathcal{C}$ is given by

$$\int |\mathcal{C}'(q)|\, dq \tag{5.9}$$

Assume that we wish to find the curve $\mathcal{C}$ that minimizes the Euclidean arclength above (even though in this simple example, the minimal such curve is a point). The

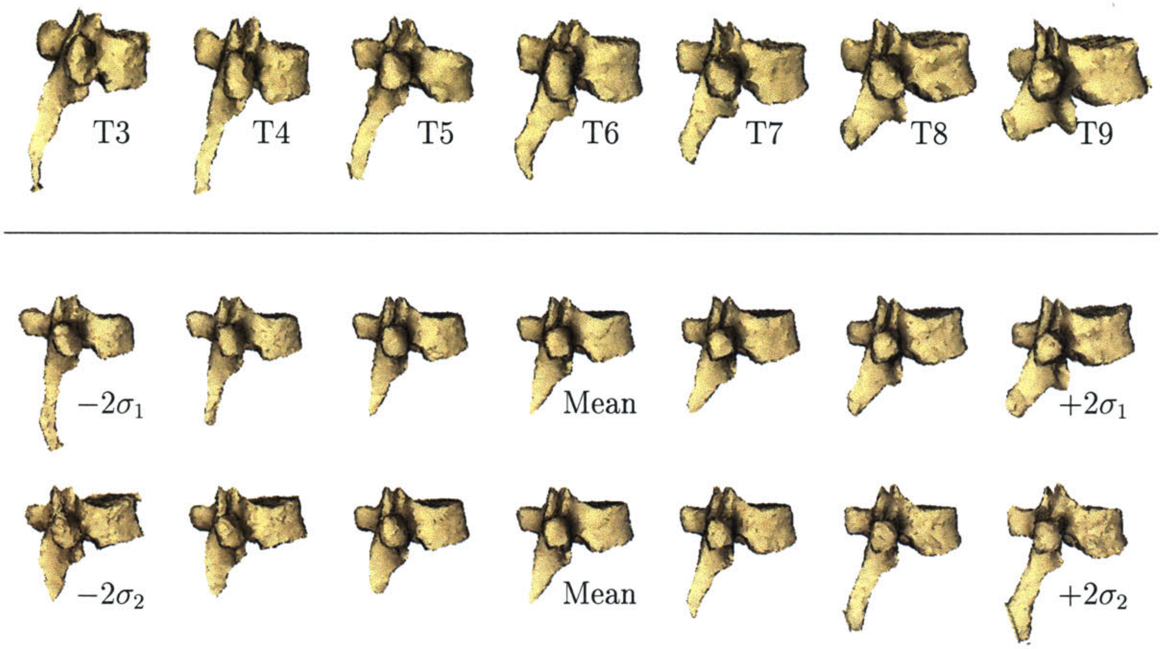


Figure 5-13: **Top:** Three-dimensional models of seven thoracic vertebrae (T3-T9) used as training data. **Bottom left and right**: Extracted zero level set of first and second largest mode of variation respectively.

Euler-Lagrange equation states that in order to minimize an equation of the form

$$\min_{\mathcal{C}(q)} \int \Phi \, dq \tag{5.10}$$

via gradient descent, the curve $\mathcal{C}$ is evolved over time based on the following equation:

$$\mathcal{C}_t = \Phi_{\mathcal{C}} - \frac{\partial}{\partial q} \Phi_{\mathcal{C}'} \tag{5.11}$$

In the example above, Φ is simply $|\mathcal{C}'(q)|$, which yields the following expression to update $\mathcal{C}$ over time:

$$\begin{aligned}
\mathcal{C}_t &= \frac{\partial}{\partial C} |\mathcal{C}'(q)| - \frac{\partial}{\partial q} \frac{\partial}{\partial \mathcal{C}'} |\mathcal{C}'(q)| && (5.12) \\
&= 0 - \frac{\partial}{\partial q} \frac{\mathcal{C}'(q)}{|\mathcal{C}'(q)|} && (5.13) \\
&= -\frac{\partial \mathbf{t}}{\partial q} && (5.14) \\
&= -\kappa \mathcal{N} && (5.15)
\end{aligned}$$

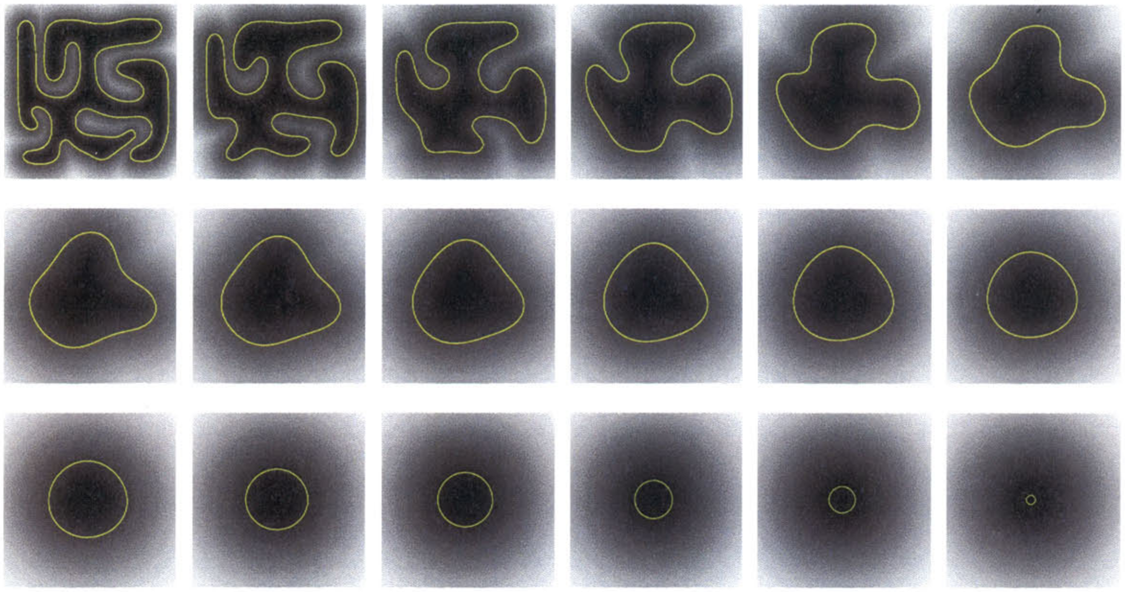


Figure 5-14: A front propagating with speed proportional to its curvature. By Grayson's Theorem [38], the curve will collapse smoothly to a circle and a point.

where $\mathbf{t}$ is the tangent to the curve, κ is the curvature, and $\mathcal{N}$ is the normal. Thus, evolving a curve with speed equal to the curvature will minimize the Euclidean arclength, shrinking the curve down to a point. This flow is also called the *Euclidean heat flow* or the *Euclidean curvature flow*[9]. Figure 5-14 illustrates this process on an example curve.

Instead of explicitly representing the curve by the parameterization $\mathcal{C}(q)$, we again choose an implicit representation of the zero level set of the signed distance map, u (Figure 5-5). When propagating the curve front at a speed β, the following equivalence holds between the Laplacian and Eulerian curve representations [8, 54, 75]:

$$\mathcal{C}_t \;=\; \beta\,\mathcal{N} \qquad \Longleftrightarrow \qquad u_t \;=\; \beta\,|\nabla u| \tag{5.16}$$

Therefore, in the previous example where a curve shrinks smoothly to a point, the entire distance surface u can be evolved by $u_t = \kappa|\nabla u|$. In this case, the zero level set of the surface will similarly shrink smoothly to a point. To implement this, we need an expression for the curvature κ of a level set as a function of the surface u.

The curvature, κ, is the divergence of the unit normal, which yields the following expression involving first and second partial derivatives of u:

$$\begin{aligned} \kappa &= \nabla \cdot \mathcal{N} && (5.17) \\ &= \nabla \cdot \frac{\nabla u}{|\nabla u|} && (5.18) \\ &= \frac{\partial}{\partial x} \frac{u_x}{(u_x^2 + u_y^2)^{\frac{1}{2}}} + \frac{\partial}{\partial y} \frac{u_y}{(u_x^2 + u_y^2)^{\frac{1}{2}}} && (5.19) \\ &= \frac{u_{xx}u_y^2 - 2u_x u_y u_{xy} + u_{yy}u_x^2}{(u_x^2 + u_y^2)^{\frac{3}{2}}} && (5.20) \end{aligned}$$

To use the idea of minimal curves to segment a boundary in an image, an energy function is derived that includes both an image gradient term and a curvature term, such as the classical snake energy function [52]:

$$\min_{\mathcal{C}(q)} \int |\mathcal{C}'(q)|^2 dq - \lambda \int |\nabla \mathcal{I}(\mathcal{C}(q))| dq \qquad (5.21)$$

Caselles and Kichenassamy simultaneously showed an equivalence between the above energy and a function of the form that can be solved using Euler-Lagrange [8, 54]:

$$\min_{\mathcal{C}(q)} \int g(|\nabla \mathcal{I}(\mathcal{C}(q))|)\ |\mathcal{C}'(q)|\ dq \qquad (5.22)$$

The first term in the integral is used to scale the arclength by a function g of the image gradient that is small when the gradient magnitude is large. In this non-Euclidean space, the arclength now depends on the image, and curves overlapping large gradients are "shorter". The function g is often of the form

$$g(|\nabla \mathcal{I}|) = \frac{1}{1 + |\nabla \mathcal{I}|^2} \qquad (5.23)$$

By again using Euler-Lagrange from Eq. 5.11, but now with the new Φ that includes g, we obtain the following expression to update the curve $\mathcal{C}$:

$$\mathcal{C}_t = g\kappa\mathcal{N} - (\nabla g \cdot \mathcal{N})\mathcal{N} \qquad (5.24)$$

The equivalence in Eq. 5.16 of evolving the curve C and the surface u results in the evolution equation for level set based segmentation:

$$u_t = g\,\kappa\,|\nabla u| + \nabla u \cdot \nabla g \tag{5.25}$$

Evolving the surface u by the above equation will find the local minimal curve based both on the image gradients and the curvature.

As described in Section 2.2.1, in many circumstances it is advantageous to initialize the curve *inside* the object to be segmented. However, without strong gradients nearby, the curvature term will cause the curve to shrink to a point. Therefore, an outward balloon force can be added to the update equation [9, 13]:

$$u_t = g\,(\nu + \kappa)\,|\nabla u| + \nabla u \cdot \nabla g \tag{5.26}$$

The balloon force can be constant throughout [9], or a function of the image gradient (such as $\nu = c\,g$ as proposed in [67]) which prevents a curve from ballooning past a boundary.

The Lagrangian representation implicitly represents the curve as the zero level set of an entire surface that is evolved to minimize a given energy function. Actually, only the region nearby the zero level set needs to be evolved, as surface points distant from boundary have no effect on the implicit curve. Therefore, for efficiency reasons, generally only a *narrow band* of the surface is updated at every step [75]. Care must be taken to assure that the band is updated so that the zero level set never leaves the valid region. Narrow banding often results in a considerable computational savings, as it effectively reduces the dimensionality of the problem back to that of the underlying curve instead of the hyperspace.

The general framework of level set segmentation described here has been extended in many ways, such as alternative regularization [68], texture models [80], global intensity statistics [113], and pedal curve evolution [40]. Further descriptions of some of these extensions can be found in Section 2.2.1.

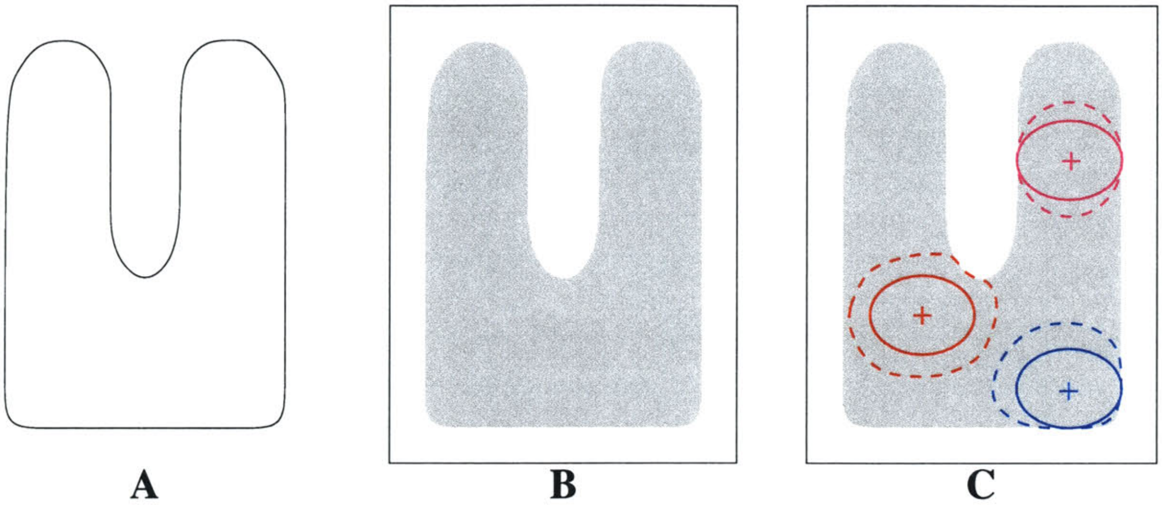


Figure 5-15: (a) The curve expected to be found in the image. (b) The image containing the shape to be segmented. (c) The same solid curve appears in three different locations. Given that the target shape is known, the direction of evolution depends on the position of the evolving curve with respect to the object. The dotted lines show a later step in evolution given the curve's position and target shape.

5.3 Shape Priors and Geodesic Active Contours

Our first extension to level set segmentation involves incorporating the shape models presented in Section 5.1 into the level set evolution process. Recall in that section, we represent a curve as the signed distance map u, and perform PCA on a set of training curves, resulting in an ability to approximate a shape by a k-dimensional vector of eigen-coefficients, α. Given the eigen-representation and a probability distribution over coefficients α, the prior shape information can be folded into the segmentation process. This section describes adding a term to the level set evolution equation to pull the surface in the direction of the maximum *a posteriori* shape and position of the final segmentation.

5.3.1 Estimation of Pose and Shape

In addition to evolving the level set based on the curvature and the image term, we include a term that incorporates information about the shape of the object being segmented. To add such a global shape force to the evolution, the pose of the evolving curve with respect to the shape model must be known (see Figures 5-15 and 5-16).

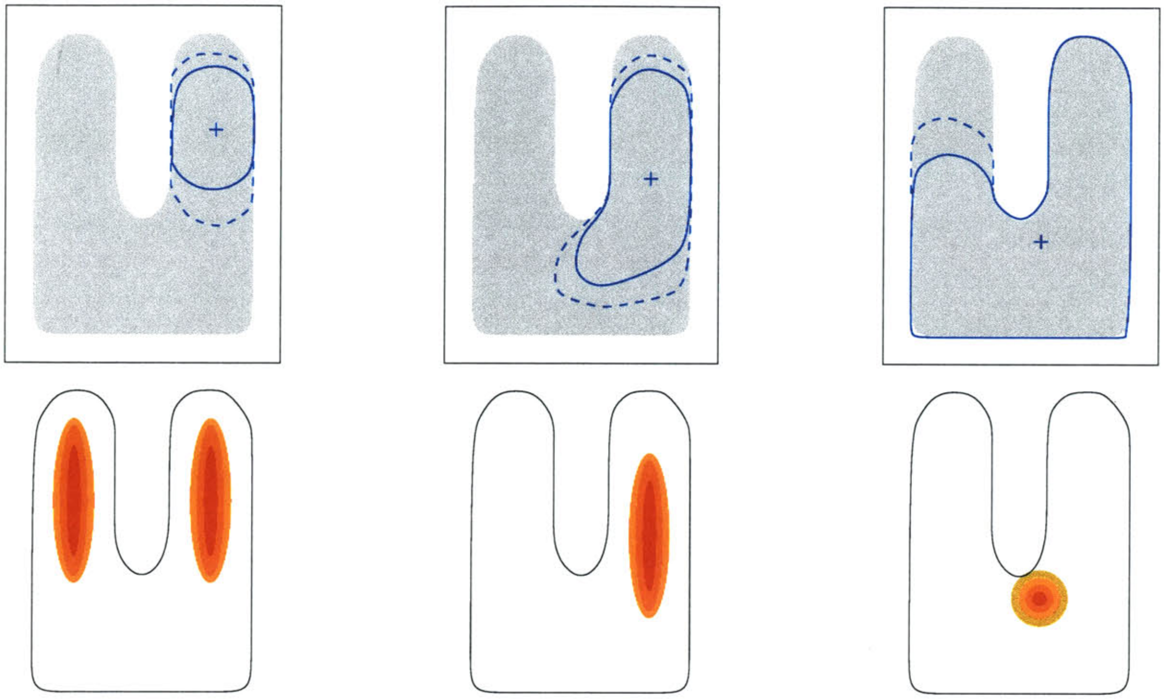


Figure 5-16: Three steps in the evolution process. The evolving curve is shown in solid blue superimposed on the image (top row). The curve is matched to the expected curve to obtain a pdf over pose (bottom row). The next evolution step (based on pose and shape) is shown as the dotted blue line.

Without an estimate of the pose, the shape model cannot adequately constrain or direct the evolution. Therefore, at each step of the curve evolution, we seek to estimate the shape parameters, α, and the rigid pose parameters, p, of the *final* curve using a maximum *a posteriori* approach.

$$\langle \alpha_{\text{MAP}}, p_{\text{MAP}} \rangle = \operatorname*{argmax}_{\alpha,p} P(\alpha, p \mid u, \nabla I) \tag{5.27}$$

In this equation, u is the evolving surface at some point in time, whose zero level set is the curve that is segmenting the object. The term ∇I is the gradient of the image containing the object to be segmented. By our definition of shape and pose, the final segmentation curve is completely determined by α and p. Let u^* be the estimated final curve, which can be computed from α and p. Therefore, we also have

$$u^*_{\text{MAP}} = \operatorname*{argmax}_{u^*} P(u^* \mid u, \nabla I) \tag{5.28}$$

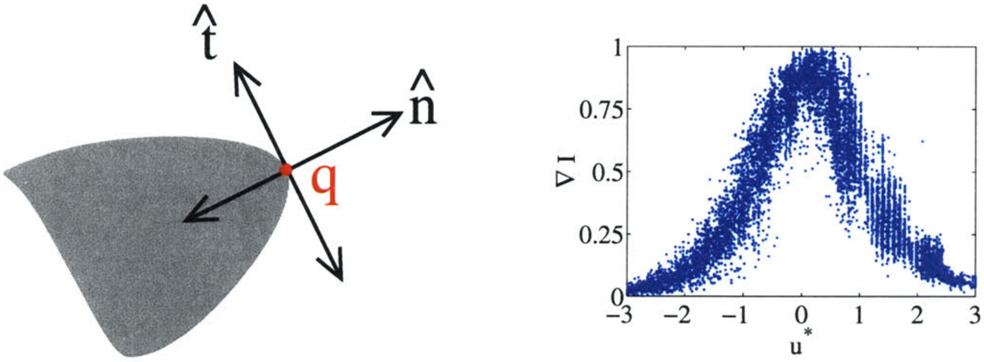


Figure 5-17: The relationship between the distance map and the image gradient.

To compute the maximum *a posteriori* final curve, we expand the terms from Eq. 5.27 using Bayes' Rule.

$$\begin{aligned} P(\alpha, p \mid u, \nabla I) &= \frac{P(u, \nabla I \mid \alpha, p) P(\alpha, p)}{P(u, \nabla I)} \qquad (5.29) \\ &= \frac{P(u \mid \alpha, p) P(\nabla I \mid \alpha, p, u) P(\alpha) P(p)}{P(u, \nabla I)} \end{aligned}$$

Note that the preceding step assumes that shape is independent from pose. Since our current model does not attempt to capture the relationships between these two quantities, this is reasonable. Future work may incorporate positional priors (and relative positional priors between objects) into our shape model. We proceed by defining each term of Eq. 5.29 in turn. We discard the normalization term in the denominator as it does not depend on shape or pose.

Inside Term. The first term in Eq. 5.29 computes the probability of a certain evolving curve, u, given the shape and pose of the final curve, u^* (or $\langle \alpha, p \rangle$). Notice that this term does not include any image information whatsoever. Given our method of initializing the curve with a point *inside* the object, it is reasonable to assume that the curve should remain inside the object throughout the evolution. Therefore, if the evolving curve lies completely inside the final curve, then it is more likely than a curve that lies partially or fully outside the final curve. We model this term as a Laplacian density function over $V_{outside}$, the volume of the curve u that lies outside the curve u^*.

$$P(u \mid \alpha, p) = \exp\left(-V_{outside}\right) \tag{5.30}$$

This term assumes that any curve u lying inside u^* is equally likely. Since the initial curve can be located at any point inside the object and the curve can evolve along any path, we do not favor any such curve.

Gradient Term. The second term in Eq. 5.29 computes the probability of seeing certain image gradients given the current and final curves. Consider the relationship between u^* and $|\nabla I|$ when u^* correctly outlines the boundary of the object (see Figure 5-17). Notice that the distance map u^* is linear along the normal direction of the curve at any boundary point, q, and $u^*(q) = 0$. Furthermore, under the assumption that the object boundary is a smoothed step edge, $|\nabla I|$ approximates a Gaussian along the normal at q. Therefore, we'd expect the relationship between $|\nabla I|$ and u^* to be Gaussian in nature. Figure 5-17 shows an example scatter plot of these quantities when u^* is aligned with the object boundary. Let $h(u^*)$ be the best fit Gaussian to the samples $(u^*, |\nabla I|)$. We model the gradient probability term as a Laplacian of the goodness of fit of the Gaussian.

$$P(\nabla I \mid u^*, u) = \exp\left(-\mid h(u^*) - |\nabla I| \mid^2\right) \tag{5.31}$$

Shape and Pose Priors. The last two terms in Eq. 5.29 are based on our prior models, as described in Section 5.1.3. Our shape prior is a Gaussian model over the shape parameters, α, with shape variance Σ_k.

$$P(\alpha) = \frac{1}{\sqrt{(2\pi)^k |\Sigma_k|}} \exp\left(-\frac{1}{2}\alpha^{\mathsf{T}} \Sigma_k^{-1} \alpha\right) \tag{5.32}$$

In our current framework, we seek to segment one object from an image, and do not retain prior information on the *a posteriori* of the object appearing in a certain

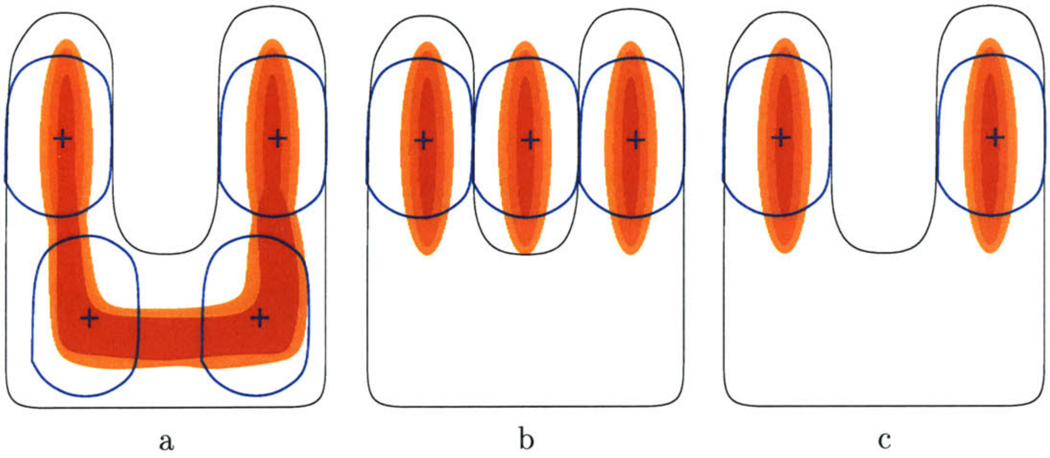


Figure 5-18: An illustration of the need for both the *inside* and *image* terms. (a) If only the inside term is used, then any pose inside the object would be equally likely. (b) If only the gradient term is used, the curve could fall outside the object. (c) When both terms are combined, the PDF over pose provides a better result.

location. Thus, we simply assume a uniform distribution over pose parameters, which can include any type of transformation, depending on application.

$$P(p) = \mathcal{U}(-\infty, \infty) \tag{5.33}$$

Currently we are modeling translation and rotation. We feel, however, that positional priors could provide a rich source of information to explore in the future, especially when segmenting multiple objects from a single image that may have clear prior relative poses, or when a distribution over pose in a fixed image-based coordinate system is known.

These terms define the maximum *a posteriori* estimator of shape and pose, which estimates the final curve or segmentation of the object. For efficiency, these quantities are computed only in a narrow band around the zero level set of the evolving surface, and the MAP pose and shape are re-estimated at each evolution step using simple gradient ascent on the log probability function in Eq. 5.29. While each ascent may yield a local maximum, the continuous re-estimation of these parameters as the surface evolves generally results in convergence on the desired maximum. Next, we

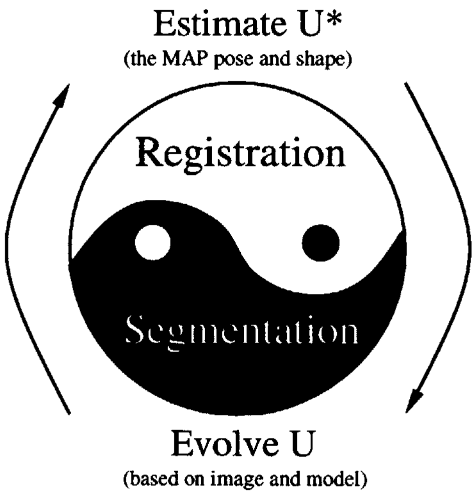


Figure 5-19: Diagram of the Registration / Segmentation "closed loop" of the algorithm. At each step, a rough estimate of the final curve and shape is computed (u^*), and then the evolving surface u is updated based on local image information and the global shape and pose.

incorporate this information into the update equation commonly used in level set segmentation.

5.3.2 Evolving the Surface

Initially, the surface, u, is assumed to be defined by at least one point that lies inside the object to be segmented. Given the surface at time t, we seek to compute an evolution step that brings the curve closer to the correct final segmentation based on local gradient and global shape information (see Figure 5-19).

The level set update expression shown in Eq. 2.12 provides a means of evolving the surface u over time towards the solution to the original curve-minimization problem stated in Eq. 2.10. Therefore, the shape of the surface at time t+1 can be computed from $u(t)$ by:

$$u(t+1) = u(t) + \lambda_1 \Big(g\,(c+\kappa)\,|\nabla u(t)| + \nabla u(t) \cdot \nabla g \Big) \tag{5.34}$$

where λ_1 is a parameter defining the update step size.

By estimating the final surface u^* at a given time t, (Section 5.3.1), we can also

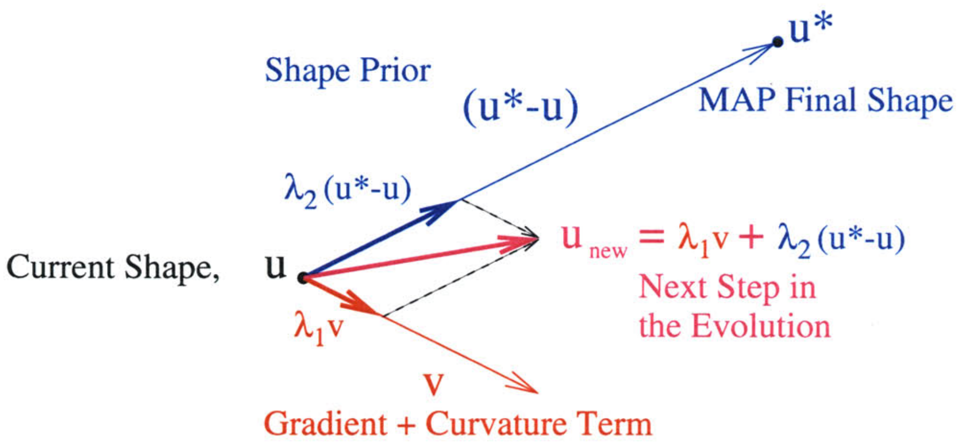


Figure 5-20: Illustration of the various terms in the evolution of the surface, u. The surface u^*, is the maximum *a posteriori* (MAP) final shape. To update u, we combine the standard gradient and curvature update term, v, and the direction of the MAP final shape, $u^* - u$.

evolve the surface in the direction of the maximum *a posteriori* final surface:

$$u(t+1) = u(t) + \lambda_2 \left(u^*(t) - u(t)\right) \tag{5.35}$$

where $\lambda_2 \in [0, 1]$ is the linear coefficient that determines how much to trust the maximum *a posteriori* estimate. Combining these equations yields the final expression for computing the surface at the next step.

$$\begin{aligned} u(t+1) = u(t) &+ \lambda_1 \left(g\,(c+\kappa)\,|\nabla u(t)| + \nabla u(t) \cdot \nabla g\right) \\ &+ \lambda_2 \left(u^*(t) - u(t)\right) \end{aligned} \tag{5.36}$$

Figure 5-20 illustrates this evolution. The two parameters λ_1 and λ_2 are used to balance the influence of the shape model and the gradient-curvature model. The parameters also determine the overall step size of the evolution. The tradeoff between shape and image depends on how much faith one has in the shape model and the imagery for a given application. Currently, we set these parameters empirically for a particular segmentation task, given the general image quality and shape properties.

The original evolution equation (Eq. 5.34), to which we added the shape influence term, was derived from an energy minimization expression (Eq. 2.10). We are cur-

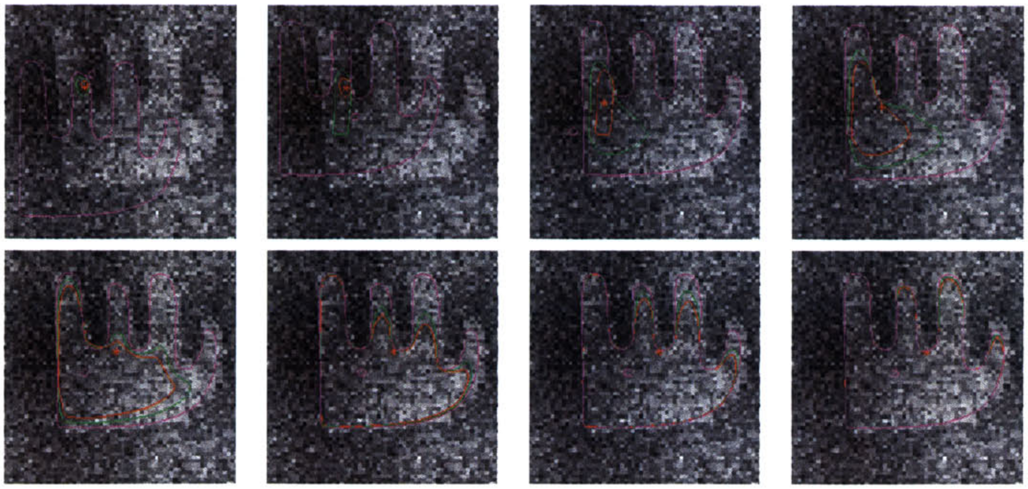


Figure 5-21: Several time steps in the curve evolution process of segmenting the hand. The red curve is the zero level-set of the evolving surface. The green curve is the next step in the curve evolution. The magenta curve is the maximum *a posteriori* estimate of the position of the final curve.

rently exploring ways of adding a "shape energy" term to the curve integral in Eq. 2.10, and then deriving the complete evolution equation using Euler-Lagrange, instead of adding the shape force after the fact. In this new framework, both processes will be fused into a single energy minimization. One difficulty is that the curve integral is inherently local and does not require any notion of correspondence, whereas global shape information involves the comparison of a shape to a model, in correspondence.

We have tested the segmentation algorithm on synthetic and real shapes, both in 2D and in 3D. For controlled testing, a training set of rhombi of various sizes and aspect ratios was generated to define a shape model. Test images were constructed by embedding the shapes of two random rhombi with the addition of Gaussian speckle noise and a low frequency diagonal bias field. Figure 5-22 illustrates several steps in the segmentation of the synthetic objects. In the first frame of each trial, the small red circle represents the initialization point. The yellow curve illustrates the MAP shape and pose at each time step. The final segmentations are shown in the last frames.

Segmentation experiments were performed on 2D slices of MR images of the femur and corpus callosum (Figures 5-23 and 5-24). For the femur experiments, the training

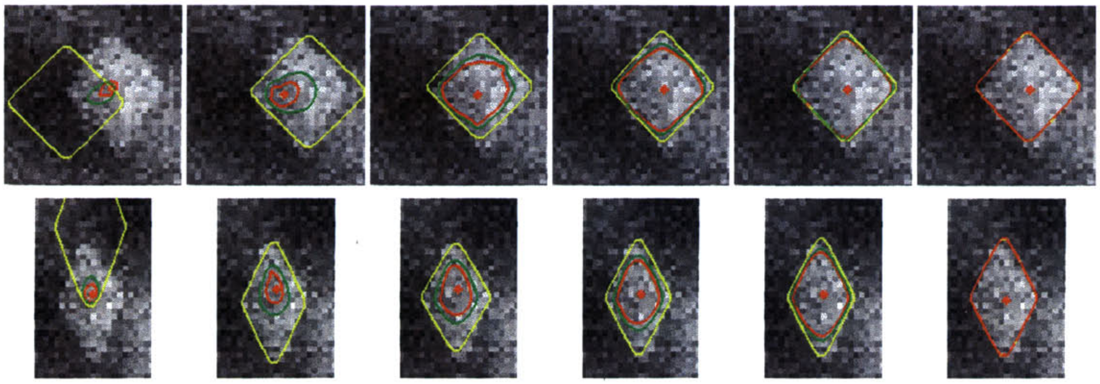


Figure 5-22: Several time steps in the curve evolution process of segmenting two rhombi. The training set for the rhombus consisted of rhombi of various sizes and aspect ratios. The red curve is the zero level set of the evolving surface. The green curve is the next step in the curve evolution. The yellow curve is the MAP estimate of the position and shape of the final curve.

set consisted of 18 nearby slices of the same femur, leaving out the slice being segmented and its neighbors. In both femur examples, the same initialization point was used to seed the evolution. As the curve evolves, the maximum *a posteriori* estimator of shape and pose locks into the shape of the femur slice.

The corpus callosum training set consisted of 49 examples like those in Figure 5-7. The segmentations of two corpora callosa are shown in Figure 5-24. Notice that while the MAP shape estimator is initially incorrect, as the curve evolves, the pose and shape parameters converge on the boundary. The segmentations of the femur slices and the corpora all converged in under a minute on a 550 MHz Pentium III.

The vertebrae example illustrates the extension of the algorithm to 3D datasets. Figure 5-25 illustrates a few steps in the segmentation of vertebra T7. The training set in this case consisted of vertebrae T3-T9, with the exception of T7. The initial surface was a small sphere placed in the body of the vertebra. The red contour is a slice through the zero level set of the evolving hyper-surface. The yellow overlay is the MAP pose and shape estimate. Segmenting the vertebra took approximately six minutes.

To validate the segmentation results, we compute the undirected partial Hausdorff distance [47] between the boundary of the computed segmentation and the boundary

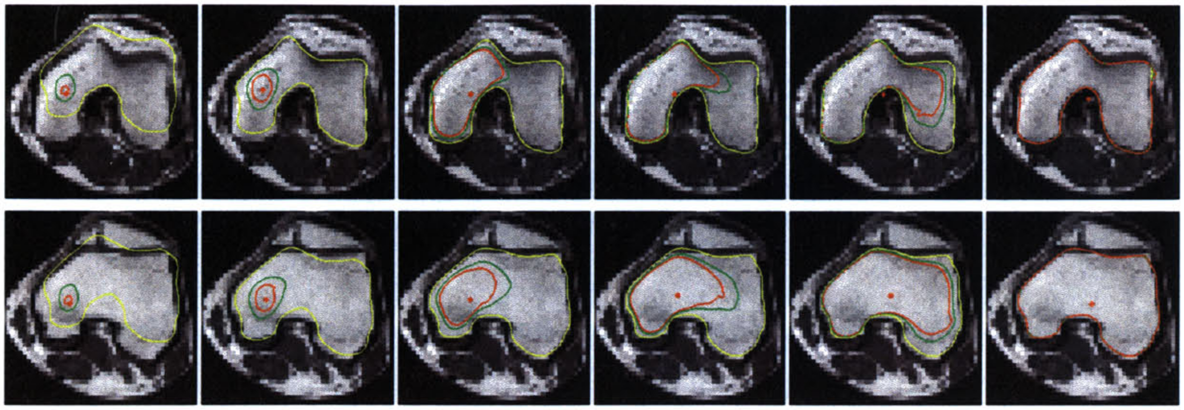


Figure 5-23: Initial, middle, and final steps in the evolution process of segmenting two slices of the femur. The training set consisted of 18 nearby slices of the same femur, leaving out the slice being segmented and its neighbors.

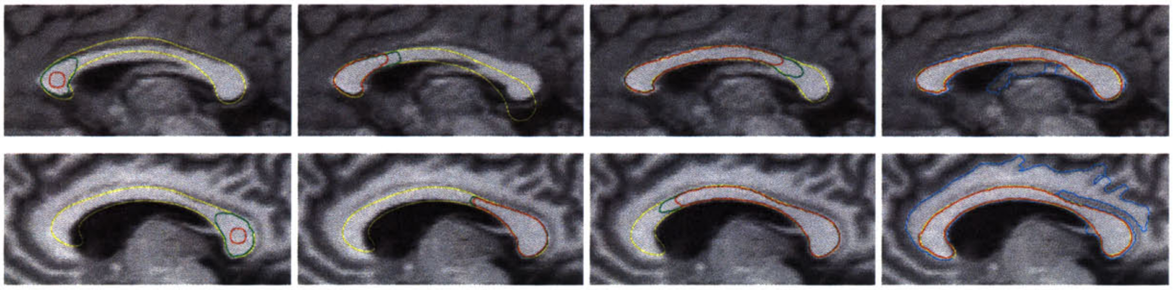


Figure 5-24: Four steps in the segmentation of two different corpora callosa. The last image in each case shows the final segmentation in red. The cyan contour is the result of the standard evolution *without* the shape influence.

of the manually-segmented ground truth. The directed partial Hausdorff distance over two point sets A and B is defined as

$$h_K(A,B) = \underset{a\in A}{\mathrm{K}^{\mathrm{th}}} \min_{b\in B} ||a-b|| \tag{5.37}$$

where K is a quantile of the maximum distance. The undirected partial Hausdorff distance is defined as

$$H_K(A,B) = \max(h_K(A,B), h_K(B,A)) \tag{5.38}$$

The results for the corpora and vertebra shown in Table 5.1 indicate that virtually

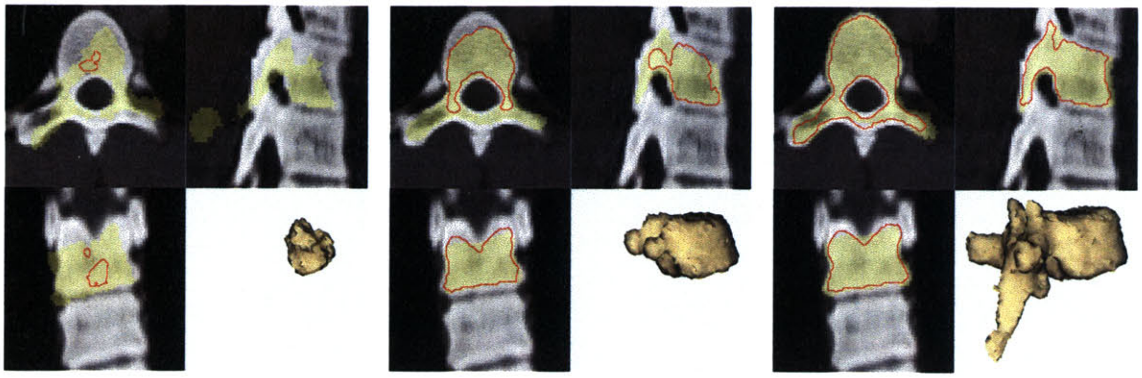


Figure 5-25: Early, middle, and final steps in the segmentation of the vertebra T7. Three orthogonal slices and the 3D reconstruction are shown for each step. The red contour is a slice through the evolving surface. The yellow overlay is a slice through the inside of the MAP final surface.

K	Corpus 1	Corpus 2	Vertebra
95%	1.3 mm	1.5 mm	2.7 mm
99%	1.6 mm	2.0 mm	4.4 mm

Table 5.1: Partial Hausdorff distance between our segmentation and the manually-segmented ground truth.

all the boundary points lie within one or two voxels of the manual segmentation.

5.4 The Inclusion of Intensity Models

The previous sections established a shape model that can be used in conjunction with the surface evolution to direct the segmentation. Intensity information has so far been included only as the classical heuristic of high gradients at the boundaries. Since our shape model already requires training data, we seek to establish a prior intensity model to further aid in segmentation.

Cootes, *et al.* extended their Active Shape Models [16] to Active Appearance Models [18] to strengthen the intensity model. Yet, recall that their approach requires point-wise correspondences throughout the training data, which is what we ideally prefer to avoid. The appearance models include intensity samples at *shape normalized* positions in the image. The shape normalization is computed from the

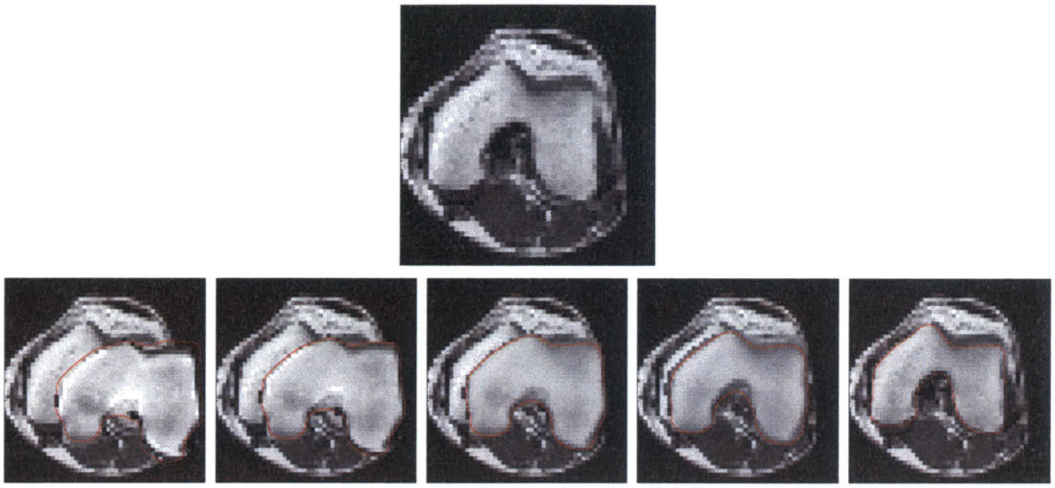


Figure 5-26: The top image is the target. The bottom five images illustrate steps in the minimization process where the combined shape and intensity model computes the best fit over the target image. Notice that the intensity values and the shape change as the model is fit.

correspondences.

Even without explicit point-wise correspondences and *a priori* shape normalization, we attempt to include grayscale information by adding the samples to the linear model. Originally, the i'th training example was represented by the distance map $\mathbf{u}_i$. In this framework, samples of the intensity image I_i are placed in a vector $\mathbf{v}_i$, and both are combined into one larger vector as our representation:

$$\mathbf{z}_i = \begin{bmatrix} \mathbf{u}_i \\ \mathbf{v}_i \end{bmatrix} \tag{5.39}$$

The Gaussian model is built as before (see Chapter 3), but now the vector eigencoefficients, $\boldsymbol{\alpha}$, represent *both* the shape and the grayscale intensities. If U_k is the matrix consisting of the first k eigenvectors, then the reconstructed information is computed from $\boldsymbol{\alpha}$ as follows:

$$\begin{bmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{v}} \end{bmatrix} = \tilde{\mathbf{x}} = U_k \boldsymbol{\alpha} + \boldsymbol{\mu} \tag{5.40}$$

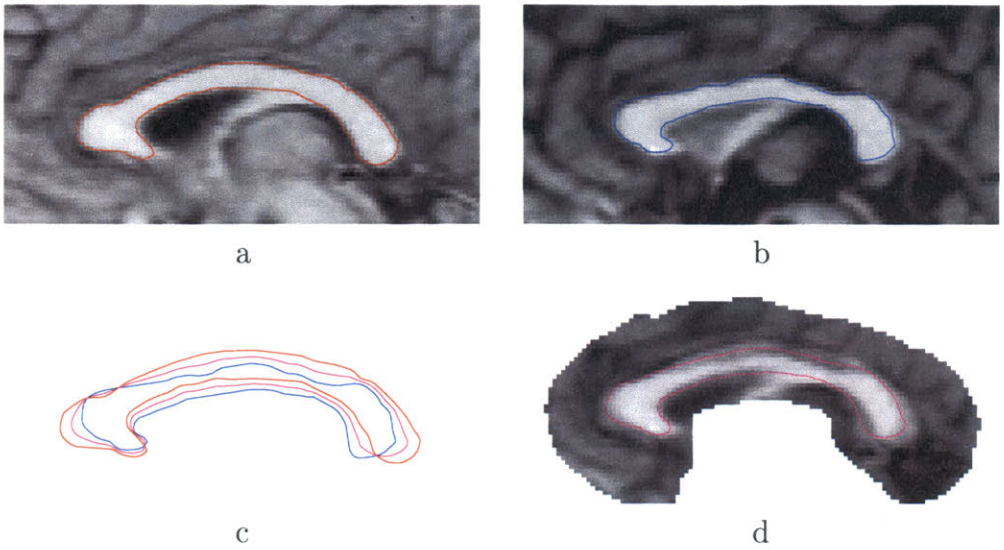
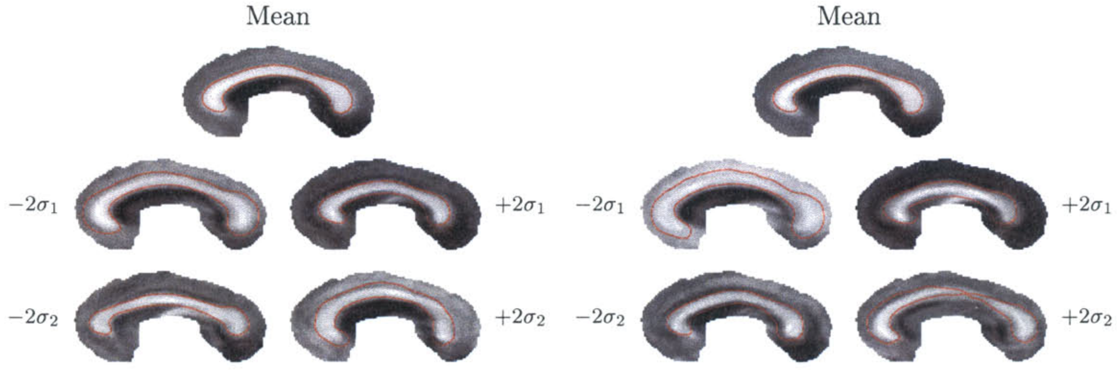


Figure 5-27: This example illustrates how averaging distance maps is more robust to misalignment than averaging grayscale. Two example corpora are shown in (a) and (b). The zero level set of the average of the distance maps is shown in (c). Image (d) shows the average of their intensities with the average contour superimposed.

where $\tilde{\mathbf{u}}$ is the reconstructed "distance" surface, and $\tilde{\mathbf{v}}$ holds the reconstructed grayscale intensities.

When segmenting a novel image, we still compute the most likely pose and shape of the model, and then evolve the curve as described in the previous section. The intensity model is incorporated when computing the maximum *a posteriori* pose and shape. Instead of a gradient-based image term (Eq. 5.31), we compute the sum of squared differences between the observed image I under the pose $\mathbf{p}$ and the expected intensity vector $\tilde{\mathbf{v}}$ computed from the current $\boldsymbol{\alpha}$. Thus, the model fitting stage searches over the eigen-coefficients to find the best match of shape and intensity. Figure 5-26 illustrates a novel image and the shape + intensity model fitting to the image.

When the model appropriately captures the intensity variance, the matching process benefits from the richer model. However, there is a flaw in this approach to incorporating grayscale information. While the distance map is robust to slight misalignment, the intensity information is not, which causes blurring and unrealistic images when the intensities are combined. Figure 5-27 illustrates the problem with



Distance Values Dominate

Intensity Values Dominate

Figure 5-28: Problems with including intensity information with the distance-map shape model. LEFT: The distance values are high in magnitude and the modes are not affected by the addition of intensity information, but the intensity reconstructions are not consistent. RIGHT: The intensity values are scaled higher relative to the distances, which adversely affects the shape model as well.

averaging (or more generally, combining) grayscale images that are not aligned. This is similar to the alignment problems in the original EigenFaces paper [99].

We further explore the problems with combining unregistered intensity values by looking at the primary modes of variance of the shape/intensity model. Figure 5-28 illustrates some of these issue. Note that when building the model, we are concatenating distances and intensities in the same vector, which implies an weighting factor between them. When the distances are given greater weight, the shape model is not significantly affected by the intensities not being in correspondence, but does lose fidelity when the intensity influence is increased.

We have also considered using the distance maps to warp all the intensity samples into a consistent coordinate system before including them in the state vector. This effectively boils down to making a hard decision on correspondences (albeit automatic). When this approach is used, correspondence errors cause very noticeable artifacts in the reconstructions when a sharp intensity edge is slightly misaligned after being warped. Aiming to keep the robustness to correspondence, we abandon this technique of building an intensity model, and consider a slightly less rich, but more robust method.

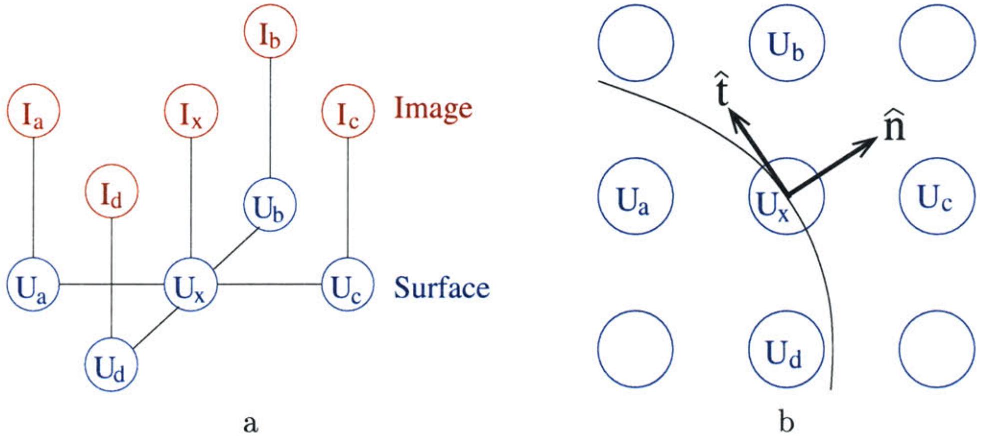


Figure 5-29: (a) Statistical dependency of node samples of the surface U and pixels of image I. (b) The directions of the normal and tangent of U are shown at the position $\mathbf{x}$. The contour curve of height $U_{\mathbf{x}}$ is drawn.

5.5 Statistical Image-Surface Relationship

Given an image $I(\mathbf{x})$, the segmentation problem involves finding the closed curve $\mathcal{C}$ that lies along the boundary the object of interest in the image. We define a surface U as the signed distance function to the curve $\mathcal{C}$. Therefore, $U(\mathbf{x})$ is both the height of the surface U at the position $\mathbf{x}$ and the signed distance from $\mathbf{x}$ to the nearest point on the curve $\mathcal{C}$. Instead of estimating the position of $\mathcal{C}$ directly, we estimate the value of U at every position $\mathbf{x}$ in the image. Once U is computed, the boundary of the object is found by extracting the zero level set of U.

We define a statistical dependency network over the surface U and the image I. To estimate the value $U_{\mathbf{x}}$ of U at a certain position $\mathbf{x}$ in the image, we maximize $P(U_{\mathbf{x}} \mid I, U \backslash U_{\mathbf{x}})$, namely the probability of the height of the surface at that point, given the entire image and the rest of the surface. This expression is difficult to model, given all the dependencies. We therefore simplify the problem to a Markov network with only local dependencies. The links of the nodes in Figure 5-29a represent the dependencies considered. We assume the height of the surface at a certain point depends only on the intensity value of the image at that point, and the neighboring heights of the surface. This assumption can be expressed as:

$$P\left(U_{\mathbf{x}} \mid I,\, U \backslash U_{\mathbf{x}}\right) = P\left(U_{\mathbf{x}} \mid I_{\mathbf{x}},\, U_{\mathcal{N}(\mathbf{x})}\right) \tag{5.41}$$

where $\mathcal{N}(\mathbf{x})$ is the neighborhood of $\mathbf{x}$.

Using properties of our network, we derive an expression for the above probability consisting of terms we will estimate from a set of training data.

$$\begin{align} P(U_{\mathbf{x}} \mid I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})}) &= \frac{P(U_{\mathbf{x}}, I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})})}{P(I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})})} \tag{5.42}\\ &= \frac{P(I_{\mathbf{x}}, U_{\mathbf{x}})}{P(I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})})} \frac{P(U_{\mathbf{x}}, I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})})}{P(I_{\mathbf{x}}, U_{\mathbf{x}})} \tag{5.43}\\ &= \frac{P(I_{\mathbf{x}}, U_{\mathbf{x}})}{P(I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})})} P(U_{\mathcal{N}(\mathbf{x})} \mid I_{\mathbf{x}}, U_{\mathbf{x}}) \tag{5.44}\\ &= \frac{P(I_{\mathbf{x}}, U_{\mathbf{x}}) P(U_{\mathcal{N}(\mathbf{x})} \mid U_{\mathbf{x}})}{P(I_{\mathbf{x}}, U_{\mathcal{N}(\mathbf{x})})} \tag{5.45} \end{align}$$

The definition of conditional probability is used in Equations 5.42 and 5.44, and we are multiplying by the identity in Equation 5.43. In the final expression, we note that when conditioned on $U_{\mathbf{x}}$, the heights of the surface at neighboring locations do not depend on $I_{\mathbf{x}}$.

The first term in Equation 5.45 is the *image term*, that relates the intensity and the surface at $\mathbf{x}$. The next term relates the neighborhood of U about $\mathbf{x}$ to $U_{\mathbf{x}}$, and will act as a *regularization term.* The following two sections describe a means of estimating these functions based on prior training data. The denominator of Equation 5.45 does not depend on $U_{\mathbf{x}}$, and thus is ignored during the estimation process.

5.5.1 Intensity Model

We define a joint distribution model, $P(i, d)$, that relates intensity values, i, to signed distances d, based on a set of training images and training segmentations. Let $\{I_1, I_2, \ldots, I_n\}$ be a set of images of the same modality containing the same anatomical structure of various subjects and $\{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_n\}$ be the set of corresponding segmented boundary curves of the anatomical structure. Let U_j be the signed dis-

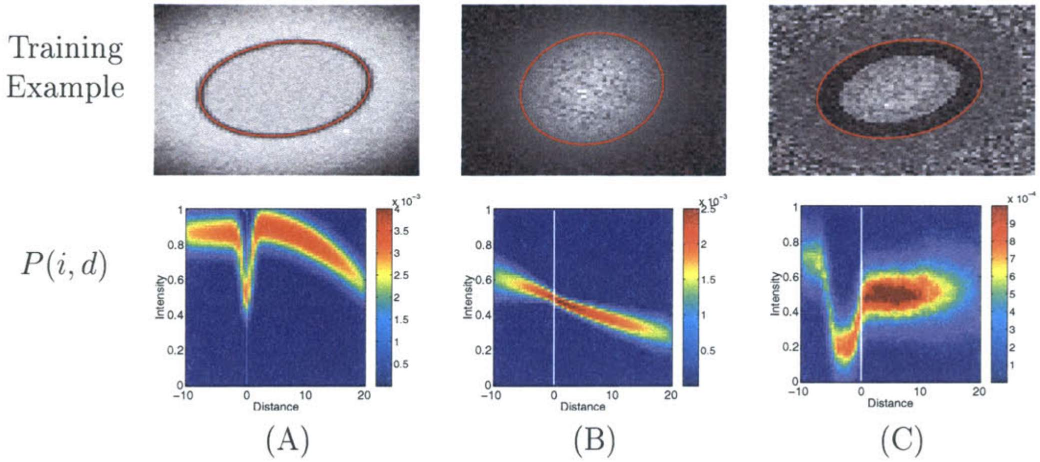


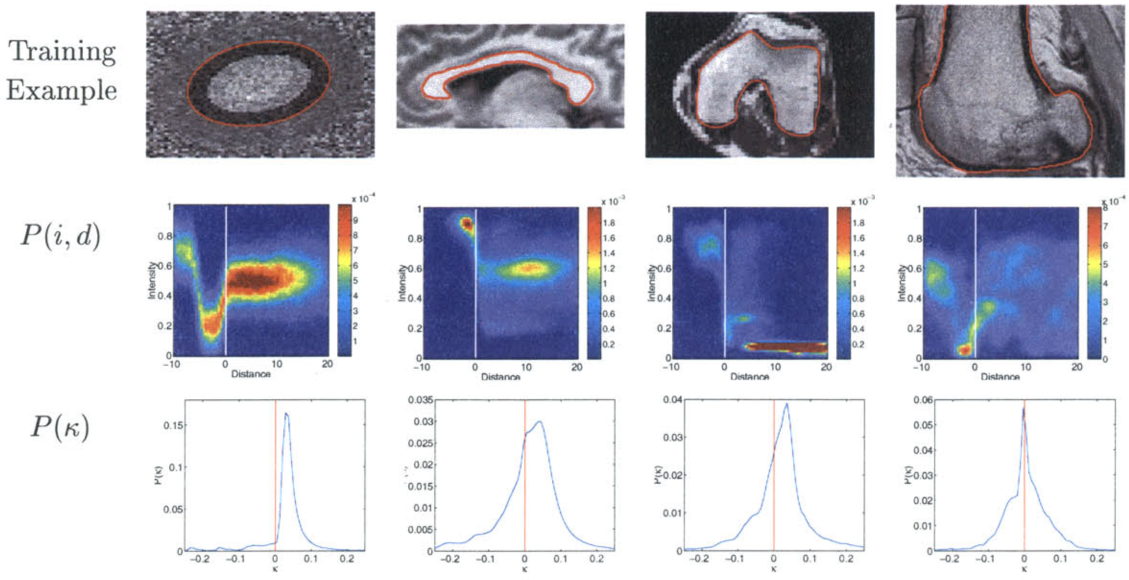
Figure 5-30: Top: One example of each of three training sets of random ellipses under different imaging conditions The red contour shows the boundary of the object. Bottom: The joint intensity/distance-to-boundary PDF derived from the training set. Left and right of the white vertical line indicates the intensity profile inside and outside the object respectively.

tance map to the closed curve $\mathcal{C}_j$. The training set $\mathcal{T}$, is a set of image-surface pairs, $\mathcal{T} = \{\langle I_1, U_1\rangle, \ldots, \langle I_n, U_n\rangle\}$. We use a Parzen window density estimator with a Gaussian windowing function to model the joint distribution. The PDF is the mean of many 2D Gaussians centered at the positions $\langle I_j(\mathbf{x}), U_j(\mathbf{x})\rangle$ for every pixel in every training image:

$$P(i,d) = \frac{1}{Z}\sum_{j=1}^{n}\sum_{\mathbf{x}\in\mathcal{X}} e^{\left(-\frac{(i-I_j(\mathbf{x}))^2}{2\sigma_i^2}-\frac{(d-U_j(\mathbf{x}))^2}{2\sigma_d^2}\right)} \tag{5.46}$$

where $\mathcal{X}$ is the set of all positions in the image, σ_i and σ_d are the variances of the Parzen kernel, and the normalization factor $Z = (2\pi\sigma_i\sigma_d n|\mathcal{X}|)^{-1}$.

Figure 5-30 illustrates the estimated joint probability density functions for random ellipses with various contrast scenarios. Notice that Ellipse A can be easily segmented using a gradient based method, but cannot be thresholded, as the intensity distribution inside and outside the object overlap significantly. On the other hand, simple gradient methods would fail to extract Ellipse B due to high texture, while a single intensity threshold isolates the object. Neither scheme would work for Ellipse C, as higher level knowledge is required to know which edge is the correct boundary.



(a) (b) (c) (d)

Figure 5-31: Top: One example of each of four training sets of objects. Middle: The joint intensity/distance-to-boundary PDF derived from the training set. Bottom: The curvature profile of each object class. Notice that the ellipse class has only positive curvature.

Figure 5-31 illustrates the estimated joint PDFs for the corpus callosum and slices of two acquisitions of the femur. In the sagittal scan of the femur (Figure 5-31d), the dark layer around the brighter, textured region is cortical bone, which gives essentially no MR signal. Without prior knowledge of the intensity distribution of the entire femur, the cortical bone could easily be missed in a segmentation. However, by training a joint distribution over intensity and signed distance, we generate a PDF that encodes the fact that the dark region is a part of the object. This example is similar to the synthetic ellipse in Figure 5-31a.

5.5.2 Curvature Model

Boundary detection methods commonly use a regularization term to uniformly penalize regions of high curvature, independent of the structure being segmented [52, 9, 113]. Our goal is to use the term $P(U_{\mathcal{N}(\mathbf{x})} \mid U_{\mathbf{x}})$ as a local regularizer, but one that can be estimated from a set of training data, and thereby tuned to the appropriate application. One method of modeling this function is to create a 5D Parzen joint PDF over $U_{\mathbf{x}}$ and its four neighbors. A drawback of this approach is that a 5D space is difficult to fill and thus requires many training samples. Furthermore, the neighborhood depends on the embedded coordinate system, and would give different results based on the pose of the object in the image (consider a square aligned with the axes versus one rotated by 45°). Therefore, instead of considering the four neighbors (two horizontal and two vertical) in the "arbitrary" embedded coordinate system, we use four (interpolated) neighbors in the direction of the local normal ($\hat{n}$) and tangent ($\hat{t}$) to the embedded curve (see Figure 5-29b). In other words, we reparameterize to an intrinsic coordinate system based on the surface:

$$\begin{aligned} P(U_{\mathcal{N}(\mathbf{x})} \mid U_{\mathbf{x}}) &= P(U_{h^+}, U_{h^-}, U_{v^+}, U_{v^-} | U_{\mathbf{x}}) && (5.47) \\ &= P(U_{\hat{n}^+}, U_{\hat{n}^-}, U_{\hat{t}^+}, U_{\hat{t}^-} | U_{\mathbf{x}}) && (5.48) \\ &= P(U_{\hat{n}^+}, U_{\hat{n}^-} | U_{\mathbf{x}}) P(U_{\hat{t}^+}, U_{\hat{t}^-} | U_{\mathbf{x}}) && (5.49) \end{aligned}$$

The last step makes the assumption that the neighbors in the normal direction and those in the tangent direction are conditionally independent given $U_{\mathbf{x}}$.

We now seek to model the quantities in Eq. 5.49 based on properties of U and the boundary to be segmented. While U is considered to be a finite network of nodes and $U_{\mathbf{x}}$ a particular node, we define $u(\mathbf{x})$ to be the analogous continuous surface over $\mathbf{x}$. Given u, the unit normal and unit tangent to the underlying level set are defined as [93]:

$$\hat{\mathbf{n}} = \frac{\nabla u}{|\nabla u|} = \frac{1}{(u_x^2 + u_y^2)^{\frac{1}{2}}} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{5.50}$$

$$\hat{\mathbf{t}} = \frac{1}{(u_x^2 + u_y^2)^{\frac{1}{2}}} \begin{bmatrix} -u_y \\ u_x \end{bmatrix} \tag{5.51}$$

We now consider the values the first and second partial derivatives of u in the directions of the tangent and normal. We first evaluate the second-order Taylor expansion of u in the direction of $\mathbf{w}$:

$$u(\mathbf{x}+\mathbf{w}) \;=\; u(\mathbf{x}) \;+\; \lambda\frac{\partial u}{\partial \mathbf{x}}\mathbf{w} \;+\; \frac{\lambda^2}{2}\mathbf{w}^{\mathsf{T}}\frac{\partial^2 u}{\partial \mathbf{x}^2}\mathbf{w} \;+\; \ldots \tag{5.52}$$

We compute a general expression for derivatives of u at $\mathbf{x}$ in the direction of $\mathbf{w}$:

$$\begin{aligned} \frac{\partial u}{\partial w} &= \lim_{\lambda\to 0}\frac{u(\mathbf{x}+\lambda\mathbf{w})-u(\mathbf{x}-\lambda\mathbf{w})}{2\lambda} && (5.53)\\ &= \frac{\partial u}{\partial \mathbf{x}}\mathbf{w} && (5.54) \end{aligned}$$

And the second derivative:

$$\begin{aligned} \frac{\partial^2 u}{\partial w^2} &= \lim_{\lambda\to 0}\frac{u(\mathbf{x}+\lambda\mathbf{w})+u(\mathbf{x}-\lambda\mathbf{w})-2u(\mathbf{x})}{\lambda^2} && (5.55)\\ &= \mathbf{w}^{\mathsf{T}}\frac{\partial^2 u}{\partial \mathbf{x}^2}\mathbf{w} && (5.56) \end{aligned}$$

Using these general expressions, we compute the first partial in the direction of $\hat{\mathbf{n}}$:

$$\begin{aligned} \frac{\partial u}{\partial \hat{n}} &= \frac{\partial u}{\partial \mathbf{x}}\hat{\mathbf{n}} && (5.57)\\ &= \frac{[u_x \;\; u_y]}{(u_x^2+u_y^2)^{\frac{1}{2}}}\begin{bmatrix} u_x \\ u_y \end{bmatrix} && (5.58)\\ &= (u_x^2+u_y^2)^{\frac{1}{2}} && (5.59)\\ &= |\nabla u| && (5.60) \end{aligned}$$

It is not surprising that $\frac{\partial u}{\partial \hat{n}} = |\nabla u|$, given that $\hat{\mathbf{n}}$ by definition lies in the direction of the gradient. When u is a signed distance map, $\frac{\partial u}{\partial \hat{n}} = |\nabla u| = 1$. Now consider the first derivative in the direction of the tangent:

$$\begin{aligned} \frac{\partial u}{\partial \hat{t}} &= \frac{\partial u}{\partial \mathbf{x}}\hat{\mathbf{t}} && (5.61)\\ &= \frac{[u_x \;\; u_y]}{(u_x^2+u_y^2)^{\frac{1}{2}}}\begin{bmatrix} -u_y \\ u_x \end{bmatrix} && (5.62)\\ &= 0 && (5.63) \end{aligned}$$

As expected, effectively dotting the tangent and normal yields a derivative of zero. More interestingly, we consider the second derivatives:

$$\begin{aligned} \frac{\partial^2 u}{\partial \hat{n}^2} &= \hat{\mathbf{n}}^{\mathsf{T}} \frac{\partial^2 u}{\partial \mathbf{x}^2} \hat{\mathbf{n}} \qquad (5.64) \\ &= \frac{1}{u_x^2 + u_y^2} \begin{bmatrix} u_x & u_y \end{bmatrix} \begin{bmatrix} u_{xx} & u_{xy} \\ u_{xy} & u_{yy} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \qquad (5.65) \\ &= \frac{u_{xx}u_x^2 + 2u_{xy}u_xu_y + u_{yy}u_y^2}{u_x^2 + u_y^2} \qquad (5.66) \end{aligned}$$

We derive the value of the above expression under the special case when u is a distance function (or constant multiple thereof):

$$u_x^2 + u_y^2 = c$$

$$\begin{array}{rcl|rcl} \frac{\partial}{\partial x}\left(u_x^2 + u_y^2\right) & = & \frac{\partial}{\partial x} c & \frac{\partial}{\partial y}\left(u_x^2 + u_y^2\right) & = & \frac{\partial}{\partial y} c \\ 2u_xu_{xx} + 2u_yu_{xy} & = & 0 & 2u_xu_{xy} + 2u_yu_{yy} & = & 0 \\ u_x^2u_{xx} + u_xu_yu_{xy} & = & 0 & u_xu_yu_{xy} + u_y^2u_{yy} & = & 0 \end{array}$$

$$u_{xx}u_x^2 + 2u_{xy}u_xu_y + u_{yy}u_y^2 = 0$$

where the last line is the result of summing the two equations from the previous line. Therefore, $\frac{\partial^2 u}{\partial \hat{n}^2} = 0$ when u is a constant multiple of a distance function. The final expression we consider is the second derivative in the direction of the tangent, which we will show is related to the curvature of the underlying level set. Recall the curvature of the level sets can be computed from the surface u by:

$$\kappa = \frac{u_{xx}u_y^2 - 2u_xu_yu_{xy} + u_{yy}u_x^2}{(u_x^2 + u_y^2)^{\frac{3}{2}}} \qquad (5.67)$$

With this equation for curvature in mind, we turn to computing the second derivative of u in the tangent direction:

$$\begin{aligned} \frac{\partial^2 u}{\partial \hat{t}^2} &= \hat{\mathbf{t}}^{\mathsf{T}} \frac{\partial^2 u}{\partial \mathbf{x}^2} \hat{\mathbf{t}} \qquad (5.68) \\ &= \frac{1}{u_x^2 + u_y^2} \begin{bmatrix} -u_y & u_x \end{bmatrix} \begin{bmatrix} u_{xx} & u_{xy} \\ u_{xy} & u_{yy} \end{bmatrix} \begin{bmatrix} -u_y \\ u_x \end{bmatrix} \qquad (5.69) \end{aligned}$$

$$= \frac{u_{xx}u_y^2 - 2u_x u_y u_{xy} + u_{yy}u_x^2}{u_x^2 + u_y^2} \tag{5.70}$$

$$= \kappa \ (u_x^2 + u_y^2)^{\frac{1}{2}} \tag{5.71}$$

Note that in cases where u is a signed distance map, $\frac{\partial^2 u}{\partial \hat{t}^2} = \kappa$.

In summary, here are the values of the first and second partial derivatives of u in the tangent and normal directions under the assumption that the surface u is a continuous signed distance map:

$$\begin{array}{cc} \frac{\partial u}{\partial \hat{n}} = 1 & \frac{\partial u}{\partial \hat{t}} = 0 \\ \\ \frac{\partial^2 u}{\partial \hat{n}^2} = 0 & \frac{\partial^2 u}{\partial \hat{t}^2} = \kappa \end{array} \tag{5.72}$$

We can now use this information about the ideal distance map and underlying curvatures to regularize U at $\mathbf{x}$. Consider the formulas for (centered) finite differences of an arbitrary 1D function.

$$\frac{df(x)}{dt} \approx \frac{f(x+t) - f(x-t)}{2t} \tag{5.73}$$

$$\frac{d^2 f(x)}{dt^2} \approx \frac{f(x+t) + f(x-t) - 2f(x)}{t^2} \tag{5.74}$$

Notice that $f(x)$ does not appear in $\frac{df(x)}{dt}$, indicating that changing f only at one position at a time cannot change the local slope. However, changing $f(x)$ does directly affect the second derivative.

We define the likelihood of the neighbors in the normal direction to be:

$$P(U_{\hat{n}^+}, U_{\hat{n}^-} | U_{\mathbf{x}}) = \frac{1}{Z_1} \exp\left(\frac{(U_{\hat{n}^+} + U_{\hat{n}^-} - 2U_{\mathbf{x}})^2}{2\sigma_1^2}\right) \tag{5.75}$$

where Z_1 is a normalization factor and σ_1 determines the strength of the constant-normal-direction constraint. This has the effect of keeping the gradient magnitude of the surface constant (however not necessarily 1), preventing the surface from evolving arbitrarily. Typically one needs to extract the zero level set and reinitialize the distance function from time to time during the evolution. This direction of regularization reduces the need to reinitialize the surface.

As for the likelihood of the neighbors in the tangent direction, we seek an expression that takes into account the curvature profile of the training data. A convex shape, for example, has non-negative curvature everywhere, so if the training set consists of all convex shapes, the level sets of the surface should stay convex throughout the estimation. The fact that a polygon has bi-modal curvature (0 and ∞) could be used to define a polygon-finder, if the surface can be constrained to match the distribution of curvatures of the training set. For each training surface, U_j, we compute a curvature map K_j using Equation 5.67. We then define a PDF which encourages the curvature as reflected in the discrete differences $(U_{\hat{t}+} + U_{\hat{t}-} - 2U_{\mathbf{x}})$ to be consistent with curvatures observed in the training data $(K_j(\mathbf{x}))$. The PDF is derived using Parzen windowing, similar to the intensity PDF defined in Section 5.5.1.

$$P(U_{\hat{t}+}, U_{\hat{t}-} | U_{\mathbf{x}}) = \frac{1}{Z_2} \sum_{j=1}^{n} \sum_{\mathbf{x} \in \mathcal{X}} \exp\left(- \frac{||(U_{\hat{t}+} + U_{\hat{t}-} - 2U_{\mathbf{x}}) - K_j(\mathbf{x})||^2}{2\sigma_2^2} \right) \qquad (5.76)$$

The third row of Figure 5-31 shows the curvature profile of training sets of various objects.

In summary, the regularization of the surface is broken down into a regularization in the local normal and tangent directions. The second derivative in the normal direction is modeled as a zero mean, low variance Gaussian to keep the surface linear in that direction. A distribution over the second derivative in the tangent direction (e.g. the curvature of the level sets) is derived from the training data and used as an object-specific curvature regularization term.

5.6 Surface Estimation

In the previous section, we defined a model of the embedding surface of the boundary that we wish to estimate given a novel image. Starting with an initial closed curve, we build the higher dimensional surface by computing the signed distance to the curve at every point. Given the prior model and the image, the height of the surface at location $\mathbf{x}$ is related to the image intensity at $\mathbf{x}$ and the local neighborhood of the surface by the following equation:

$$P\left(U_{\mathbf{x}} \mid I_{\mathbf{x}},\, U_{\mathcal{N}(\mathbf{x})}\right) \propto \underbrace{P(I_{\mathbf{x}}, U_{\mathbf{x}})}_{\text{Image Term}}\ \underbrace{P(U_{\hat{t}^+}, U_{\hat{t}^-} | U_{\mathbf{x}})}_{\text{Curvature Term}}\ \underbrace{P(U_{\hat{n}^+}, U_{\hat{n}^-} | U_{\mathbf{x}})}_{\text{Linearity Term}} \tag{5.77}$$

We use this equation to re-estimate each surface point independently, maximizing its log probability while assuming the rest of the surface is constant.

$$\tilde{U}_{\mathbf{x}} = \max_{U_{\mathbf{x}}} \log P(U_{\mathbf{x}} \mid I_{\mathbf{x}},\, U_{\mathcal{N}(\mathbf{x})}) \tag{5.78}$$

Instead of finding the global maximum of the PDF for each $U_{\mathbf{x}}$, we adjust $U_{\mathbf{x}}$ in the direction of increasing probability towards the local maximum by differentiating the log probability in Equation 5.77. At each time step, $U_{\mathbf{x}}$ is updated as:

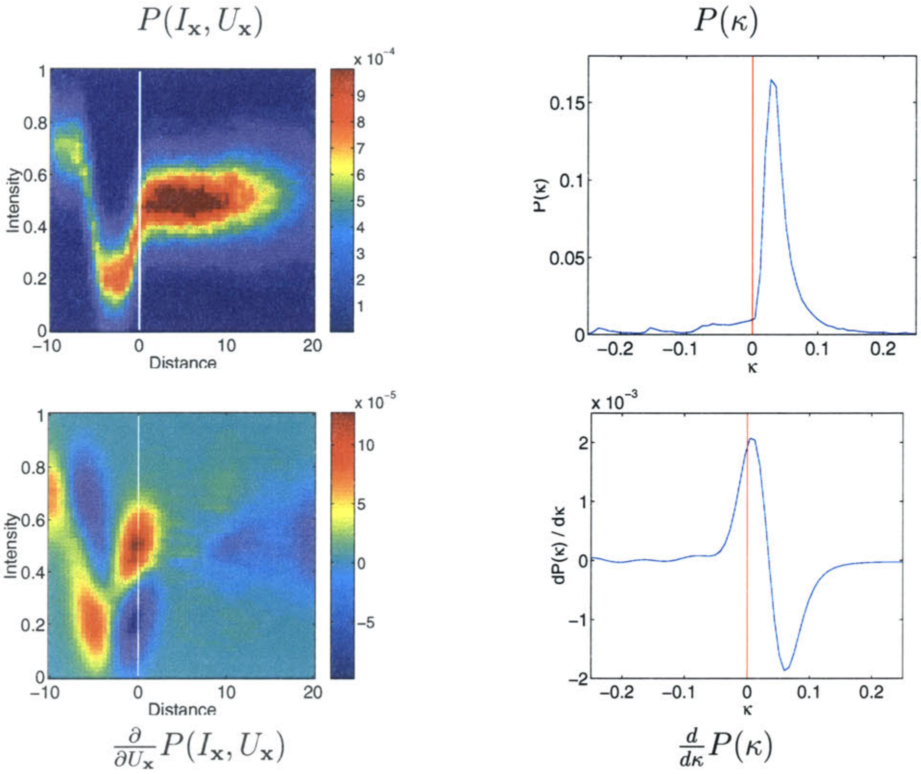
$$\tilde{U}_{\mathbf{x}} = U_{\mathbf{x}} + \lambda D_{\mathbf{x}} \tag{5.79}$$

where D is defined at every point $\mathbf{x}$ by

$$\begin{aligned} D_{\mathbf{x}} &= \frac{d}{dU_{\mathbf{x}}} \log P(U_{\mathbf{x}} \mid I_{\mathbf{x}},\, U_{\mathcal{N}(\mathbf{x})}) && (5.80) \\ &= \frac{d}{dU_{\mathbf{x}}} \log P(I_{\mathbf{x}}, U_{\mathbf{x}}) \ + \ \frac{d}{dU_{\mathbf{x}}} \log P(U_{\hat{t}^+}, U_{\hat{t}^-} | U_{\mathbf{x}}) && (5.81) \\ &\quad + \frac{d}{dU_{\mathbf{x}}} \log P(U_{\hat{n}^+}, U_{\hat{n}^-} | U_{\mathbf{x}}) \end{aligned}$$

This update is repeated until there is little change in the surface. While small, local oscillations in the surface can occur if the step size λ is too high, in practice, an empirical value of λ can easily be found such that the surface evolves in reasonable time without oscillations.

The image and curvature terms in Eq. 5.81 are derived from the training set using a Parzen density estimator. Therefore, the derivative of the log is computed by taking the gradient of the sampled probability in the direction of $U_{\mathbf{x}}$. Figure 5-32 illustrates these gradients. The linearity term is not based on training data, and thus can be



$P(I_{\mathbf{x}}, U_{\mathbf{x}})$

$P(\kappa)$

$\frac{\partial}{\partial U_{\mathbf{x}}} P(I_{\mathbf{x}}, U_{\mathbf{x}})$

$\frac{d}{d\kappa} P(\kappa)$

Figure 5-32: Top Row: The joint intensity vs. distance PDF and the curvature PDF from the synthetic image in Figure 5-31a. Bottom Row: the (partial) derivatives of the PDFs.

cleanly differentiated analytically:

$$\frac{d}{dU_{\mathbf{x}}} \log P(U_{\hat{n}^+}, U_{\hat{n}^-} | U_{\mathbf{x}}) = k \frac{d}{dU_{\mathbf{x}}} \left(U_{\hat{n}^+} + U_{\hat{n}^-} - 2U_{\mathbf{x}}\right)^2 \tag{5.82}$$

$$= -4k \left(U_{\hat{n}^+} + U_{\hat{n}^-} - 2U_{\mathbf{x}}\right) \tag{5.83}$$

which has a minimum when $U_{\mathbf{x}} = \frac{1}{2}\left(U_{\hat{n}^+} + U_{\hat{n}^-}\right)$ which implies that the second derivative in the normal direction is zero. The linearity regularization term that acts on the surface in the direction normal to the level set keeps the gradient magnitude locally constant, but in general the surface does not remain a true distance function. We therefore extract the boundary and reinitialize the distance map when the gradient magnitudes stray from 1. The regularization term greatly reduces the need to reinitialize, but it does not eliminate it. Typically reinitialization occurs once or twice during the segmentation, or about every 50 iterations.

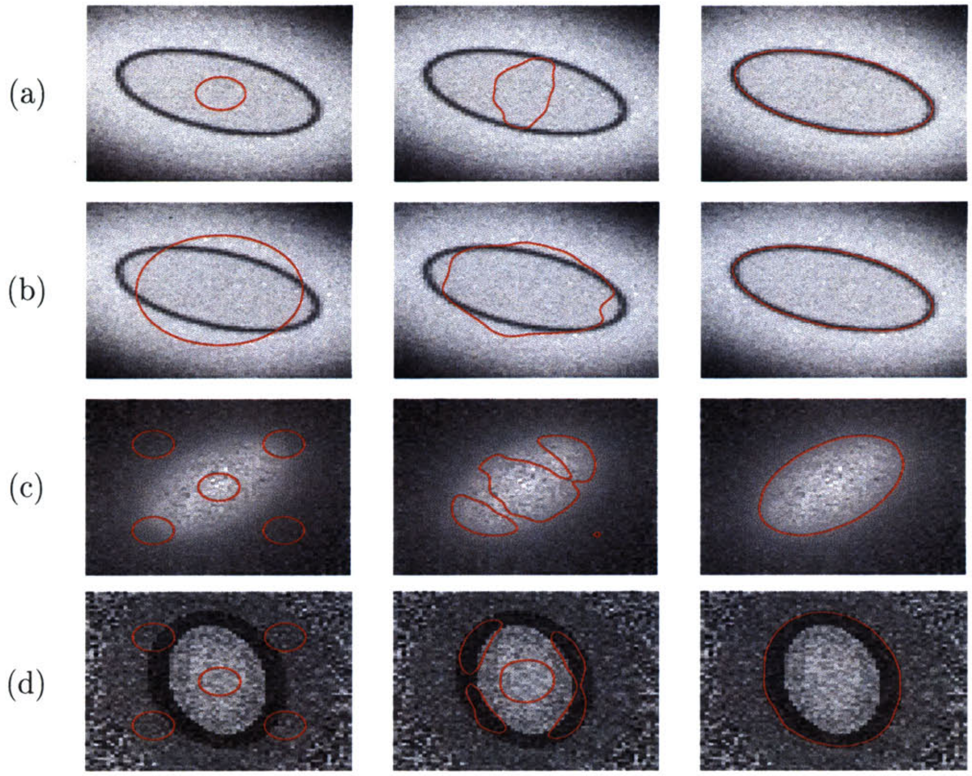


Figure 5-33: The results of segmenting various ellipses. Each group consists of the initial, middle, and final steps in the evolution with the zero level set of the surface overlaid. In each case, the training set consisted of random ellipses with similar rendering. A balloon force was added only for (a) as described in the text.

For efficiency, traditional level set based evolutions do not evolve the higher dimensional surface at every point over the image, but just in a narrow band around the boundary of the object [9]. As most level set segmentation methods are equivalent to curve energy minimization, the band size should not affect the results. Banding in this way can also be applied to our method of surface estimation, but clearly anything outside the band would not contribute to the surface estimation. In the 2D examples presented here, banding was not used, but for efficiency, banding will most likely be a necessity when the method is extended to 3D segmentation.

Some level set segmentation methods require a balloon force be added to the evolution when the initial curve is placed inside the object [9]. This is due to the fact that the regularization term by itself will shrink the evolving curve to a point if the boundaries are not close enough to the edge. A balloon force counteracts

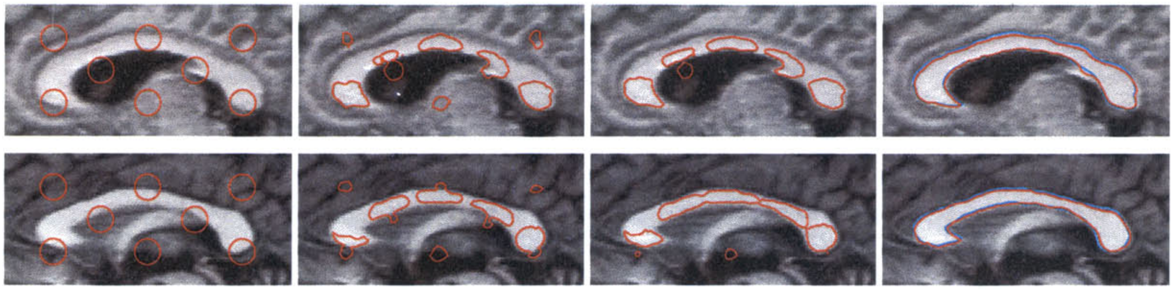


Figure 5-34: Initial, middle, and final steps in the segmentation two corpora callosa. The training set consisted of 48 other corpora callosa with the same MR imaging parameters. The cyan curve in the last frame of each is the manually-segmented ground truth.

the regularization force and pushes the curve outward. In our method, a balloon force is generally not necessary even when starting only from inside the object, as the intensity vs. distance distribution will force the curve to grow or shrink as appropriate. However, in cases where the intensity distribution inside and outside the object are very similar (e.g. Figure 5-30A), only a force at the boundary is felt, which may not be strong enough to pull the zero level set to the edge. For cases such as this, a balloon force can be added to the system to force the expansion.

We report results of testing this method on synthetic data and 2D slices of MR data. Three sets of ellipses were used to test the algorithm under varying contrast conditions. In each case, a training set of 25 random ellipses was generated and used to derive the intensity and curvature model. A novel random ellipse was generated using similar rendering for segmentation. Figure 5-33 shows the results of the segmentation. In the top row, there are two starting curves for the ellipse with the strong edge. In the first case only, (a), a small balloon force was required to push the curve outward, as described above. In case (b), the initial curve crossed the edge, so no balloon force was necessary. Figure 5-33c–d show the segmentations of the other two renderings. In (d), both the interior bright region and the dark boundary are included in the segmentation due to the prior intensity model, consistent with the training images.

The segmentations of two corpora callosa are shown in Figure 5-34. The training set consisted of 48 segmented corpora callosa. Notice that the intensity distribution

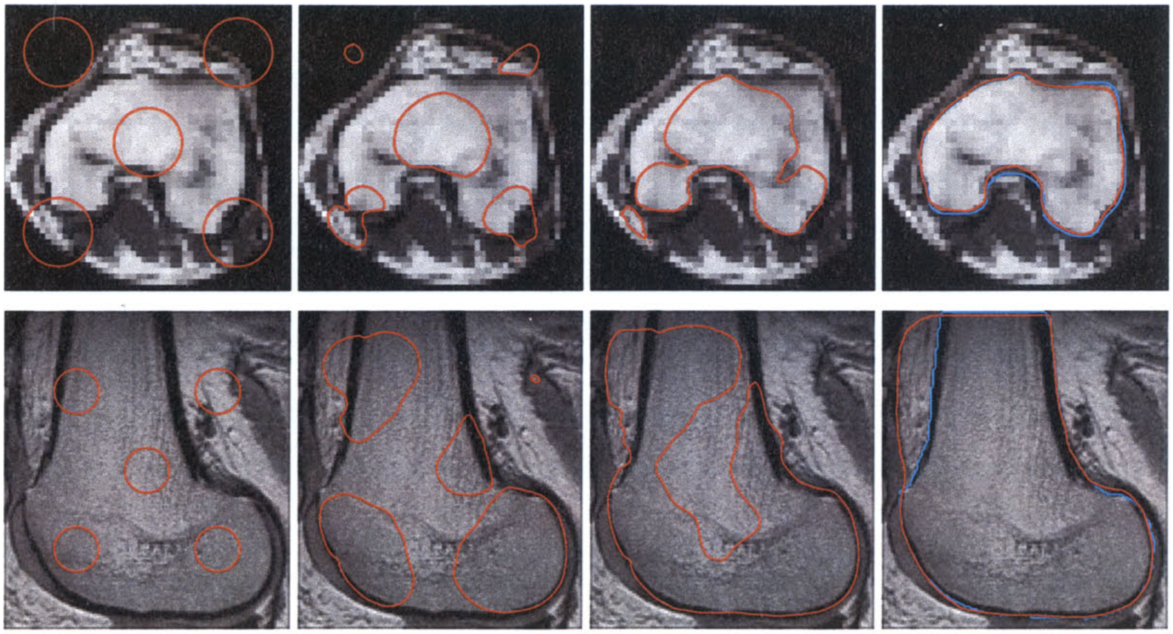


Figure 5-35: Initial, middle, and final steps in the 2D segmentation two acquisitions of the femur. In each case, the training set consisted of the ten neighboring slices of the test slice. The cyan curve in the last frame of each is the manually-segmented ground truth.

in Figure 5-31b is consistent and compact, despite the large number of images. The cyan boundary in the final frame shows a manual segmentation of each corpus. Due to the topological invariance of the level set method, the contours grow, shrink, combine, and split to reach the final segmentation. The algorithm is generally robust to initialization, as long as some portion of the object lies inside the initial curve.

Figure 5-35 shows the segmentation of two slices of femurs from two different MR acquisitions. In each case, the training set consisted of the ten neighboring slices of the test slice. In the axial slices (a), the surface evolves such that the zero level set converges on the boundary of the femur. Notice that in the third image, the two dark spots in the femur create a "snag" for the zero level set as the distances for the dark intensities are much more likely to be outside the object. However, the neighborhood term distributes the information and pulls the region inside the object. Also note that there are other regions in the image with intensity values in the range of those inside the femur. One such region can be seen surrounded by the boundary in the lower left area of the third frame. Again, the neighboring information and curvature

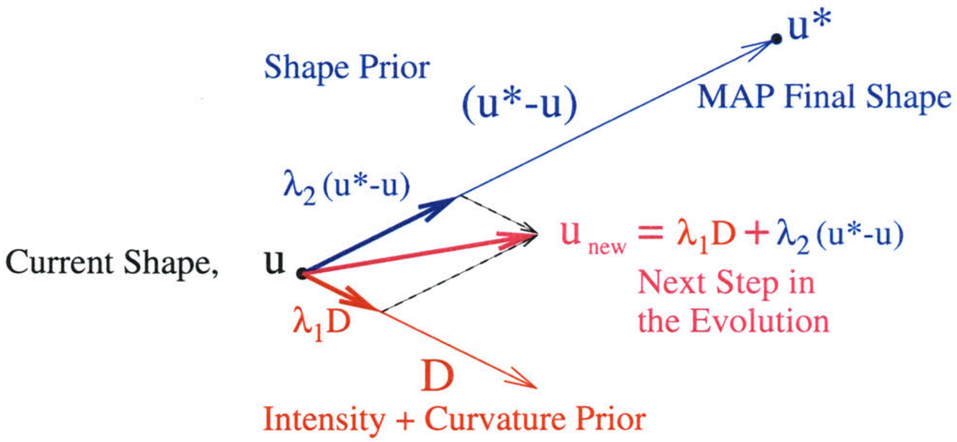


Figure 5-36: Illustration of the various terms in the evolution of the surface, u. The surface u^*, is the maximum *a posteriori* (MAP) final shape. To update u, we combine the local image and curvature update term, $D(\mathbf{x})$, and the direction of the MAP final shape, $u^* - u$.

prior cause this region to correctly be left out of the segmentation.

The sagittal slice of the femur in (b) illustrates both a strength and a weakness of the method. The final boundary successfully captures both the marrow inside the bone and the dark region of cortical bone at the boundary. Furthermore, the segmentation excludes the cartilage, which is the mid-tone semi-circular strip at the bottom of the image. However, the boundary does leak into the muscle located in the upper left as the intensities very closely match those of the interior of the femur. Using a texture measure in addition to the intensity could prevent this leakage.

5.7 Unifying the Statistical Segmentation Models

The preceding sections separately describe the use of global shape models and local intensity and curvature models for segmentation. Thus far, the global shape model introduced in Section 5.3 only uses local gradient information to evolve the surface, but does not use any local priors. The local priors described in Section 5.4 comprise a much stronger intensity model than simply using gradients, allowing for more distant initializations of the curve and faster convergence. However, when using only local priors, other structures nearby with similar intensities, such as the fat next to the femur or the fornix next to the corpus callosum, may be included in the segmentation.

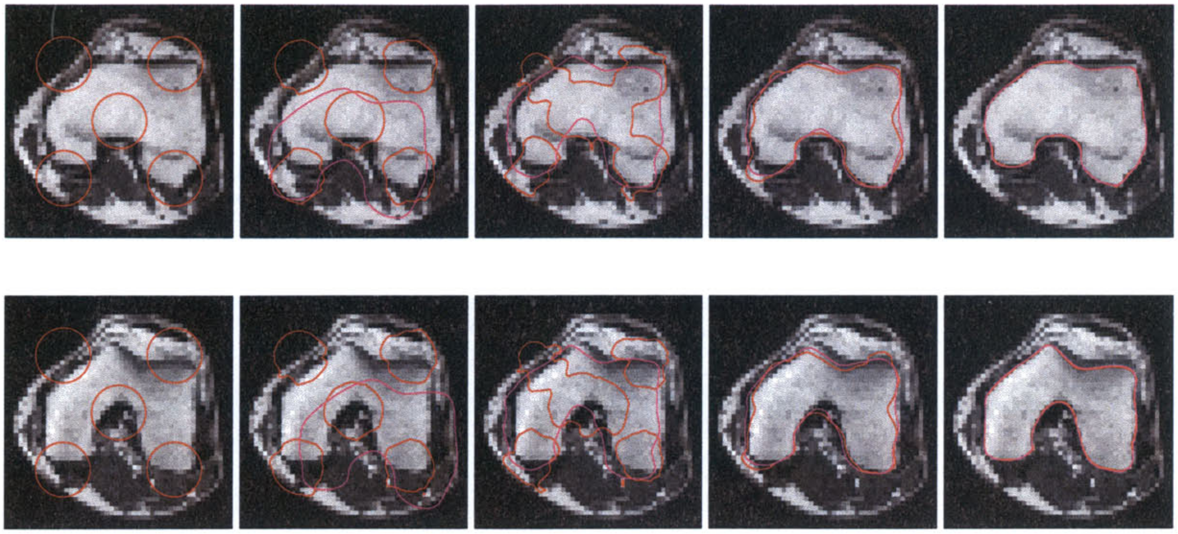


Figure 5-37: Segmentation of two axial slices of the femur. The red curve is the zero level set of the evolving surface, U. The magenta curve is the zero level set of the MAP estimate U^*.

The combination of the global shape model and the local intensity and curvature prior in one framework yields an algorithm more robust to noise and intensity overlap.

The final update rule for the surface $U(\mathbf{x})$ is derived from the local priors in Eq. 5.79 and the global shape influence in Eq. 5.35:

$$U_{\text{new}}(\mathbf{x}) \quad = \quad U(\mathbf{x}) + \overbrace{\lambda_1 D(\mathbf{x})}^{\text{Local Term}} + \underbrace{\lambda_2(U^*(\mathbf{x}) - U(\mathbf{x}))}_{\text{Global Shape}} \tag{5.84}$$

The choice of λ_1 and λ_2 determine the tradeoff between the local and global terms and the step size of the iteration. Thus far, values for these parameters have been chosen empirically. In our experiments, we have found that the local models generally capture the intensity and curvature profiles well, and thus there is a large range (1-2 orders of magnitude) of λ_1 values that work well, and primarily affect only the speed of the evolution. The algorithm is more sensitive to the choice of the λ_2 parameter mainly because the U^* estimate can be quite noisy early in the evolution process. One approach to choosing λ_2 which showed promise in our initial experiments is to have λ_2 be a function of the goodness of fit in estimating U^*. Clearly, when there is confidence in the estimate of U^*, the influence should be higher, so λ_2 is increased.

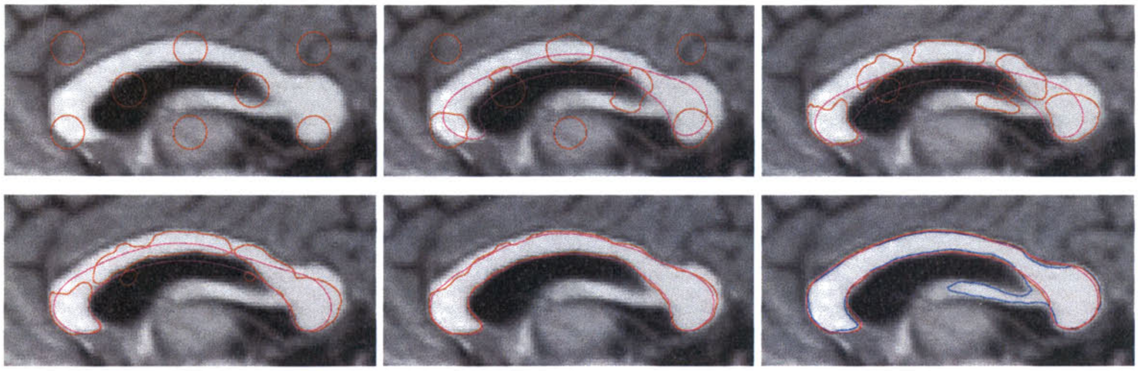


Figure 5-38: Six steps in the segmentation of a corpus callosum using the combined algorithm. The red curve is the zero level set of the evolving surface, U. The magenta curve is the zero level set of the MAP estimate U^*. The blue curve in the final frame is a result of segmenting the corpus with the local intensity and curvature term, but without the shape prior. The strength of the local and global priors make the algorithm insensitive to the choice of the initial curve.

Unfortunately, as often is the case with selecting evolution step sizes, this approach to selecting the parameters may seem a bit *ad hoc*. In Chapter 6, we discuss future work in attempts to fuse the local intensity and global shape influences more naturally in to one process.

Results of the unified segmentation algorithm are shown for axial slices of the femur in Figure 5-37. The initialization consists of the five red circles covering the image. Notice how the red curve evolves as the magenta curve changes pose and shape to localize the femur. Notice that our segmentation algorithm actually returns two answers: the final evolving curve U and the final model estimate U^*. In the cases illustrated here, U and U^* usually converge to within a pixel of one another. We report the curve U as the final answer, due to the fact that our global shape model may not always have enough degrees of freedom to capture pixelwise (or sub-pixelwise) fluctuations in the boundary. Even though the final U is reported as the label map, the final U^* can provide the user with important information. Since U^* is restricted to only feasible shapes from the prior model, if there is a large discrepancy between the two surfaces at convergence, this can indicate one of two important occurrences: (1) the segmentation algorithm may have been distracted by noise or other objects in the image and may have converged on the incorrect boundary, or (2) the anatomical

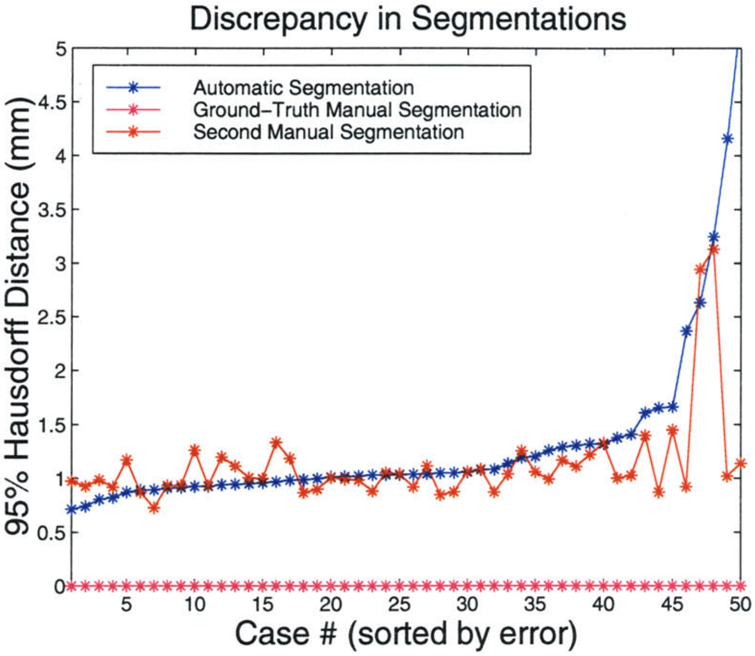


Figure 5-39: A plot comparing the automatic segmentation of 50 corpora callosa to the variability found between two manual segmentations. On manual segmentation (in magenta) was chosen as the ground truth. The cases were sorted by the automatic segmentation, shown in blue, with corresponding second manual segmentation shown in red.

structure in the image may have some type of pathology that had not been previously observed when building the shape model.

Figure 5-38 illustrates the results of segmenting the corpus callosum from an MR image using both the local and global priors. Again, the red curve was initialized using a grid of circles distributed over the image, and evolves during the segmentation process. The magenta curve gets pulled towards the object boundary and influences the evolution. The blue curve in the final image is the result of segmenting the corpus without the global shape prior. The bright protrusion captured by the blue curve is an anatomical structure called the fornix that has an intensity profile almost identical to that of the corpus callosum. Therefore, no algorithm using only local intensity information would be able to detach that structure from the corpus callosum. However, using the global shape model, the protrusion is extremely unlikely and is eliminated from the final segmentation.

In order to evaluate the algorithm, fifty corpora callosa were segmented and compared to manual segmentations. A "leave-one-out" cross-validation method was used, in that each image being labeled was left out of the training set in turn. Two sets of manual segmentations were performed on the fifty images. The first set is considered ground truth and the second set is used to show the amount of variance typically found when manually labeling the boundaries of objects. Since the corpus callosum is a very narrow structure, comparing segmentations via volume overlap can be misleading, as a disproportionate fraction of the voxels lie on the boundary. Therefore, the Hausdorff metric [47] previously described in Equation 5.38 is again chosen to compare segmentations. The results of the segmentations are shown in Figure 5-39. Notice that the automatic segmentation performs comparably to variance found in manual outlining. In five of the cases, the automatic segmentation did not completely extract the correct boundary due to confusion with a thin band of bright white matter adjacent to the corpus callosum, causing the higher errors. However, in two of those cases, the operators also disagreed on the segmentation, indicating the difficulty in extracting the exact boundary. Overall, the segmentation algorithm performed well and was insensitive to the initial starting curve, and showed robustness to the noise and other structures present in the images.

5.8 Summary

This chapter explored methods of incorporating both local and global prior knowledge into the process of medical image segmentation. Representing shape variation across a population of objects can be quite difficult due to issues of correspondence between the training shapes. One approach to the correspondence problem is assume that it has been solved, and that either dense flow fields [49] or corresponding feature points [16] have already been identified. Automatically computing accurate correspondences, especially in three dimensions, is still an open research problem. The other extreme is to ignore correspondences even though the representation cannot truly handle any misalignment, as in Eigenfaces [99], which can detract from the accuracy of the model. We have proposed using the signed distance function as a representation of shape since the boundary information is propagated spatially without loss of fidelity, which provides robustness to slight misalignments in the training set, without requiring full correspondence.

To incorporate intensity information, we derived a model that relates intensity and distance to the object, which captures the intensity profile inside, outside, and at the object boundary. An object-specific local curvature model was also developed and folded into the segmentation process, eliminating the need to hand craft the boundary smoothness term as is traditionally done. The distance surface estimation using the local terms effectively combine the boundary detection and voxel classification methods of segmentation. The global shape influence is effectively an atlas-matching approach. The complete framework therefore unifies three common approaches to segmentation.

Chapter 6

Conclusions and Future Work

We have presented novel registration and segmentation algorithms that incorporate prior information via statistical models derived from training data. The prior models provide application-specific context to help direct the search process towards the best solution. The algorithms and results we have presented suggest various directions of future work, in both the registration and segmentation domains.

6.1 Joint Intensity Registration

The multi-modal registration technique using joint intensity priors has demonstrated sub-voxel accuracy and large region of convergence. To date, we have tested this registration algorithm only on MR images, but here is interest in acquiring more datasets of different modalities, including MR Angiogram, CT, and PET to further examine this registration technique.

Another area of further investigation is to include additional statistical models to the current framework. Non-linear bias fields present in the MR data can cause mismatches in intensity histograms between the training and test images. Registration using a prior on joint intensity information can be sensitive to these differences. Thus, there is interest in integrating the statistical intensity correction work of Wells, *et al.* [108] into this registration technique to both provide a more reliable intensity correspondence between training and test data, and perhaps assist in the process of segmenting the anatomical structures in the images.

Additional information such as prior knowledge of the shapes or relative positions of internal structures may help in the multi-modal registration process. Such information certainly aids in segmentation, by offering guidelines such as what structures are most likely to appear next to other structures. Given that segmentation and registration are related in that the knowledge of one greatly assists in the computation of the other, this would imply that the addition of these types of priors may also assist in registration.

6.2 Segmentation using Statistical Priors

We presented a means of incorporating prior knowledge into the geodesic active contour method of medical image segmentation. The shape representation of using PCA on the signed distance map was chosen with the intention of being robust to slight misalignments without requiring exact point-wise correspondences. Our extension to active contours that estimates the model parameters and then evolves the curve based on that estimation provides a method of robustly converging on the boundary even with noisy inputs. The representation and the curve evolution technique merge well together since the evolution requires a distance map of the evolving curve, which is inherent in the shape model. However, these two modules need not be coupled. A different statistical shape model could be tied into the evolution method, or a different method of model-based matching could be used with the proposed shape model.

6.2.1 Representing Shape Variation

In the analysis of shapes presented in this thesis, an $n \times n$ distance map is represented as a point in $\Re^{n^2}$. However, most points in $\Re^{n^2}$ are not valid distance maps. Furthermore, the set of all distance functions is not closed under linear operations. Averaging two distance functions or taking convex combinations of a set of them will generally not result in a distance function. Yet these are exactly the operations that occur when we perform principal component analysis, as described in Chapter 5. The following two questions arise: (1) When is it reasonable to combine distance functions in this way? (2) Can the distance maps be represented in different space that ensures closure over such operations?

A surface u is a distance function if $|\nabla u|$ is one everywhere, except on the skeleton, where ∇u is undefined. Therefore, we consider situations when the convex combination of two distance functions yield a function whose gradient magnitude is one everywhere. Let u and v be two distance functions such that the *skeletons* of both u and v are identical. Define the function w to be an arbitrary weighted average of u and v:

$$w = \lambda u + \bar{\lambda} v \tag{6.1}$$

where $\lambda + \bar{\lambda} = 1$ and $0 \leq \lambda, \bar{\lambda} \leq 1$. We compute the gradient magnitude of w to show that it is one everywhere:

$$\begin{aligned} |\nabla w| &= |\nabla(\lambda u + \bar{\lambda} v)| && (6.2) \\ &= |\lambda \nabla u + \bar{\lambda} \nabla v| && (6.3) \\ &= |(\lambda + \bar{\lambda}) \nabla u| && (6.4) \\ &= (\lambda + \bar{\lambda}) |\nabla u| && (6.5) \\ &= 1 && (6.6) \end{aligned}$$

where Eq. 6.4 holds because $\nabla u = \nabla v$ since they have identical skeletons. Note that when the skeletons of u and v are identical, there is a one-to-one correspondence between points on the zero level sets of u and v. The greater the correspondence mismatch, the more the gradient magnitude of $\lambda u + \bar{\lambda} v$ will deviate from one. In the extreme, the average of distance functions of a point located at each of all possible positions in the image will have gradient magnitude zero everywhere.

Averaging two distance maps whose skeletons are not alike in general does not produce a distance map. However, the zero level set of the average does have a nice property: It is a subset of the skeleton of the XOR of the two initial shapes (see

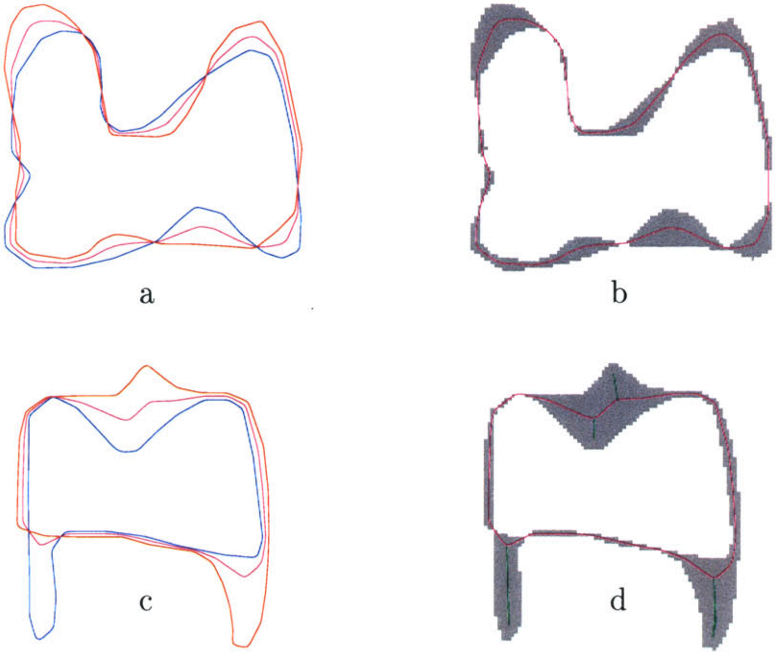


Figure 6-1: (a,c) Consider averaging the distance functions of the red and blue curves. The zero level set of the average distance function is shown in magenta. (b,d) The zero level set of the average two distance functions is a subset of the skeleton of the XOR of the objects (shown in gray). The green skeleton lines in (d) are the regions of the XOR's skeleton where the two closest boundaries belong to the same object.

Figure 6-1). The skeleton of the XOR (like all skeletons) occurs half way between the two boundaries. When the two XOR boundaries are due to the two different shapes, the value of the average of the distance surfaces is zero ($\frac{(+d)+(-d)}{2}$) exactly on the skeleton of the XOR. When the XOR boundaries are due to only one of the shapes, the zero level set is not contained in the skeleton (shown in green in Figure 6-1). Furthermore, these regions also correspond to feature mismatches where the average of the distance functions does not have gradient magnitude equal to one. Further work is required to fully understand when combining distance functions is reasonable, but empirically when the training objects seem to be roughly aligned, the convex combinations produce results close enough to a distance map for our needs.

Another question that would help understand the uses of distance maps relates to constraining operations to a subspace: What is the dimensionality of the subspace of all $n \times n$ distance maps? If the subspace of valid distance maps could be represented analytically, then perhaps the operations we wish to perform on distance maps could be defined to be closed under that subspace, ensuring that we always remain on the manifold of distance functions when representing shapes or evolving surfaces.

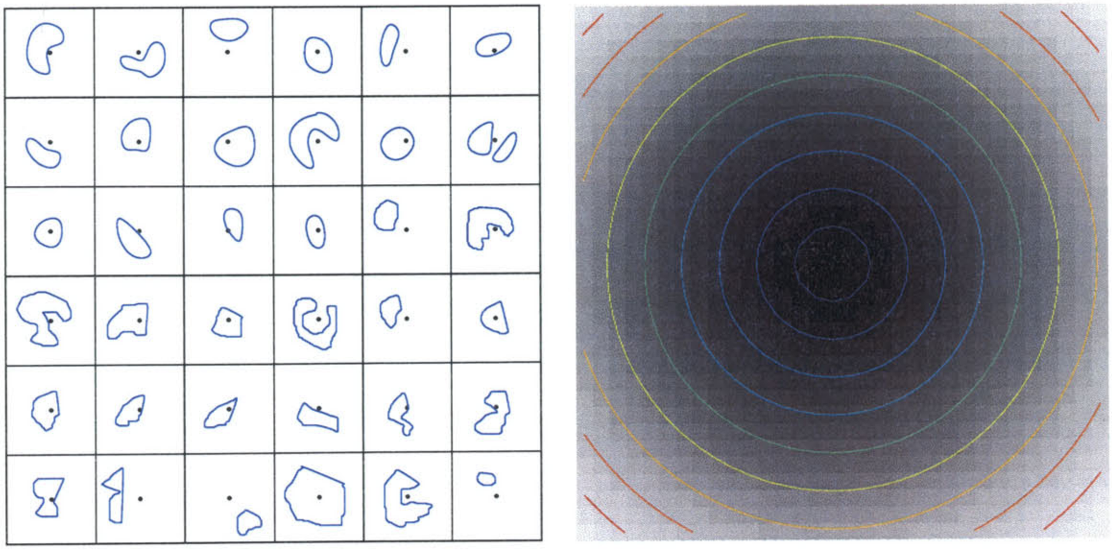
Clearly distance maps are generally extremely redundant, with neighboring values highly correlated. This is why it is possible to represent distance functions of a class of shapes using a relatively small number of principal components, as shown in Chapter 5. The inherent redundancy begs the following question: Is there a k-dimensional manifold of all distance functions embedded in $\Re^{n^2}$ (with $k < n^2$)? Not surprisingly, the dimensionality of such a manifold is a function of the complexity of the shapes being represented. If arbitrarily complex shapes are permitted then the answer to the previous question is no, as illustrated in Figure 6-2a. However, in special cases when the shapes to represent are constrained in some way, then the answer is yes (Figure 6-2b). Since the existence and also properties of a sub-manifold depend on the population of shapes to be considered, it may be rather difficult to compute the manifold in practice without knowing types of shape variation present. However, being able to analytically represent such a manifold would be very useful in the manipulation of distance functions.

6.2.2 Local Statistical Models

Currently, our imaging model consists of relating intensity to distance from the boundary. However, other measures derived from the image can be used in this framework, such as image gradient (∇I), texture measurements, or even the image gradient in the intrinsic coordinate system of the level sets ($\nabla I \cdot \nabla U$). These other measures may be more appropriate for segmentation of some anatomical structures.

The local curvature model thus far has been fully derived and implemented for 1D curves embedded in 2D. Recall that the curvature profile enters the evolution equation by means of the neighborhood term of the distance surface:

$$P\left(U_{\mathcal{N}(\mathbf{x})} \mid U_{\mathbf{x}}\right) = \underbrace{P(U_{\hat{t}^+}, U_{\hat{t}^-} | U_{\mathbf{x}})}_{\text{Curvature Term}} \; \underbrace{P(U_{\hat{n}^+}, U_{\hat{n}^-} | U_{\mathbf{x}})}_{\text{Linearity Term}} \tag{6.7}$$



a b

Figure 6-2: Exploring the dimensionality of the subspace of $n \times n$ distance functions. (a) A "worst case scenario" where the complex boundary (shown in blue) has islands at arbitrary positions inside each of the n^2 pixels. Each sample can vary continuously from $-\sqrt{2}$ to $\sqrt{2}$ independently of any other sample, resulting in n^2 degrees of freedom. (b) A simple case where the manifold of the distance functions of all centered circles of varying radii is just one dimensional.

To model the curvature term in this equation, the probability of observing a certain curvature value, $P(\kappa)$, is derived from training data. As shown in Figure 5-31, the curvature profile is quite different for various shapes, and therefore having a rich model of curvature has potential over simply penalizing high curvatures empirically.

The derivation of the neighborhood term into the linearity (distance) term and the curvature is different when dealing with 3D imagery. In this case, the zero level set is a 2D surface, and the function U is a 3D hypersurface. Furthermore, the 2D boundary surface now has two principal curvatures κ_1 and κ_2. The neighborhood of U now has three orthogonal directions instead of two. Once again, we define the neighborhood to depend on the orientation of the underlying level set with one direction equal to the normal, but in this case, the tangent space is two dimensional. We therefore consider the two principal directions of the tangent space, $\hat{t_1}$ and $\hat{t_2}$, and model the

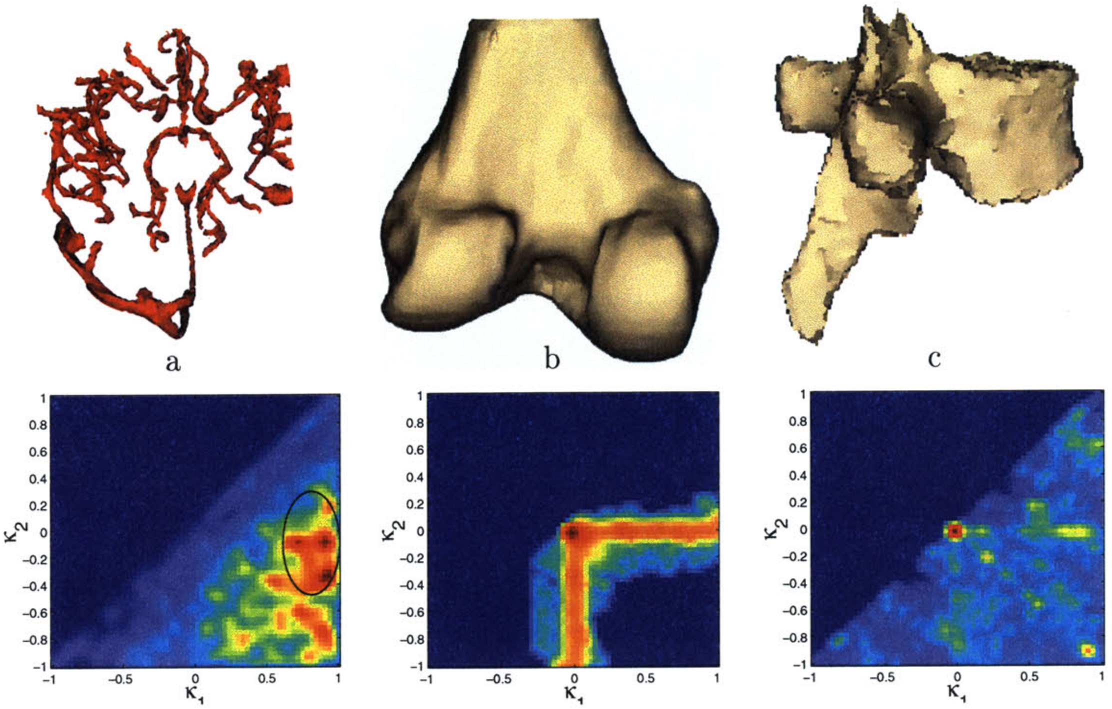


Figure 6-3: Three dimensional models and joint histogram of the two principal curvatures of various structures. (a) Blood vessels in the brain. (b) Base of the femur. (c) Vertebra of the spine. Notice that in (a) much of the probability mass occurs with larger maximal curvature (κ_1) because of the tube-like structure of vessels. (Segmentation and distance map of the vessels provided by Liana Lorigo [69].)

regularization probability as follows:

$$P\left(U_{\mathcal{N}(\mathbf{x})} \mid U_{\mathbf{x}}\right) = \underbrace{P(U_{\hat{t_1}+}, U_{\hat{t_1}-}, U_{\hat{t_2}+}, U_{\hat{t_2}-}, |U_{\mathbf{x}})}_{\text{Principal Curvatures}} \; \underbrace{P(U_{\hat{n}+}, U_{\hat{n}-}|U_{\mathbf{x}})}_{\text{Linearity Term}} \tag{6.8}$$

In the 2D case, we derive in Eq. 5.71 that the derivative of u (the continuous version of the network U) in the direction of $\hat{t}$ is proportional to the curvature κ of the underlying level set. In the 3D case, we expect the following relationships to similarly hold:

$$\frac{\partial^2 u}{\partial \hat{t_1}^2} \propto \kappa_1 \qquad\qquad \frac{\partial^2 u}{\partial \hat{t_2}^2} \propto \kappa_2 \tag{6.9}$$

Therefore, when modeling a curvature prior of a 2D boundary, we can derive the *joint* probability distribution of the two principal curvatures. Figure 6-3 shows an example of such PDFs derived from various structures. Note that the joint curvature profiles are quite different for each object. In the blood vessels, which are effectively tubes, the maximal curvature, which depends on the tubular radius, is *expected* to be large, especially for thin vessels. The minimal curvature, which corresponds to the "bending" of the tube, should be smaller, as found to be the case in the joint curvature profile. This observation by Lorigo *et al.*, led to their derivation of an alternative regularization force for tubular structures that only penalizes the minimal curvature [68, 69]. Their results show marked improvement over the classical method of penalizing either the mean or Gaussian curvature.

While tubular objects generally have high maximal curvature and low minimal curvature, other structures may also have a certain expected joint distribution of the principal curvatures, but not quite as simple as those of tubes. By incorporating the joint curvature prior into the segmentation process (as we had incorporated the 1D curvature prior for curves), the segmentation will favor shapes whose principal curvatures match that of the model, which can help eliminate noise and other distracters.

6.2.3 Multi-Structure Segmentation

Throughout this work, we have considered segmenting only one object at a time from the image. However, often the goal is to segment multiple structures in a region for a full reconstruction of the anatomy. While each object could be segmented in turn, there is useful information in segmenting some or all structures simultaneously. For example, Zeng, *et al.* [114] use level set methods to evolve two surfaces simultaneously, one for gray matter and one for white matter, with a prior on the thickness of the cortical mantle. Kapur [51] also uses a prior based on relative distance between structures. While distance measures can help constrain the segmentation of multiple structures, they cannot provide strong priors on the full 3D pose or shape of neighboring objects.

In our approach, we estimate the alignment of the shape model to the evolving curve at every iteration. The registration provides a correspondence between our

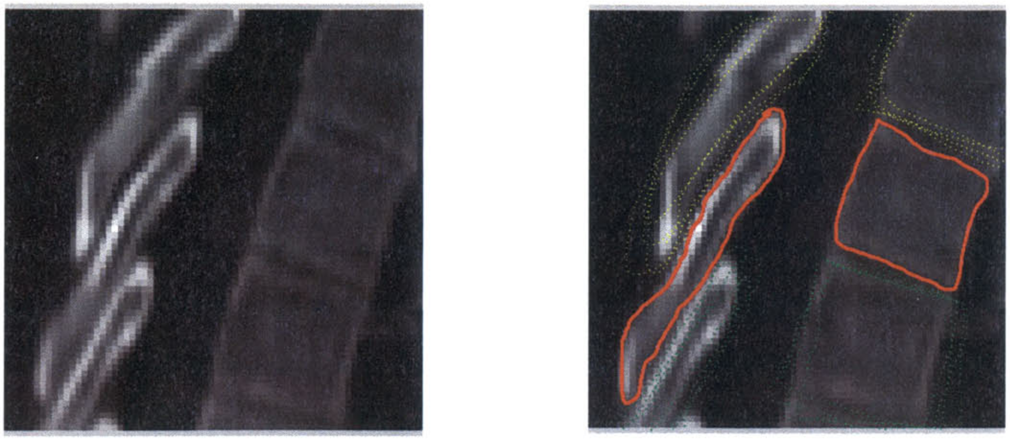


Figure 6-4: LEFT: Image of vertebrae of the spine. RIGHT: Once the pose and shape of one vertebra (U_6^*) has been estimated (red curve), a distribution of the pose and shape of neighboring vertebrae (U_5^* in yellow and U_7^* in green) is induced.

model and the image. If our model is extended to include a distribution of relative poses of neighboring anatomical structures (again, learned from training data), then these priors can be expressed in image coordinates and thus can influence the evolving surfaces corresponding to the other objects. Figure 6-4 illustrates this idea when segmenting the vertebrae of the spine.

6.2.4 Combining Local and Global Information

There are clear benefits to both local and global approaches to image segmentation. Local algorithms can capture precise pixelwise variations in the boundary of an object that would be difficult to model globally. Global methods provide context and high level guidance to the search process, and can relate information from spatially distant regions. The local and global terms presented here are essentially disjoint processes that are combined in the final evolution update equation. However, ideally one energy function could be developed that that unifies both the local and global information, and the final evolution equation can be computed analytically by gradient descent. Many local energy functions are defined to be integrals around the evolving curve, making it more difficult to encode global information such as shape. Furthermore, as observed in Chapter 5, a registration step is required to bring the global model into alignment with the image. Therefore, the correspondence between the prior shape model and the evolving curve must be somehow encoded in the unified energy function for any non pose-invariant global representation. Investigating methods of encoding this correspondence into an energy function is an interesting area of future work.

Bibliography

[1] L. Ambrosio and H. Soner. Level set approach to mean curvature flow in arbitrary codimension. Journal of Differential Geometry **43**:693–737, 1996.

[2] F.L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Patt. Anal. Mach. Intell.*, 11:567-585.

[3] A.J. Bell and T.J. Sejnowski. "An information-maximisation approach to blind separation." In *Advances in Neural Information Processing* **7**, Denver, 1994.

[4] P. Besl and N. McKay A Method For Registration of 3D Shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **14**, 239-256, 1992.

[5] C.M. Bishop. Neural Netowks for Pattern Recognition. Clarendon Press, Oxford, 1995.

[6] Andrew Blake and Michael Isard. *Active Contours.* Springer-Verlag, 1998.

[7] G. Borgefors. Distance transformations in digital images. *CVGIP: Image Understanding*, 34:344–371, 1986.

[8] V. Caselles, F. Catte, T. Coll, and F. Dibos A geometric model for active contours. Numerische Mathematik **66**:1–31, 1993.

[9] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *Int'l Journal of Computer Vision* **22(1)**:61–79, 1997.

[10] G. Christensen, R. Rabbitt, and M. Miller. Deformable templates using large deformation kinematics. *IEEE Trans. on Image Processing* **5(10)**:1435–1447, 1996.

[11] G. Christensen. Consistent Linear-Elastic Transformations for Image Matching. *Information Processing in Medical Imaging* 1999.

[12] H. Cline, W. Lorensen, R. Kikinis, and F. Jolesz. Three-Dimensional Segmentation of MR Images of the Head Using Probability and Connectivity. *Journal of Computer Assisted Tomography*, 14(6):1037–1045, 1990.

[13] L. Cohen On active contour models and balloons. *CVGIP: Image Understanding*, 53(2):211–218, 1991.

[14] Laurent D. Cohen and I. Cohen. Finite element methods for active contour models and balloons for 2d and 3d images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15(11):1131–1147, 1993.

[15] A. Collignon, D. Verndermeulen, P. Suetens, and G. Marchal. "3D multimodality medical image registration using feature space clustering". In *First Conf. on Computer Vision, Virtual Reality and Robotics in Medicine.* Springer, 1995.

[16] T. Cootes, C. Taylor, D. Cooper, and J. Graham. Active shape models - their training and application. *Computer Vision and Image Understanding* 1995.

[17] T. Cootes and C. Taylor. A Mixture Model for Representing Shape Variation. *Image and Vision Computing* 17(8), 567-574, 1999.

[18] T. Cootes, C. Beeston, G. Edwards, and C. Taylor. Unified Framework for Atlas Matching Using Active Appearance Models. *Information Processing in Medical Imaging* 1999.

[19] T. Cootes. Statistical Models of Appearance for Computer Vision. Working report, available from `http://www.wiau.man.ac.uk/~bim/`.

[20] T.M. Cover, J.A. Thomas. Elements of Information Theory. Wiley-Interscience, New York, 1991.

[21] P-E. Danielsson. Euclidean distance mapping. *Computer Graphics and Image Processing*, 14:227–248, 1980.

[22] A. Dempster, N. Laird, and D. Rubin. Maximal Likelihood from Incomplete Data via the EM Algorithm. *Proc. Royal Statistical Society, Series B*, 1977.

[23] Duda, R. and Hart, P. *Pattern Classification and Scene Analysis.* John Wiley & Sons, 1973.

[24] 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* 1997.

[25] P. Edwards, D. Hill, J. Little, D. Hawkes. Deformation for image-guided interventions using a three component tissue model. *Medical Image Analysis*, 2(4): 355-367, 1998.

[26] G. Ettinger. Hierarchical Three-Dimensional Medical Image Registration. MIT Ph.D. Thesis, 1997.

[27] M. Fleute and S. Lavalle. Building a complete surface model from spearse data using statistical shape models: Application to computer assisted knee surgery. In *MICCAI*, 878-887, 1998.

[28] W. T. Freeman and J. B. Tenenbaum. Learning bilinear models for two-factor problems in vision. *IEEE Conference on Computer Vision and Pattern Recognition*, 1997.

[29] W. Freeman and E. Pasztor. Learning low-level vision MERL Tech Report TR-99-12, 1999.

[30] G. Gerig, W. Kuoni, R. Kikinis, and O. Kübler. Medical Imaging and Computer Vision: an Integrated Approach for Diagnosis and Planning. *Proc. 11'th DAGM Symposium*, pages 425–443. Springer, 1989.

[31] G. Gerig, J. Martin, R. Kikinis, O. Kbler, M. Shenton, F. Jolesz. Unsupervised tissue type segmentation of 3D dual-echo MR head data. Image and Vision Computing, 10:349–360 (1992).

[32] S. Gibson. Constrainted Elastic SurfaceNets: generating smooth surfaces from binary segmented data. In *MICCAI*, 1998.

[33] S. Gibson, J. Samosky, A. Mor, *et al.* Simulating Arthroscopic Knee Surgery using Volumetric Object Representations, Real-Time Volume Rendering and Haptic Feedback. *MERL Tech Report*, TR96-19, 1996.

[34] P. Golland, W.E.L. Grimson, R. Kikinis. Statistical Shape Analysis Using Fixed Topology Skeletons: Corpus Callosum Study. *In Inf. Proc. and Med. Imaging (IPMI)*, 1999.

[35] P. Golland, R. Kikinis, C. Umans, M. Halle, M.E. Shenton, J.A. Richolt. AnatomyBrowser: A Framework for Integration of Medical Information. *Medical Image Computing and Computer-Assisted Intervention (MICCAI'98).* 1998.

[36] J. Gomes and O. Faugeras. Reconciling distance functions and level sets. *Proc. Int'l Conf. Scale-Space*, pages 70–81, 1999.

[37] U. Grenander and M. Miller. Representations of knowledge in complex systems. *Journal of the Royal Statistical Society B*, 56:249-603, 1993.

[38] M. Grayson. The heat equation shrinks emedded plane curves to round points. Journal of Differential Geometry **26**:285, 1987.

[39] W.E.L. Grimson, G.J. Ettinger, S.J. White, T. Lozano-Pérez, W.M. Wells III, and R. Kikinis. "An Automatic Registration Method for Frameless Stereotaxy, Image Guided Surgery, and Enhanced Reality Visualization". *IEEE Transactions on Medical Imaging*, **15**(2), April 1996.

[40] Y. Guo and B. Vemuri. Hybrid geometric active models for shape recovery in medical images. In *Int'l Conf. Inf. Proc. in Med. Imaging*, pages 112–125. Springer-Verlag, 1999.

[41] C.R.G. Guttmann, R. Kikinis, M.C. Anderson, M. Jakab, S.K. Warfield, R.J. Killiany, H.L. Weiner, F.A. Jolesz. Quantitative follow-up of patients with multiple sclerosis using MRI: reproducibility. *JMRI* 9:509-518, 1999.

[42] N. Hata, T. Dohi, S. Warfield, W. Wells III, R. Kikinis, F. Jolesz. Multimodality Deformable Registration of Pre- and Intraoperative Images for MRI-Guided Brain Surgery. First International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'98), 1067–1074, 1998.

[43] K. Held, et al. Markov Random Field Segmentation of Brain MR Images. IEEE Transactions on Medical Imaging 16:878–887, 1998.

[44] B.K.P. Horn. *Robot Vision.* MIT Press, Cambridge, MA, 1986.

[45] B.K.P. Horn, "Closed-Form Solution of Absolute Orientation Using Unit Quaternions", *Journal of the Optical Society of America A*, **4**, pp. 629–642, 1987.

[46] N.R. Howe, M. Leventon, W.T. Freeman. Bayesian Reconstruction of 3D Human Motion from Single-Camera Video. In *Neural Information Processing Systems*, 1999.

[47] D. Huttenlocher, G. Klanderman, W. Rucklidge, "Comparing images using the Hausdorff distance" in *IEEE Trans PAMI*, **15**:850-863, 1993.

[48] H. Jiang, R. Robb, and K. Holton. A new approach to 3-D registration of multimodality medical images by surface matching. *Visualization in Biomedical Computing 1992*, Proc. SPIE 1808:196-213, 1992.

[49] M. Jones and T. Poggio. Multidimensional Morphable Models. *Int'l Conf. Computer Vision*, 1998.

[50] T. Kapur, W.E.L. Grimson, W.M. Wells, R. Kikinis. Segmentation of Brain Tissue from Magnetic Resonance Images. *Medical Image Analysis*, 1(2) 1996.

[51] T. Kapur. Model Based Three Dimensional Medical Image Segmentation. MIT Ph.D. Thesis, 1999.

[52] M. Kass, A. Witkin, D. Terzopoulos. Snakes: Active contour models. *Int'l Journal on Computer Vision*, **1**, 321-331, 1988.

[53] M. Kaus, S. Warfield, F. Jolesz, and R. Kikinis. Adaptive template moderated brain tumor segmentation in MRI. *Bildverarbeitung fur die Medizin*, pages 102–106. Springer Verlag, 1999.

[54] A. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum and A. Yezzi. Gradient flows and geometric active contour models. In Proc. IEEE Int'l Conf. Computer Vision, 810–815, 1995.

[55] R. Kikinis, M. Shenton, D. Iosifescu, R. McCarley, P. Saiviroonporn, H. Hokama, A. Robatino, D. Metcalf, C. Wible, C. Portas, R. Donnino, and F. Jolesz. A Digital Brain Atlas for Surgical Planning, Model Driven Segmentation and Teaching. IEEE Transactions on Visualization and Computer Graphics, 2(3), 1996.

[56] A. Kotcheff and C. Taylor. Automatic Construction of Eigenshape Models by Genetic Algorithm. *Information Processing in Medical Imaging*, 1997.

[57] D. Lemoine, D. Liegeard, E. Lussot, and C. Barillot. Multimodal registration system for the fusion of MRI, CT, MEG, and 3D or stereotatic angiographic data. *Medical Imaging 1994: Image Capture, Formatting, and Display*, Proc. SPIE 2164:46-56, 1994.

[58] H. Lester, S. Arridge, K. Jansons, L. Lemieux, J. Hajnal, and A. Oatridge. Non-linear Registration with the Variable Viscosity Fluid Algorithm. *In Inf. Proc. and Med. Imaging (IPMI)*, 1999.

[59] F. Leymarie and M.D. Levine. Faster raster scan distance propagation on the discrete rectangular lattice. *CVGIP: Image Understanding*, 55(1):84–94, 1992.

[60] M. Leventon A Registration, Tracking, and Visualization System for Image Guided Surgery. MIT Master's Thesis, May, 1997.

[61] M. Leventon and W.E.L. Grimson. Multi-Modal Volume Registration Using Joint Intensity Distributions. *MICCAI*, 1998.

[62] M. Leventon and S. Gibson. Model Generation from Multiple Volumes using Constrained Elastic SurfaceNets. *In Inf. Proc. and Med. Imaging (IPMI)*, 1999.

[63] M. Leventon, W.E.L. Grimson, O. Faugeras. Statistical Shape Influence in Geodesic Active Contours. *Comp. Vision Pat. Recog.*, 2000.

[64] M. Leventon, O. Faugeras, W.E.L. Grimson, W.M. Wells. Level Set Based Segmentation with Intensity and Curvature Priors. *Mathem. Meth. in Biomed. Image Anal.*, 2000.

[65] J. Little, D. Hill, and D. Hawkes. Deformaions Incorporating Rigid Structures. *Mathematical Methods in Biomedical Image Analysis*, 1996.

[66] W.E. Lorensen, H.E. Cline, "Marching Cube: A High Resolution 3-D Surface Construction Algorithm", *Computer Graphics* **21**(3), pp. 163–169, 1987.

[67] L. Lorigo, O. Faugeras, W.E.L. Grimson, R. Keriven, R. Kikinis. Segmentation of Bone in Clinical Knee MRI Using Texture-Based Geodesic Active Contours. In *MICCAI*, 1998.

[68] L. Lorigo, O. Faugeras, W. E. L. Grimson, et al. Co-dimension 2 Geodesic Active Contours for MRA Segmentation. *Int'l Conf. Information Processing in Medical Imaging*, 1999.

[69] L. Lorigo. Curve Evolution for Medical Image Segmentation. MIT Ph.D. Thesis, 2000.

[70] T. McInerney and D. Terzopoulos. Deformable models in medical image analysis: a survey. *Medical Image Analysis*, 1(2):91-108, 1996.

[71] T. McInerney and D. Terzopoulos. Medical image segmentation using topologically adaptable surfaces. *Conf. Computer Vision, Virtual Reality, and Robotics in Medicine and Medical Robotics and Computer-Assisted Surgery (CVRMed-MRCAS)*, 23–32, 1997.

[72] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. "Multimodality image registration by maximization of mutual information." *Mathematical Methods in Biomedical Image Analysis.* IEEE Computer Society Press, 1996.

[73] G. Maguire, Jr., M. Noz, H. Rusinek, J. Jaeger, E. Kramer, J. Sanger, and G. Smith. Graphics applied to medical image registration. *IEEE Comput. Graphics Appl.*, 11:20-29, 1991.

[74] J.B. Antoine Maintz, P.A. van den Elsen, and M.A. Viergever. "Comparison of Feature-Based Matching of CT and MR Brain Images." *Computer Vision, Virtual Reality and Robotics in Medicine*, Nice France, pp. 219–228, 1995.

[75] R. Malladi, J. Sethian, and B. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Trans. Patt. Analysis and Mach. Intell.*, 17(2):158–175, February 1995.

[76] V.R. Mandava, J.M. Fitzpatrick, C.R. Maurer, Jr., R.J. Maciunas, and G.S. Allen. "Registration of multimodal volume head images via attached markers." *Medical Imaging VI: Image Processing*, Proc. SPIE 1652:271-282, 1992.

[77] L. Matejic. Group Cascades for Representing Biological Variability in Medical Images. Brown University Ph.D. Thesis, 1997.

[78] M. Miller, A. Banerjee, G.Christensen, S. Joshi, N. Khadeja, U. Grenander, L. Matejic. Statistical methods in computational anatomy. *Staistical Methods in Medical Research*, 1997.

[79] N. Paragios and R. Deriche. A PDE-based Level Set approach for Detection and Tracking of moving objects. *Proc. IEEE Int'l Conf. Comp. Vision*, 1998.

[80] N. Paragios and R. Deriche. Geodesic active regions for supervided texture segmentation. *Proc. IEEE Int'l Conf. Comp. Vision*, 1999.

[81] E. Parzen. On estimation of a probability density function and mode. *Annals of Mathematical Statistics* **33**:1065-1076, 1962.

[82] C. Pelizzari, G. Chen, D. Spelbring, R. Weichselbaum, and C. Chen. Accurate three-dimensional registration of CT, PET, and/or MR images of the brain. *J. Comput. Assist. Tomogr.*, 13:20-26, 1989.

[83] A. Pentland and S. Sclaroff. Closed-form solution for physically based shape modeling and recognition. *PAMI* **13(7)**:715–729, 1991.

[84] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C (2nd Edition).* Cambridge University Press, 1992.

[85] R. Ronfard. Region-based strategies for active contour models. *Int'l J. Comp. Vision*, 13(2):229-251, 1994.

[86] W. Rucklidge. *Efficient computation of the minimum Hausdorff distance for visual recognition.* Cornell University Ph.D. Thesis, 1995.

[87] G. Sapiro. Vector-valued active contours. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 680–685, 1996.

[88] W. Schroeder, K. Martin, and B. Lorensen. *The Visualization Toolkit: an object oriented approach to 3D graphics.* Prentice Hall PTR, Upper Saddle River, New Jersey, 1998.

[89] S. Sclaroff and A. Pentland. Modal Matching for Correspondence and Recognition. *PAMI* **17(6)**:545–561, 1995.

[90] S. Sclaroff and J. Isidoro. Active blobs. *Int'l Conference on Computer Vision*, 1998.

[91] SensAble Technologies, Woburn, MA. `http://www.sensable.com` .

[92] C.R. Shelton. Three-Dimensional Correspondence. Massachusetts Institute of Technology, Masters Thesis. May 1998.

[93] J. Sethian. Level Set Methods. Cambridge University Press, 1996.

[94] L. Staib and J. Duncan. Boundary Finding with Parametrically Deformable Models *PAMI* **14(11)**:1061–1075, 1992.

[95] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, B*, **36(1)**:111-147, 1974.

[96] G. Szekely, A. Keleman, C. Brechbuler, and G. Gerig. Segmentation of 2D and 3D objects from MRI volume data using constrained elastic deformations of flexible Fourier contours and surface models. *Medical Image Analysis*, 1(1):19-34, 1996.

[97] R. Szeliski. *Bayesian Modeling of Uncertainty in Low-Level Vision.* Kluwer Academic Publisher, Boston, 1989.

[98] R. Szeliski, S. Lavallee, "Matching 3D Anatomical Surfaces with Non-Rigid Deformations using Octree-Splines", in *IEEE Workshop on Biomedical Image Analysis*, pp. 144-153, 1994.

[99] M. Turk and A. Pentland. Eigenfaces for recognition *J. of Cogn. Neuroscience* **3(1)**:71–86, 1991.

[100] S. Ullman and R. Basri. Recognition by linear combinations of models. *Comp. Vision Patt. Recogn.*, 1991.

[101] G. Stetten and S. Pizer. Medial-node models to identify and measure objects in real-time 3-D echocardiography. *IEEE Transactions on Medical Imaging*, 18(10):1025–1034, 1999.

[102] P. van den Elsen, E. Pol, T. Sumanaweera, P. Hemler, S. Napel, and J. Adler. Grey value correlation techniques used for automatic matching of CT and MR brain and spine images. *Visualization in Biomedical Computing 1994* Proc SPIE 2359:227-237, 1994.

[103] D. Vandermeulen, X. Descombes, P. Suetens, and G. Marchal. Unsupervised regularized classification of multi-spectral MRI. *Proceedings of the Fifth Conference on Visualization in Biomedical Computing*, SPIE, 1996.

[104] Vannier, M., Butterfield, R., Jordan, D., Murphy, W., et al. Multi-Spectral Analysis of Magnetic Resonance Images. *Radiology*, (154):221–224, 1985.

[105] Y. Wang and L. Staib. Boundary Finding with Correspondence Using Statistical Shape Models. *Comp. Vision Patt. Recogn.*, 1998.

[106] Y. Wang and L. Staib. Elastic Model Based Non-Rigid Registration Incorporating Statistical Shape Information. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 1998.

[107] S. Warfield, A. Robatino, J. Dengler, F. Jolesz, and R. Kikinis. Nonlinear Registration and Template Driven Segmentation. In *Brain Warping*. Ed. Arthur W. Toga, (Progressive Publishing Alternatives) Ch.4:67-84, 1998.

[108] W. Wells, W.E.L. Grimson, R. Kikinis, F. Jolesz. Statistical intensity correlation and segmentation of MRI Data. *Visualization in Biomedical Computing*, 1994.

[109] W.M. Wells III, P. Viola, H. Atsumi, S. Nakajima, R. Kikinis. "Multi-Modal Volume Registration by Maximization of Mutual Information". *Medical Image Analysis*, **1**(1):35–51, 1996.

[110] J. West, J. Fitzpatrick, *et al.* "Comparison and evaluation of retrospective intermodality image registration techniques." In *Medical Imaging: Image Processing*, volume 2710 of *Proc. SPIE*, Newport Beach, California, February 1996.

[111] R. Whitaker, D. Breen. Level-Set Models for the Deformation of Solid Objects. *SIGGRAPH*, 1999.

[112] A. Yezzi, S. Kichenassamy, A. Kumar, P. Olver, and A. Tannenbaum. A geometric snake model for segmentation of medical imagery. *IEEE Trans. Medical Imaging*, 16(2):199–209, 1997.

[113] A. Yezzi A. Tsai, A. Willsky. A Statistical Approach to Snakes for Bimodal and Trimodal Imagery. Seventh IEEE International Conference on Computer Vision, 1999.

[114] X. Zeng, L. Staib, R. Schultz, J. Duncan. Segmentation and Measurement of the Cortex from 3D MR Images. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 1998.