

THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Technical Report No. 415

Towards Intelligent Deformable Models for Medical Image Analysis

by

GHASSAN HAMARNEH



Department of Signals and Systems
School of Electrical and Computer Engineering
Chalmers University of Technology
Göteborg, Sweden 2001

GHASSAN HAMARNEH
Towards Intelligent Deformable Models
for Medical Image Analysis

Copyright © 2001, Ghassan Hamarneh
All rights reserved
ISBN 91-7291-082-8

Doktorsavhandlingar vid Chalmers tekniska högskola
Ny serie nr 1765
ISSN 0346-718x

Technical Report No. 415
Department of Signals and Systems
School of Electrical and Computer Engineering
Chalmers University of Technology
SE-412 96 Göteborg, Sweden
Telephone + 46 (0)31-772 1000
<http://www.s2.chalmers.se>

Cover image: 'Deformable Organisms' progressing through a sequence of behaviors to locate the lateral ventricles, caudate nuclei, and putamina structures in a transversal brain MRI slice (Chapter 8).

Printed in Sweden by Chalmers Reproservice
Göteborg, September 2001

To My Parents Laurice and Saleh Hamarneh

Abstract

Medical imaging continues to permeate the practice of medicine, but automated yet accurate segmentation and labeling of anatomical structures continues to be a major obstacle to computerized medical image analysis (MIA). Deformable models, with its profound roots in estimation theory, optimization, and physics-based dynamical systems, represent a powerful approach to the general problem of medical image segmentation. This Thesis presents a number of novel contributions to the field of deformable modeling, and includes theory as well as application. In the first part of the Thesis, a modified Active Contour Model (ACM), utilizing adaptive inflation reversal and damping, is applied to segmenting oral lesions in color images. In the second part, the amalgamation of Active Shape Models (ASM) and ACM into a technique, that harnesses the powers of both, is applied to locating the left ventricular boundary in echocardiographic images. The third part of the Thesis discusses the development of two methodological extensions for spatio-temporal image analysis: Optical flow-based contour deformations, applied to contrast agent tracking in echocardiographic image sequences, and deformable spatio-temporal shape models for extending 2D ASM to 2D+time. The fourth part describes the use of a new Hierarchical Regional Principal Component Analysis, and presents two methods for interactive and learned, localized and multiscale, controlled shape deformation: medial-based shape profiles and physics-based shape deformations. In the final part of the Thesis, we develop Deformable Organisms: a robust decision-making framework for MIA that combines bottom-up, data-driven deformable models with top-down, knowledge-driven processes in a layered fashion inspired by Artificial Life modeling concepts. We present different segmentation and labeling examples of various anatomical structures from medical images and conclude that deformable organisms represent a promising new paradigm for MIA.

Keywords

Medical image analysis, segmentation, deformable models, shape modeling, shape deformation, physics-based modeling, artificial life, spatio-temporal shape analysis, statistical shape variation, principal component analysis, active contour models, snakes, active shape models, optical flow, dynamic programming, echocardiography, magnetic resonance imaging, digital color images, oral lesions, medial axis, spring-mass model, hierarchical regional principal component analysis, deformable organisms, deformable spatio-temporal shape models, medial-based shape profiles.

Contents

Abstract	i
Contents	iii
Acknowledgement	vii
Abbreviations and Acronyms	ix
Chapter 1. Introduction	1
1.1 Medical Imaging.....	1
1.2 Medical Image Analysis.....	5
1.2.1 Medical Image Segmentation.....	6
1.2.2 Deformable Models.....	7
1.2.3 Brief Overview of Deformable Models for MIA.....	8
1.2.4 Statistical Prior Knowledge of Shape.....	10
1.3 Thesis Outline and Contributions.....	11
1.4 Auto-bibliography.....	13
Chapter 2. Oral Lesion Detection in Color Images	17
2.1 Introduction.....	17
2.2 Methods.....	19
2.2.1 ACM with Adaptive Inflation Reversal and Damping.....	19
2.2.2 Single Band Generation from Color Images.....	19
2.2.3 Comparing Single bands.....	22
2.3 Results.....	22
2.4 Conclusion.....	25
Chapter 3. Statistically Constrained Snakes	27
3.1 Introduction.....	27
3.2 Methods.....	28
3.2.1 Overview.....	28
3.2.2 Representing Contours by DCT Coefficients.....	29
3.2.3 Principal Component Analysis.....	30
3.2.4 Constraining Contour Deformation.....	30
3.3 Results.....	31
3.4 Conclusion.....	36
Chapter 4. Optical Flow Snake Forces	37
4.1 Introduction.....	37

4.1.1	Clinical Motivation	37
4.1.2	Medical Imaging Procedure	37
4.1.3	Contrast Echocardiography and Videodensitometry.....	38
4.2	Tracking the Contrast Front	42
4.2.1	Optical Flow	43
4.2.2	Optical Flow Snake Forces	44
4.3	Results	44
4.4	Conclusion.....	49
Chapter 5.	Deformable Spatio-Temporal Shape Models	51
5.1	Introduction	51
5.2	Methods.....	54
5.2.1	Statistical ST-Shape Variation	54
5.2.2	Gray-Level Training.....	58
5.2.3	ST-Shape Segmentation Algorithm	59
5.3	Results	66
5.3.1	Synthetic ST-Shapes and Frame Sequences	66
5.3.2	Frame Sequence Imperfections	68
5.3.3	Gray-Level Training.....	69
5.3.4	Results on Synthetic Data	69
5.3.5	Results on Real Data	80
5.4	Conclusion.....	80
Chapter 6.	Controlled Shape Deformation via Medial Profiles	83
6.1	Introduction	83
6.2	Shape Representation and Deformation via Medial Profiles	85
6.2.1	Medial Profiles for Shape Representation.....	85
6.2.2	Shape Reconstruction from Medial Profiles	87
6.2.3	Shape Deformation using Medial-Based Operators.....	88
6.2.4	Statistical Shape Analysis by Hierarchical Regional PCA	91
6.2.5	Combining Operator- and Statistics-Based Deformations	93
6.3	Application and Results	93
6.4	Conclusion.....	93
Chapter 7.	Physics-Based Shape Deformation	97
7.1	Introduction	97
7.2	The Dynamic Mesh Model.....	98
7.3	Shape Deformation.....	100
7.3.1	Deformation using External Forces	100
7.3.2	Deformation using Spring Actuation	100
7.3.3	Affine Transformations.....	106
7.4	Hierarchical Regional PCA.....	107
7.5	Mesh Generation from Real Data	110

7.6	Conclusion.....	111
Chapter 8.	Deformable Organisms: An ALife Approach to MIA	113
8.1	Introduction	113
8.1.1	ALife for Computer Graphics	116
8.1.2	An ALife Modeling Paradigm for MIA: Motivation	118
8.2	Deformable Organisms for Medical Image Analysis.....	122
8.2.1	Shape Representation: Geometry	123
8.2.2	Motor System	123
8.2.3	Perception System	124
8.2.4	Behavioral/Cognitive System.....	125
8.3	Results	125
8.4	Conclusion.....	136
Chapter 9.	Future Research	139
9.1	Specific Improvements.....	139
9.2	An ALife Paradigm for Medical Image Analysis	142
9.3	Towards Intelligent Deformable Models for MIA.....	142
9.3.1	Knowledge and Knowledge Representation	143
9.3.2	Deformable ‘Anatomical’ Models.....	143
9.3.3	Use of Knowledge	144
Appendices		147
Appendix A.	Deformable Models.....	149
Appendix B.	Active Shape Models	157
Appendix C.	Principal Component Analysis.....	159
Appendix D.	Aligning 2D Shapes	161
Appendix E.	Aligning Spatio-Temporal Shapes	165
Appendix F.	Skeleton Pruning for Medial Axis Extraction.....	169
Appendix G.	Physics-Based Shape Deformation Tool.....	171
Bibliography		179

Acknowledgement

Many people have played important roles during the course of my doctoral studies. It is time to reflect on those whose support and encouragement were essential components for a successful completion of this thesis.

I am grateful to my supervisor Prof. Tomas Gustavsson for introducing me to the field of (medical) image analysis and for his continuous supervision, encouragement, and support throughout the years.

I am also fortunate for the time I spent at the Vision Group, Computer Science Department, University of Toronto. I owe Prof. Demetri Terzopoulos and Prof. Tim McInerney a great deal of gratitude for their supervision and assistance, and for being a source of scientific inspiration.

I wish to thank Dr. Tim Cootes (Imaging Science and Biomedical Engineering/Wolfson Image Analysis Unit, University of Manchester) for his helpful remarks during our discussions on Active Shape Models.

Special thanks to Dr. Martha Shenton (Harvard Medical School) and to Dr. Erik Houltz (Department of Anesthesia and Intensive Care, Sahlgrenska University Hospital) for their help in preparing and providing us with medical image data.

I also wish to acknowledge other collaborators and co-authors: Dr. Marie Beckman Suurkula (Department of Clinical Physiology, Sahlgrenska University Hospital) and sonographer Carina Olausson. Dr. Ann Hellström (Eye Clinic, Paediatric Ophthalmology, Sahlgrenska University Hospital), Nina Lundberg, and Mohammad Afshar-Beroz (Department of Informatics, Göteborg University).

I am also grateful to Prof. Mats Viberg; an endless source of scientific guidance and motivation.

All the members of the Image Analysis Group deserve special recognition for the friendly atmosphere, supportive environment, and scientific assistance they provided.

I am greatly indebted to my funding programme VISIT (Visual Information and Technology) supported by the Swedish Foundation for Strategic Research.

I appreciate the administrative assistance offered by Gunnel Edström and Stina Lindqvist and the technical support of the system administrators Lars Jansson, Lars Kollberg, Berndt Andersson, Lars Börjesson, and Hans Wahlström.

Thanks to my friends in Sweden, Sorour, Harald, Sabine, Jonas, Ziad,...

Finally, I would like to show my deepest appreciation to my Parents; Laurice and Saleh, my family, and my wife Rafeef for their unparalleled love and support.

Abbreviations and Acronyms

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
AAM	Active Appearance Model
ACM	Active Contour Model
AI	Artificial Intelligence
ALife	Artificial Life
ARVD	Arrhythmogenic Right Ventricular Dysplasia
ASD	Allowable Shape Domain
ASM	Active Shape Model
ASTSD	Allowable Spatio-Temporal Shape Domain
CAD	Computer Aided Diagnosis
CAT	Computed Axial (Computer Assisted) Tomography
CC	Corpus Callosum
CN	Caudate Nucleus
CT	Computed Tomography
DCT	Discrete Cosine Transform
DWT	Discrete Wavelet Transform
DM	Deformable Model
DSA	Digital Subtraction Angiography
DSTS	Deformable Spatio-Temporal Shape
FEM	Finite Element Method
FSPD	Fourier Series of Polar Development
GA	Genetic Algorithms
GVF	Gradient Vector Field
HCI	Human-Computer Interaction
HRPCA	Hierarchical Regional Principal Component Analysis
HSI	Hue-Saturation-Intensity
IDCT	Inverse Discrete Cosine Transform
IDM	Intelligent Deformable Model
LV	Left Ventricle
MC	Motor Controller
MIA	Medical Image Analysis
MRI	Magnetic Resonance Imaging
OF	Optical Flow
PB	Physics-Based

PCA	Principal Component Analysis
PDM	Point Distribution Model
PET	Positron Emission Tomography
RF	Radio Frequency
RGB	Red-Green-Blue
RL	Reinforcement Learning
ROI	Region Of Interest
RV	Right Ventricle
RVIT	Right Ventricular Inflow Tract
SNR	Signal to Noise Ratio
SO	Sensory Organ
SPECT	Single Photon Emission Computed Tomography
ST	Spatio-Temporal
SVD	Singular Value Decomposition

Chapter 1. INTRODUCTION

This Chapter introduces the reader to medical imaging, medical image analysis, and applications thereof. Medical image segmentation using deformable models and models incorporating prior knowledge of anatomical shapes are emphasized. Additionally, this Chapter presents a summary of the contributions and contents of this Thesis.

1.1 Medical Imaging

The advancements in medical imaging over the past decades are enabling physicians to non-invasively peer inside the human body for the purpose of diagnosis and therapy. With the advent of medical imaging modalities that provide different measures of internal anatomical structure and function, physicians are now able to perform typical clinical tasks such as patient diagnosis and monitoring more safely and effectively than before such imaging technologies existed. Applications of imaging in medicine include computer-aided diagnosis (CAD), image guided therapy and therapy evaluation, computer assisted intervention, surgical simulation, planning, and navigation, medical telepresence and telesurgery, functional brain mapping, etc. Evidently, this introduction of a number of advanced internal, in vivo medical imaging technologies, which allow for the acquisition of high-resolution cross-sectional images of the human body, has significantly improved the quality of medical care available to patients. Short descriptions of some of the common modalities follow.

Planar (2D) X-ray images, as in mammography and chest X rays, are projection (shadow) images of a patient's 3D region of interest. The images are produced from X rays passing through the patient's body tissues and attenuated according to the varying tissue densities (Figure 1.1).

Computed Tomography (CT) or Computed Axial (Computer Assisted) Tomography (CAT) is based on the same principle as conventional X-ray radiography. However, stacks of axial slices or mathematically reconstructed volume (3D) images are produced. X-ray based imaging is useful for the investigation of bone structure and fat tissue. For adequate acquisition of soft tissue images, invasive contrast agents are required which cause allergic reactions in some patients (Figure 1.2).

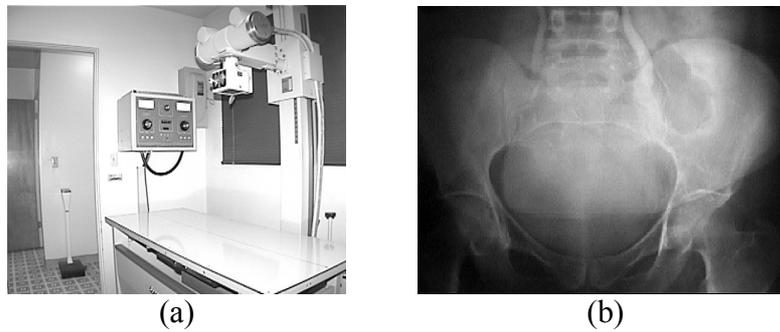


Figure 1.1. (a) X-ray scanner. (b) Pelvis X-ray.

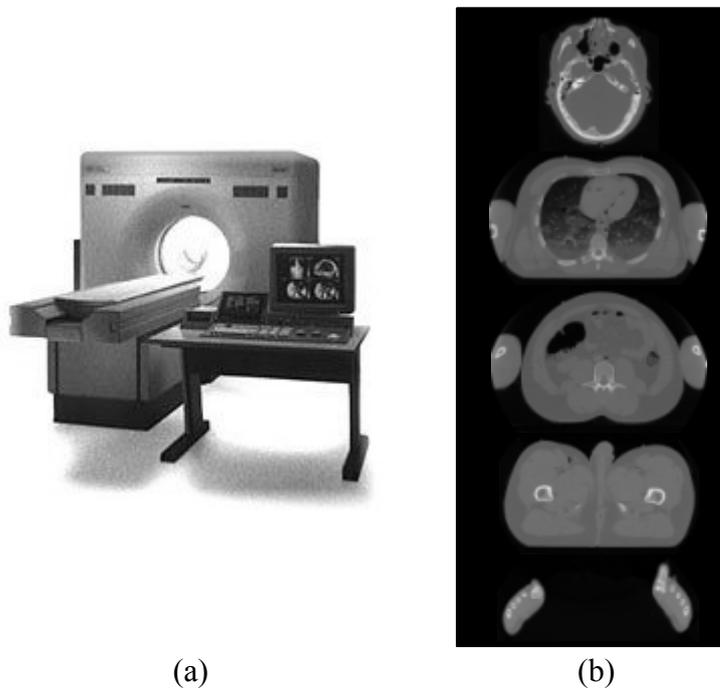


Figure 1.2. (a) CT scanner. (b) Male CT data from the Visible Human Project® [VIS] (from top to bottom): Head, thorax, abdomen, pelvis, and feet.

Magnetic Resonance Imaging (MRI) is based on the principal of resonance (the absorption of energy from a source at a particular frequency, the resonant or natural frequency). In MRI, Radio Frequency (RF) pulses modify the net magnetization of groups of protons (hydrogen nuclei) while in an external magnetic field. An MR signal is the RF energy released when nuclei return to their original state. Different characteristics of the emitted MR signal, along with spatial localization procedures via external magnetic field gradients,

are used to produce images of tissue hydrogen concentration that reflect the different structures imaged. MRI is noninvasive, provides high-resolution images, and the use of radio waves is much safer than imaging using X rays. However it is an expensive procedure with typically longer scanning time than CT, during which the patient should ideally lie motionless inside a narrow tube. Magnetic resonance angiography, a specific type of MRI, is used to produce an image of blood flow for the visualization of arteries and veins (Figure 1.3).

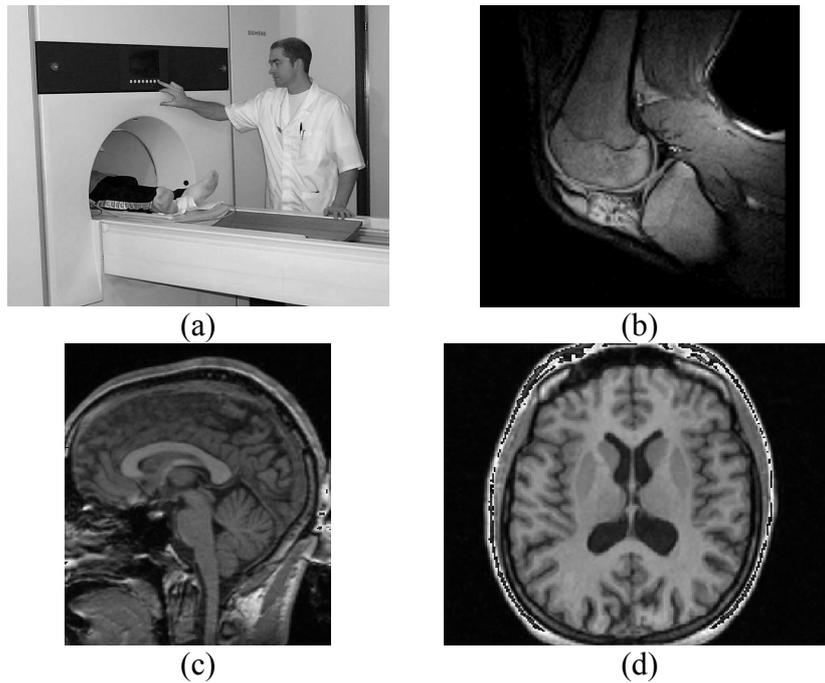


Figure 1.3. (a) MRI scanner. (b) Knee MRI. (c) Sagittal and (d) transversal MRI brain slices (see Figure 1.4).

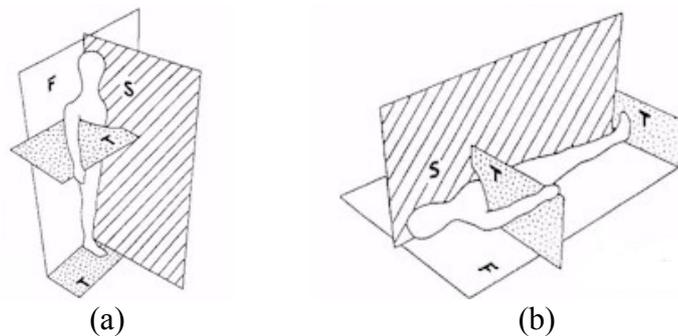


Figure 1.4. Cardinal planes of the body: Sagittal (S), frontal (F), and transversal (T). (b) Planes in supine position.

Digital Subtraction Angiography (DSA) produces images of a patient's blood vessels as the difference image between a post- and a pre-contrast injection images. Since the contrast medium injected flows only in the vessels, the image data arising from other structures does not change in the two images and are eliminated by the subtraction (Figure 1.5).

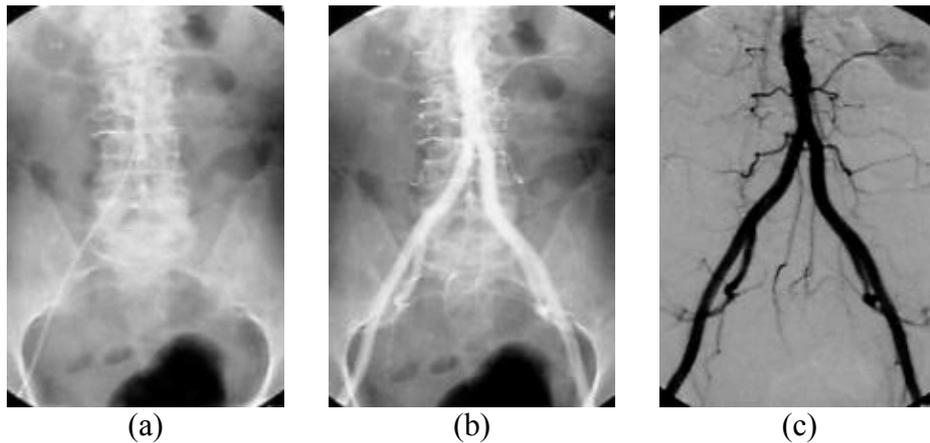


Figure 1.5. Digital subtraction angiography. (a) Before and (b) after contrast injection. (c) Image of vessels after subtraction.

Ultrasound imaging (such as B-mode and Doppler) uses pulsed or continuous high-frequency sound waves to image internal structures by recording the different reflecting signals. Among others, ultrasound imaging is used in echocardiography for studying heart function and in prenatal assessment. Although ultrasonographic images are typically not high-resolution as images obtained through CT or MRI, they are widely adopted because of ultrasound's invasiveness, cost effectiveness, acquisition speed, and harmlessness (Figure 1.6).



Figure 1.6. (a) Ultrasound examination. (b) B-mode ultrasound image of the carotid artery.

Nuclear medicine acquisition methods such as Single Photon Emission Computed Tomography (SPECT), and Positron Emission Tomography (PET) are functional imaging techniques. They use radioactive isotopes to localize the physiological and pathological processes rather than anatomic information (Figure 1.7).

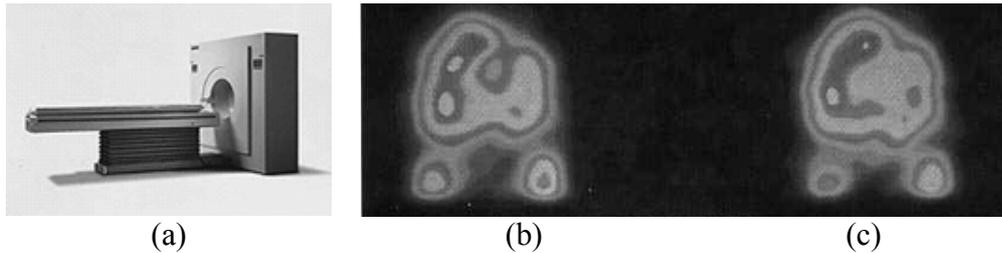


Figure 1.7. (a) PET scanner. Example PET brain scans (b) before and (c) after therapy.

1.2 Medical Image Analysis

Medical imaging is an important source of anatomical and functional information and is indispensable for the diagnosis and treatment of disease. However, huge amounts of high-resolution three-dimensional spatial and temporal data cannot be effectively processed and utilized with traditional visualization techniques. It is generally insufficient or inefficient for physicians to only visually inspect the medical image data collected from MR, CT, PET and other modalities. The role of medical imaging is expanding and the medical image analysis community has become preoccupied with the challenging problem of creating quantification algorithms that make full use of the information in the flood of image data.

Among the primary tasks of medical image analysis are image segmentation, registration, and matching. Medical image analysis directly impacts applications such as image data fusion, quantitative and time series analysis, biomechanical modeling, generating anatomical atlases, visualization, virtual and augmented reality, instrument and patient localization and tracking, etc. Medical images, for example, are analyzed to ascertain the detailed shape and organization of anatomic structures, in an effort to enable a surgeon to preoperatively plan an optimal approach to some target structure. Medical images can also be analyzed for examining relationships between structural abnormalities and deformations and certain functional abnormalities and diseases. In radiotherapy, medical image analysis is crucial for allowing the delivery of a necrotic dose of radiation to a tumor with minimal collateral damage to healthy tissue.

The reader is referred to [Duncan2000] for a recent overview of the medical image analysis field. See also, for example, [Ayache1995, Pun1993] for general overview articles related to medical image computing.

1.2.1 Medical Image Segmentation

Segmentation is nearly always the crux of any problem in computer assisted medical image computing. Segmenting an anatomical structure in a medical image amounts to identifying the region or boundary in the image corresponding to the desired structure. In the classical approach of segmentation by image labeling, image features are extracted and used to obtain a sparse collection of locations and data, which are then interpolated to form a representation and possible segmentation. Desired regions are identified by labeling each volume element (voxel) in a 3D scan, or picture element (pixel) in 2D, based on the anatomical structure to which it corresponds. In more recent approaches, an initial curve or surface estimate of the structure boundary is provided and optimization methods are used to refine the initial estimate based on image data.

A fully segmented 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. Detailed segmentation and subsequent 3D models can be used to generate an anatomical atlas for visualization, teaching, and as training data for other algorithms. Segmentation is beneficial when applied to image data of both patients with pathology and normal volunteers. Scans of people without pathological abnormalities can be used as a method for comparison to define abnormality.

The output of manual segmentation of medical images, by knowledgeable medical experts, can sometimes be considered optimal. Unfortunately, expert segmentation is far from recommended in many clinical situations. For example, in manual segmenting a structure in a three-dimensional volume data, experts cannot visualize the entire volume simultaneously and typically resort to outlining the structure of interest manually in a series of consecutive two-dimensional slices of the original 3D volume. This slice-by-slice segmentation suffers from errors due to the difficulty in maintaining consistency across slices. Furthermore, manual tracing of object boundaries generally suffers from poor reproducibility of results (inter- and intra-operator variability). It is also tedious and time consuming thus becoming questionable given the large number of data sets usually required. Naturally, segmentation is performed automatically whenever possible. Most applications still require at least some amount of manual intervention and some are performed completely manually.

Although exceptional views of internal anatomy can be provided by modern medical imaging devices, efficient computer-assisted analyses of

internal anatomy that produce accurate results is limited. Accurate, repeatable, quantitative data must be efficiently extracted in order to support the spectrum of biomedical investigations and clinical activities, from diagnosis, to radiotherapy, to surgery. Medical imagery may be exceptional, but it is far from being ideal. The shortcomings typical of sampled data, such as sampling artifacts, spatial aliasing, and noise are a cause for the less than perfect performance of current medical image segmentation tools. Additionally, the similar appearance of different tissue in images, the shape complexity and variability of anatomical shapes, and the appearance of structure boundaries as indistinct and disconnected, together render accurate and efficient segmentation tools difficult to obtain.

Subsequent analysis and interpretation of segmented objects is hindered by voxel-level (or pixel-level in 2D) structure representations, generated by most traditional low-level image processing techniques. Low-level segmentation techniques that consider only local information can make incorrect assumptions during the integration process and generate infeasible object boundaries. These model-free techniques usually require considerable amounts of expert intervention. The challenge is to extract boundary elements belonging to the same structure and integrate these elements into a coherent, consistent, and compact model representation of the structure.

1.2.2 Deformable Models

Although segmenting objects in high contrast, noise-free images can be done with simple low-level techniques, problems do arise when medical images are corrupted with noise and the structure itself is not clearly or completely visible in the image. This may result in detecting erroneous object regions or boundaries, or failing to detect true ones. Furthermore, in medical applications the structures to be analyzed, segmented, or tracked are generally anatomical structures that are natural (not man-made), non-rigid, and usually dynamic; changing their shape in time and/or between observations. To analyze such noisy images and to provide a coherent representation for variable structure shapes, deformable models were introduced, with some ideas dating back to the early 70's (rubber mask technique [Widrow1973] and spring-loaded templates [Fischler1973]). Deformable models are curves or surfaces defined within an image domain. They are designed to be attracted to external image features (such as edges) while maintaining internal shape constraints (such as smoothness), thus progressively changing their shape in an effort to locate a desired structure in the image. By constraining the extracted boundaries of the target object shape to be smooth, and by incorporating other prior information about the object shape, deformable models offer robustness to both image noise and boundary gaps. Deformable models allow integrating boundary elements into a coherent and consistent mathematical description readily available for

subsequent applications. Furthermore, deformable models can be implemented on the continuum and achieve subpixel accuracy, a highly desirable property for medical imaging applications.

Snakes or Active Contour Models (ACM) [Terzopoulos1987, Kass1987], the seminal work on deformable models, has attracted the most attention and has been widely used for segmenting non-rigid objects in 2D in a wide range of applications. The mathematical formulation of snakes, dynamic deformable models, numerical simulation, and probabilistic deformable models are treated in Appendix A.

The work presented in this Thesis, focuses on the development of deformable models and their application to medical image analysis.

1.2.3 Brief Overview of Deformable Models for MIA

Current research on deformable models for medical image analysis is extensive. Many variations, extension, and alternative formulations appeared since the introduction of snakes. For general reviews the reader is referred, for example, to [Terzopoulos1988, McInerney1996, Gibson1997, Singh1998a, Blake1998, Xu2000]. The following paragraphs summarize some of these advancements.

Different methods that proposed additional energy (or force) terms were reported, mainly to increase the capture range of deformable models. For example, in [Cohen1991] an inflation force is incorporated and the contour curve is treated as a balloon that is inflated in order to avoid local minima solutions, i.e. the curve passes over weak edges and is stopped only if the edge is strong. [Xu1998] proposed Gradient Vector Fields (GVF). GVF extend the gradient map farther away from the edges into homogenous regions using a computational diffusion process. The attraction potential can also be defined through the use of Chamfer distance to edge points [Borgefors1984]. Additionally, a scale-space implementation was originally suggested in [Kass1987] where the snake is allowed to come to equilibrium on a very blurry energy function and then slowly reduce the blurring. The use of medial-ness (or medial axis-related features and energy terms) was also proposed [Pizer1999]. This has the effect of increasing the capture range and reducing the models' sensitivity to initializations. In [Gunn1997] (see also [Gunn1994]) a dual active contour model (or dual snake) that overcomes the primary problems of sensitivity due to initialization was presented. In dual snakes, two inter-linked contours are used, one expanding from inside the target and the other contracting from the outside, until locking onto the object.

Different physics-based formulations were reported. [Terzopoulos1991] proposed Deformable Superquadrics and others proposed Finite Element Methods (FEM) formulations [Pentland1991, Cohen1993].

Methods for dealing with topological changes also appeared in the literature such as the topologically adaptable snakes [McInerney2000], surfaces [McInerney1999], and meshes [Lachaud1999].

Hierarchically organized models, which shift their focus from structures associated with stable image features to those associated with less stable features, were also reported [McInerney1998, Shen2000].

Different numerical methods for model parameter optimization were introduced, including the use of dynamic programming [Amini1990, Gunn1996] (as in the “live-wire” technique [Mortenssen1992], which has been incorporated into united snakes [Liang1999a, Liang1999b]), simulated annealing [Ruekert1995, Grzeszczuk1997], genetic algorithms [MacEachern1998, Ballerini1998], and Bayesian frameworks [Storvik1994].

Motion tracking using deformable models has been used for tracking non-rigid structures, such as blood cells [Laymarie1993]. Much attention has also been given to tracking the left ventricle in both 2D and 3D [Singh1993, McInerney1995]. Using a B-spline active contour model, [Stark1996] tracked the silhouette of 3D object in 2D, using Kalman filtering in conjunction with a 3D object pose tracker and an underlying 3D geometric model. [Curwen1994] used a Kalman B-spline snake model to track coronary vessel motion using a linear motion model. In ‘Kalman snakes’ [Terzopoulos1992], the contour’s motion equation is used to describe the expected evolution of the contour’s shape parameters, i.e. a time varying prior (see Appendix A).

Different shape representations for deformable models were also adopted. For example [Rueckert1995] proposed an adaptive spline model. The accuracy of the model is gradually increased during segmentation by inserting new control points yielding faster and more efficient computations. [Menet1990] introduced B-spline snakes (B-snakes) and deformable models based on elliptic Fourier descriptors were proposed by [Staib1992]. Fourier coefficients obtained from a Fourier series of polar development (FSPD) were also proposed [Bonciu1998]. Level-set and minimal path techniques for finding the global minimum of active contour models were also formulated [Cohen1996]. [Caselles1995] introduced Geodesic Active Contours and [Leventon2000] proposed an extension that incorporates statistical shape information. [Szekely1996] developed Fourier contours with constrained elastic deformation, and [Lobregt1995] formulated a polygonal or discrete dynamic contour model. Wavelets-based deformable contours were also reported [Yoshida1997]. [Delingette1999] developed deformable simplex meshes and [Hug1999] introduced ‘Tamed snakes’, particle-based snakes with adaptive subdivision. Loop free snakes were also proposed [Ji1999]. Furthermore, medial-based deformable models have been recently investigated [Fritsch1997, Pizer1998, Pizer1999, Pizer2000, Joshi2001].

Solving the registration and matching problems that usually follow segmentation has resulted in methods that simultaneously determine the object

boundary and the spatial correspondence between similar structures of different subjects or anatomical atlases [Wang1998, Wang2000, Cootes1995a]. Additionally, there has been a great deal of work on multi-modality image fusion, warping, and registration in a deformable anatomy context. Some methods, for example, are based on the maximization of mutual information [Viola1997, Pluim2000] or on elastic [Kelemen1999] and fluid deformations [Christensen1996]. For more information on registration techniques, the reader is referred to survey papers on medical image registration, including [Antoine1998, Audette2000]. It's worthwhile mentioning that non-rigid deformation methods [Sederberg1986, Bookstein1989, Little1996, MacCracken1996, Moccozet1997, Singh1998b] are valuable tools for many of the medical image registration techniques.

The incorporation of statistical prior knowledge in deformable models has also attracted much attention and research. We devote the following section (Section 1.2.4) for discussing this topic emphasizing on Active Shape Models [Cootes1995a] and related work.

1.2.4 Statistical Prior Knowledge of Shape

The original snakes formulation may be too general to give acceptable results when dealing with images where shape and appearance abnormalities are present due to occlusions, closely located but irrelevant structures, or noise. This led to several techniques that utilize prior knowledge of object shape for segmentation, pioneered by the work on Active Shape Models (ASM) [Cootes1995a]. While introducing a priori knowledge generally improves the segmentation results, nevertheless, the model will require training and thus becomes less general.

ASM is a deformable shape modeling technique that is used for segmentation of objects in digital images and has been used for locating anatomical structures in medical images [Cootes1993, Cootes1994b, Cootes1995c]. In ASM the statistical variation of shapes pertaining to a specific class of objects is modeled beforehand from a training set. An initial model guess is then applied and the model is allowed to deform according to image data. Proposed deformations, which are chosen to minimize a certain energy (cost) function, are constrained to be consistent with the prior knowledge about the target object. The energy function is chosen in a way that the model will be attracted to certain image features extracted from the intensity (or gray-level) values of the image. Appendix B describes the steps involved in ASM in more detail.

Several enhancement and additions to the basic ASM method were developed. An automatic landmark generation algorithm was proposed in [Hill1994, Hill1997, Hill2000]. A multi-resolution implementation of ASM was presented in [Cootes1994a]. [Hill1992] suggested to tackle the ASM

parameter optimization problem via genetic algorithms. [Baumberg1994, Lanitis1994b] used Kalman filtering for tracking in an ASM framework. [Cootes1995b] formulated the combination of physical vibrational modes and statistical variational modes. [Hill1993] applied ASM to 3D data. [Cootes1997] presented a method for kernel-based estimation of the shape density distribution. The use of ASM for classification (face recognition) was investigated in [Lanitis1994a, Lanitis1995, Lanitis1997]. Combined Appearance Models and Active Appearance Models (AAM)¹ were introduced in [Cootes1998, Cootes1999]. View-Based Active Appearance Models, in which a set of statistical models is built for several distinct view points, were introduced [Cootes2000a] and used for tracking. More recently, [Cootes2000b] presented the work on combining elastic and statistical models of appearance variation.

To model the non-linearity that can be present in point distribution models, non-linear statistical models were proposed [Bowden2000]. [Sozou1994] attacked this problem by fitting a high order polynomial to the non-linear axis of the training set. [Sozou1995] modeled the non-linearity via back propagation neural networks. [Bowden1997] approximated the non-linearity by a combination of multiple smaller linear models. [Chennubhotla2001] proposed Sparse PCA, a modified version of PCA, as a means to trade off the correlation among coefficients for sparsity.

1.3 Thesis Outline and Contributions

In this section we introduce the contributions of the Thesis to the field of deformable models for Medical Image Analysis (MIA). The paragraphs below present our work in a general context and point to different chapters within the Thesis where the reader can find detailed information about specific contributions. Note that minor overlap in the chapters may be noticed since they were written to be self-contained.

Active contour models gained large acceptance within the medical image analysis community and their use covered a wide range of applications. In Chapter Two of this Thesis, a modified version of snakes, which uses adaptive inflation reversal and damping, is applied to the problem of detecting oral lesions in digital color images.

Many extensions to the original snakes formulation were developed. Among the most notable are the incorporation of additional energy or force terms. In Chapter Four, an additional optical flow-based force is introduced and utilized for tracking the leading edge of injected contrast agent in an echocardiographic image sequence.

¹ ‘Active Blobs’, an approach similar to AAM, was presented in [Scarloff1998].

Another important development to the original active contour formulation is the utilization of prior knowledge. The leading work on including statistical prior knowledge is Cootes' Point Distribution Models (PDM) and Active Shape Models (ASM). In Chapter Five the reader is introduced to our work on extending 2D ASM to 2D+time.

A necessary nuisance for generating PDM is the need for labeling a training-set of images with spatial correspondence. In Chapter Three we present our contribution to overcoming the spatial labeling by means of performing the statistical analysis in the frequency domain.

The main modes of variation captured by a PDM via Principal Component Analysis (PCA) are spatially global. Changing the weight of a single variational mode of the PDM generally causes the whole shape to change. In Chapter Six and Chapter Seven we use our Hierarchical Regional PCA as a means for performing multiscale and spatially localized learned shape deformations.

A substantial amount of knowledge is often available about structures of interest. However, the use of high-level contextual knowledge in current deformable models is either largely ineffective because it is intertwined much too tightly with the low-level optimization, or non-automatic relying on the knowledgeable users' interaction. Chapter Eight presents Deformable Organisms, a novel approach for MIA, which incorporates a higher-level cognitive layer on top of the original physics and geometry layers of traditional deformable models. Deformable Organisms are architected in an artificial-life modeling framework.

Since high-level segmentation strategies (of deformable organisms or other) eventually need to trigger low-level geometrical and physical shape deformations, methods that provide controlled deformation 'handles' are needed. In Chapter Six and Chapter Seven we present a geometry- and a physics-based controlled shape deformation methods, respectively, to be used for such lower level layers.

Chapter Nine presents a future outlook. The Thesis also includes seven appendices dealing with the mathematical formulation of deformable models (Appendix A), active shape models (Appendix B), principal component analysis (Appendix C), 2D and spatio-temporal shape alignment (Appendix D and Appendix E), a proposed pruning algorithm (Appendix F), and details of a physics-based shape deformation tool that we developed (Appendix G).

1.4 Auto-bibliography

This Thesis is based primarily on the following publications:

Chapter 2.

G. Hamarneh, A. Chodorowski, T. Gustavsson. "Active Contour Models: Application to Oral Lesion Detection in Color Images". *IEEE Proceedings of the International Conference on Systems, Man, and Cybernetics*. Vol. 4, pp. 2458-2463, Nashville, Tennessee, USA, October 8-11, 2000.

Chapter 3.

G. Hamarneh, T. Gustavsson. "Combining Snakes and Active Shape Models for Segmenting the Human Left Ventricle in Echocardiographic Images". *IEEE Proceedings on Computers in Cardiology*. Vol. 27, pp. 115-118, Cambridge, USA, September 24-27, 2000.

A related version appears as:

G. Hamarneh, T. Gustavsson. "Statistically Constrained Snake Deformations". *IEEE Proceedings of the International Conference on Systems, Man, and Cybernetics*. Vol. 3, pp. 1610-1615, Nashville, Tennessee, USA, October 8-11, 2000.

Chapter 4.

K. Althoff, G. Hamarneh, T. Gustavsson. "Tracking Contrast in Echocardiography by a Combined Snake and Optical Flow Technique". *IEEE Proceedings on Computers in Cardiology*. Vol. 27, pp. 29-32, Cambridge, USA, September 24-27, 2000.

An earlier version appeared as:

G. Hamarneh, K. Althoff, T. Gustavsson. "Snake Deformations Based on Optical Flow Forces for Contrast Agent Tracking in Echocardiography". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 2000, pp. 45-48, Halmstad, Sweden, March 7-8, 2000.

Chapter 5.

G. Hamarneh, T. Gustavsson. "Deformable Spatio-Temporal Shape Models: Extending ASM to 2D+Time". *Proceedings of the British Machine Vision Conference*. BMVC 2001, Vol. 1, pp. 13-22, Manchester, UK, September 10-13, 2001.

Chapter 6.

G. Hamarneh, T. McInerney. "Controlled Shape Deformations via Medial Profiles". *Proceedings of Vision Interface*. VI 2001, pp. 252-258, Ottawa, Canada, June 7-9, 2001.

Chapter 7.

G. Hamarneh, T. McInerney. "Physics-Based Shape Deformations for Medical Image Analysis". Technical report CSRG-436, Department of Computer Science, University of Toronto, 2001, <ftp://ftp.cs.toronto.edu/cs/ftp/csrq-technical-reports/>.

Chapter 8.

G. Hamarneh, T. McInerney, D. Terzopoulos. "Deformable Organisms for Automatic Medical Image Analysis". *Proceedings of the Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention*. MICCAI 2001, Utrecht, The Netherlands, October 14-17, 2001.

Other related publications:

1. G. Hamarneh, R. Abu-Gharbieh, T. Gustavsson. "Review - Active Shape Models - Part I: Modeling Shape and Gray Level Variation". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 1998, pp. 125-128, Uppsala, Sweden, March 16-17, 1998.
2. G. Hamarneh, T. Gustavsson. "A Method for Modeling and Segmentation of Spatio-Temporal Shapes". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 1999, pp. 49-52, Göteborg, Sweden, March 9-10, 1999.
3. G. Hamarneh, T. Gustavsson. "Constraining Contour Deformations Using Statistical A Priori Knowledge of Shape Without Requiring Point-to-Point Correspondence". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 2000, pp. 33-36, Halmstad, Sweden, March 7-8, 2000.
4. G. Hamarneh, T. McInerney, D. Terzopoulos. "Intelligent Deformable Organisms: An Artificial Life Approach to Medical Image Analysis", Technical report CSRG-432, Department of Computer Science, University of Toronto, 2001, <ftp://ftp.cs.toronto.edu/cs/ftp/csr-g-technical-reports/>.
5. G. Hamarneh. "Digital Image Analysis of Fundus Photographs on the WWW". Technical Report R002/1999 (S2-IAG-99-1), Department of Signals and Systems, Chalmers University of Technology, Sweden, February 1999.
6. G. Hamarneh. "Active Shape Models, Modeling Shape Variations and Gray Level Information and an Application to Image Search and Classification". Technical Report R005/1998 (S2-IAG-98-1), Department of Signals and Systems, Chalmers University of Technology, Sweden, September 1998.
7. G. Hamarneh. *Deformable Spatio-Temporal Shape Modeling*. Technical report 311L, Licentiate Thesis, Department of Signals and Systems, Chalmers University of Technology, 1999.

8. G. Hamarneh, "Implementation and Comparison of Four Different Boundary Detection Algorithms for Quantitative Ultrasonic Measurements of the Human Carotid Artery". Masters thesis, Technical report EX0NA/1996, Department of Applied Electronics, Chalmers University of Technology, December 1996.
9. G. Hamarneh. "Image Segmentation with Constrained Snakes". *Swedish society for image analysis newsletter*. SSABlaskan Nr. 8, pp. 5, 2000.
10. T. Gustavsson, R. Abu-Gharbieh, G. Hamarneh, Q. Liang. "Implementation and Comparison of Four Different Boundary Detection Algorithms for Quantitative Ultrasonic Measurements of the Human Carotid Artery". *IEEE Proceedings on Computers in Cardiology*. Vol. 24, pp. 69-72, Lund, Sweden. September 1997.
11. R. Abu-Gharbieh, G. Hamarneh, T. Gustavsson, C. Kaminski. "Flame front tracking by laser induced fluorescence spectroscopy and advanced image analysis". *Journal of Optics Express*. Vol. 8(5), pp. 278-287, 2001.
12. R. Abu-Gharbieh, C. Kaminski, T. Gustavsson, G. Hamarneh. "Flame Front Matching and Tracking in PLIF Images Using Geodesic Paths and Level Sets". *IEEE Proceeding of the workshop on Variational and Level Set Methods in Computer Vision*. pp. 112-118, Vancouver, Canada, July 13, 2001.
13. R. Abu-Gharbieh, G. Hamarneh, T. Gustavsson. "Review - Active Shape Models - Part II: Image Search and Classification". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 1998, pp. 129-132. Uppsala, Sweden, March 16-17, 1998.

Chapter 2. ORAL LESION DETECTION IN COLOR IMAGES

This chapter¹ presents the application of active contour models (snakes) to the segmentation of oral lesions in medical color images acquired from the visual part of the light spectrum. The aim is to assist the clinical expert in locating potentially cancerous cases for further analysis (e.g. classification of cancerous vs. non-cancerous lesions). We apply a modified version of snakes, which uses adaptive inflation reversal and damping, to single-band images derived from the original color images. A number of different single-bands were evaluated including those resulting from the original and normalized RGB, perceptual HSI space, $I_1I_2I_3$, and the Fisher discriminant function. Examples of segmentation results of oral lesions are presented.

2.1 Introduction

The human oral mucosa is a site of a variety of disorders. Numerous diseases or lesions have been clinically classified [Pindborg1992]. In particular, there exist lesions that have a potential to develop into oral cancer. The American Cancer Society estimated 30,200 new cases and 7,800 deaths in the US in the year 2000 of oral cancer [ACS]. The preliminary diagnosis of oral disease is based on ocular inspection and registration of the patient's oral cavity as true-color digital images. Although complementary techniques exist, based e.g. on infrared or fluorescence spectroscopy [DhIngra1996], in clinical practice the decision about further treatment of the patient is predominantly based on lesion appearance from the visual part of light spectrum. The automatic detection (segmentation) of color images of the oral mucosa is thus an important part of computer-aided oral lesion diagnosis systems (CADx). It is of great interest for the medical community working with oral lesions to have an automatic (or semi-automatic) method for segmenting the lesions in true-color images, since by doing that the next step of extracting the different features and the consequent classification (examining the potentiality of a malignant cancerous lesion) can be immediately performed and evaluated. A previous study evaluated the classification of lesions based on different color features with the lesions being manually segmented by medical experts [Chodorowski1999, Chodorowski2000]. The oral specialists usually agree on the position of the

¹ This chapter is based primarily on [Hamarneh2000d].

lesion boundaries in the recorded images. However, this is still a challenging computer vision problem due to the shape and appearance variability of oral lesions. On the other hand, the machine is usually more efficient, after supervised learning, than humans in discrimination of different oral diseases. The automatic segmentation algorithm will simplify analysis of oral lesions and can be used in clinical practice to assist in the diagnosis of potentially cancerous lesions.

Currently our image database includes cases of two common oral lesions, the potentially cancerous lesions called leukoplakia and the usually harmless lesions called lichenoid reactions. Furthermore, the lichenoid reactions can be divided into atrophic, plaque-formed and reticular lesions. Thus the subsequent classification problem can be studied as a 2-class problem: cancerous vs. non-cancerous, or a 4-class problem: complete classification (Figure 2.1). Both of the lesion types appear reddish-whitish to the human observer and are not easily differentiated. From a clinical viewpoint the boundaries of the lesions form a closed contour with no gaps. Most of the research in this field arises from dermatology and skin cancer detection [Ercal1993, Round1997]. In contrast to skin lesions, the oral lesions are predominantly reddish and occupy a narrow band of hue-spectrum.

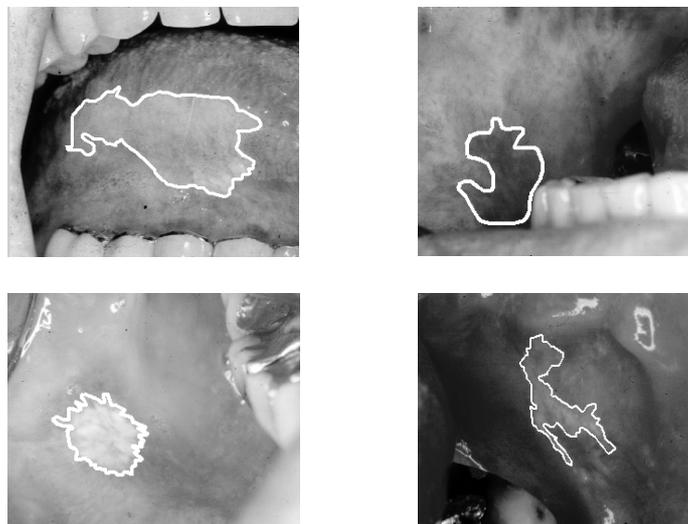


Figure 2.1. Examples of the four classes of oral lesions.

2.2 Methods

Most of the previous work on deformable models was directed towards scalar-valued (intensity or gray level) images. Nevertheless, attempts have been made to modify, extend, and/or apply snakes to detecting and segmenting objects in multi-band images [Sapiro1996, Chiou1996, Vandenbroucke1997, Zhu1996, Sobottka1996, Rasmussen1998]. For our application, we derived different single bands from the original multi-band (true-color, RGB) images and compared their suitability for semi-automatic detection of oral lesion boundaries. We used a modified version of snakes, including adaptive inflation reversal and damping.

2.2.1 ACM with Adaptive Inflation Reversal and Damping

Using a single value for the inflation force for all the nodes proved insufficient in our experiments. It caused the snake to ‘leak’ at regions where the snake nodes reached the target boundary earlier than others, since those regions were still being inflated. In order to dampen the inflation force when the snake nodes reach the target boundary, we associate a node-specific inflation weight $q_i(t)$ and the equation for updating the position of snake node i becomes (compare with equation (A.15) in Appendix A)

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) - \frac{\Delta t}{\gamma} (\alpha \mathbf{F}_i^{\text{tensile}}(t) + \beta \mathbf{F}_i^{\text{flexural}}(t) - \mathbf{F}_i^{\text{external}}(t) - q_i(t) \mathbf{F}_i^{\text{inflation}}(t)). \quad (2.1)$$

When a node reaches the target boundary the inflation direction is reversed (inflation becomes deflation and vice versa). If a certain number of inflation reversals occurred within a limited number of past iterations then the inflation force is dampened for this particular node only. We also implemented an adaptive resampling scheme, where the polygonal snake nodes are re-sampled based on the distance between nodes and the curvature along the snake. Our snakes implementation was also equipped with a facility that allows the user to place certain forced nodes on the target boundary through which the snake must pass.

2.2.2 Single Band Generation from Color Images

In order to apply the discussed ACM formulation on color images without using complex multi-band forces, we derive a number of single bands from the original three-band (RGB) images. We have investigated the use of the single bands shown in Table 2.1, seeking the band in which the detected edges are most pronounced and coincide with the true lesion boundaries.

Table 2.1. Single bands

<i>Color space*</i>	<i>Single bands</i>		
RGB	R: red	G: green	B: blue
HSI	H: hue	S: saturation	I: intensity
Normalized RGB	$R_n=R/(R+G+B)$	$G_n=G/(R+G+B)$	$B_n=B/(R+G+B)$
$I_1I_2I_3$	$I_1=(R+G+B)/3$	$I_2=R-B$	$I_3=(2G-R-B)/2$
Other	F: Fisher projection	M: modified Fisher projection	

* [Ledley1990, Ohta1980]

The different single bands generated from one example color image are shown in Figure 2.2. All single bands used are either linear or nonlinear transformations of the original RGB values to other color coordinates. However, the Fisher (F) and the modified Fisher (M) single bands require training. In order to generate the F and M single bands, manually segmented lesions (by clinical experts) in true-color images were supplied (such as those in Figure 2.1). For a single image two classes were formed representing the two regions near the boundary; inside the lesion (*in*) and outside (*out*) (Figure 2.3). The Fisher single band image, I_F , is calculated as ([Blake1998])

$$I_F(x, y) = \mathbf{f}^T \mathbf{I}_{RGB}(x, y) \quad (2.2)$$

where $\mathbf{I}_{RGB}(x, y) = [I_R(x, y) \ I_G(x, y) \ I_B(x, y)]^T$ are the original RGB values,

$$\mathbf{f} = (S^{in} + S^{out})^{-1} (\bar{\mathbf{I}}_{RGB}^{out} - \bar{\mathbf{I}}_{RGB}^{in}) \quad (2.3)$$

$$S^{in} = \sum_{(x,y) \in in} (\mathbf{I}_{RGB}(x, y) - \bar{\mathbf{I}}^{in})(\mathbf{I}_{RGB}(x, y) - \bar{\mathbf{I}}^{in})^T \quad (2.4)$$

$$\bar{\mathbf{I}}^{in} = \frac{1}{N_{in}} \sum_{(x,y) \in in} \mathbf{I}_{RGB}(x, y) \quad (2.5)$$

and similarly for S^{out} and $\bar{\mathbf{I}}^{out}$. In the modified fisher the within-class scatter matrices (S^{in} , S^{out}) used in the original Fisher formulation were ignored (i.e. S replaced by an identity matrix). This firstly simplifies calculations and secondly gives complete emphasis on generating a single band that possesses high mean contrast along the boundary edge, i.e. giving maximum separation of the means of the two classes with no regards to their respective variance.

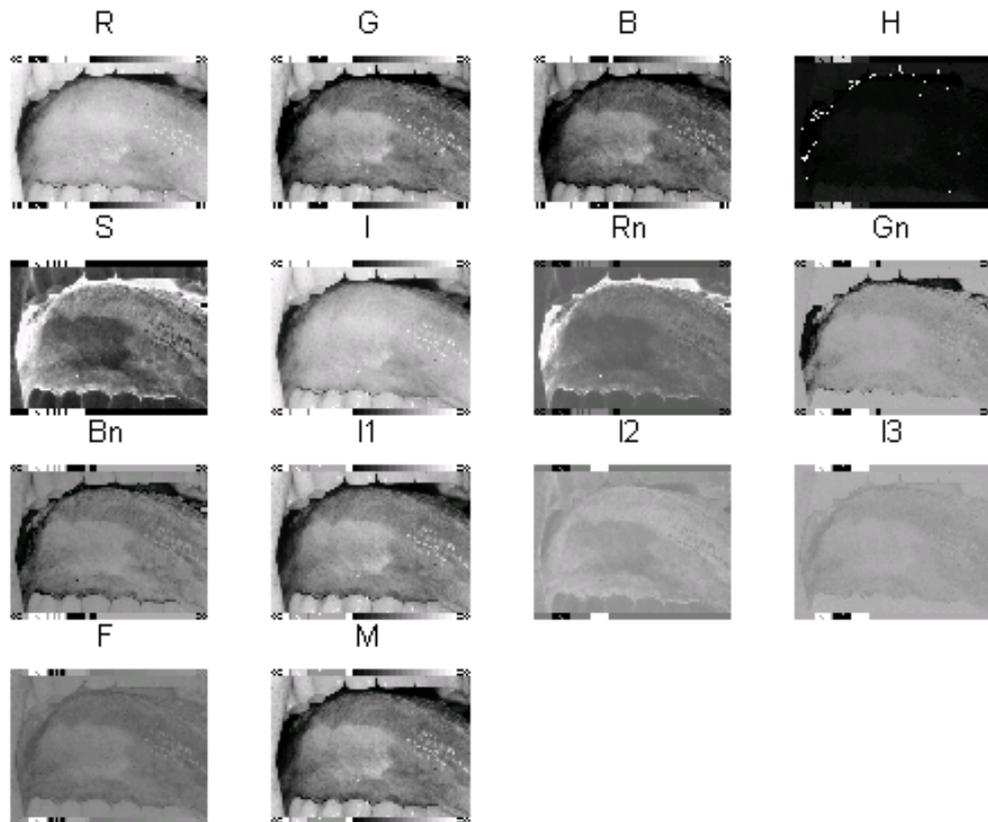


Figure 2.2. Different single bands derived from an RGB image of an oral lesion (see Table 2.1).

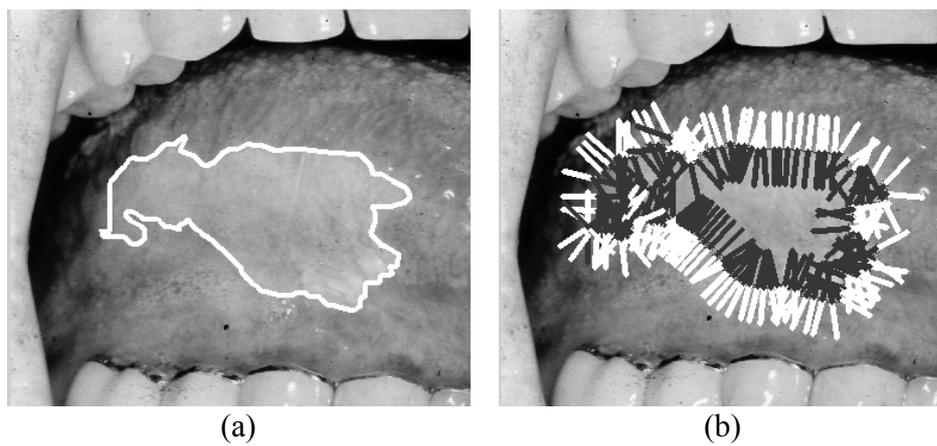


Figure 2.3. (a) Original oral lesion image with expert delineation. (b) Inner (dark) and outer (white) samples used for the Fisher training.

2.2.3 Comparing Single bands

In the snake implementation there are different weighting factors and parameters to be set. Many of these depend mainly on the shape of the lesion, for example the tensile and flexural weights and the resampling parameters. The threshold value T (see equation (A.13)) on the other hand, is directly linked to the intensity of the image and hence to the single band under investigation. In order to compare the performance of snakes using the different single bands, we fixed the values of all the parameters except for the threshold value. For each single band generated we performed a fully automated snake segmentation (without any manual intervention) over a feasible range of threshold values. To quantify the difference between the manually delineated boundary, M , and the snake-segmented boundary, S , we defined the following error measure

$$\varepsilon = \frac{A(S) \cup A(M) - A(S) \cap A(M)}{A(M)} \quad (2.6)$$

where $A(M)$ and $A(S)$ are the areas enclosed within M and S , respectively. This error measure was then used to determine which bands perform well to be used in the segmentation procedure. Figure 2.4 illustrates the various error measures for a selection of single bands obtained as described above for an example color image.

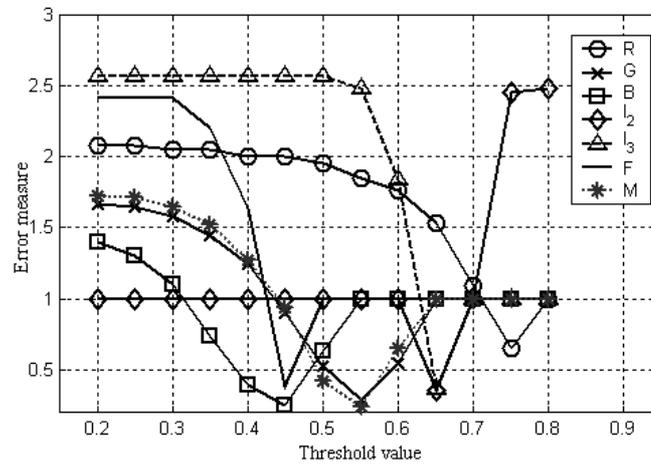


Figure 2.4. Error measures for various single bands vs. the threshold value calculated for one cancerous lesion.

2.3 Results

In this section we present our results of the snake segmentation of the oral lesions performed on single band images generated from the true color digital

images. Figure 2.5(a) shows a single band image with only five snake nodes used for initialization and placed inside the target lesion region. Figure 2.5(b) shows the final result of the snake segmentation with the forced points used to constrain the snake shown as circles. The expert manual tracings of the oral lesion are shown in Figure 2.5(c). Figure 2.6 shows a similar snake segmentation result on a different lesion image using four initial snake nodes. Figure 2.7 depicts the deformation of the snake and the progress of the segmentation. The four snake nodes used for initialization are shown in Figure 2.7(a) while Figure 2.7(b) shows the snake after deformation and subdivisions. Figure 2.7(c) shows the snake in a later stage of the deformation where it is stuck on an erroneous edge, not being able to reach the correct left side of the lesion boundary. The placement of a single forced point at the correct lesion boundary (the part that the snake couldn't latch to) and how this improves the segmentation is depicted in Figure 2.7(d). The final segmentation result is shown in Figure 2.7(e) and the expert delineation of the oral lesion is shown in Figure 2.7(f). Notice how the number of snake nodes is adaptively increased to accommodate the complexity of the lesion boundary. Figure 2.8 illustrates the calculation of the error term in equation (2.6). The binary image in Figure 2.8(a) shows the area $A(S)$ of the snake-segmented oral lesion of Figure 2.8(e) and Figure 2.8(b) shows the area $A(M)$ of the manually segmented lesion. Figure 2.8(c) shows the area described by the numerator of the error term in (2.6).

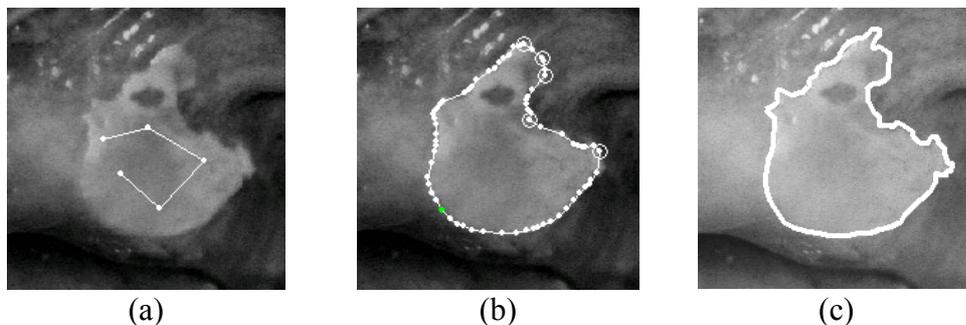


Figure 2.5. Segmentation example using the Green band: (a) Initial snake nodes. (b) Final segmentation result (snake nodes shown as white dots and forced points as white circles). (c) The manual expert delineation of the oral lesion overlaid on the original lesion image.

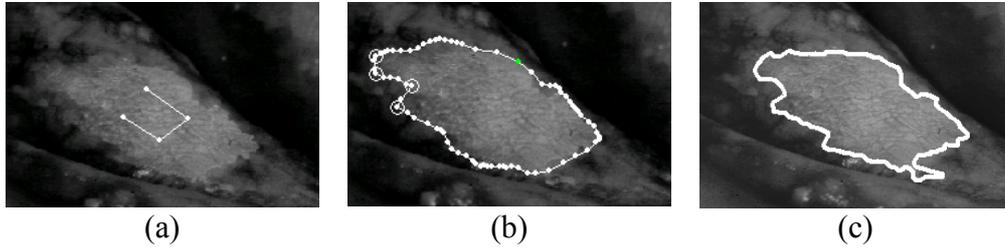


Figure 2.6. Segmentation example using the Blue band: (a) Initial snake nodes. (b) Final segmentation result (snake nodes shown as white dots and forced points as white circles). (c) The manual expert delineation of the oral lesion overlaid on the original image.

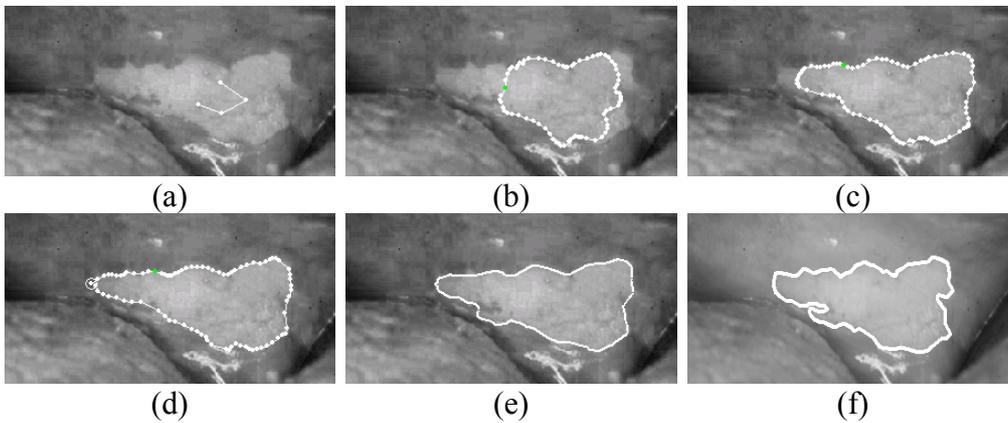


Figure 2.7. Segmentation example using the original Fisher projection band: (a) Initial snake nodes. (b) Progress of snake deformation (snake nodes shown as white dots). (c) Snake stuck on erroneous edge (left-side of lesion). (d) Addition of a forced point (white circle). (e) Final segmentation result. (f) The manual expert delineation of the oral lesion.

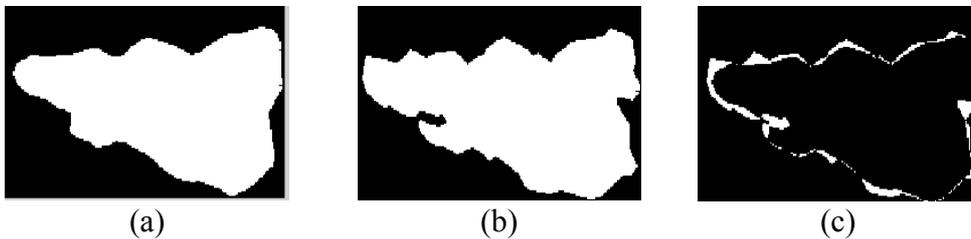


Figure 2.8. Error calculation: Area of (a) snake-segmented lesion, (b) manual delineation of the same lesion. (c) The erroneous area, $\varepsilon = 0.0953$ (9.53%).

2.4 Conclusion

We have applied snakes for semi-automatic segmentation of oral lesions in color images of the human oral cavity. Snakes proved to be a valuable method for the segmentation of oral lesions by guaranteeing continuous and smooth lesion boundaries (by edge linking) and led to small segmentation errors. However some operator interaction was still needed due to the large variability of the objects and images in this application. It is important that a user (or an add-hoc method) assists in the segmentation procedure by initializing the snake in the vicinity of the region in the image where the target lesion exists (done here by specifying a few initial snake nodes). The user should also be ready to intervene by placing constrained (forced) points to assist the snake if it clings to erroneous edges. Such assistance is generally accepted in clinical practice, sometimes even preferred and therefore we conclude that our segmentation method based on snakes will contribute to the clinical toolbox.

Chapter 3. STATISTICALLY CONSTRAINED SNAKES

In this chapter¹ we present a method for constraining the deformations of snakes in a way that is consistent with a training set of images. The method we propose is similar to both Active Shape Models (ASM) but without the landmark identification and correspondence requirement, and to Active Contour Models (ACM), but enforced with a priori information about shape variation. Rather than representing the object boundaries by spatially corresponding landmarks, we employ a frequency-based boundary representation. The Principal Component Analysis (PCA), which is central to ASM, is applied to the set of frequency-domain shape descriptors. An average object shape is extracted along with a set of significant shape variational modes. Armed with this model of shape variation we find the boundaries in unknown images by placing an initial ACM and allowing it to deform only according to the examined shape variations. The described methodology was applied to the problem of locating the left ventricular boundary in echocardiographic images. A training set of 105 expert-segmented echocardiographic images was used to train the model.

3.1 Introduction

Ultrasound echocardiography is a valuable non-invasive and relatively non-expensive tool for clinical diagnosis and analysis of heart function including ventricular wall motion. An important step towards this analysis is the segmentation of endocardial boundaries of the left ventricle (LV) [Hunter1993, Parker1994, Taine1994, Papadopoulos1995, Mikic1998, Malassiotis1999]. Although segmenting anatomical objects in high SNR images can be done with simple techniques, problems do arise when the images are corrupted with noise and the object itself is not clearly or completely visible in the image. This is clearly the case in heart images obtained by ultrasonography, which are characterized by weak echoes, echo dropouts and high levels of speckle noise. These image artifacts often result in detecting erroneous object boundaries or failing to detect true ones. Snakes or Active Contour Models [Kass1987] and its variants [Amini1990, Cohen1991, Grzeszczuk1997, McInerney2000] overcome parts of these limitations by considering the boundary as a single, inherently connected, and smooth structure, and also by supporting intuitive, interactive mechanisms for guiding the segmentation. In our application of locating the human LV boundary in echocardiography, human guidance is

¹ This chapter is based primarily on [Hamarneh2000c] (see also [Hamarneh2000e, Hamarneh2000b, Hamarneh2000f]).

often needed to guarantee acceptable results. A remedy is to present the snake with a priori information about the typical shape of the LV. Statistical knowledge about shape variation can be obtained using Point Distribution Models (PDM), which are central to the Active Shape Models (ASM) segmentation technique [Cootes1995a]. PDM, which are obtained by performing Principal Component Analysis (PCA) on landmark coordinates labeled on many example images, have been applied to the analysis of echocardiograms [Parker1994]. However, this procedure is problematic since manual labeling of corresponding landmark points is required. In our application it is tedious to obtain a training data set delineated by experts with point correspondence, let alone the fact that defining a sufficient number of landmarks on the LV boundary is a challenging task by itself. Hence, we adopt an approach similar to PDM for capturing the main modes of ventricular shape variation, however, in our method shapes are represented by descriptors that eliminate the need for spatial point correspondence, namely the Discrete Cosine Transform (DCT) coefficients. We use snakes as the underlying segmentation technique but with constrained deformations derived from the prior knowledge of ventricular shape (A similar approach using Fourier descriptors and applied to locating the corpus callosum in 2D MRI images was reported in [Székely1996]). Results of segmenting the human LV in real echocardiographic data using the discussed methodology are presented.

3.2 Methods

3.2.1 Overview

This section presents a general overview of the method (Figure 3.1). We used snakes as the underlying segmentation technique. In order to arm the snake model with a priori information about the typical shape variations of the LV that may be encountered during the segmentation stage, a training set of images is provided. This set is manually delineated by medical experts without the requirement of complete landmark correspondence between different images. The entire set of manually traced contours is then studied to model the typical ventricular shape variations. This is done by first applying a re-parameterization of the contours, which gives a set of DCT coefficients replacing the spatial coordinates. We then apply PCA to find the strongest modes of shape variation. This results in an average ventricular shape, represented by a set of average DCT coefficients, in addition to the principal components along with the fraction of variation each component explains. To segment a new image of the LV, we initialize a snake and, unlike classical snakes, do not allow it to freely deform according to internal and external energy terms, but instead we constrain its deformations in such a way that the resulting contour is similar to the training set. To attain the constrained

deformations we obtain the vector of DCT coefficients for the active contour coordinates, project it onto an allowable snake space defined by the main principal components, and then perform an Inverse DCT (IDCT) that converts the constrained DCT coefficients back to spatial coordinates. This is repeated until convergence, which is reached when the majority of snake nodes do not change their locations significantly. The shape models generated are normalized with respect to the similarity transformation parameters: rotation angle, scaling factor, and two translation parameters.

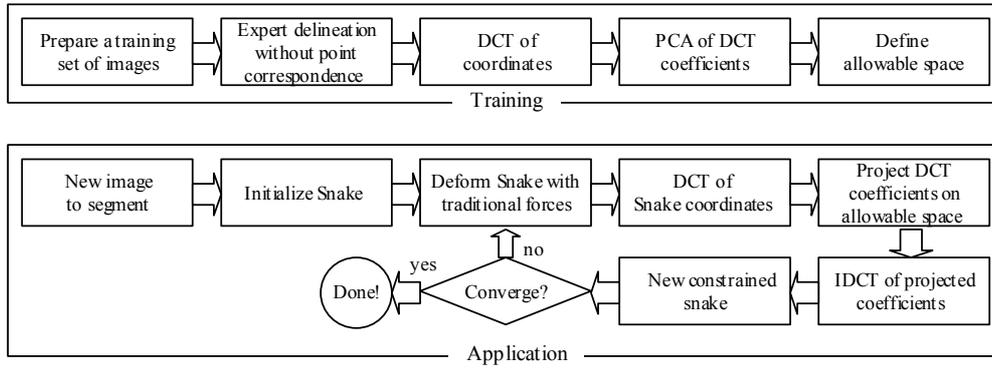


Figure 3.1. Flowchart depicting the main steps involved in the use of a statistically constrained snake for image segmentation.

3.2.2 Representing Contours by DCT Coefficients

We use snakes as the underlying segmentation technique. A snake contour is originally represented by a set of N of nodes $\{\mathbf{v}_i(t) = (x_i(t), y_i(t))\}$, $i = 1, 2, \dots, N$, and is deformed according to internal and external forces (more details can be found in Appendix A). Snake contour re-parameterization is obtained via the use of the one-dimensional Discrete Cosine Transform (DCT) of the snake coordinates. The 1D DCT of the sequence x_i of snake contour coordinates is defined as

$$X(k) = w(k) \sum_{i=1}^N x_i \cos \frac{\pi(2i-1)(k-1)}{2N}, k = 1, \dots, N \quad (3.1)$$

and the inverse DCT is give as

$$x_i = \sum_{k=1}^N w(k) X(k) \cos \frac{\pi(2i-1)(k-1)}{2N}, i = 1, \dots, N \quad (3.2)$$

where

$$w(k) = \begin{cases} \frac{1}{\sqrt{N}}, & k = 1 \\ \sqrt{\frac{2}{N}}, & 2 \leq k \leq N \end{cases} \quad (3.3)$$

and $X(k)$ are the DCT coefficients. Similar equations are used for the y_i coordinates and the $Y(k)$ DCT coefficients. The DCT was favored as the new frequency domain shape parameterization because it produces real coefficients, has excellent energy compaction properties, and the correspondence between the coefficients (when transforming contours with no point correspondence) is readily available. The latter property stems from the fact that corresponding DCT coefficients capture specific spatial contour frequencies.

3.2.3 Principal Component Analysis

In order to identify the main modes of shape variation found in the training contours, we perform a PCA on the DCT coefficients representing them. The same number, say M , of DCT coefficients is obtained for the set of x and y coordinates that represent each shape in the training set. This is done either by interpolating the spatial coordinates or truncating the DCT coefficients. PCA yields the principal components (PCs) or main variation modes, \mathbf{a}_j , and the variance explained by each mode, λ_j . The first t PCs, sufficient to explain most of the variance, are used, i.e. $j = 1, 2, \dots, t$. The average of the coefficient vectors, \bar{X} , is also calculated. The same procedure is performed for the y coordinates (further details on PCA can be found in Appendix C).

3.2.4 Constraining Contour Deformation

Subsequent to providing a set of images containing the object of interest, the training set of tracings is obtained (contours represented by coordinate-vectors of varying length with no point correspondence). DCT coefficients (X) are then obtained followed by PCA. Presented with a new image, a snake contour is first initialized by specifying the starting and end points of the contour, and then allowed to deform by applying forces that minimize traditional energy terms. In order to guarantee a snake contour resembling an acceptable shape (similar to those in the training set), we constrain the resulting deformed contour, $\{\mathbf{v}_i(t), i = 1, \dots, N\}$, by projecting the vector X (consisting of M DCT coefficients) onto the subspace of principal components (the allowable shape space) according to

$$X_{proj} = \bar{X} + \mathbf{A}\mathbf{b} \quad (3.4)$$

where \mathbf{b} is a vector of scalars weighing the main variation modes in \mathbf{A} and is calculated as

$$\mathbf{b} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T (X - \bar{X}) \quad (3.5)$$

and $\mathbf{A} = [\mathbf{a}_1 \ \mathbf{a}_2 \ \dots \ \mathbf{a}_t]$. Prior to performing the IDCT, we restrict the projected coefficients (X_{proj}) to lie within $\pm 3\sqrt{\lambda_j}$ since in this application the population typically lies within three standard deviations from the mean. Again, the same procedure is performed for the y coordinates. The statistical constraints can be applied after each snake deformation step or only when the resulting DCT coefficients of the snake contour are different (using some norm and threshold) from the mean coefficients. Note that the DCT coefficients are obtained and constrained for shapes normalized with respect to similarity transformation parameters utilizing corresponding starting and ending contour points.

3.3 Results

The described methodology was applied for segmenting the LV ventricle in real echocardiographic images. We collected 105 images of the human LV. The ventricular boundaries were manually traced by a medical expert. There was no point correspondence between the frames, with the number of traced points varying between 28 and 312 (Figure 3.2 and Figure 3.3). The DCT of the manual tracings was then obtained.

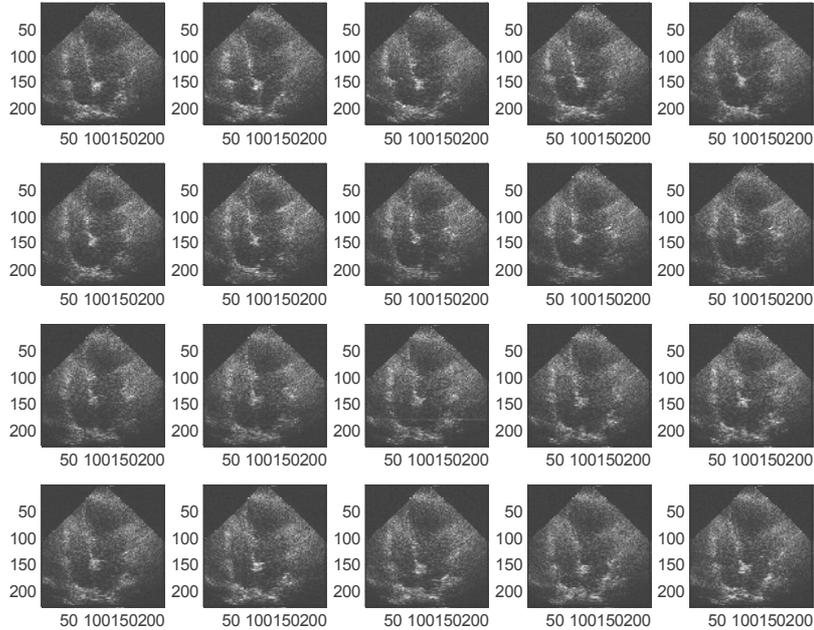


Figure 3.2. Sample of the echocardiographic training image set.

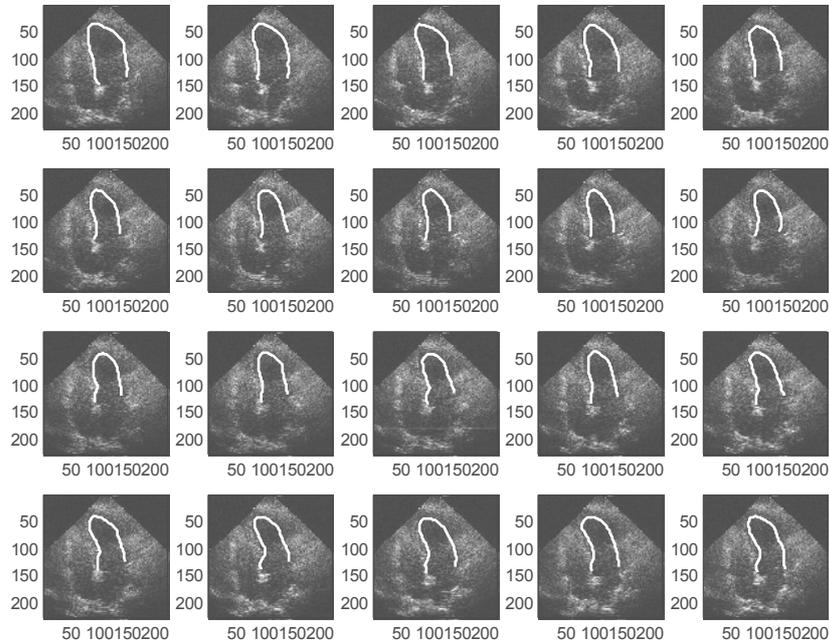


Figure 3.3. Manual tracings of LV boundary in the training images.

Figure 3.4 shows an example of the manual tracings and the resulting contour after IDCT of the truncated DCT coefficients. The ratio ‘energy of truncated contour’/‘energy of the original contour’ for increasing numbers of DCT coefficients was examined in order to determine how many DCT coefficients to use. This was followed by a PCA of the truncated DCT coefficients. Five variation modes of 56 possible were enough to explain 95% of the total variation, 12 were enough for 99%, and 24 for 99.9%. Figure 3.5 depicts the first and second shape variation modes found in the training set.

To illustrate applying shape constraints we used test examples. We began with one of the manual tracings, added Gaussian noise, performed DCT, truncated certain DCT coefficients, projected the remaining coefficients on the allowable shape space, and then performed the IDCT. It was visually obvious how the constrained contour resembles a much more plausible boundary of the LV than the noisy one (Figure 3.6).

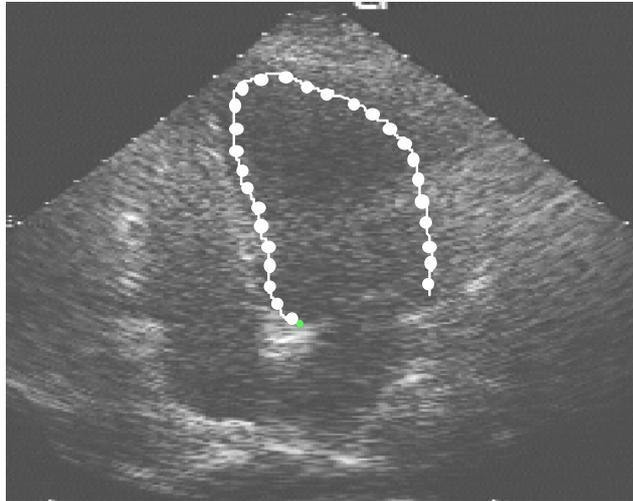


Figure 3.4. Ultrasound image with the manual tracing (continuous) and the contour after IDCT of truncated DCT coefficients (dots).

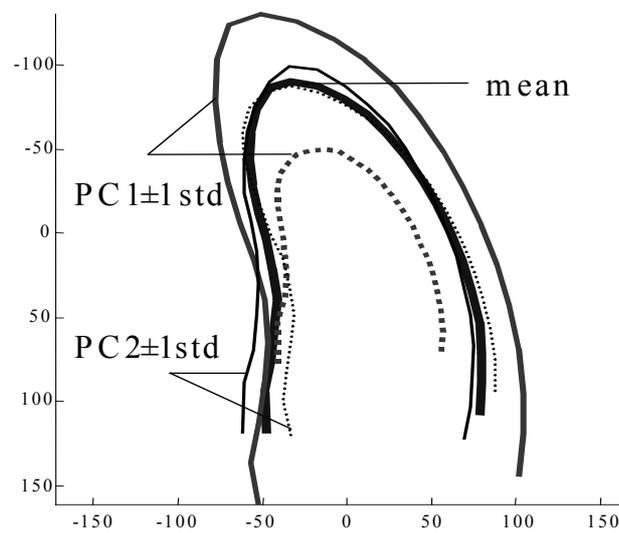


Figure 3.5. Mean contour and the first and second variation modes (weighted by ± 1 std).

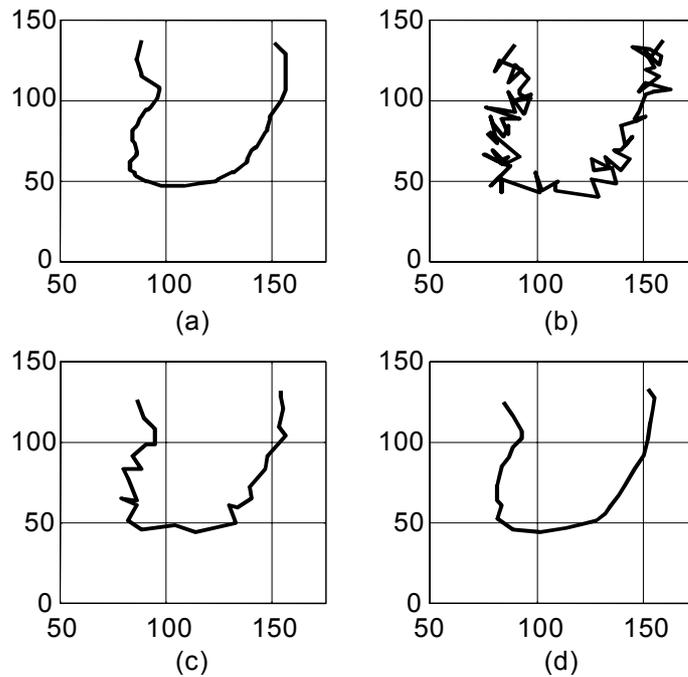


Figure 3.6. (a) Manual tracing. (b) Noisy version of (a). (c) IDCT of truncated DCT coefficients of (b). (d) The projection of (c) on the allowable shape space (note the similarity to (a)).

More importantly, results on real echocardiographic data were obtained by initializing the snake on an image that wasn't included in the training set (i.e. cross validation was used) and then allowing it to deform under forces that minimize its energy. This was followed by a DCT-Truncation-Projection-IDCT procedure. The outcome of the snake segmentation alone, due to noise and echo dropouts in the image, often gave unreasonable and unacceptable shape of the LV. Conversely, employing constrained deformations resulted in acceptable LV boundaries (Figure 3.7, Figure 3.8).

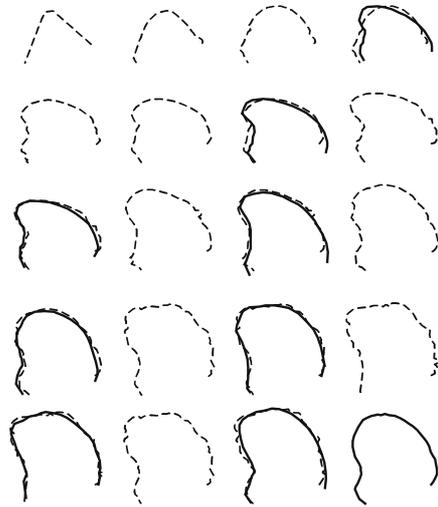


Figure 3.7. Snake contours (dashed) and the constrained contours (continuous) with increasing number of iterations (left to right, top to bottom).

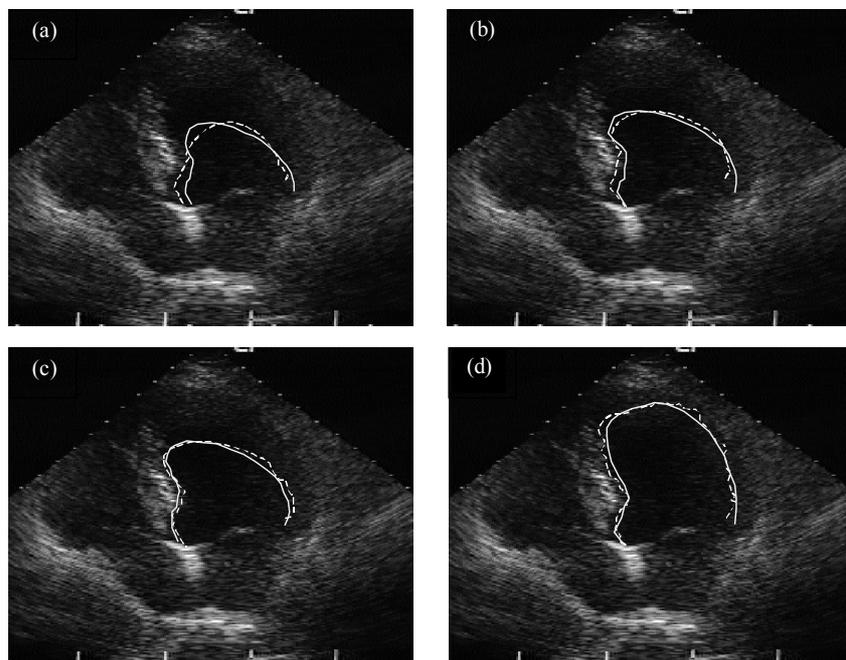


Figure 3.8. The progress (a to d) of a snake overlaid on an ultrasound image of the left ventricle (dashed) and the result of the DCT-Truncation-Projection-IDCT (continuous).

3.4 Conclusion

We presented a method for constraining the deformations of an active contour according to training examples and applied it to segmenting the human left ventricle in echocardiographic (ultrasound) images. To capture the typical shape variations of the training set, principal component analysis was performed on frequency-domain shape descriptors in order to avoid the drawbacks associated with labeling corresponding landmarks (only the starting and ending points of the contours correspond). The method utilizes the strength of ACM in producing smooth and connected boundaries along with the strength of ASM in producing shapes similar to those in a training set. More plausible LV shapes resulted when employing the new method compared to classical snakes.

Chapter 4. OPTICAL FLOW SNAKE FORCES

In this chapter¹ we present an extension to the formulation of Active Contour Models (snakes) by including an additional contour-deforming force. The new force is derived from the optical flow field calculated between two time-consecutive frames in an image sequence. The addition of the new force assists the snake in tracking desired dynamics while the traditional snake forces guarantee the contour's smoothness and attraction to edges. The method is applied to the problem of tracking the leading edge of injected contrast agent in an echocardiographic image sequence and is shown to improve the tracking speed. A clinical motivation and previous work on echocardiography and videodensitometry are initially presented.

4.1 Introduction

4.1.1 Clinical Motivation

The assessment of human right ventricular (RV) function is of great importance in many diseases afflicting the heart. For example, disturbances in filling and elimination patterns of the RV hemodynamics can be interpreted as signs of abnormal RV function. Arrhythmogenic Right Ventricular Dysplasia (ARVD) is a rare but clinically important disease, which afflicts young adults and may play a role in the etiology of sudden death among young people [McKenna1994]. The impairments of the RV function in this group of patients can be described in terms of wall motion abnormalities, or as localized bulging and sacculations. These abnormalities are mainly located at the inflow, outflow or apical regions. To study these abnormalities we use sequences of contrast echocardiographic images.

4.1.2 Medical Imaging Procedure

Thirty patients with biopsy-verified ARVD and 18 healthy volunteers (control group) were investigated by use of contrast echocardiography. The investigations were performed with an Acuson XP 128 computed system or a Sequoia system equipped with multiHertz transducers. As a contrast agent, 2 ml of Haemaccel® (Hoechst) was injected intravenously. Transthoracic apical four-chamber view with focus on the right ventricle was used and continuously recorded during and after the injection. The video sequence (Figure 4.1) of the

¹ This chapter is based primarily on [Hamarneh2000a] (see also [Althoff2000]).

filling and elimination of the contrast agent was then digitized using a PC with a frame grabber (Matrox, Meteor II), giving about 600 images for each sequence.

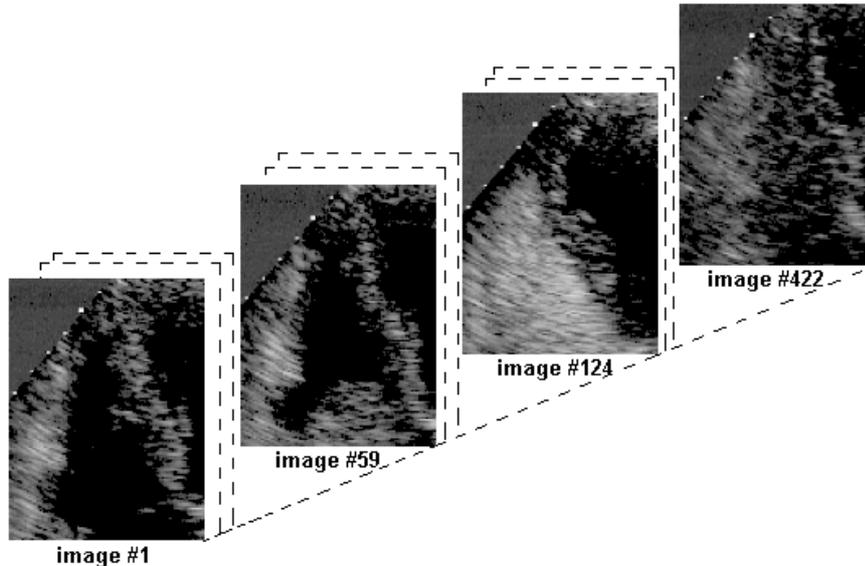


Figure 4.1. Sample frames from a digitized image sequence. In frame #1, #59, #124 and #422 the contrast agent hasn't reached, just reached, totally filled and washed out from the RV, respectively.

4.1.3 Contrast Echocardiography and Videodensitometry

Contrast-echocardiography in conjunction with real-time videodensitometry [Suurkula1997] is a method to describe the RV hemodynamics by intravenously injecting a contrast agent and continuously imaging, recording, and analyzing its flow. This can be an effective means of studying right ventricular (RV) structural changes, e.g. in patients diagnosed with ARVD, since it can be used to illustrate such hemodynamics by calculating wash-in and wash-out time indices (parameters) corresponding to the time needed for filling and elimination of the contrast agent in the RV. The time indices are calculated for 5×5 pixels regions of interest (ROI) in each sequence, located in the right ventricular inflow tract (RVIT) and in the apex (Figure 4.2).

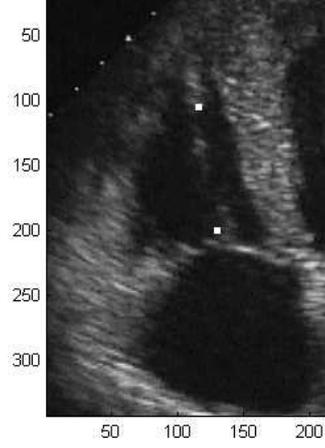


Figure 4.2. The RVIT and apex ROIs (shown as white squares). In this example the contrast agent has not yet reached the RV.

The changes over time in image intensity, I , due to the flow of the contrast agent at those regions were studied. A total of 48 frame sequences were analyzed from two groups: 30 ARVD-diagnosed patients and 18 healthy volunteers. The mean intensity of each ROI in the sequence was calculated and analyzed. Three time parameters T_o , T_a and T_e were obtained (Figure 4.3). The parameters are defined as

- T_a : time of maximum contrast agent intensity, I_{max} .
- T_o : time for the intensity to rise to $0.2 \times I_{max}$, (wash-in).
- T_e : time for the intensity to drop to $0.5 \times I_{max}$, (wash-out).

A comparison of the following time-differences, rather than the absolute values of T_o , T_a and T_e , is performed for different sequences

$$\begin{aligned}
 \Delta_{eo} &= T_e - T_o \\
 \Delta_{ao} &= T_a - T_o \\
 \Delta_{ea} &= T_e - T_a.
 \end{aligned} \tag{4.1}$$

The mean and standard deviation of the time differences are found to be larger for the ARVD-group (Table 4.1, Figure 4.4).

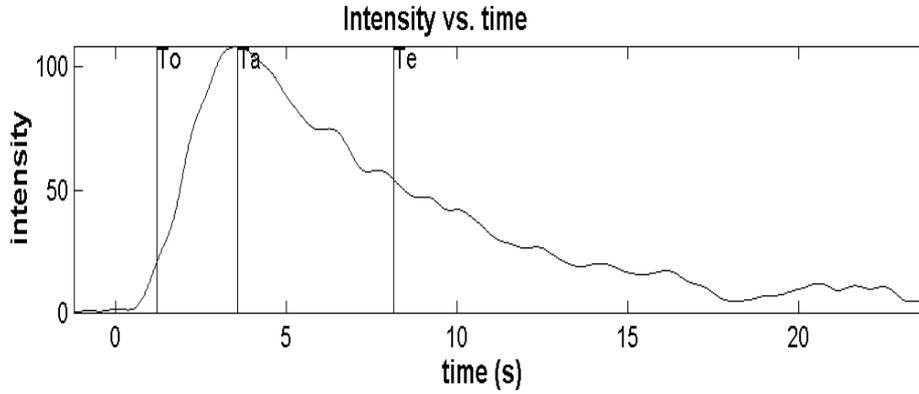


Figure 4.3. Image intensity at a particular ROI vs. time. Also shown in the figure are locations of the calculated time parameters, T_o , T_a and T_e .

Table 4.1. Mean (in seconds) and standard deviation values for the ARVD-group and the control group.

Test	ARVD-group	Control-group
Δ_{eo} RVIT	11.1 ± 6.9	5.5 ± 1.7
Δ_{eo} apex	11.0 ± 5.0	6.3 ± 1.7
Δ_{ea} RVIT	8.6 ± 6.6	3.9 ± 1.6
Δ_{ea} apex	7.1 ± 4.2	4.0 ± 1.5
Δ_{ao} RVIT	2.6 ± 1.4	1.7 ± 0.8
Δ_{ao} apex	3.8 ± 1.9	2.3 ± 0.6

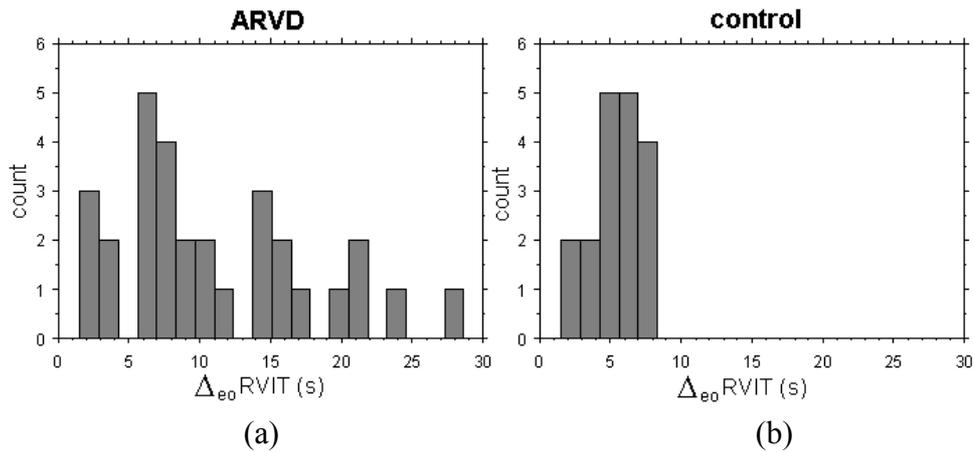


Figure 4.4. Δ_{eo} RVIT histograms for (a) the ARVD group and (b) the control group.

The P-values resulting from a Mann-Whitney U-test² show that the ARVD and the control group are statistically well separated (Table 4.2) and that the best separation measure is Δ_{eo} .

Table 4.2. Resulting P-values after a Mann-Whitney U test

Test	P-value
Δ_{eo} RVIT	0.0019
Δ_{eo} apex	0.0002
Δ_{ea} RVIT	0.0109
Δ_{ea} apex	0.0031
Δ_{ao} RVIT	0.0203
Δ_{ao} apex	0.0019

Furthermore, a pseudo color map can be obtained by analyzing the image intensities at different regions of the right ventricle (Figure 4.5) [Suurküla1997].

² About the Mann-Whitney U-test: Basically, the P value answers this question: If the populations really have the same median, what is the chance (probability) that random sampling would result in the separation observed in this experiment? For example, low probability P-value (e.g. 0.0019) implies rejecting the hypothesis that the populations come from the same median, with high confidence, $(1-0.0019) \times 100\%$. The test doesn't require that the groups have normal distribution. The test involves the following steps: Ranking all the values from low to high, paying no attention to which group each value belongs, if two values are the same, then they both get the average of the two ranks for which they tie. The smallest number gets a rank of 1. The largest number gets a rank of N, where N is the total number of values in the two groups. Summing the ranks in each group and reporting the two sums. If the sums of the ranks are very different, the P value will be small.

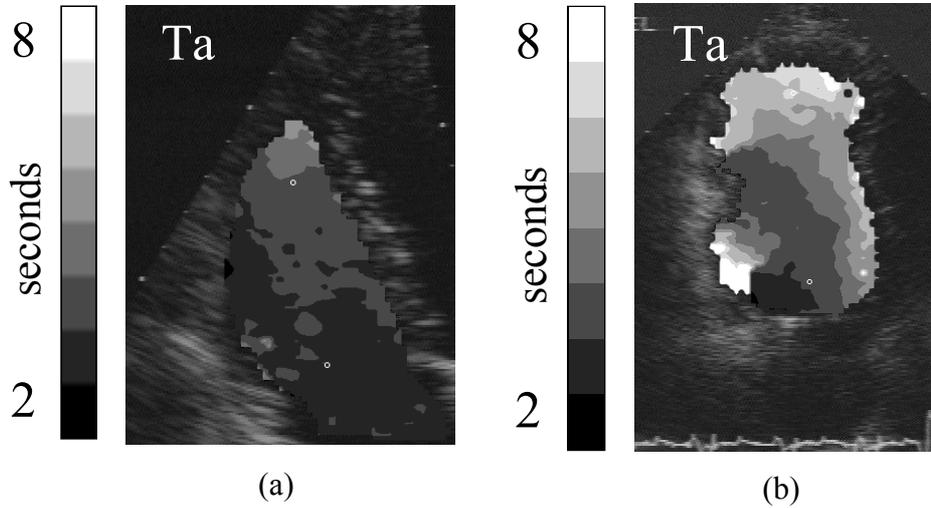


Figure 4.5. The time (T_a) needed for the contrast agent to produce maximum intensity in different regions of the RV in both (a) normal and (b) ARVD cases shown as color-coded time maps. The bright values in the right figures are indicative of hindered flow (longer times) relating to ARVD.

4.2 Tracking the Contrast Front

Here we attempt a different approach to characterize RV flow pattern, which involves tracking the front (the leading edge) of the contrast agent during the RV filling. Examining the flow of the contrast agent in a typical echocardiographic image sequence (Figure 4.6) reveals weak ultrasound echoes, echo dropouts and high levels of noise. Thus, the application of simple edge detectors to locate the front of the contrast would result in detecting erroneous edges and gaps.

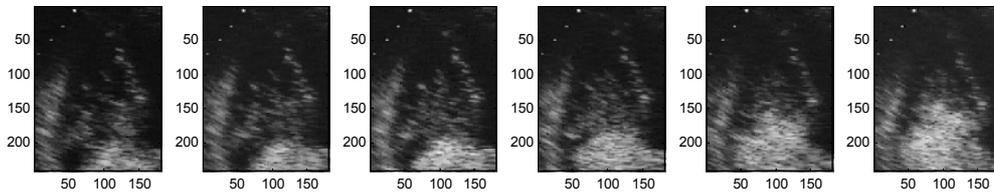


Figure 4.6. Successive frames of the contrast agent entering the right ventricle of the heart.

The strength of active contour models in integrating low-level image information and guaranteeing a smoothly connected contour makes them suitable for our purposes (refer to Section 1.2.2 for further details). For

tracking the contrast front in a sequence we make use of the motion field obtained by Optical Flow (OF) computation as an additional contour deforming force, in order to speed up the tracking and to influence the snake nodes to match corresponding contrast front regions in-between frames.

Other authors have investigated similar approaches. In [Mikic1998] a method for segmenting and tracking cardiac structures in ultrasound image sequences is presented. In integrating the contour's equation of motion, the method sets the initial velocities of the contour vertices to OF estimates, and sets their positions to the final position from the preceding frame. [Peterfreund1999] used Kalman filter-based active contours that calculate OF along the contour as system measurement to detect and reject measurement that may belong to other objects. [Akgul1998] presents an application of tracking 2D contours of tongue surfaces from digital ultrasound image sequences. The proposed method makes use of OF to reduce the computational complexity involved when searching for optimal snake node locations in a dynamic programming setting. This is done by considering only a subset of pixels in a search window. The subset is chosen on the basis of the first OF constraint, namely that the intensity of an object's point in a dynamic image doesn't change with time.

4.2.1 Optical Flow

Optical flow [Horn1981] is a well-established method for calculating the velocity field $(u(x, y), v(x, y))$ of the apparent 2D motion of pixels in a dynamic image $I(x, y, t)$ due to the 3D motion of imaged objects, by examining the spatial and temporal changes in intensity values. Classical OF is based on two main constraints. The first states that the brightness of any object point is constant over time. This can be written as:

$$I(x + dx, y + dy, t + dt) = I(x, y, t). \quad (4.2)$$

Using Taylor series expansion and neglecting higher order terms gives the first OF constraint equation:

$$I_x u + I_y v + I_t = 0 \quad (4.3)$$

where $u = dx/dt, v = dy/dt$ are the desired velocity field components, I_x and I_y are the spatial image derivatives, and I_t is the temporal image derivative. Equation (4.2) by itself is insufficient to calculate (u, v) hence a second constraint, the velocity field smoothness constraint, is introduced. The velocity field can now be calculated as that which best satisfies both constraints by minimizing the following square error function

$$\xi^2(x, y) = (I_x u + I_y v + I_t)^2 + \lambda(u_x^2 + u_y^2 + v_x^2 + v_y^2) \quad (4.4)$$

where λ is a Lagrange multiplier. The following iterative algorithm detailed in [Horn1981] is used to find the velocity field (Figure 4.7).

Initialize: $u(x, y) = v(x, y) = 0$ for all x, y

$$\text{do: } u = \bar{u} - I_x \frac{P}{D}, \quad v = \bar{v} - I_y \frac{P}{D} \quad \text{while } \xi^2(x, y) > \varepsilon \quad (4.5)$$

where $P = I_x \bar{u} + I_y \bar{v} + I_t$, $D = \lambda^2 + I_x^2 + I_y^2$ and ε is a small number.

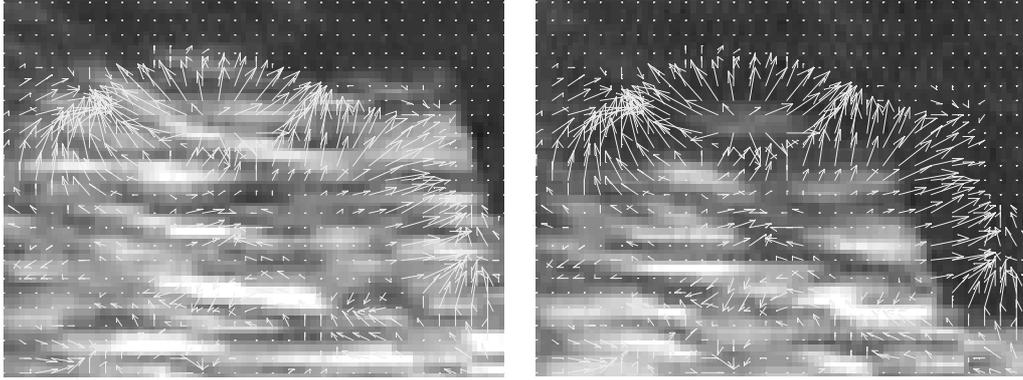


Figure 4.7. Optical flow (velocity) field shown on two consecutive frames.

4.2.2 Optical Flow Snake Forces

In order to track the contrast agent front in an echocardiographic image sequence we need to accomplish two tasks. The first is to locate the region where the contrast front has moved from one frame to the next, and the second is to detect this front as a smooth and connected boundary. We use the optical flow to address the first task and snakes to address the second (see Appendix A for more details on snakes). To combine the two techniques we include an additional force term $\mathbf{F}_i^{flow}(t)$ proportional to the calculated velocity field at the current snake node position $\mathbf{v}_i(t) = (x_i(t), y_i(t))$, yielding (compare with equation (A.15)):

$$\mathbf{v}_i(t) = \mathbf{v}_i(t-1) + w_1 \mathbf{F}_i^{tensile}(t) + w_2 \mathbf{F}_i^{flexural}(t) + w_3 \mathbf{F}_i^{external}(t) + w_4 \mathbf{F}_i^{flow}(t) \quad (4.6)$$

where w_4 is a weighting factor,

$$\mathbf{F}_i^{flow}(t) \propto (u(x_i(t-1), y_i(t-1)), v(x_i(t-1), y_i(t-1))) \quad (4.7)$$

and $u(x_i, y_i), v(x_i, y_i)$ are obtained using algorithm (4.5) in Section 4.2.1.

4.3 Results

We tracked the leading edge of synthesized frame sequences and of contrast agent filling the RV in real ultrasonic image sequences. Images were first smoothed using non-linear diffusion filtering [Perona1990]. Initially we present an example of tracking in a synthesized sequence (Figure 4.8). The

snake nodes are shown after latching onto the leading edge. In Figure 4.8(a) the snake deformed without OF forces and detected the leading edge after 7, 8, 8, 11, 13, and 21 iterations, whereas in Figure 4.8(b) OF forces were used and the edge was reached in *only* 2, 5, 3, 2, 3, and 2 iterations.

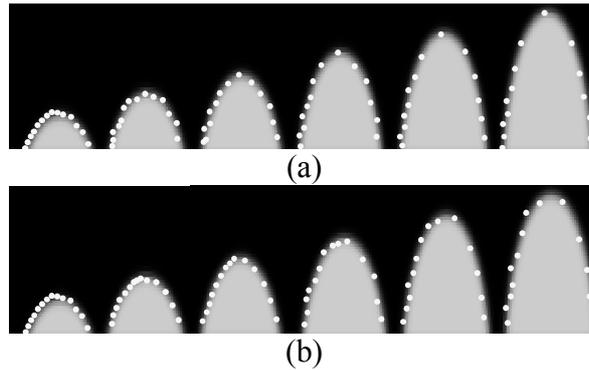


Figure 4.8. Results of tracking a synthetic sequence. (a) Without using optical flow forces. (b) With optical flow forces. In (a) 7, 8, 8, 11, 13, and 21 iterations were required compared to 2, 5, 3, 2, 3, and 2 iterations in (b).

The front of the contrast agent was tracked in eight sequences during the RV filling process, five from the ARVD-group and three from the control group. In each sequence, the front was tracked, on average, in about eight images. In the example depicted in Figure 4.9, the snake without OF forces needed 19, 20, 23, 22, 22, and 16 iterations (Figure 4.9(a)) whereas the snake with OF forces needed 5, 8, 10, 10, 6 and 10 iterations (Figure 4.9(b)).

We also show an example of both snakes, with and without OF forces, deforming towards the leading edge of the contrast agent in a single frame (Figure 4.10). The OF snake (with larger nodes in the figure) progresses faster towards the edge and locates it in only 10 iterations compared to 23 iterations needed for the snake without OF forces.

Histograms of the number of iterations needed for the contour to find the edge for all tested frames are also calculated (Figure 4.11). The mean number of iterations needed when using and without using information about the OF was 6.3 and 12.3, respectively.

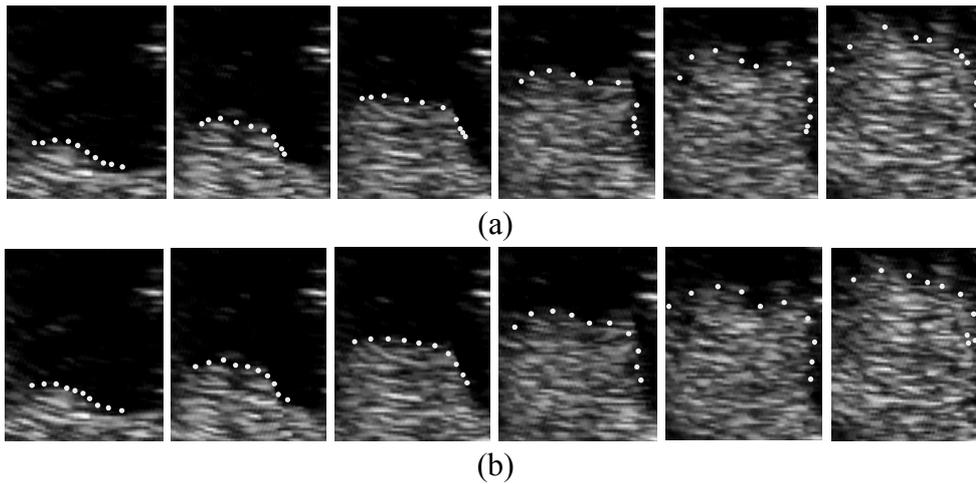


Figure 4.9. Results of tracking a real sequence. Upper frames: without using optical flow forces obtained after 19, 20, 23, 22, and 16 iterations. Lower frames: with optical flow forces obtained after 5, 8, 10, 10, 6 and 10 iterations.

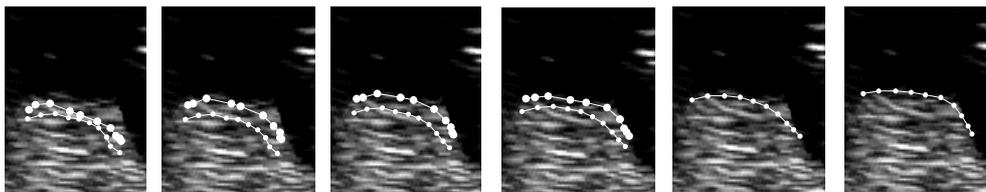


Figure 4.10. The snake with optical flow forces (large nodes) progresses faster towards the contrast front compared to the snake without optical flow forces. The snake nodes are shown after 1 (left-most), 2, 6, 10, 15, and 23 (right-most) iterations.

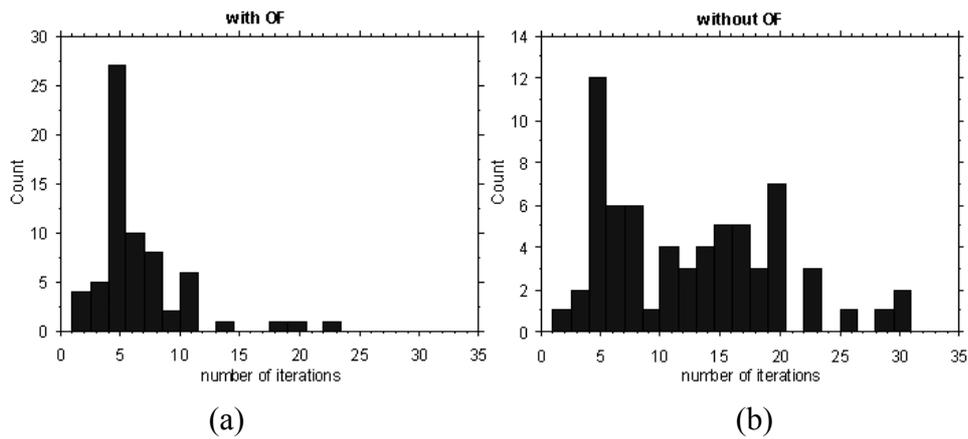


Figure 4.11. Histogram of the total number of iterations needed for the contour to latch to the contrast front (a) with optical flow forces, (b) without optical flow forces.

An investigation into the use of the methodologies described (contrast echocardiography, videodensitometry and tracking the contrast front) for the discrimination between ARVD patients and healthy subjects can be found in [Althoff2001]. For example, in the case when the calculated time indices do not clearly belong to either the ARVD patients or the healthy subjects groups, the characteristics of the contrast front dynamics may provide the additional needed clues or features (Figure 4.12).

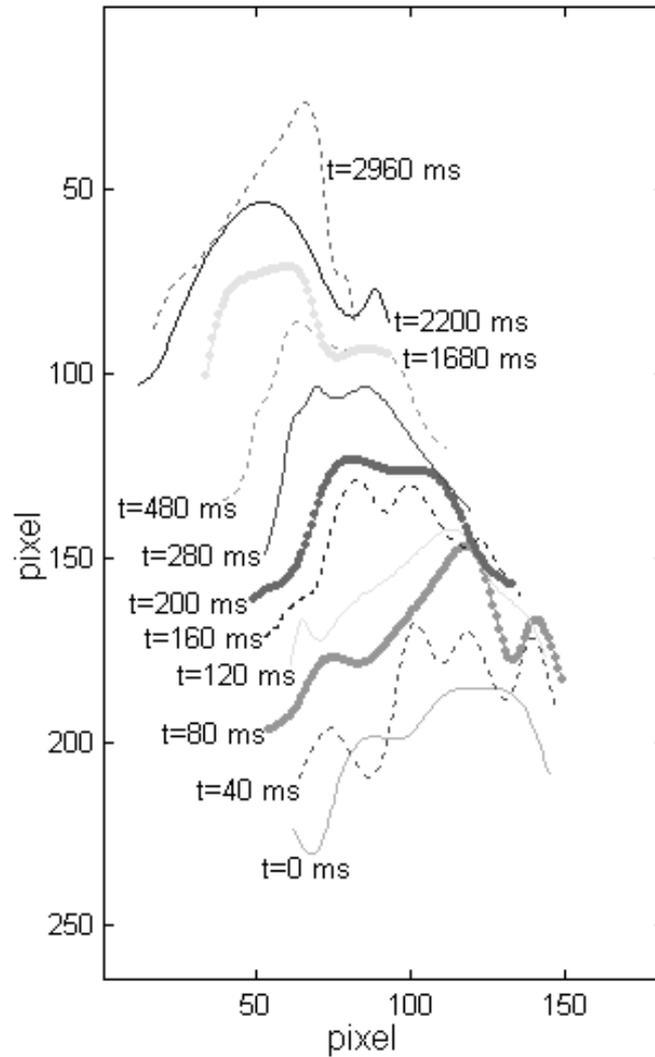


Figure 4.12. The result of tracking the leading edge in one of the sequences from the ARVD-group. Each contour represents the contrast agent front at different times indicated beside each contour. The contrast front enters the RV ($t=0$) until the RV is totally filled ($t=2960$ ms). Note how the contrast front in the initial phase of the filling ($t=0$ to 480ms) moves faster than the final phase ($t=480$ to 2960ms). This is indicative of the inhomogeneous operation of the RV, identified via contrast front tracking.

4.4 Conclusion

In this chapter we motivated the use of contrast echocardiography for studying the blood flow in RV. We discussed the use of videodensitometry for extracting time indices and pseudo color time maps describing RV hemodynamics. We then presented an extension to active contour models in the form of an additional optical flow-based force. The method was used to track the front of the injected contrast agent while filling the RV. The front dynamics may provide additional information for discriminating between healthy subjects and ARVD patients.

Chapter 5. DEFORMABLE SPATIO-TEMPORAL SHAPE MODELS

In this chapter¹ 2D Active Shape Models are extended to 2D+time by presenting a method for modeling and segmenting spatio-temporal shapes (ST-shapes). The modeling part consists of constructing a statistical model of ST-shape parameters. The model obtained describes the principal modes of variation of the ST-shape in addition to certain constraints on the allowed variations. A deformable model approach is used in segmentation; an initial ST-shape is deformed to better fit the data and the optimal proposed deformation is calculated using dynamic programming. Resulting ST-shapes are forced to conform to the training set. Different spatio-temporal segmentation results are presented.

5.1 Introduction

In many image analysis applications there is a need for modeling and locating non-rigid time-varying objects. One approach for dealing with such objects is the use of deformable models. Deformable models (DM) [Singh1998a], such as snakes [Kass1997] and its variants [Cohen1991, Grzeszczuk1997, Herlin1992, Lobregt1995, McInerney2000], have attracted considerable attention and are widely used for segmenting non-rigid objects in 2D and 3D (volume) images. Nevertheless, there are several problems associated with snakes. They were designed as interactive models often relying upon user intervention to guide the segmentation. They were also designed to be a general model showing no preference for a particular object shape other than those that are smooth. This generality can cause unacceptable results when snakes are used to segment objects with shape abnormalities arising from occlusion, closely located but irrelevant structures, or noise. Thus, techniques which incorporate a priori knowledge of object shape led by Active Shape Models (ASM) [Cootes1995a] were introduced. In ASM the statistical variation of shapes is modeled beforehand in accordance with a training set of known examples. In order to attack the problem of tracking non-rigid time-varying objects, DM were extended to dynamic deformable models [Leymarie1993, Singh1993, McInerney2000]. These describe the shape changes (over time) in a single model that evolves through time to reach a state of equilibrium where internal

¹ This chapter is based primarily on [Hamarneh2001d] (see also [Hamarneh1999a, Hamarneh1999b]).

forces, representing constraints on shape smoothness, balance the external image forces and the contour comes to rest. DM have been constructed by applying a probabilistic framework and led to techniques such as ‘Kalman snakes’ [Terzopoulos1992]. Motion tracking using deformable models has been used for tracking non-rigid structures such as blood cells [Leymarie1993] and much attention has been given to the human heart and the tracking of the left ventricle in both 2D and 3D [Leymarie1993, McInerney2000]. In addition to tracking rigid objects, previous work focused on arbitrary non-rigid motion and gave little attention to tracking objects moving in specific motion patterns, without the incorporation of statistical prior knowledge in both 2D and time [Black1997].

In this chapter, we present a new method for locating spatio-temporal shapes (ST-shapes) in image sequences. We extend ASM [Cootes1995a] (see also Appendix B) to include knowledge of temporal shape variations and present a spatio-temporal shape modeling and segmentation technique. We believe the method is suited for modeling and segmenting objects with specific motion patterns, as in cardiography, optical signature motion recognition, and lip-reading for Human-Computer Interaction (HCI), and others.

In order to model a certain class of ST-shapes (Figure 5.1), a representative training set of known shapes is collected. The set should be large enough to include most of the ST-shape variations we wish to model. The ST-shapes in the training set are then parameterized. A data dimensionality reduction stage is then performed by capturing only the main modes of ST-shape variation. In addition to constructing the ST-shape model, the training stage also includes the modeling of gray-level information. The task is then to locate an ST-shape given a new unknown image sequence. An average ST-shape is first initialized then ‘optimal’ deformations are proposed and forced to agree with the training data. The proposed changes minimize a cost function that takes into account both temporal and appearance constraints. The search for the optimal proposed change is done using dynamic programming.

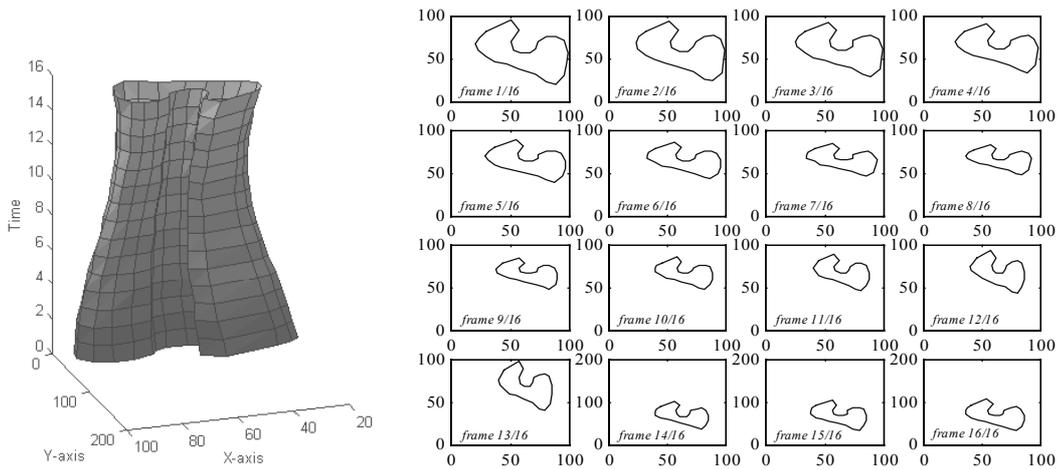


Figure 5.1. Spatio-temporal shape example (consisting of 16 frames).

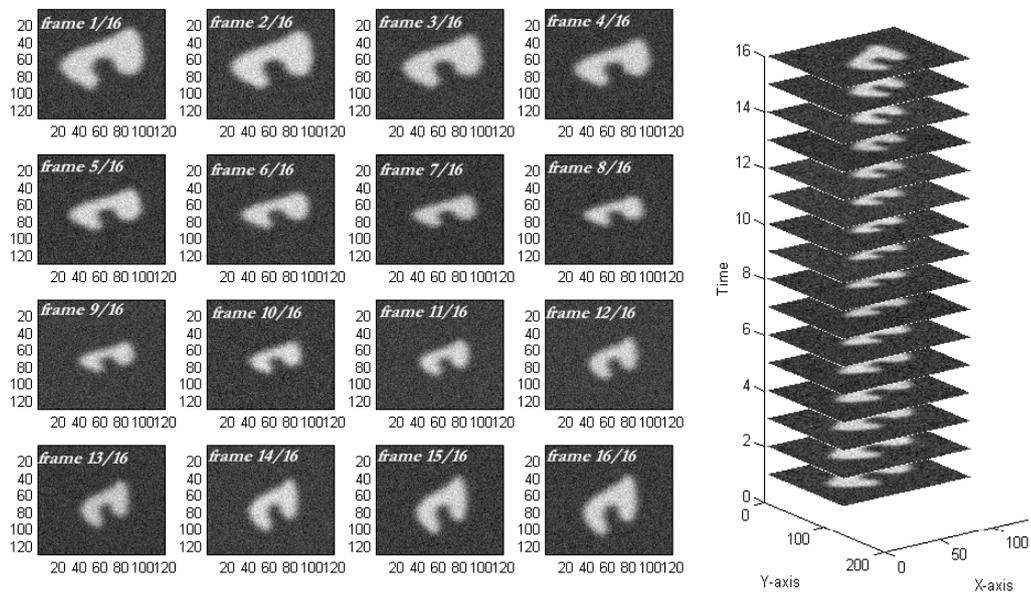


Figure 5.2. Frame sequence example (consisting of 16 frames).

5.2 Methods

5.2.1 Statistical ST-Shape Variation

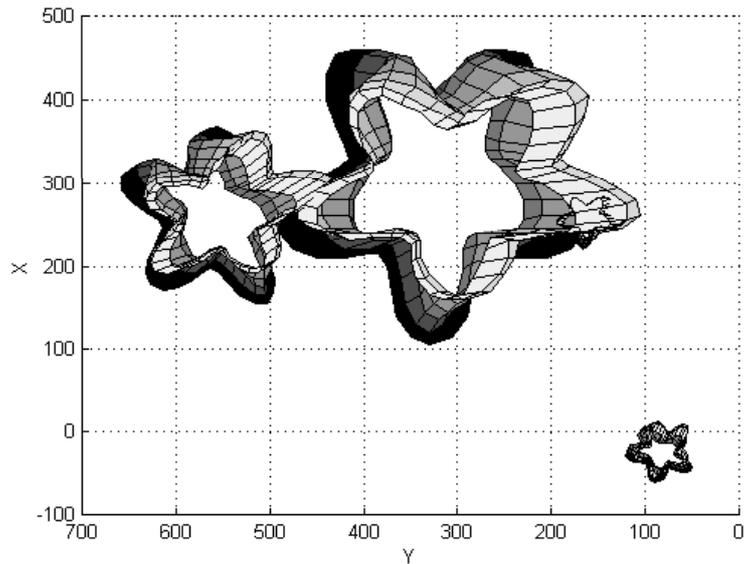
The training set. N training frame-sequences each with F frames are collected. The training set, $\Phi_V = [V_1, V_2, \dots, V_N]$, displays similar object motion patterns. $\Phi_V(i) = V_i = [f_{i1}, f_{i2}, \dots, f_{iF}]$ is the i^{th} frame-sequence containing F frames (Figure 5.2) and $V_i(j) \equiv \Phi_V(i, j) = f_{ij}$ is the j^{th} frame of the i^{th} frame-sequence containing the intensity value $f_{ij}(r, c) \equiv \Phi_V(i, j, r, c)$ at the r^{th} row and c^{th} column of the frame.

The ST-shape parameters. S_i is introduced to denote the parameter vector representing the i^{th} ST-shape. Parameterization is done using landmarks² that are labeled either manually, as when a cardiologist labels the heart chamber boundaries [Cootes1995a, Hill1993], or (semi-) automatically as reported, for example, in [Hill1994]. Each landmark point is represented by (x, y) coordinates. Using L landmarks per frame and F frames per sequence, the training set of ST-shapes is written as $\Phi_S = [S_1, S_2, \dots, S_N]$, where $\Phi_S(i) = S_i = [p_{i1}, p_{i2}, \dots, p_{iF}]$ is the i^{th} ST-shape containing F shapes and $S_i(j) \equiv \Phi_S(i, j) = p_{ij}$ is the j^{th} shape of the i^{th} ST-shape. p_{ij} is written as $p_{ij} = [x_{ij1}, y_{ij1}, x_{ij2}, y_{ij2}, \dots, x_{ijL}, y_{ijL}]$, where $x_{ijk} = p_{ij}(k, 1) \equiv \Phi_S(i, j, k, 1)$ and $y_{ijk} = p_{ij}(k, 2) \equiv \Phi_S(i, j, k, 2)$ are the x and y coordinates of the k^{th} landmark of the shape p_{ij} .

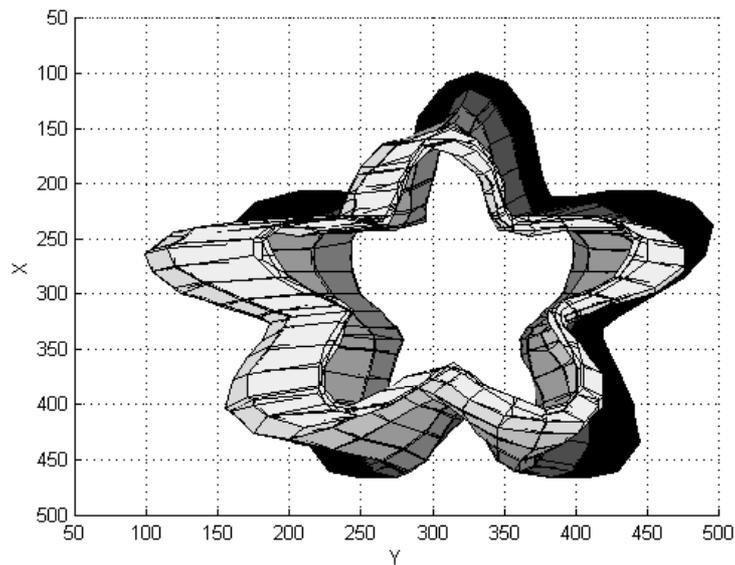
ST-shapes alignment. ST-shapes are aligned to allow comparing equivalent points from different ST-shapes. This is done by rotating, scaling and translating the shape in each frame by an amount that is fixed within one ST-shape (Figure 5.3 and Figure 5.4). A weighted least-squares approach is used for aligning two sequences and an iterative algorithm is used to align all the ST-shapes (details of aligning ST shapes can be found in Appendix E). If the observed motion patterns in the training sequences span different time

² Other shape parameterization methods may be utilized, e.g. Fourier descriptors [Bonciu1998] or B-Splines [Stark1996].

intervals, temporal re-sampling or aligning that incorporates temporal scaling might be needed. If these differences are insignificant, their effects may be interpreted and modeled as shape variations.



(a)



(b)

Figure 5.3. ST-shapes before alignment (a) and after alignment (b), shown projected on the x - y plane.

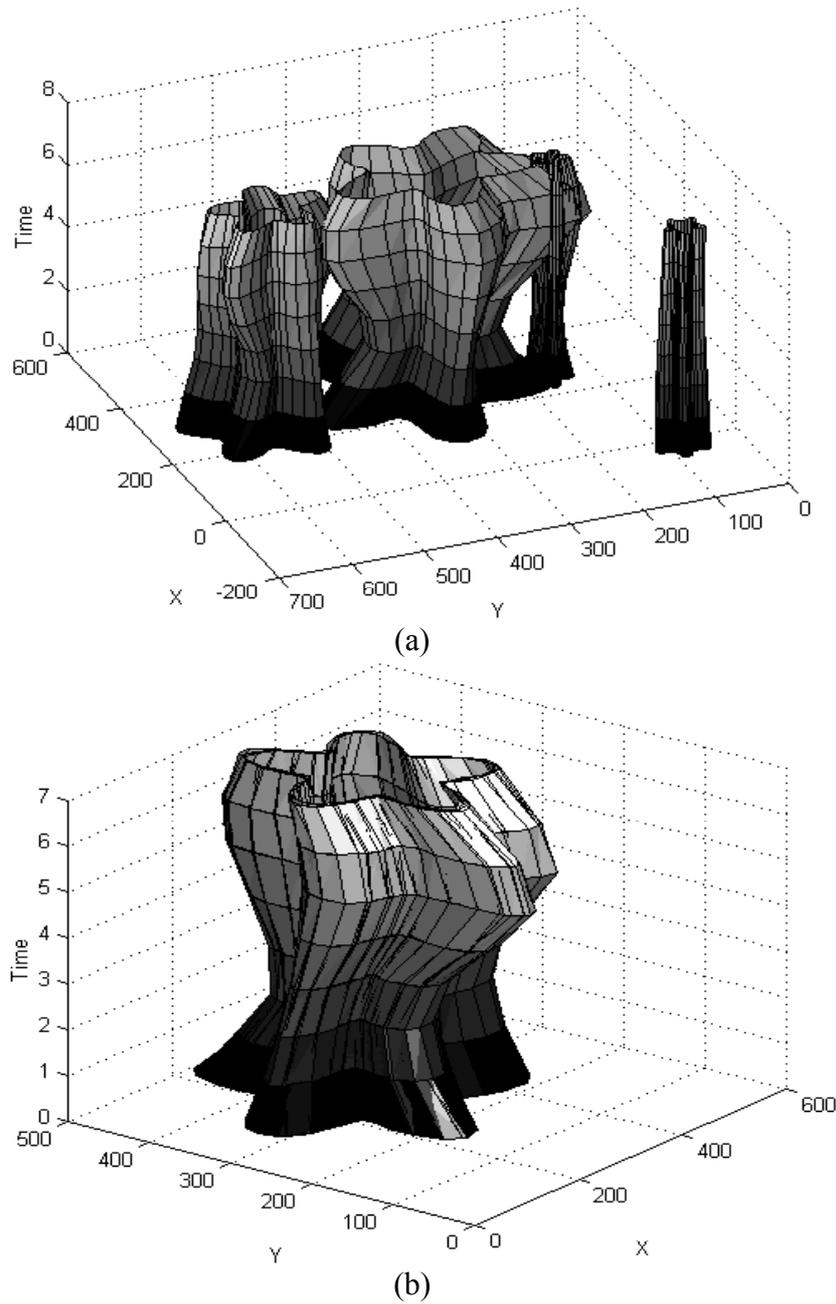


Figure 5.4. ST-shapes before alignment (a) and after alignment (b), shown in 3D (2D-time).

Main ST-shape variation modes. The N aligned ST-shapes, each of length $2FL$ and now represented by $\{S_1, S_2, \dots, S_N\}$, map to a ‘cloud’ of N points in a $2FL$ dimensional space. It is assumed that these N points are contained within a region of this $2FL$ dimensional space we dub the Allowable ST-Shape Domain (ASTSD). Principal Component Analysis (PCA) of the aligned ST-shapes is performed to find the main modes of ST-shape variation (PCA is explained in Appendix C). The resulting PCs are the eigenvectors \mathbf{p}_k , $1 \leq k \leq 2FL$, of the observations covariance matrix, C_S

$$C_S \mathbf{p}_k = \lambda_k \mathbf{p}_k \quad (5.1)$$

where

$$C_S = \frac{1}{N-1} \sum_{i=1}^N (S_i - \bar{S})^T (S_i - \bar{S}) \quad (5.2)$$

λ_k is the k^{th} eigenvalue of C_S ($\lambda_k \geq \lambda_{k+1}$) and is equal to the variance along the k^{th} PC. The mean ST-shape is calculated as

$$\bar{S} = \frac{1}{N} \sum_{i=1}^N S_i. \quad (5.3)$$

The principal components are normalized to unit length and are mutually orthogonal.

Model representation. Each ST-shape, S_i , is expressed as the sum of the mean ST-shape, \bar{S} , and a linear combination of the principal modes of variation, $\mathbf{P}\mathbf{b}_i$, i.e.

$$S_i = \bar{S} + \mathbf{P}\mathbf{b}_i \quad (5.4)$$

where $\mathbf{b}_i = [b_{i,1} \ b_{i,2} \ \dots \ b_{i,2FL}]^T$ and $\mathbf{P} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \dots \ \mathbf{p}_{2FL}]$. b_l is constrained to $b_{l_{\min}} \leq b_l \leq b_{l_{\max}}$ with $b_{l_{\min}} = -b_{l_{\max}}$ and $1 \leq l \leq 2FL$. $b_{l_{\max}}$ is chosen to be proportional to λ_l .

Assuming that the observations form a hyper-ellipsoid in $2FL$ dimensions, then the eigenvectors of the covariance matrix corresponding to the largest eigenvalues describe the longest axes of the ellipsoid. Those eigenvectors describe the most significant variation modes of the variables. The variance explained by each eigenvector is equal to its corresponding eigenvalue. The $2FL$ dimensional ellipsoid can be approximated by a t dimensional one by taking the first t eigenvectors. The N vectors representing the ST-shapes are expressed as the sum of the mean shape and a weighted sum of some of the

principal components. We assume that these t (out of $2FL$) principal components explain a sufficiently high percentage of the total variance in the original data. Intuitively, this means that the $2FL$ dimensional cloud of ST-shapes has a ‘narrow’ width in the direction of the k^{th} principal components for $k = t + 1, t + 2, t + 3, \dots, 2FL$. The fundamental model equation now becomes

$$S = \bar{S} + \mathbf{P}\mathbf{b} \quad (5.5)$$

where $\mathbf{b} = [b_1 \ b_2 \ \dots \ b_t]^T$, $\mathbf{P} = [\mathbf{p}_1 \ \mathbf{p}_2 \ \dots \ \mathbf{p}_t]$, and t is the smallest value satisfying

$$\frac{\sum_{i=1}^t \lambda_i}{\sum_{i=1}^{2FL} \lambda_i} \geq f. \quad (5.6)$$

f is the fraction of explained variance. The constraints on \mathbf{b} become $b_{l_{\min}} \leq b_l \leq b_{l_{\max}}$, where $1 \leq l \leq t$.

5.2.2 Gray-Level Training

The ST-shape model alone is typically not enough for spatio-temporal segmentation. Therefore, additional representative information about the intensities or gray-levels relating to the object is also desired and collected in the gray-level training stage. In the search stage, new estimates of the ST-shape are sought that will better match the gray-level prior knowledge. Different gray-level representative information can be used, e.g. gathering the intensity values in the entire patch contained within the object ([Cootes1998]) or using a parametric description of the profiles or patches around the landmark. In this implementation we follow [Cootes1995a] and use a mean normalized derivative (difference) profile, passing through each landmark and perpendicular to the boundary created by the neighboring ones. The profile of the k^{th} landmark in the j^{th} shape of the i^{th} ST-shape can be written as

$$\mathbf{g}_{ijk} = [g_{ijk1} \ g_{ijk2} \ \dots \ g_{ijkG}] \quad (5.7)$$

where $1 \leq i \leq N$, $1 \leq j \leq F$, $1 \leq k \leq L$, and G is the length of the profile. The derivative profile (of length $G - 1$) is written as

$$d\mathbf{g}_{ijk} = [g_{ijk2} - g_{ijk1} \ g_{ijk3} - g_{ijk2} \ \dots \ g_{ijkG} - g_{ijk(G-1)}] \quad (5.8)$$

and the normalized derivative profile as

$$\mathbf{y}_{ijk} = \frac{d\mathbf{g}_{ijk}}{G-1} \cdot \frac{1}{\sum_{l=1}^G |d\mathbf{g}_{ijkl}|} \quad (5.9)$$

The mean normalized derivative profile for each landmark is calculated as

$$\bar{\mathbf{y}}_k = \frac{1}{FN} \sum_{j=1}^F \sum_{i=1}^N \mathbf{y}_{ijk} \quad (5.10)$$

where \mathbf{y}_{ijk} is the representative profile for the k^{th} landmark in the j^{th} shape of the i^{th} ST-shape. Using the gray-level information along with temporal and shape constraints, the model is guided towards a better estimate of the dynamic object hidden in the new frame-sequence. Here it is assumed that a landmark's profile doesn't change considerably between frames, otherwise we could obtain a different mean profile for each landmark in each frame (resulting in FL mean profiles). $\bar{\mathbf{y}}_k$ is used in the search stage for locating better positions of the landmarks.

5.2.3 ST-Shape Segmentation Algorithm

Given a new frame-sequence, the task is to locate the object in all the frames or equivalently, locate the ST-shape. An initial estimate of the ST-shape parameters is chosen at first, then changes to the parameters are proposed. The pose of the current estimate is then modified and suitable weights for the modes of variation are chosen in order to fit the model to the proposed changes. This is done with the restriction that the changes can only be made in accordance with the model (with reduced dimensionality) and the training set. New changes are then proposed and so on. Here we present a detailed discussion of these steps.

Initial estimate. An initial ST-shape $\hat{S}^{(0)}$ is used

$$\hat{S}^{(0)} = M(s^{(0)}, \theta^{(0)})[\bar{S} + \mathbf{P}\mathbf{b}^{(0)}] + \mathbf{t}^{(0)} \quad (5.11)$$

where $\mathbf{t} = [t_x \ t_y \ t_x \ t_y \ \dots \ t_x \ t_y]$ is of length $2FL$. $M(s, \theta)[S] + \mathbf{t}$ scales, rotates, and translates S by s , θ , and \mathbf{t} , respectively. Both \bar{S} and \mathbf{P} are obtained from the training stage. A typical initialization would set $\mathbf{b}^{(0)}$ to zero, and $s^{(0)}$, $\theta^{(0)}$, and $\mathbf{t}^{(0)}$ to values that put the initial sequence in the vicinity of the target.

Proposing a new sequence. The next step is to calculate a proposed change in the ST-shape. This can be done in different ways; the one we describe is based on the minimization of a cost function using dynamic programming. In

summary, we make use of the gray level information in addition to time-smoothness constraints to find where each landmark of the ST-shape is suggested to move. We deal with a specific landmark, in all the frames, separately. The idea is to search in the neighborhood of the landmark in all the frames in order to find the ‘best’ combination of changes to their positions. The best changes minimize a cost function comprising two terms: a cost due to large landmark position changes with time, and a cost reflecting the mismatch between the gray levels surrounding the current landmark and expected values found from training. The minimization is done using dynamic programming.

Specifically, for each landmark, say the k^{th} landmark in the j^{th} frame, a search profile $\mathbf{h}_{jk} = [h_{jk1} \quad h_{jk2} \quad \dots \quad h_{jkH}]$ is defined. It is differentiated and normalized as in the training profiles (Figure 5.5(a-b)). This gives H^F possibilities for the proposed positions of the k^{th} landmarks in the F frames (Figure 5.6). Since locating the new positions (one out of H^F possible) is computationally demanding, the problem is formulated as a multi-stage decision process and dynamic programming ([Amini1990, Wendelhag1997, Gustavsson1997]) is used to find the optimal positions. In the following paragraphs, we detail our implementation of dynamic programming.

A gray-level mismatch value $M_k(j, l)$ is calculated for each point along each search profile in all the frames according to

$$M_k(j, l) = (\mathbf{h}_{jk}(l) - \bar{\mathbf{y}}_k)^T \mathbf{W}^T \mathbf{W} (\mathbf{h}_{jk}(l) - \bar{\mathbf{y}}_k) \quad (5.12)$$

where $1 \leq k \leq L$, $1 \leq j \leq F$, $1 \leq l \leq H$, $\mathbf{h}_{jk}(l)$ is a sub-profile of length $G - 1$ anchored at the l^{th} location of the search profile \mathbf{h}_{jk} , and \mathbf{W} is a weighting matrix (Figure 5.5(d-e)). A temporal discontinuity value, $d_{jk}^2(l_j, l_{j-1})$, is calculated, corresponding to moving the k^{th} landmark in frame $j - 1$ to location l_{j-1} and the k^{th} landmark in frame j to location l_j , each along its respective search profile (Figure 5.5(f-h)). d_{jk} is given by

$$d_{jk}^2(l_j, l_{j-1}) = (\mathbf{c}_{jkx}(l_j) - \mathbf{c}_{j-1kx}(l_{j-1}))^2 + (\mathbf{c}_{jky}(l_j) - \mathbf{c}_{j-1ky}(l_{j-1}))^2 \quad (5.13)$$

where $\mathbf{c}_{jkx} = [x_{jk1} \quad x_{jk2} \quad \dots \quad x_{jkH}]$ and $\mathbf{c}_{jky} = [y_{jk1} \quad y_{jk2} \quad \dots \quad y_{jkH}]$ are the search profile coordinates relating to the k^{th} landmark in the j^{th} frame (Figure 5.5(b)).

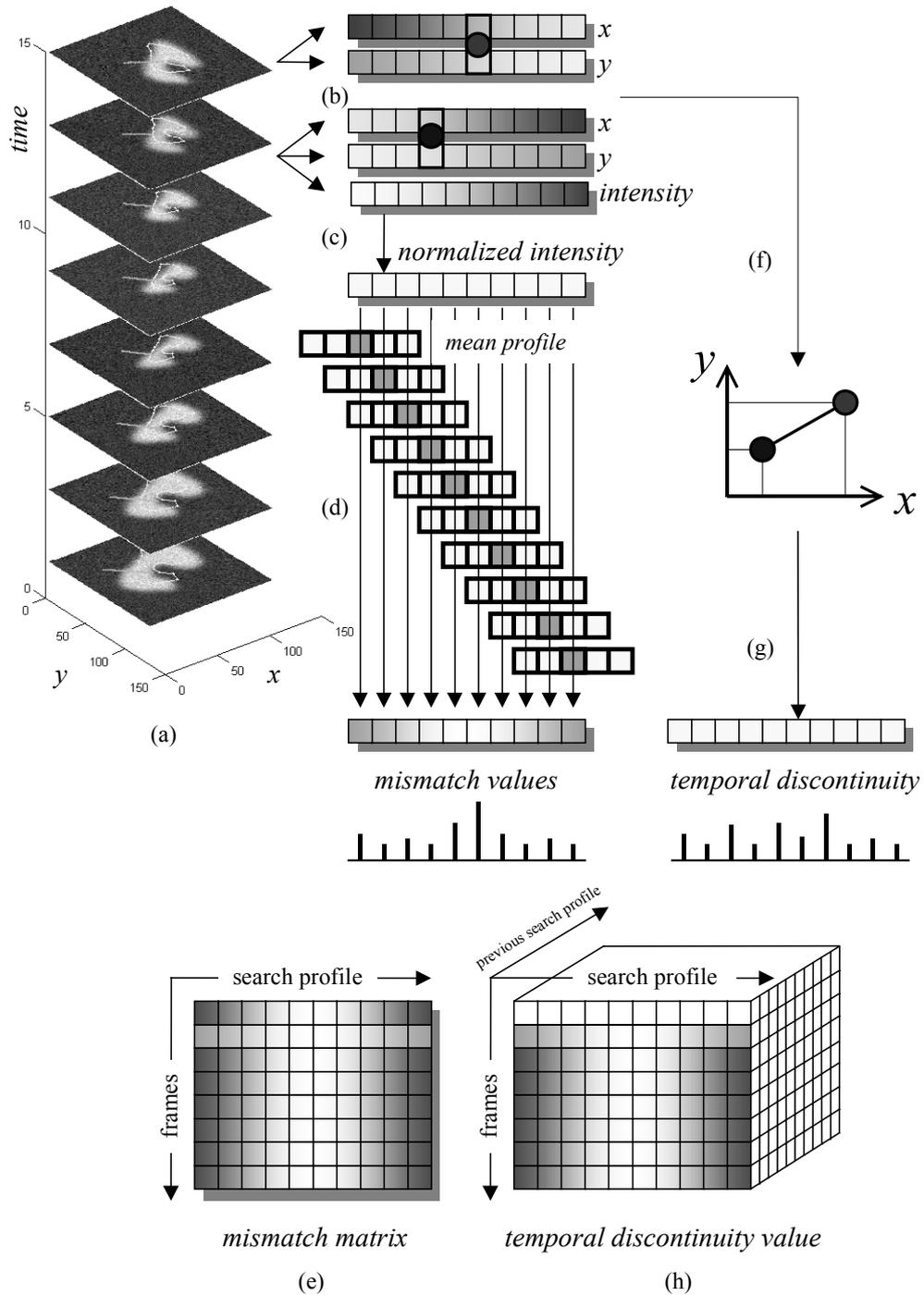


Figure 5.5. Schematic diagram: Obtaining the mismatch matrix and the temporal discontinuity values.

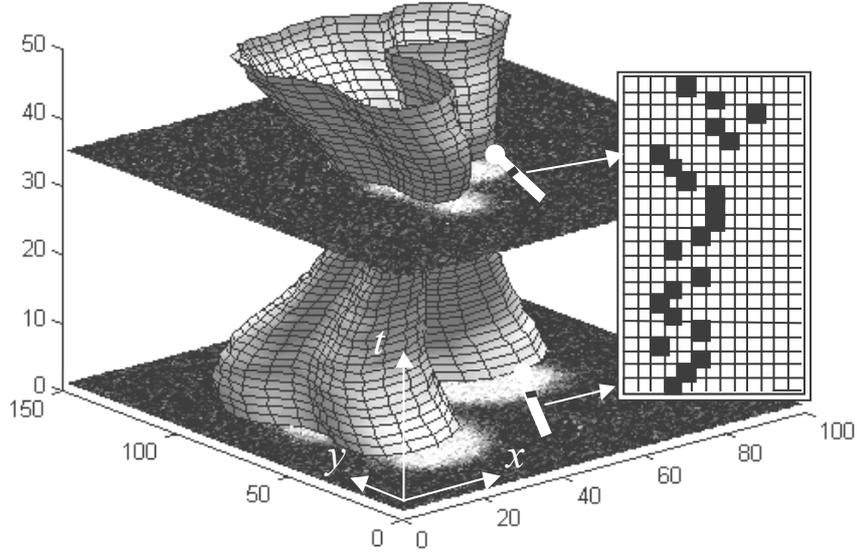


Figure 5.6. An illustration of an ST-shape overlaid on an image sequence. The search profiles of one landmark in 2 frames are shown in white. Examples of proposed landmark positions are shown as black squares.

The accumulated costs of moving the k^{th} landmark to the l^{th} position in the j^{th} frame, $2 \leq j \leq F$, from any of the H positions in frame $j-1$ is compared and the least value is assigned to $A_k(j, l)$, i.e.

$$A_k(j, l) = \min \{t_{jkl1}, t_{jkl2}, \dots, t_{jklH}\} \quad (5.14)$$

where

$$t_{jklm} = w_d d_{jk}(l, m) + w_m M_k(j, l) + A_k(j-1, m). \quad (5.15)$$

w_d and w_m satisfy $w_d + w_m = 1$ and control the relative importance of temporal discontinuity and gray-level mismatch (Figure 5.7(a-d)). Furthermore, an index or a pointer, $P_k(j, l)$, is assigned to the location of the best landmark in the previous frames (Figure 5.7(e)). The same procedure is applied to the k^{th} landmark in all the F frames yielding $F \times H$ accumulated values and $F \times H$ pointers (no temporal discontinuity cost is associated with the first frame).

To find the proposed positions of the k^{th} landmark in all the frames, the location, m_F , of the minimum accumulated cost, along the search profile of

the landmark in the last frame F , is found. m_F is used to find the proposed landmark position in the second last frame, frame $F - 1$, as $m_{F-1} = P_k(F, m_F)$ (Figure 5.7(f)). Its coordinates will be $(\mathbf{c}_{F-1kx}(m_{F-1}), \mathbf{c}_{F-1ky}(m_{F-1}))$. In general the proposed coordinates of the k^{th} landmark of the j^{th} frame are given by

$$(x, y) : (\mathbf{c}_{jkx}(m_j), \mathbf{c}_{jky}(m_j)) \quad (5.16)$$

with

$$m_j = P_k(j + 1, m_{j+1}) \quad (5.17)$$

Tracking back to the first frame, the coordinates of the proposed positions of the k^{th} landmark in all frames are acquired (Figure 5.7(g)). In the same way, the proposed positions for all the landmarks ($1 \leq k \leq L$) defining the ST-shape changes $d\hat{S}_{proposed}^{(0)}$, are obtained.

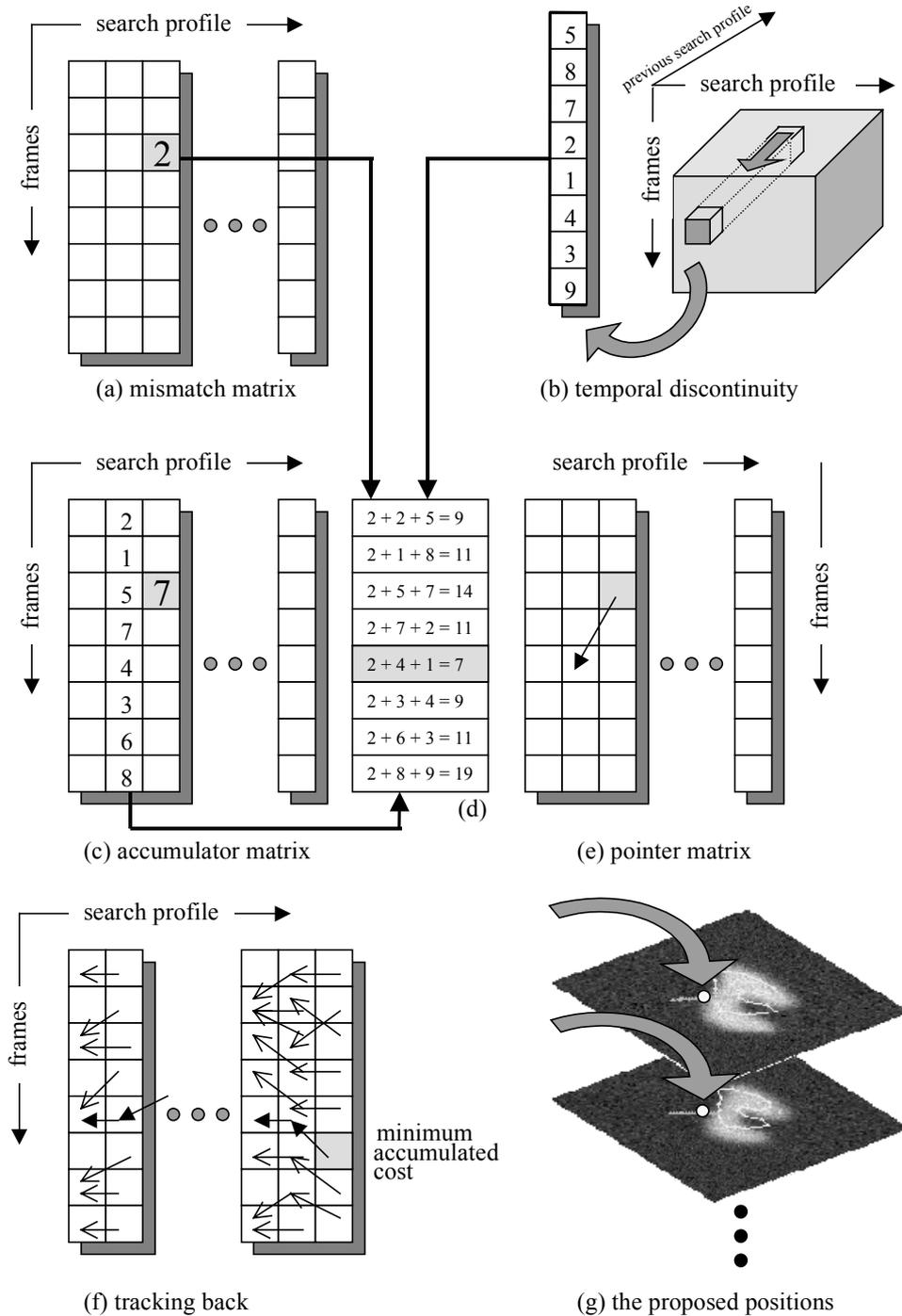


Figure 5.7. Schematic diagram: Using dynamic programming to obtain the proposed landmark positions in all frames.

Limiting the proposed ST-shape. Since, in general, the proposed ST-shape ($\hat{S}^{(0)} + d\hat{S}_{proposed}^{(0)}$) will not conform to our model of reduced dimensionality and will not lie in the allowable spatio-temporal shape domain (ASTSD), it cannot be accepted as an ST-shape estimate. Therefore, an acceptable ST-shape that is closest to the proposed one is sought. This is done by first calculating the pose parameters ($s^{(1)}, \theta^{(1)}$, and $\mathbf{t}^{(1)}$) that will align \bar{S} to $\hat{S}^{(0)} + d\hat{S}_{proposed}^{(0)}$ by mapping \bar{S} to $M(s^{(1)}, \theta^{(1)})[\bar{S}] + \mathbf{t}^{(1)}$, then finding the extra ST-shape modifications $dS^{(1)}$ which, when combined with the pose parameters, will map exactly to $\hat{S}^{(0)} + d\hat{S}_{proposed}^{(0)}$. The latter is done by solving the following equation for $dS^{(1)}$

$$M(s^{(1)}, \theta^{(1)})[\bar{S} + dS^{(1)}] + \mathbf{t}^{(1)} = \hat{S}^{(0)} + d\hat{S}_{proposed}^{(0)} \quad (5.18)$$

hence obtaining

$$dS^{(1)} = M(s^{(1)}, \theta^{(1)})^{-1}[\hat{S}^{(0)} + d\hat{S}_{proposed}^{(0)} - \mathbf{t}^{(1)}] - \bar{S} \quad (5.19)$$

where $M(s^{(1)}, \theta^{(1)})^{-1} = M((s^{(1)})^{-1}, -\theta^{(1)})$. In order to find the new shape parameters, $\mathbf{b}^{(1)}$ we need to solve $dS^{(1)} = \mathbf{P}\mathbf{b}^{(1)}$, which, in general, has no solution since $dS^{(1)}$ lies in a $2FL$ dimensional space whereas \mathbf{P} spans only a t dimensional space. The best solution in a least-squares sense is obtained as

$$\mathbf{b}^{(1)} = \mathbf{P}^T dS^{(1)} \quad (20)$$

Finally, using the constraints discussed earlier, $b_{l_{\min}} \leq b_l \leq b_{l_{\max}}$ where $1 \leq l \leq t$, the ST-shape variations are constrained to obtain an acceptable or allowable shape within the ASTSD.

Updating the estimate and reiterating. Similarly, new ST-shape estimates can be obtained, i.e.,

$$\begin{aligned} \hat{S}^{(i)} &= M(s^{(i)}, \theta^{(i)})[\bar{S} + \mathbf{P}\mathbf{b}^{(i)}] + \mathbf{t}^{(i)} \\ &\rightarrow \hat{S}^{(i+1)} = M(s^{(i+1)}, \theta^{(i+1)})[\bar{S} + \mathbf{P}\mathbf{b}^{(i+1)}] + \mathbf{t}^{(i+1)} \end{aligned} \quad (5.21)$$

for $i = 1, 2, 3, \dots$. Checking for convergence can be done by examining the changes in parameters. If the new estimate is not much different (according to some predefined threshold) then the search is completed, otherwise we reiterate.

5.3 Results

5.3.1 Synthetic ST-Shapes and Frame Sequences

We tested the method on synthetic data (Figure 5.8). A single synthetic example consisted of an ST-shape and a frame-sequence. The ST-shape data is first calculated and then used to generate the frame-sequence. The ST-shapes are represented by a set of coordinates describing the shapes in all the frames. Each synthetic ST-shape consists of F frames. Each frame contains L landmark coordinates. Both the x and the y coordinates of each landmark move within a sequence according to sinusoidal functions with certain amplitudes and frequencies. The positions of the landmarks in the first frame and the amplitudes and frequencies of the sinusoidal functions are sampled from Gaussian distributed functions with given means and variances. This is done to produce similar ST-shapes to be used in the training stage. After the ST-shapes are produced, binary images are generated for all the frames in the sequences, by ‘filling’ the polygon areas generated from the landmark coordinates. Then the binary frame-sequences are smoothed by convolution with a Gaussian kernel. Noise and occlusions are added when producing a frame-sequence for testing the search algorithm.

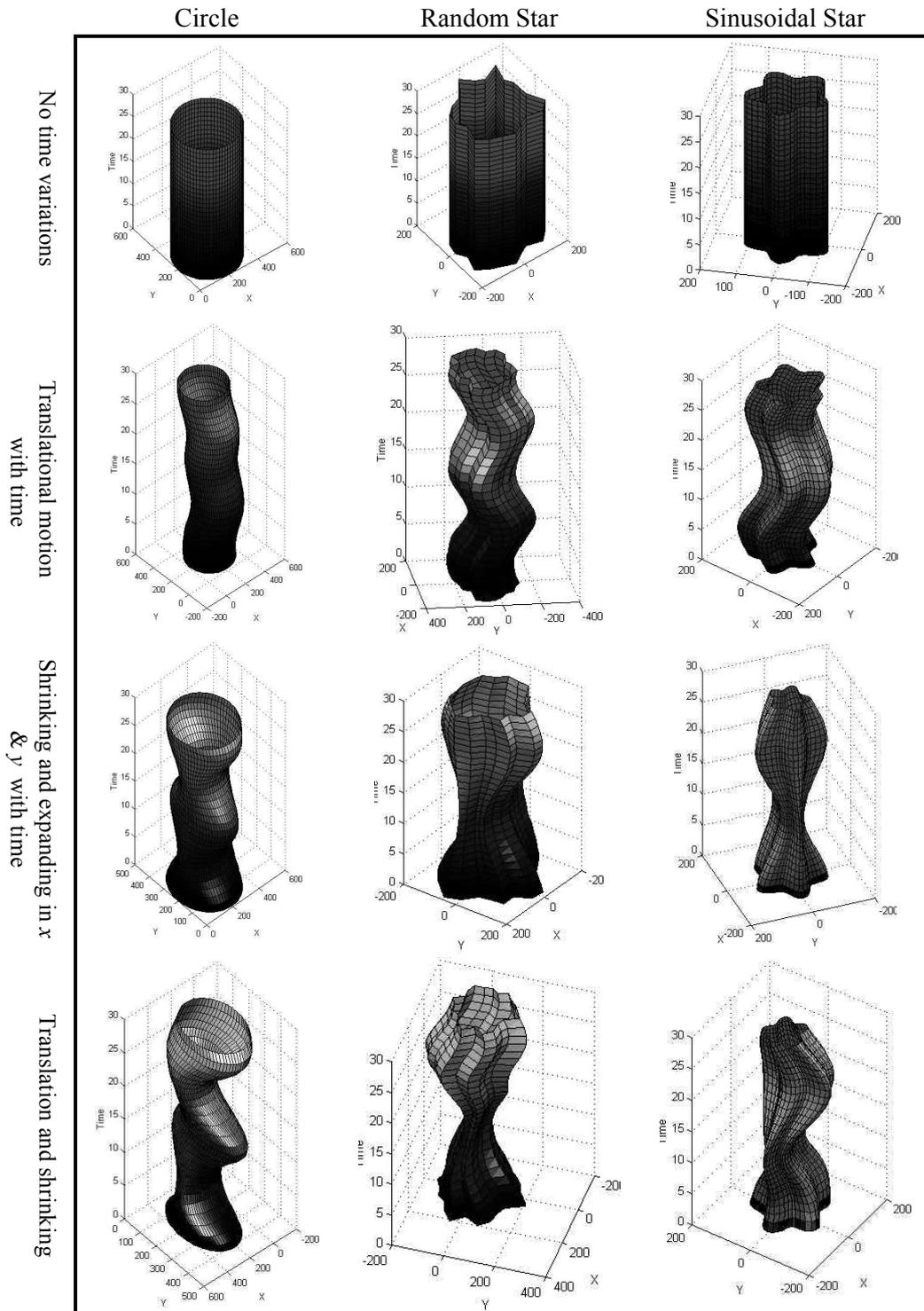


Figure 5.8. Examples of synthetic ST-shapes.

5.3.2 Frame Sequence Imperfections

For the purpose of producing image sequences that imitate real-life imagery including artifacts, the synthetically generated image sequences used for both training and testing were deteriorated in different ways as listed below.

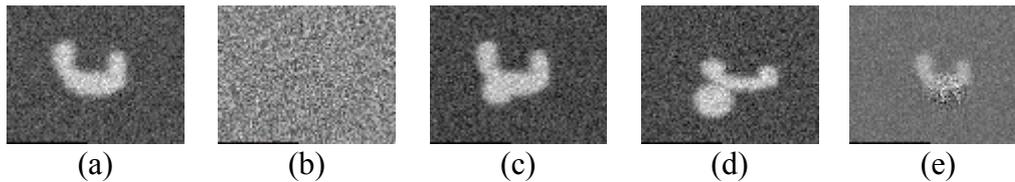


Figure 5.9. Examples of synthetic frames with imperfections due to (a) global noise, (b) missing frame, (c) overlapping occlusion, (d) touching occlusion, and (e) local noise.

Global noise. For each pixel in the 3D test data sets, white Gaussian noise was added (Figure 5.9(a)). The choice of Gaussian type noise is somewhat arbitrary. It should, however, sufficiently well serve the purpose of evaluating the robustness of the ST-shape segmentation in presence of noise.

Missing frames. In many applications dealing with analyzing image sequences, the sequence digitization may be disturbed by badly synchronized frames, frames including strong artifacts, or missing frames. Therefore, it is important that studies can be done on data sets even if there are some missing frames (Figure 5.9(b)).

Overlapping occlusion. In most tomographic techniques it is hard to select the view so that the object under examination can be seen clearly throughout its cross-section. Given a specific view, the object maybe partly occluded by overlapping occlusions. Therefore, in the synthetic data set we have constructed circular objects having varying radii and appearing at varying positions throughout the spatio-temporal space (Figure 5.9(c)).

Touching occlusion. In many cases, the object under examination is not free-lying but has neighboring objects touching at the boundary, as can be found in MRI scans. This is problematic in segmentation especially when the touching object has similar texture. Therefore, we have constructed a data set that, besides the ST-shape under consideration, includes a touching circular object appearing at varying positions throughout the spatio-temporal space (Figure 5.9(d)).

Local noise. Besides global noise due to shortcomings in the imaging procedures, there may also exist local noise deteriorating the image of the object under examination. In ultrasonic cross-sectional imaging of the left ventricle, for example, there are often noisy areas where the ultrasound beam is almost tangential to the ventricular boundary. Therefore, we have constructed a data set with local white additive noise of Gaussian type appearing at specific regions in the frames (Figure 5.9(e)). These regions have been chosen to be circular and the noise strength decays radially from the center of the circle following a Gaussian function.

5.3.3 Gray-Level Training

In practice, the gray level profile training takes place on digitized images contaminated with noise. This means that the profile variations found are not only due to variations in the appearance of the object alone but also includes variations due to other factors. In order to test the influence of different factors we constructed and trained our model on the following data sets.

An ideal case: Object perfectly reconstructed by the imaging procedure but with the presence of random noise. The resulting data set is a noised binary image sequence in which the binary object is subject to noise but otherwise perfect.

The sub-ideal case: Object subject to degradations such as smoothing through the imaging procedure as well as random noise. The resulting data set is a smoothed and noised binary image sequence.

The realistic case: Object subject to degradations such as smoothing as well as some imperfections as those described in Figure 5.9. Resulting in a smoothed, occluded, and noised binary image sequence.

5.3.4 Results on Synthetic Data

The test data sets. In all the test cases, training was performed using 10 image sequences. Each sequence consisted of 16 frames ($F=16$). The size of each frame was 160×182 pixels. Each ST-shape that generated an image sequence consisted of 16 contours (one at each frame). Each contour was represented by 25 landmarks (x, y coordinates), ($L=25$). The gray level search was conducted on a profile of length 41 pixels (20 inside the object and 20 outside it) and the gray level training profile was of length 13 (7 inside the object and 5 outside it). The testing was performed on an image sequence different from those in the training set but produced similarly, i.e. a cross validation procedure was applied. In our model we used 6 ST-shape parameters sufficient to describe

98% of the ST-shape variations. Smoothing the binary images was done using a Gaussian kernel of size 17×17 .

This section summarizes the details of the data used for both training and testing. Seven types of tests are presented. In each type we try to detect an ST-shape hidden in a test image sequence. The test image sequence suffers from different types of degradations specified under Imperfections in Table 5.1. The degradations may appear in only one frame, in some consecutive frames, in some separated frames, or in all the frames. The table also lists which frames are affected. Different image sequences are used for training as mentioned in 5.3.3. The table includes the type of training used, as well.

Table 5.1. Summary of the test data sets.

Test case	Imperfection (global noise in all cases)	Frames affected (global noise in all)	Training set (global noise in all)
1	Missing frames	Separated frames	Smoothed and noised
2	Overlapping occlusion	All frames	Smoothed and noised
3	Overlapping occlusion	Separated frames	Smoothed and noised
4	Touching occlusion	Consecutive frames	Smoothed and noised
5	Local noise	All frames	Smoothed and noised
6	Overlapping occlusion	All frames	Noised
7	Overlapping occlusion	All frames	Smoothed, occluded and noised

Note: the frame sequences are displayed in the order shown below.

1/16	2/16	3/16	4/16
5/16	6/16	7/16	8/16
9/16	10/16	11/16	12/16
13/16	14/16	15/16	16/16

Following are the different test cases. For each case a table summarizing the test conditions, a figure depicting the segmentation results, and comments are presented.

Test Case One: Missing Frames - A

Table 5.2. Missing frames: Example 1A. See Figure 5.1.

Imperfection	Missing frames (and global noise in all frames)
Frames affected	Frame 5/16
Training set	Smoothed and noised
Number of iterations	16
Global noise variance	100
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30

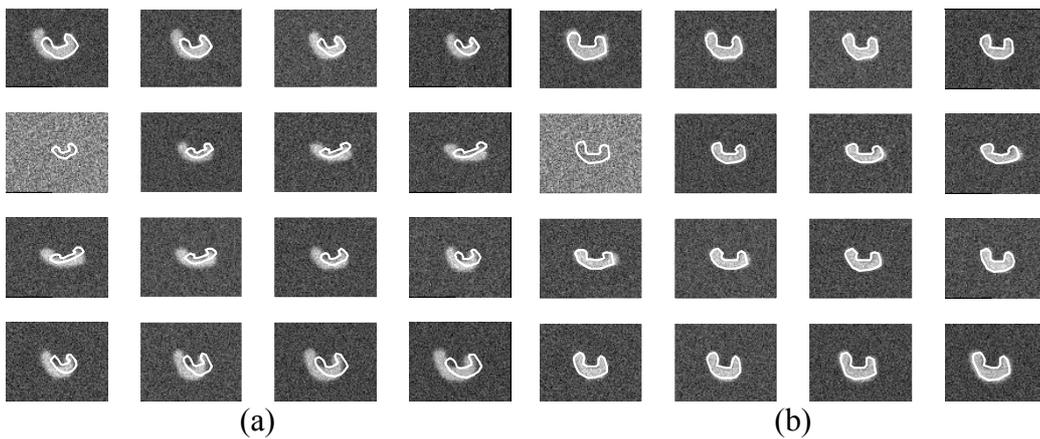


Figure 5.1. Missing frames: Example 1A. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object in all the frames and a reasonable guess was produced for the missing frame.

Test Case One: Missing Frames - B

Table 5.3. Missing frames: Example 1B. See Figure 5.2.

Imperfection	Missing frames (and global noise in all frames)
Frames affected	Separated frames: 3, 9, and 14/16
Training set	Smoothed and noised
Number of iterations	23
Global noise variance	100
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30

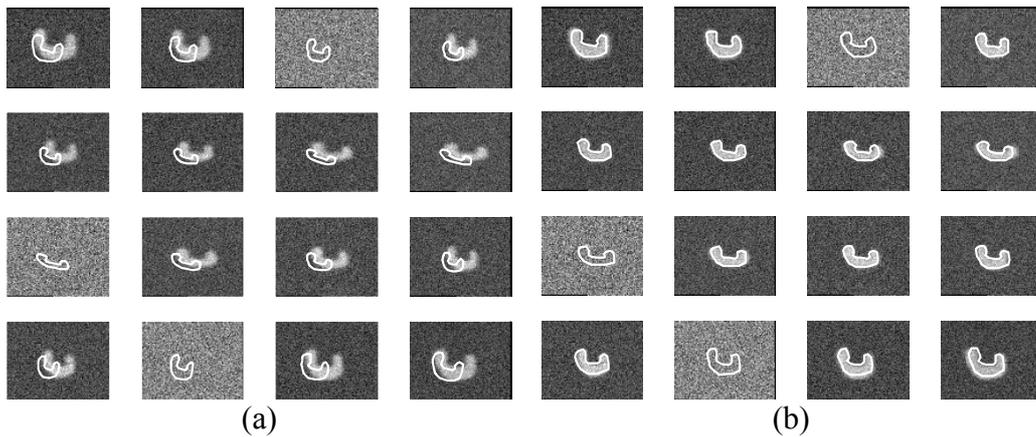


Figure 5.2. Missing frames: Example 1B. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object in all the frames and reasonable guesses were produced for the separated missing frames.

Test Case Two: Overlapping Occlusion I

Table 5.4. Overlapping occlusion: Example 2. See Figure 5.3.

Imperfection	Overlapping occlusion (and global noise in all frames)
Frames affected	All frames
Training set	Smoothed and noised
Number of iterations	15
Global noise variance	100
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30
Radius of occlusion	15 pixels

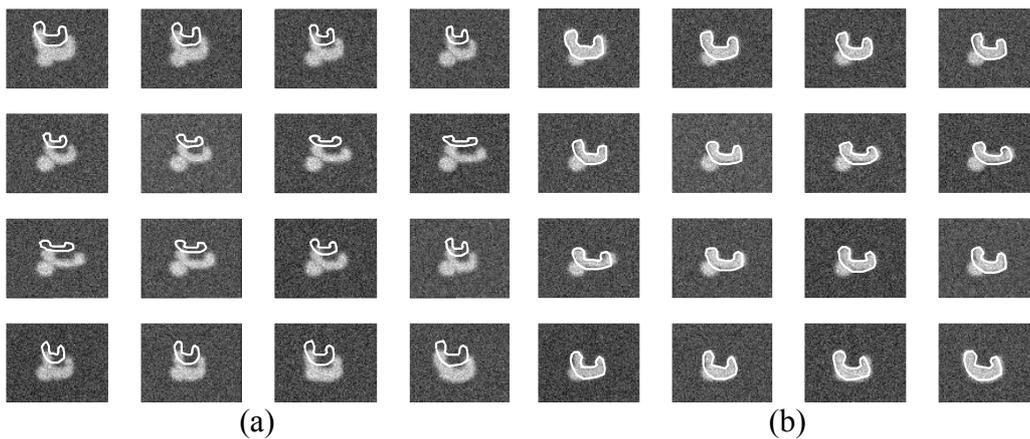


Figure 5.3. Overlapping occlusion: Example 2. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object overcoming the problem of overlapping occlusions that appeared in all the frames.

Test Case Three: Overlapping Occlusion II

Table 5.5. Overlapping occlusion: Example 3. See Figure 5.4.

Imperfection	Overlapping occlusion (and global noise in all frames)
Frames affected	Separated frames: 3, 6, 11 and 14/16
Training set	Smoothed and noised
Number of iterations	14
Global noise variance	100
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30
Radius of occlusion	20, 18, 30, and 15 pixels, respectively

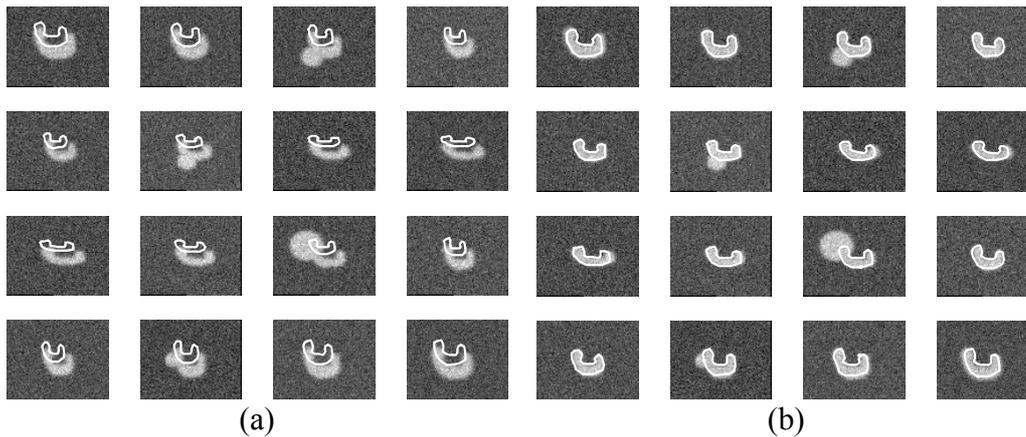


Figure 5.4. Overlapping occlusion: Example 3. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object in all the frames and overcame the problem of overlapping occlusions of different sizes that existed in some separated frames.

Test Case Four: Touching Occlusion

Table 5.6. Touching occlusion: Example 4. See Figure 5.5.

Imperfection	Touching occlusion (and global noise in all frames)
Frames affected	Consecutive frames: 7, 8, 9 and 10/16
Training set	Smoothed and noised
Number of iterations	15
Global noise variance	100
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30
Radius of occlusion	20 pixels

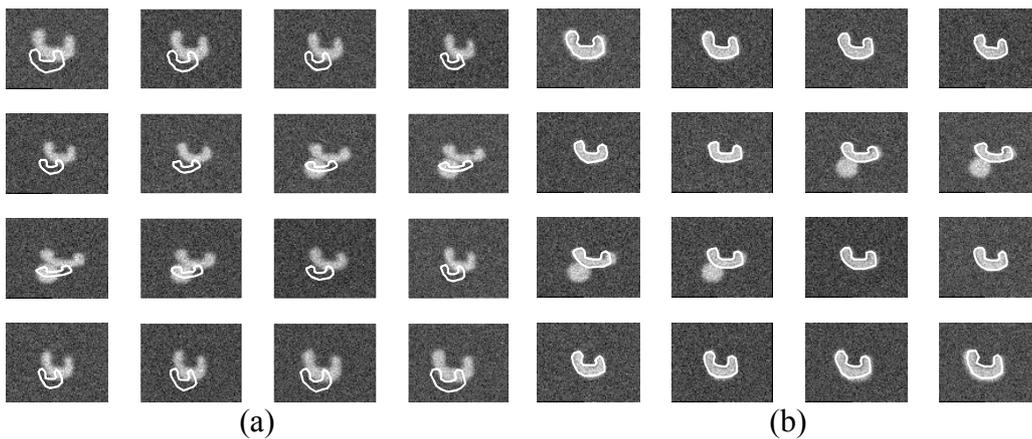


Figure 5.5. Touching occlusion: Example 4. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object in all the frames and overcame the problem of touching occlusions that appeared in some consecutive frames.

Test Case Five: Local Noise

Table 5.7. Local noise: Example 5. See Figure 5.6.

Imperfection	Local noise (and global noise in all frames)
Frames affected	All frames
Training set	Smoothed and noised
Number of iterations	18
Global noise variance	100
Local noise variance	2500
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30
Radius of local noise	25 pixels
Spatial variance of local noise	200 pixels

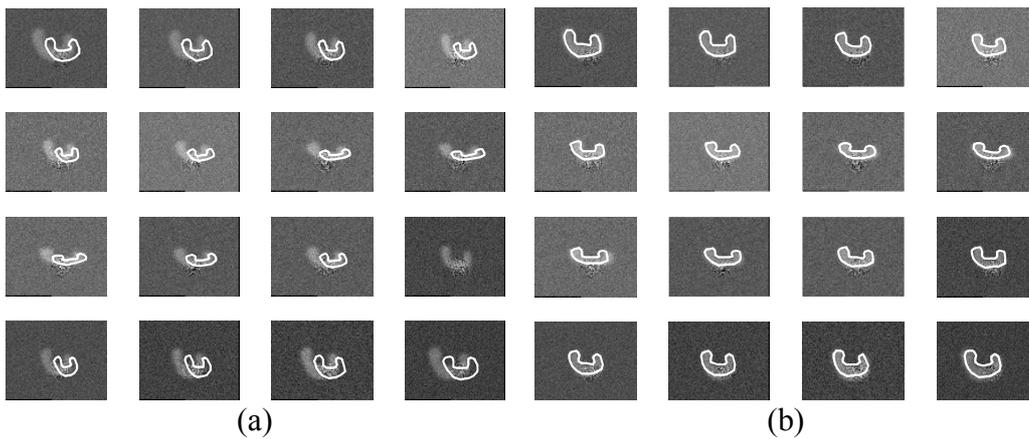


Figure 5.6. Local noise: Example 5. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object in spite of the presence of strong local noise in all the frames.

Test Case Six: Gray Level Training on Perfect Data - A

Table 5.8. Binary training: Example 6A. See Figure 5.7.

Imperfection	Global noise in all frames
Training set	Noised
Number of iterations	10
Global noise variance	0.04
Object intensity	1 (binary)

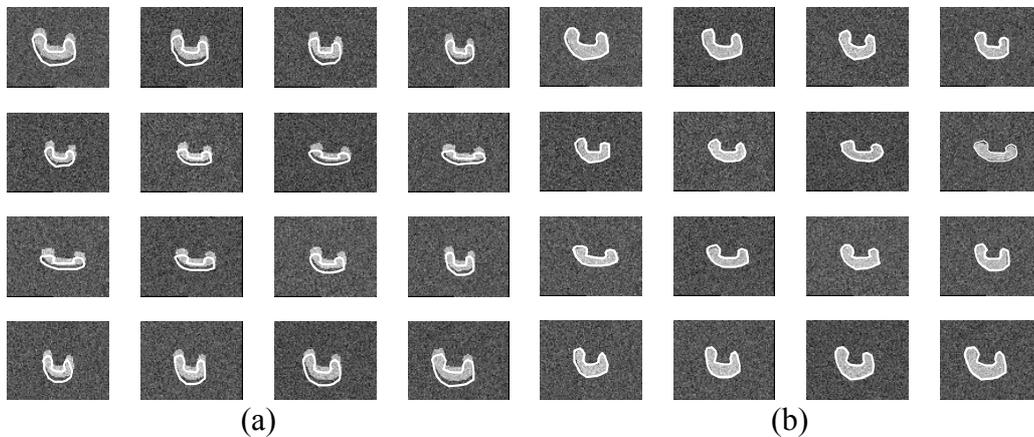


Figure 5.7. Binary training: Example 6A. (a) Initial and (b) final state.

Comments: The results here may suggest that the deformable ST-shape fits better to the target object if compared to the results given in other sections. This is explained by the fact that the training in this case is done on noised binary image sequences which are not smoothed, hence the original contours generating the images correspond exactly to the boundaries of the objects in the noised binary images. In the other cases, the definition of the boundary points, on which the model was trained, was taken before the smoothing was applied.

Test Case Six: Gray Level Training on Perfect Data - B

Table 5.9. Binary training: Example 6B. See Figure 5.8.

Imperfection	Overlapping occlusion (and global noise in all frames)
Frames affected	All frames
Training set	Noised
Number of iterations	18
Global noise variance	0.04
Object intensity	1 (binary)
Radius of occlusion	16 pixels

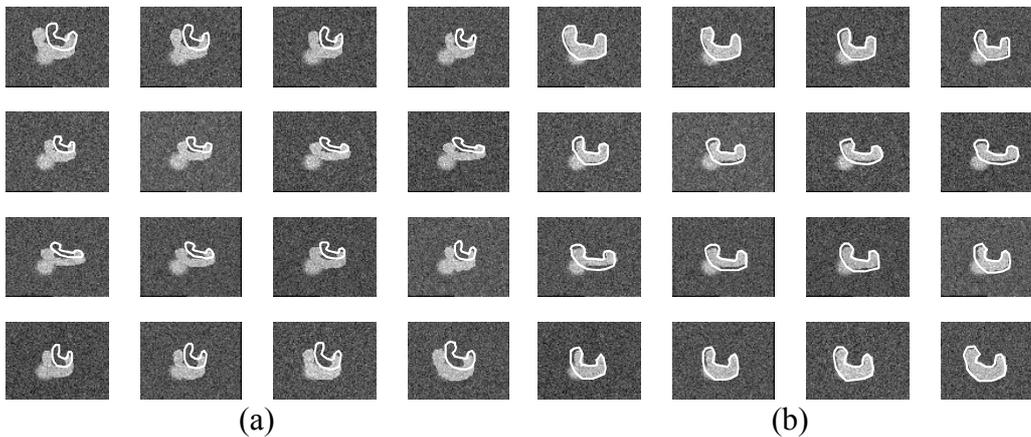


Figure 5.8. Binary training: Example 6B. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape converged to the target object overcoming the problem of overlapping occlusions that appeared in all the frames.

Test Case Seven: Occluded Noisy Training

Table 5.10. Occluded noisy training: Example 7. See Figure 5.9.

Imperfection	Overlapping occlusion (and global noise in all frames)
Frames affected	All frames
Training set	Smoothed, occluded and noised
Number of iterations	15
Global noise variance	100
Minimum object intensity	0
Maximum object intensity	60
Mean object intensity	30
Radius of occlusion	16 pixels

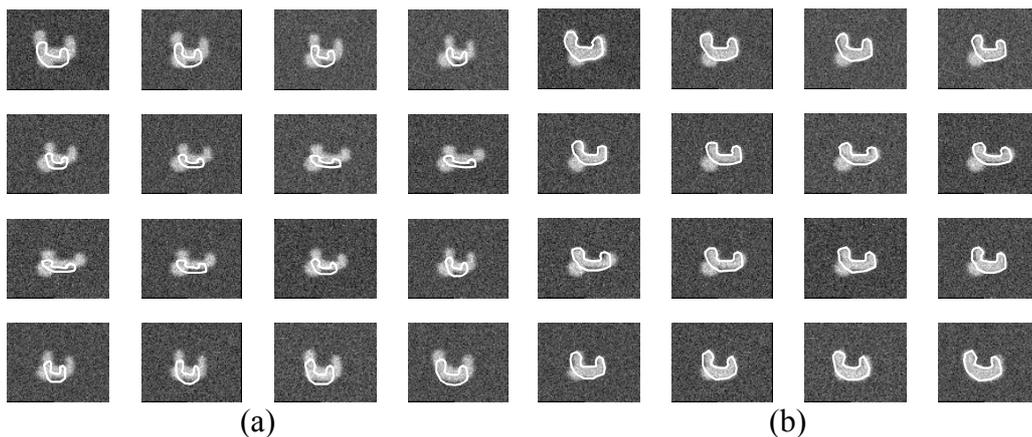


Figure 5.9. Occluded noisy training: Example 7. (a) Initial and (b) final state.

Comments: The result shows that the deformable ST-shape, which is trained on smoothed, occluded and noised data, converged to the target object overcoming the problem of overlapping occlusions that appeared in all the frames.

5.3.5 Results on Real Data

Here we present results of applying the technique to real data (see Figure 5.10 and Figure 5.11).

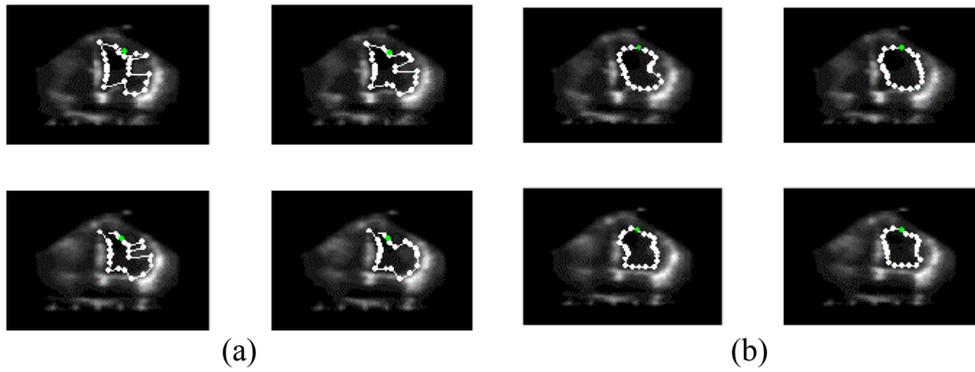


Figure 5.10. Spatio-temporal left-ventricular segmentation result on smoothed real echocardiographic data. Four frames are shown with the ST-shape overlaid in white (a) before and (b) after projection onto the ASTSD.

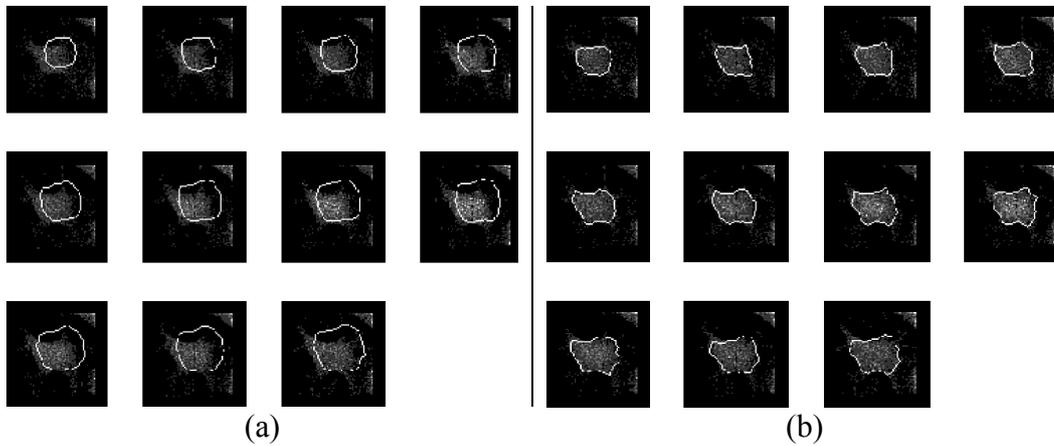


Figure 5.11. Segmenting a 3D astrocyte cell (spatial z-axis replaces time). (a) The initial shape model and (b) the segmentation result are overlaid (in white) on eleven slices of a 3D fluorescence image.

5.4 Conclusion

Motivated by the fact that many image analysis applications require robust methods for representing, locating, and analyzing time-varying shapes, we

presented Deformable Spatio-Temporal Shape Models: an extension of 2D Active Shape Models to 2D+time. This method models the spatio-temporal shape variations and the gray-level information of a time-varying object in a training set. The model is then used for locating similar objects in a new image sequence. The segmentation technique is based on deforming a spatio-temporal shape model to better fit the image sequence data only in ways consistent with the training set. The proposed deformations are calculated by minimizing an energy function using dynamic programming. The energy function includes terms reflecting temporal smoothness and gray-level information constraints. The method was elaborately tested and succeeded in segmenting synthetic spatio-temporal shapes in noisy image sequences. The method is suited for segmenting objects with specific motion patterns, hence potential applications include cardiography, optical signature motion recognition, and lip-reading for Human-Computer Interaction (HCI).

Chapter 6. CONTROLLED SHAPE DEFORMATION VIA MEDIAL PROFILES

Robust, automatic segmentation and analysis of medical images requires powerful and flexible models of anatomical structures. In this chapter¹ we present a multiscale, medial-based approach to shape representation and controlled deformation in an effort to meet these requirements. We use medial-based profiles for shape representation, which follow the geometry of the structure and describe general, intuitive, and independent shape measures (length, orientation, and thickness). Controlled shape deformations (stretch, bend, and bulge) are obtained either as a result of applying deformation operators at certain locations and scales on the medial profiles, or by varying the weights of the main variation modes obtained from a hierarchical (multiscale) and regional (multi-location) principal component analysis of the medial profiles. We demonstrate the ability to produce controlled shape deformations on a medial-based representation of the corpus callosum.

6.1 Introduction

Controlling the deformations of an object's shape in a way that is based on the natural geometry of the object is highly desirable in image interpretation tasks, especially in the segmentation of natural objects from medical images. This intuitive deformation ability reflects the flexibility of clay to be shaped in a sculptor's hands and naturally lends itself to guidance by high-level controllers. Furthermore, the performance of the controllers can be greatly enhanced by keeping the deformations consistent with prior knowledge about the possible object shape variations.

Most deformable shape models (see [McInerney1996] for a comprehensive survey) are boundary-based and although provide excellent local shape control, lack the ability to undergo intuitive global deformation. As a result, it is difficult to incorporate intelligent deformation control operating at the right level of abstraction into the typical deformable model framework of energy minimization. Consequently, these models remain sensitive to initial conditions and spurious image features in image interpretation tasks.

Various hierarchical versions of boundary-based deformable models have been developed [Miller1991, Montagnat1997, Mandal1998, Lachaud1999] but again fail to provide a natural global description of an object - the multiscale deformation control is constructed upon arbitrary boundary point sets and not

¹ This chapter is based primarily on [Hamarneh2001a].

upon object-relative geometry. Several global or “volume-based” shape representation or deformation mechanisms do exist [Barr84, Sederberg86, Coquillart1990, Singh1998b, Terzopoulos1991] but are limited either by the type of objects they can model, or the type and intuitiveness of the deformations they can carry out. They are also typically not defined in terms of the object but rather the object is unnaturally defined (or deformed) in terms of the representation or deformation mechanism.

Deformable models based on medial shape representations of objects are emerging as a powerful alternative to boundary-based and volume-based techniques, primarily led by the work of Pizer’s group at the University of North Carolina at Chapel Hill [Fritsch1997, Pizer1998, Pizer1999]. Medial representations provide both a local and global description of shape. Deformations defined in terms of a medial axis are natural and intuitive and can be limited to a particular scale and location along the axis.

In this chapter, we utilize medial-based *profiles* for shape representation and define deformation operators in terms of these shape profiles. Our goal is the ability to intelligently control the different types and extents of model deformations during the model-to-data fitting process in an effort to focus on the extraction of stable image features before proceeding to object regions with less well-defined features.

To this end, we construct a model of an anatomical structure with a set of profiles that are based on the medial axis of the structure, where each profile describes general and intuitive shape measures (length, orientation, and thickness). Structure deformations (stretch², bend, and bulge³) are then implemented as deformation operators acting on the shape profiles, where each operator can have a different shape and scale and can be applied at any point along a profile.

In addition to the general deformation operators, we would also like to use as much knowledge as possible about the object itself and to generate statistically-proven feasible deformations from a training set. We would like to control these statistical deformations locally along the medial shape profiles to support our goal of intelligent deformation scheduling. Since general statistically-derived shape models only produce global shape variation modes [Cootes1995a, Szekely1996], we have developed spatially-localized feasible deformations at desired scales by utilizing hierarchical (multiscale) and regional principal component analysis to capture shape variation statistics.

In the following sections, we demonstrate the ability to produce controlled shape deformations by applying them to medial-based representations of the corpus callosum (CC), derived from 2D mid-sagittal MRI slices of the brain. We begin by describing the generation and use of

² Stretch or compress.

³ Bulge or squash.

medial-based profiles for shape representation and describe a set of general operators that act on the medial shape profiles to produce controlled shape deformations. We then present a technique for performing a multiscale multi-location statistical analysis of the shape profiles and describe statistics-based deformations based on this analysis. In Section 6.3, we present a simple application of the controlled shape deformations and demonstrate their use in an automatic medical image analysis system.

6.2 Shape Representation and Deformation via Medial Profiles

To control shape deformation intuitively requires a shape representation that, among other things, describes global shape variation intuitively. To meet this requirement, we represent the shape with a set of profiles that are based on a sampled medial axis of an object. Each profile captures an intuitive measure of shape: length, orientation, and thickness. Once the profiles are constructed, various deformation functions or *operators* can be applied to a profile, producing intuitive, controlled deformations: stretching, bending, and bulging.

6.2.1 Medial Profiles for Shape Representation

We use a boundary representation of an object to generate the medial-based profiles. Generation of the profiles begins with the extraction of a sampled (semi-automatically) pruned skeleton of the object to obtain a set of medial nodes (the proposed pruning algorithm is explained in Appendix F). Four medial profiles are constructed: a length profile $L(m)$, an orientation profile $O(m)$, a left (with respect to the medial axis) thickness profile $T^l(m)$, and a right thickness profile $T^r(m)$, where $m = 1, 2, \dots, N$, N is the number of medial nodes, and nodes 1 and N are the terminal nodes. The length profile represents the distances between consecutive pairs of medial nodes, and the orientation profile represents the angles of the edges connecting consecutive pairs of medial nodes (measured with respect to the horizontal). The thickness profiles represent the distances between medial nodes and their corresponding boundary points on both sides of the medial axis (Figure 6.1). Corresponding boundary points are calculated by computing the intersection of a line passing through each medial node in a direction normal to the medial axis, with the boundary representation of the object. Example medial profiles are shown in Figure 6.2 and Figure 6.3.

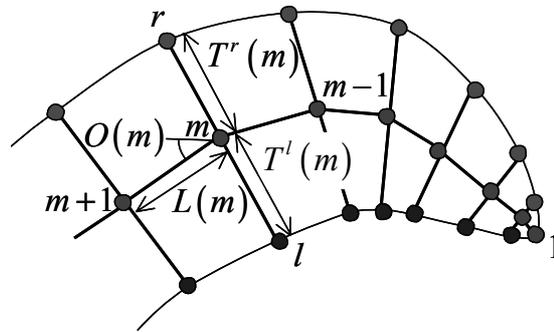


Figure 6.1. Diagram of shape representation.

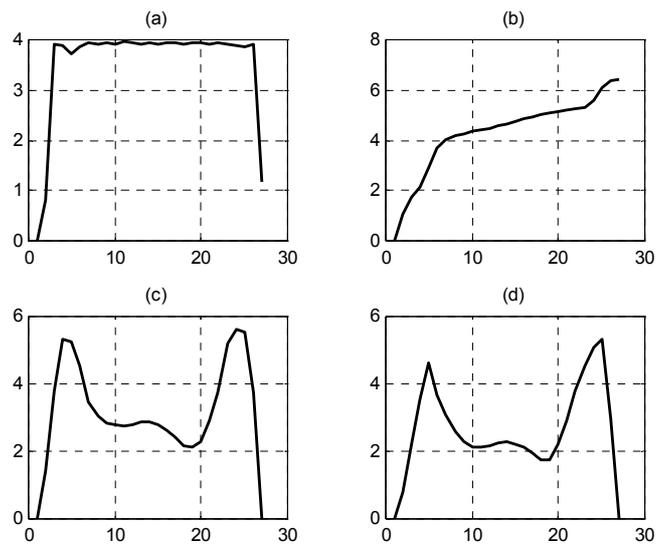


Figure 6.2. Example medial shape profiles: (a) length profile $L(m)$, (b) orientation profile $O(m)$, (c) left thickness profile $T^l(m)$, and (d) right thickness profile $T^r(m)$.

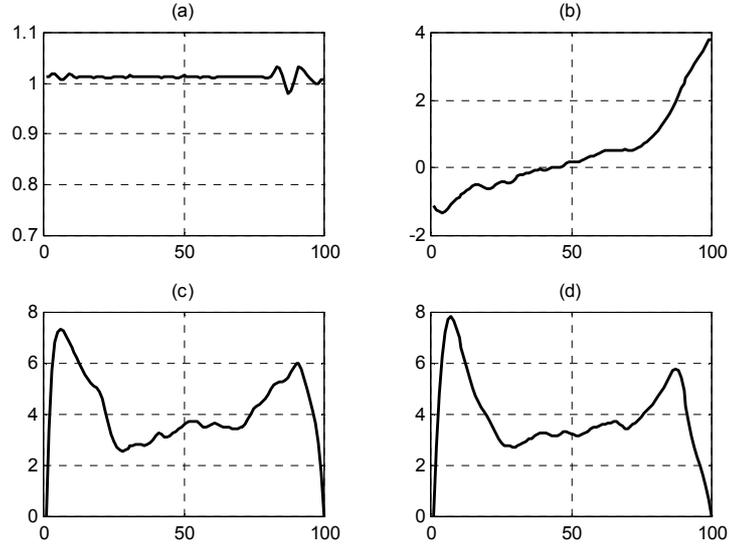


Figure 6.3. Another medial shape profiles example: (a) $L(m)$, (b) $O(m)$, (c) $T^l(m)$, and (d) $T^r(m)$.

6.2.2 Shape Reconstruction from Medial Profiles

To reconstruct the object's shape given its set of medial profiles, we calculate the positions of the medial and boundary nodes by following these steps:

1. Specify affine transformation parameters: orientation angle θ , translation values (t_x, t_y) , and scale (s_x, s_y) .
2. Using medial node 1 as the base or reference node, place it at location $x_1 = (t_x, t_y)$.
3. Repeat steps 4 and 5 for $m = 1, 2, \dots, N$.
4. Compute the locations x_m^l and x_m^r of the boundary points l and r at either side of the m^{th} medial node (Figure 6.1) as

$$x_m^l = x_m + T^l(m) \begin{pmatrix} s_x \cos\left(\theta + O(m) + \frac{\pi}{2}\right) \\ s_y \sin\left(\theta + O(m) + \frac{\pi}{2}\right) \end{pmatrix} \quad (6.1)$$

and similarly,

$$x_m^r = x_m + T^r(m) \begin{pmatrix} s_x \cos\left(\theta + O(m) - \frac{\pi}{2}\right) \\ s_y \sin\left(\theta + O(m) - \frac{\pi}{2}\right) \end{pmatrix}. \quad (6.2)$$

5. If $m < N$, compute the location x_{m+1} of the next medial node $m + 1$ as

$$x_{m+1} = x_m + L(m) \begin{pmatrix} s_x \cos(\theta + O(m)) \\ s_y \sin(\theta + O(m)) \end{pmatrix}. \quad (6.3)$$

Examples of shape reconstruction are shown in Figure 6.4 and Figure 6.5. Note that we have generalized the reconstruction algorithm so that any medial node may serve as the base or reference node.

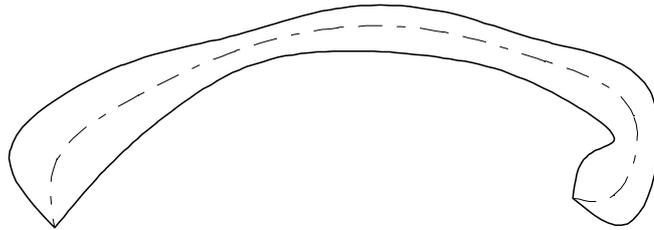


Figure 6.4. Object reconstruction resulting from the shape profiles in Figure 6.2.

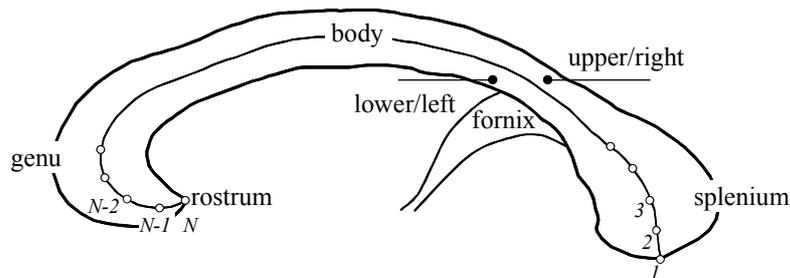


Figure 6.5. CC anatomical feature labels overlaying a reconstruction of the CC using the medial shape profiles shown in Figure 6.3.

6.2.3 Shape Deformation using Medial-Based Operators

Once the shape profiles have been generated, we can construct deformation operators and apply these operators to the shape profiles. This results in intuitive deformations of the object upon reconstruction. That is, by applying

an operator to the length, orientation, or thickness shape profile, we obtain a stretch, bend, or bulge deformation, respectively.

Each deformation operator is implemented by defining a medial-based operator profile, $k(m)$, of a particular type (Figure 6.6) [Bill1995] and specifying an amplitude, location, and scale.

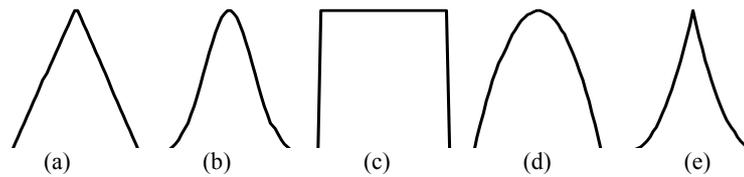


Figure 6.6. Examples of operator types: (a) Triangular, (b) Gaussian, (c) flat, (d) bell, and (e) cusp.

The operator profile is then added to (or blended with) the medial shape profile corresponding to the desired deformation. For example, to introduce a bulge on the right boundary, an operator profile with a specific amplitude, type, location, and scale is generated and added to the right thickness medial profile $T^r(m)$ to obtain $T^r(m) + k(m)$ (Figure 6.7).

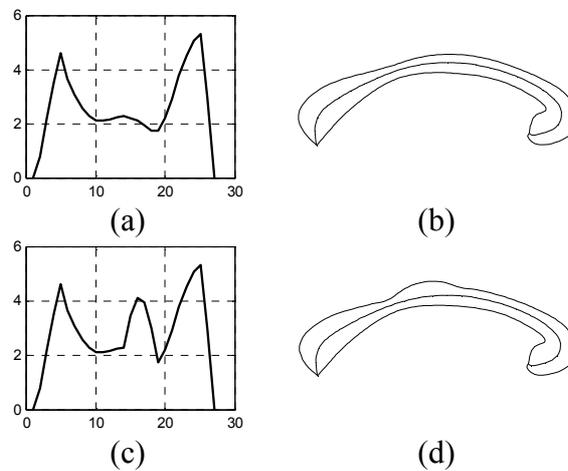


Figure 6.7. Introducing a bulge on the right boundary by applying a deformation operator on the right thickness profile: (a) $T^r(m)$ before and (c) after applying the operator. (b) The reconstructed shape before and (d) after the operator.

In general the application of a deformation operator $k(m)$ alters the desired shape profile according to

$$p_d(m) = \bar{p}_d(m) + \alpha_{dlst} k_{dlst}(m) \quad (6.4)$$

where

p shape profile

d deformation type (stretch, bend, left/right bulge),

i.e. $p_d(m) : \{L(m), O(m), T^l(m), T^r(m)\}$

\bar{p} average shape profile

k operator profile (with unity amplitude)

l location

s scale

t operator type (Gaussian, triangular, ..., etc.)

α operator amplitude.

Altering one shape profile only affects the shape property associated with that profile and does not affect any other object shape properties. For example, applying an operator to the orientation profile results in a bend deformation only and does not result in a stretch or bulge. This implies the ability to perform successive operator-based object deformations of varying amplitudes, types, locations or scales, which can be expressed as

$$p_d(m) = \bar{p}_d(m) + \sum_l \sum_s \sum_t \alpha_{dlst} k_{dlst}(m). \quad (6.5)$$

Examples of operator-based deformations are shown in Figure 6.8(a-d).

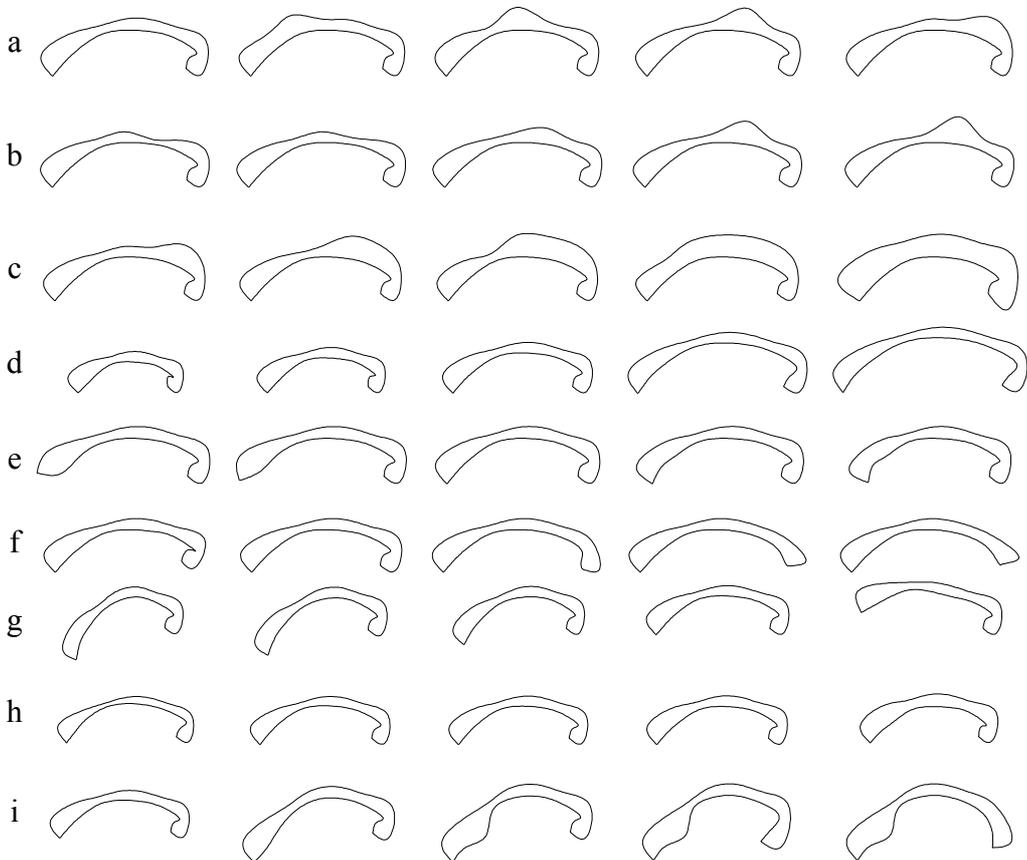


Figure 6.8. Examples of controlled deformations: (a)-(c) Operator-based bulge deformation at varying locations, amplitudes, and scales. (d) Operator-based stretching with varying amplitudes over the entire CC. (e)-(g) Statistics-based bending of the left end, the right end, and the left half of the CC. (h) Statistics-based bulge of the left and right thickness over the entire CC. (i) from left to right: (1) mean shape, (2) statistics-based bending of the left half, followed by (3) locally increasing the left thickness using an operator, followed by (4) applying an operator-based stretch and (5) an operator based bend to the right side of the corpus callosum.

6.2.4 Statistical Shape Analysis by Hierarchical Regional PCA

In many applications, prior knowledge about object shape variability is available or can be obtained by studying a training set of shape examples. The training set is typically created by labeling corresponding landmark points in each shape example. Principal Component Analysis (PCA) is then applied to the training set, resulting in a point distribution model (PDM) [Cootes1995a] (see Appendix B and Appendix C for more details). The PDM describes the

main modes of variation of the landmark positions and the amount of variation each mode explains. A drawback of this original approach is that the result of varying the weight of a single variation mode generally causes all the landmark positions to change. In other words, although the original PDM model produces only feasible shape deformations, a desirable trait, it generally produces global deformations over the entire object.

Our goal is to utilize prior knowledge and produce feasible deformations, while also controlling the scale and location of these deformations. Towards this end we perform a multiscale (Hierarchical) multi-location (Regional) PCA (HRPCA) on a training set of medial shape profiles.

To achieve this, we collect spatially corresponding sub-profiles from the shape profiles. The length of a sub-profile reflects the scale over which the analysis is performed. The principal component analysis is now a function of the location, scale, and type of shape profile (length, orientation, or thickness). Thus, for each location, scale, and shape profile type, we obtain an average medial sub-profile, the main modes of variation, and the amount of variation each mode explains. The result is that we can now generate a feasible stretch, bend, or bulge deformation at a specific location and scale⁴.

A shape profile can now be written as the sum of the average profile and the weighted modes of variation as follows

$$p_d(m) = \bar{p}_d(m) + M_{dls} w_{dls} \quad (6.6)$$

where $p, d, \bar{p}, p_d(m), l, s$ are defined in (6.4), and

M_{dls} variation modes (columns of M) for a specific d, l , and s ,

w_{dls} weights of the variation modes, where the weights are typically set such that the variation is within three standard deviations.

For any shape profile type, multiple variation modes can be activated by setting the corresponding weighting factors to non-zero values. Each variation mode acts at a certain location and scale, hence we obtain

$$p_d(m) = \bar{p}_d(m) + \sum_l \sum_s M_{dls} w_{dls} \quad (6.7)$$

In summary, varying the weights of one or more of the variation modes alters the length, orientation, or thickness profiles and generates statistically feasible stretch, bend, or bulge deformations at specific locations and scales upon reconstruction. Examples of statistics-based deformations are shown in Figure 6.8(e-h).

⁴ HRPCA was also applied for generating statistics-based shape deformations in a physics-based shape representation and deformation technique (see Section 7.4).

6.2.5 Combining Operator- and Statistics-Based Deformations

In general, operator- and statistics-based deformations (see equations (6.5) and (6.7)) can be combined as

$$p_d = \bar{p}_d + \sum_l \sum_s \left(M_{dls} w_{dls} + \sum_t \alpha_{dlst} k_{dlst} \right). \quad (6.8)$$

It is worth noting that several deformations, whether operator- or statistics-based, may spatially overlap (something that we currently do not restrict). Furthermore, adding profiles of different scales, hence different vector lengths, is possible by padding the profiles with zeros. Figure 6.8(i) shows an example of combining operator- and statistics-based deformations.

6.3 Application and Results

To demonstrate the potential of our statistics- and operator-based controlled deformations, we handcrafted a deformation schedule for fitting the CC shape model to a mid-sagittal MRI slice of the brain. Figure 6.9 shows the resulting medial shape profiles after applying the fitting schedule (compare with the initial profiles in Figure 6.2). The initial and final CC shapes are shown in Figure 6.10. The schedule steps are shown in Table 6.1 and the resulting deformed CC shapes for each step of the schedule are shown in Figure 6.11.

Furthermore, in Chapter 8 (see also [Hamarneh2001c, Hamarneh2001b]) of this Thesis we construct a model-based system that automatically and robustly interprets medical images (i.e. segmentation, registration, matching, analysis) by explicitly searching for and fitting to stable image features. A key component of this system is the ability to intelligently schedule and control the type, location, extent, and order of intuitive model deformations during the fitting process something made possible by making full use of the controlled deformation offered by the medial-based shape profiles representations.

6.4 Conclusion

In this chapter we have presented ‘medial profiles’, a medial-based shape representation that provides controlled shape deformation. Based on these profiles, we are able to construct deformation operators and generate intuitive localized and multiscale deformation types (stretch, bend, bulge). Furthermore, by introducing hierarchical regional PCA we are able to perform a multiscale multi-location statistical analysis of the shape profiles thus generating statistically feasible versions of these deformations. We demonstrate our approach to fit a CC model to a to mid-sagittal brain MRI slices both manually (in this chapter) and automatically (as will be presented in Chapter 8).

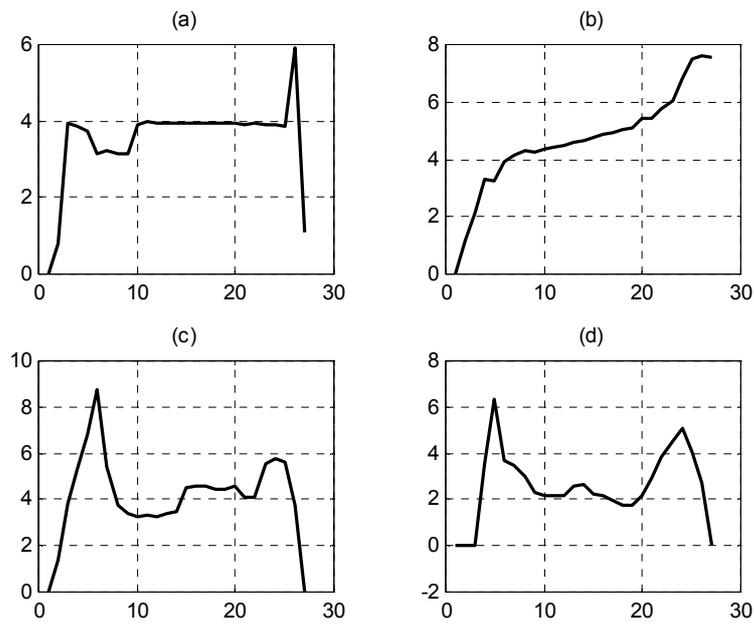


Figure 6.9. The resulting medial shape profiles after applying the fitting schedule: (a) length profile $L(m)$, (b) orientation profile $O(m)$, (c) left thickness profile $T^l(m)$, and (d) right thickness profile $T^r(m)$.

Table 6.1. Deformation schedule used to fit the corpus callosum shape model to the MRI data (see Figure 6.11).

Step	Deformation	Location	Scale	Variation mode/ Operator type	Variation mode weight/ Operator amplitude
1	Translation by ($\blacktriangledown 74, \blacktriangleright 24$)				
2	Rotation by $\circlearrowleft 10^\circ$				
3	Scaling by 1.2				
4	Bend	1	8	2	w=0.5
5	Bend	20	8	2	w=-0.8
6	Bend	22	6	2	w=-0.75
7	Bend	24	4	1	w=2.2
8	Bend	1	4	2	w=1
9	Stretch	6	4	1	w=-1.5
10	Stretch	26	1	1	w=2
11	Left-bulge	15	7	1	w=3
12	Left-bulge	18	3	1	w=2
13	Left-bulge	6	12	1	w=3
14	Left-bulge	5	3	1	w=3
15	Right-squash	9	3	1	w=-1
16	Right-bulge	13	2	1	w=0.5
17	Left-bulge	21	3	Gaussian	$\alpha=0.3$
18	Left-bulge	21	7	Gaussian	$\alpha=0.1$
19	Right-squash	24	2	Gaussian	$\alpha=-0.5$
20	Right-bulge	4	2	Bell	$\alpha=1.7$
21	Right-bulge	6	3	Gaussian	$\alpha=0.4$
22	Right-squash	1	3	Gaussian	$\alpha=-2.2$
23	Right-squash	25	1	Gaussian	$\alpha=-0.8$

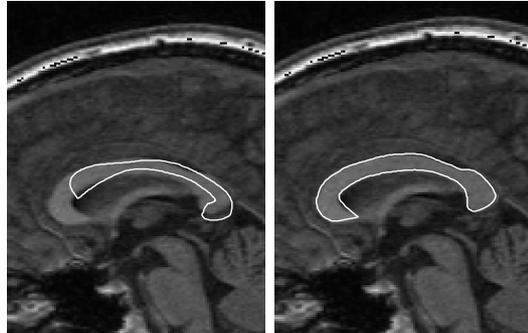


Figure 6.10. Close up of the initial and final stages of the handcrafted fitting schedule.

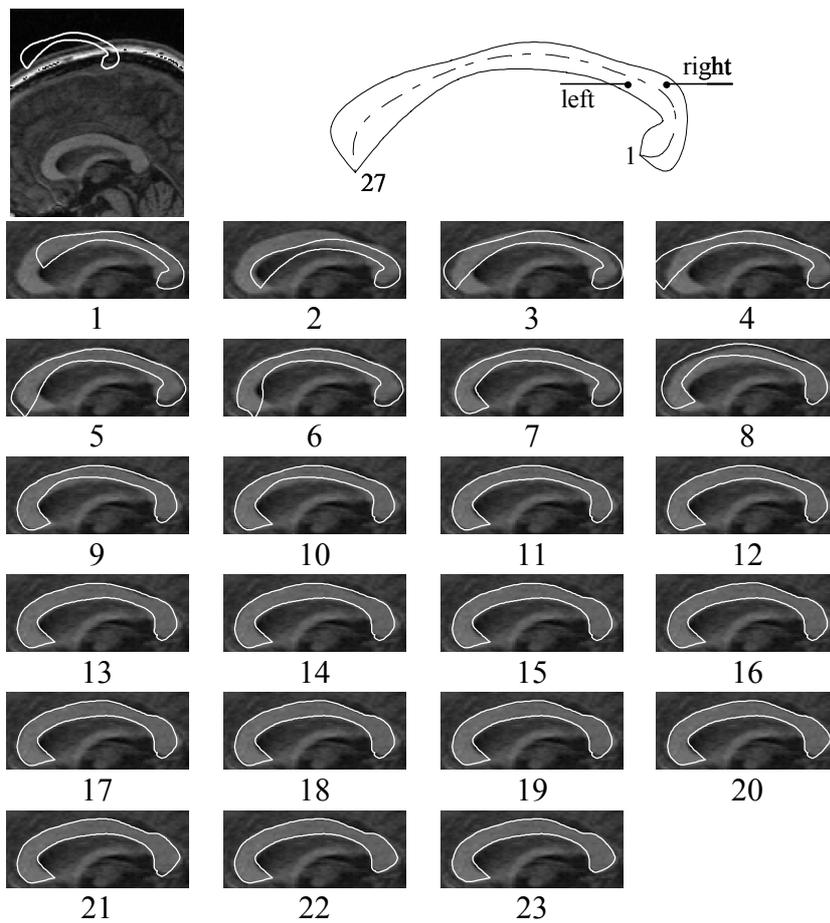


Figure 6.11. Progress of the handcrafted fitting schedule (fitting steps are listed in Table 6.1).

Chapter 7. PHYSICS-BASED SHAPE DEFORMATION

Powerful, flexible shape models of anatomical structures are required for robust, automatic analysis of medical images. In this chapter¹ we investigate a physics-based shape representation and deformation method in an effort to meet these requirements. Using a medial-based spring-mass mesh model, shape deformations are produced via the application of external forces or internal spring actuation. The range of deformations includes bulging, stretching, bending, and tapering at different locations, scales, and amplitudes. Springs are actuated either by applying deformation operators or by activating statistical modes of variation obtained via a hierarchical regional principal component analysis. We demonstrate results on both synthetic data and on a spring-mass model of the corpus callosum, obtained from 2D mid-sagittal brain MRI slices².

7.1 Introduction

Controlling non-rigid object deformation at multiple locations and scales in an interactive and intuitive manner is highly desirable in medical image analysis tasks such as segmentation and registration. Most current deformable shape models [McInerney1996], are boundary-based and although provide excellent local shape control, lack the ability to undergo intuitive global deformation. As a result, it is difficult to incorporate intelligent deformation control operating at the right level of abstraction into the typical deformable model framework of energy minimization. Consequently, these models remain sensitive to initial conditions and spurious image features in image interpretation tasks.

Various hierarchical versions of boundary-based deformable models have been developed [Miller1991, Montagnat1997, Mandal1998, Lachaud1999] but again fail to provide a natural global description of an object - the multi-scale deformation control is constructed upon arbitrary boundary point sets and not upon object-relative geometry. Several global or “volume-based” shape representation or deformation mechanisms do exist [Barr1984, Sederberg1986, Coquillart1990, Singh1998b, Terzopoulos1991] but are limited either by the type of shapes they can represent, or the type and intuitiveness of the deformations they can carry out. They are also typically not defined in terms of the object but rather the object is unnaturally defined (or deformed) in terms of the representation or deformation mechanism.

¹ This chapter is based primarily on [Hamarneh2001e].

² A physics-based shape deformation tool was also developed and is detailed in Appendix G.

Emerging trends in deformable shape modeling include medial-based approaches, which we believe are powerful techniques since they follow the geometry of the object and provide natural and intuitive deformations. [Pizer2000, Hamarneh2001a]. Additionally, physics based deformable shape models have been developed [Terzopoulos1991, Molloy2000]. The attractiveness of these models stems from their ability to inherently handle smoothness and continuity constraints. Furthermore, statistically derived shape models [Cootes1995, Szekely1996] are gaining wide acceptance within the medical image analysis community since they constrain the global shape deformations according to the statistical shape variations observed in a training set.

The shape representation and deformation method presented in this chapter is motivated by the following desirable characteristics of a deformable model for medical image analysis tasks. *First*, implementing the deformations within a physics-based framework that inherently handles smoothness and continuity constraints and facilitates intuitive user interaction. *Second*, using shape representations and deformations that follow the naturally geometry of the object. *Third*, controlling the deformations of an object shape at multiple locations and multiple scales. *Fourth*, restricting the deformations to produce only feasible shapes.

In this chapter we investigate a method that addresses all of the above points. *First*, the deformable shapes are modeled using physics-based meshes of connected nodes (mass-spring models) that maintain the structural integrity of the body as it deforms and are suitable for intuitive user interaction. *Second*, the mesh nodes and connectivity are based on the medial axis of the object. *Third*, we use either operator- or statistics-based deformations to control the different types of deformation at multiple locations and scales. *Finally*, statistics-based feasible deformations are derived from a Hierarchical (multi-scale) Regional (multi-location) Principal Component Analysis (HRPCA).

7.2 The Dynamic Mesh Model

We use mesh models to represent object shapes (Figure 7.1 and Figure 7.2(b)). A mesh is made up of nodes (masses or particles) and springs (elastic links or connecting segments). A Mass m_i , position x_i , velocity v_i , and acceleration a_i are associated with each node n_i . Two terminal nodes n_i and n_j , Hook's spring constant k_s , damping constant k_d , and rest length r_{ij} are associated with each spring s_{ij} .

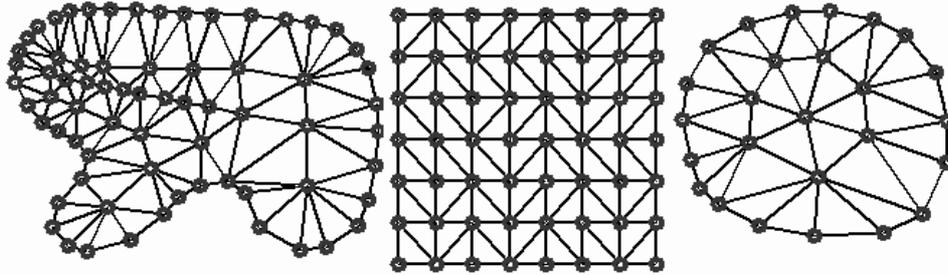


Figure 7.1. Examples of different spring-mass structures.

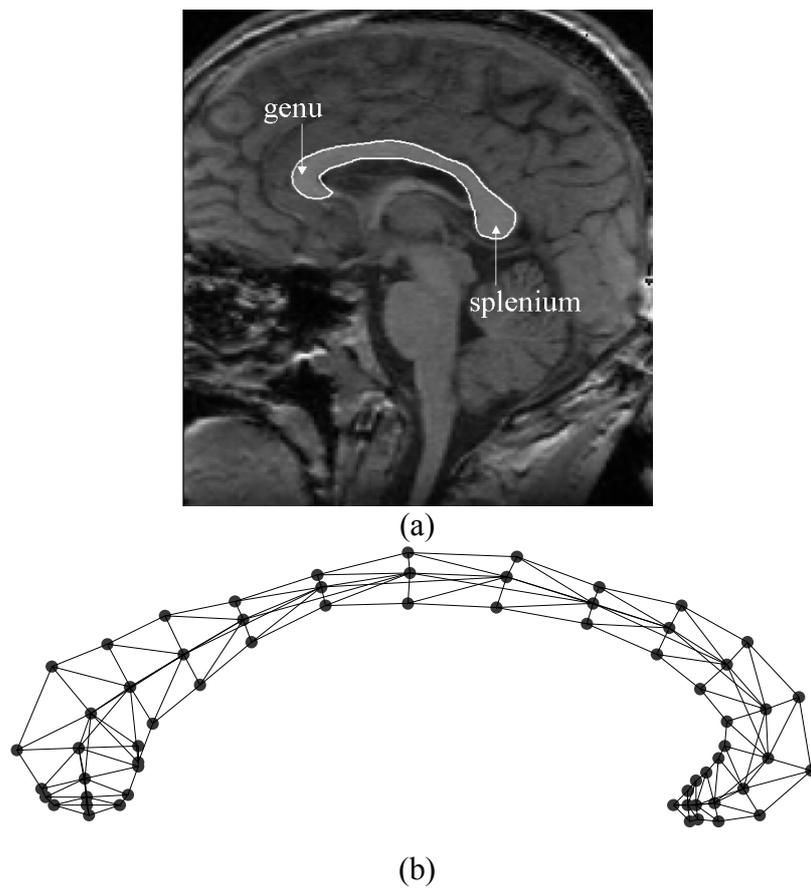


Figure 7.2. (a) Mid-sagittal MRI brain image with the corpus callosum (CC) outlined in white. (b) CC mesh model.

By applying Newton's second law of motion and simulating the dynamics by time integration, the mesh nodes move deforming the object's shape. Newton's second law of motion for the node n_i , states that $a_i = f_i/m_i$, where f_i is total force acting on n_i

$$f_i = f_i^{Hook} + f_i^{viscous} + f_i^{user} + f_i^{image}. \quad (7.1)$$

A spring s_{ij} will cause

$$f_i^{Hook} = -k_s \left(\|x_i - x_j\| - r_{ij} \right) \frac{x_i - x_j}{\|x_i - x_j\|} - \left(k_d (v_i - v_j)^T \frac{x_i - x_j}{\|x_i - x_j\|} \right) \frac{x_i - x_j}{\|x_i - x_j\|} \quad (7.2)$$

to be exerted at n_i and $-f_i^{Hook}$ on n_j . Viscous drag at n_i is given by $f_i^{viscous} = -k_v v_i$. A single user applied force f_i^{user} is implemented as the dynamic force resulting from a spring connecting a mesh node to the (varying) position of the user's point of application. Image forces can be implemented as

$$f_i^{image} \propto k_{ext} \nabla (\| \nabla I_s(x_i) \|) \quad (7.3)$$

where $I_s(x_i)$ is the intensity of a pixel at the location of node n_i in a smoothed version of the image. Image forces that attract the model to an image boundary are calculated only for boundary mesh nodes (similarly image forces that attract medial model nodes to medial features can also be applied).

Following the calculation of the node forces we compute the new acceleration, velocity, and position of each node given the old velocity and position values, as follows (explicit Euler solution with time step Δt)

$$\left. \begin{aligned} a_i &= f_i/m_i \\ v_i &= v_i^{old} + a_i \Delta t \\ x_i &= x_i^{old} + v_i \Delta t \end{aligned} \right\} \quad (7.4)$$

7.3 Shape Deformation

7.3.1 Deformation using External Forces

As explained in section 2, deformations can be applied via external forces such as user interaction (Figure 7.3) or image forces.

7.3.2 Deformation using Spring Actuation

Other forces result from spring actuation (in a manner analogous to muscle actuation in animals, see Figure 7.4). Two nodes connected by a spring will normally change position until the spring is at its rest length. To actuate a spring we change its rest length while continuously simulating the mesh dynamics.

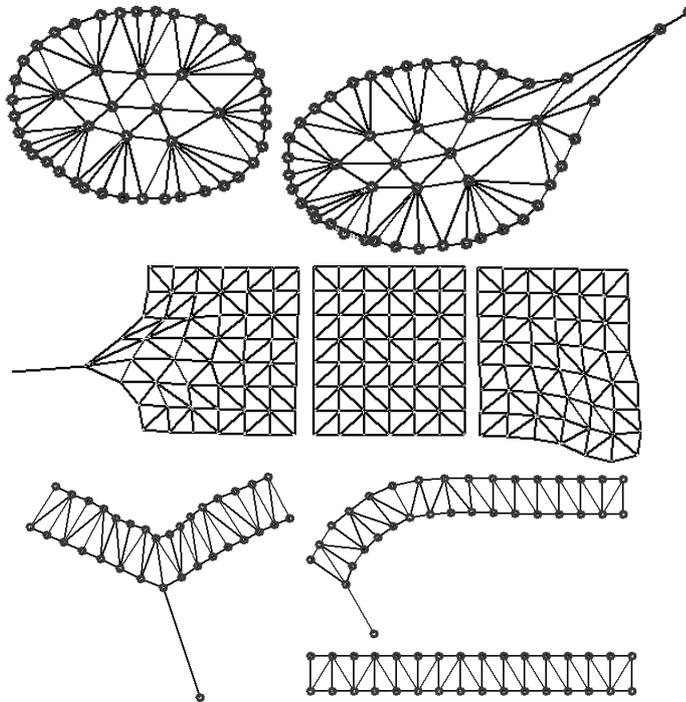


Figure 7.3. Examples of deformations via user interaction ('mouse' forces).

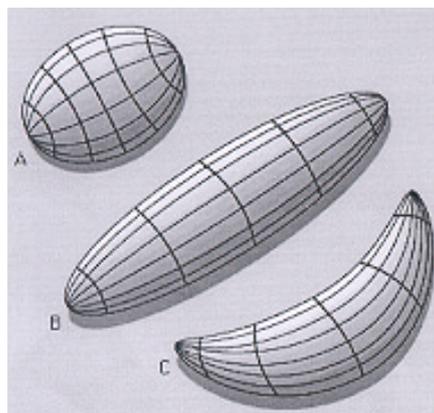


Figure 7.4. Changes in body form in wormlike soft-bodied animals. (A) The longitudinal muscles contracting. (B) The circular muscles contracting. (C) Bend deformation by contraction of the longitudinal muscles above while stretching the longitudinal muscles below [EB].

Operator-based localized deformations. Bulging (radial bulge), stretching (directional bulge), bending, tapering, and scaling deformations are implemented using spring actuation. These operator-based deformations can be applied at different locations and scales with varying amplitudes.

To perform a (radial) bulge deformation we specify a center C and a radius R of a deformation region (Figure 7.5(a)) as well as a deformation amplitude K . We then update the rest length r_{ij} of each spring s_{ij} if at least one of its terminal nodes, n_i or n_j , lies within the deformation region, as follows

$$r_{ij} = \left(\left(1 - \frac{d}{R} \right) \left(1 - \frac{2\theta}{\pi} \right) (K - 1) + 1 \right) r_{ij}^{old} \quad (7.5)$$

where $\theta \in \left[0, \frac{\pi}{2} \right]$ is the angle between s_{ij} and the line L connecting the midpoint of the spring with the C and d is the length of L (Figure 7.5(a)). The resulting effect of the above equation is that springs closer to C and with directions closer to the radial direction are affected more (Figure 7.6).

To perform a stretch (directional bulge) we again specify a deformation region and amplitude as well as a direction \vec{D} (Figure 7.5(b)). We update the rest length of each spring as in equation (7.5) where $\theta \in \left[0, \frac{\pi}{2} \right]$ is now defined as the angle between s_{ij} and \vec{D} (Figure 7.5(b)). The resulting effect in this case is that springs closer to C and with directions closer to the stretch deformation direction are affected more (Figure 7.6).

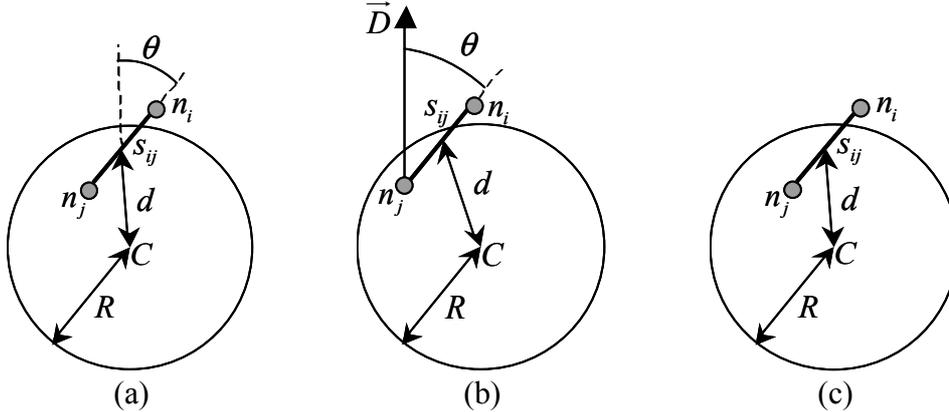


Figure 7.5. Definition of variables for (a) radial bulge, (b) directional bulge, and (c) localized scaling. A single is spring is shown as a thick line segment connecting two nodes.

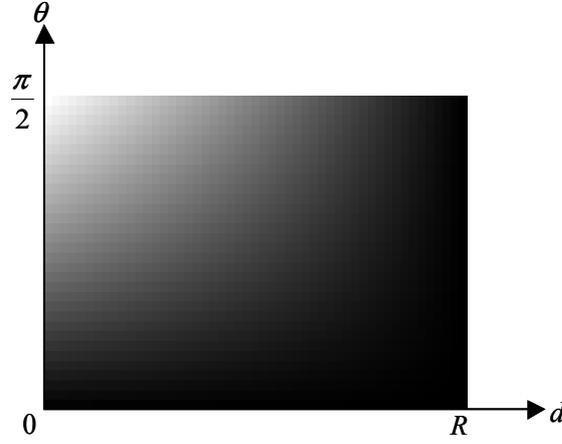


Figure 7.6. The coefficient (white= K , black=1) by which r_{ij}^{old} is multiplied as a function of θ and d .

A localized scaling deformation is independent of direction and requires only the specification of a deformation region and amplitude (Figure 7.5(c)). The rest length update equation then becomes

$$r_{ij}^1 = \left(\left(1 - d/R\right)(K - 1) + 1 \right) r_{ij}^{old}. \quad (7.6)$$

To perform localized bending, we specify a bending amplitude K and two regions surrounding the medial axis (Figure 7.7). The rest lengths of the springs on one side of the medial are increased according to

$$r_{ij}^1 = \left(\frac{d_1}{R_1} \left(1 - \frac{2\theta_1}{\pi}\right)(K - 1) + 1 \right) r_{ij}^{1,old} \quad (7.7)$$

while the rest lengths on the other side are decreased according to

$$r_{ij}^2 = \left(\frac{d_2}{R_2} \left(1 - \frac{2\theta_2}{\pi}\right) \left(\frac{1}{K} - 1\right) + 1 \right) r_{ij}^{2,old}. \quad (7.8)$$

To perform localized tapering, we specify a tapering amplitude K and a region with a base (Figure 7.8). The rest lengths on one side of the base are increased according to

$$r_{ij}^1 = \left(\frac{d_1}{R_1} (K - 1) + 1 \right) r_{ij}^{1,old} \quad (7.9)$$

while those on the other side are decreased according to

$$r_{ij}^2 = \left(\frac{d_2}{R_2} \left(\frac{1}{K} - 1\right) + 1 \right) r_{ij}^{2,old} \quad (7.10)$$

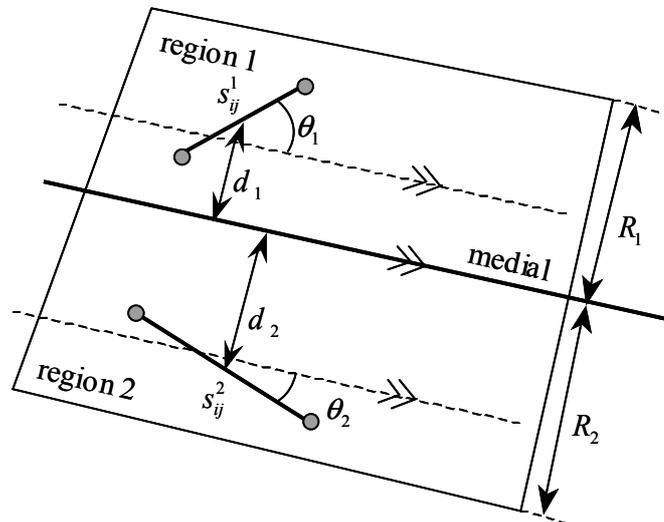


Figure 7.7. Definition of variables for localized bending deformation operator.

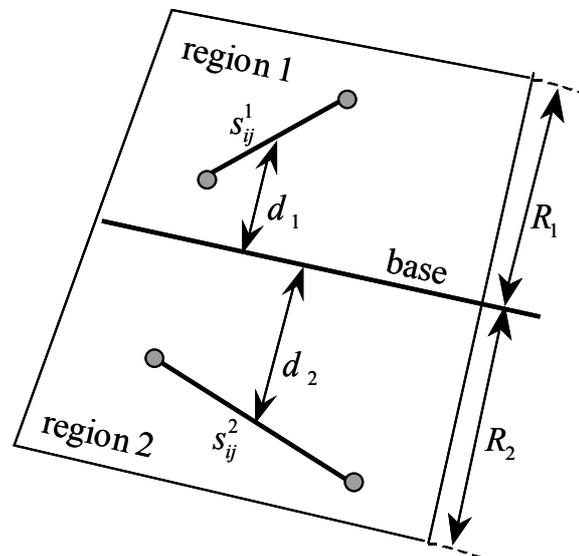


Figure 7.8. Definition of variables for tapering deformation operator.

Different examples of localized operator-based deformations are shown in Figure 7.9.

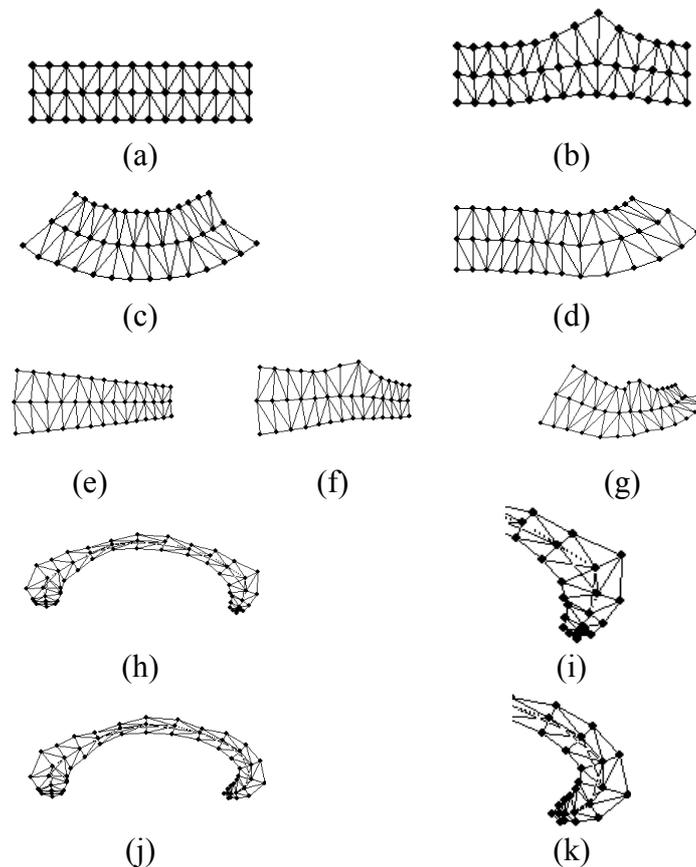


Figure 7.9. Examples of localized deformations. (a) Initial synthetic object, (b) bulge, (c) bend, (d) bend at another location, (e) tapering, (f) tapering followed by a bulge, and (g) tapering followed by a bulge and a bend deformations. CC model (h) before and (j) after a localized bend. (i-k) Close up versions of (h-j).

Learned deformations. Learned or statistical deformations are also implemented via spring actuation. To facilitate intuitive deformations, springs are designed to be of different types: stretch springs, bend springs, or thickness springs. Stretch springs connect neighboring medial nodes, bending springs are hinge springs that connect non-consecutive medial nodes, and thickness springs connect medial nodes with boundary nodes (Figure 7.10). Actuating the stretch

springs causes stretch deformations, actuating hinge springs causes bend deformations, and actuating thickness springs causes bulging, squashing, or tapering deformations.

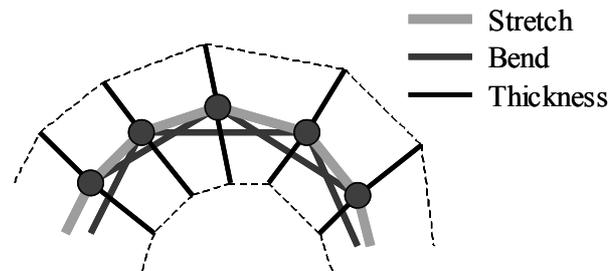


Figure 7.10. Spring types used for statistics-based deformations.

Feasible mesh deformations are obtained by actuating springs according to the outcome of a statistical analysis performed on the spring lengths of a training set (discussed in Section 7.4).

7.3.3 Affine Transformations

Rotation and translation are implemented via the application of external forces. Scaling is implemented by muscle actuation. Scaling by a factor of S is performed by changing the rest length of all the springs, i.e.

$$r_{ij} = S \cdot r_{ij}^{old}. \quad (7.11)$$

Rotation forces are applied on all nodes in a direction normal to the line connecting each node with the center of mass of the model, with a consistent clockwise/counter clockwise direction (Figure 7.11(a)). Translation forces are applied on all nodes in the direction of the desired translation (Figure 7.11(b)). Examples are shown in Figure 7.12.

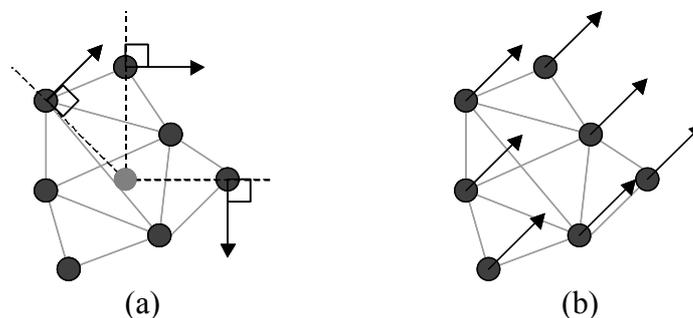


Figure 7.11. External forces for performing a (a) rotation (light gray circle marks center of mass) and a (b) translation.

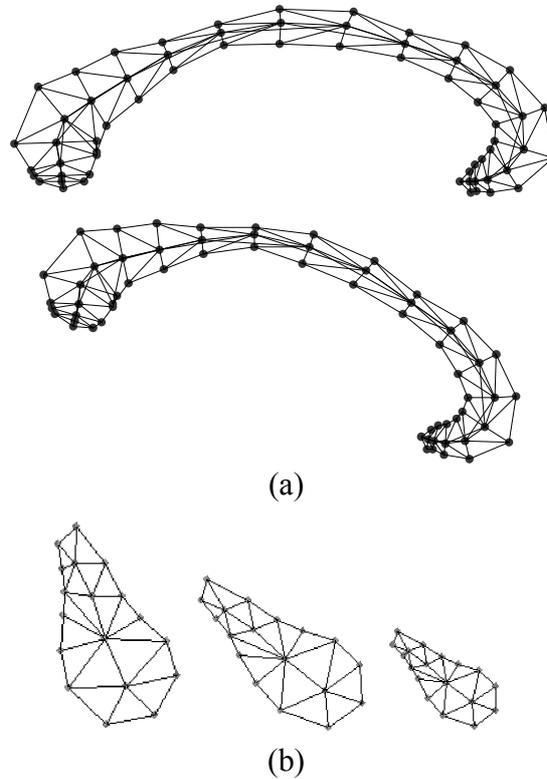


Figure 7.12. Affine transformation via external forces. (a) Rotating a model of the corpus callosum. (b) Rotating and scaling a synthetic model.

7.4 Hierarchical Regional PCA

To produce feasible (i.e. similar to what has been observed in a training set) shape deformations at different locations and scales, we use Hierarchical Regional Principal Component Analysis (HRPCA). HRPCA involves performing traditional PCA on the spring lengths corresponding to each of the desired localized deformation as explained below³.

The set of rest lengths for the stretch springs (Figure 7.10) in a single example model are collected in a vector \mathbf{r}_S , i.e.

$$\mathbf{r}_S = \{r_{ij} \forall i, j : s_{ij} \in \text{stretch springs}\} \quad (7.12)$$

³ HRPCA was also applied for generating statistics-based deformations using medial profiles (see Section 6.2.4).

and similarly for the bending and left and right thickness springs (Figure 7.10)

$$\begin{aligned}\mathbf{r}_B &= \{r_{ij} \forall i, j : s_{ij} \in \text{bend springs}\} \\ \mathbf{r}_{TL} &= \{r_{ij} \forall i, j : s_{ij} \in \text{left thickness springs}\} \\ \mathbf{r}_{TR} &= \{r_{ij} \forall i, j : s_{ij} \in \text{right thickness springs}\}.\end{aligned}\quad (7.13)$$

This gives

$$\begin{aligned}\mathbf{r}_S &= [\mathbf{r}_S^1, \mathbf{r}_S^2, \dots, \mathbf{r}_S^{N_S}] \\ \mathbf{r}_B &= [\mathbf{r}_B^1, \mathbf{r}_B^2, \dots, \mathbf{r}_B^{N_B}] \\ \mathbf{r}_{TL} &= [\mathbf{r}_{TL}^1, \mathbf{r}_{TL}^2, \dots, \mathbf{r}_{TL}^{N_T}] \\ \mathbf{r}_{TR} &= [\mathbf{r}_{TR}^1, \mathbf{r}_{TR}^2, \dots, \mathbf{r}_{TR}^{N_T}]\end{aligned}\quad (7.14)$$

where N_S, N_B , and N_T are the numbers of stretch, bend, and left/right thickness springs, respectively. The springs are numbered according to their spatial order (i.e. moving from one end of the medial to the other we encounter $\mathbf{r}_S^1, \mathbf{r}_S^2, \dots, \mathbf{r}_S^{N_S}$).

Performing global (traditional) PCA (see Appendix C) on the corresponding rest lengths of the springs in a training set gives

$$\begin{aligned}\mathbf{r}_S &= \bar{\mathbf{r}}_S + M_S \mathbf{w}_S \\ \mathbf{r}_B &= \bar{\mathbf{r}}_B + M_B \mathbf{w}_B \\ \mathbf{r}_{TL} &= \bar{\mathbf{r}}_{TL} + M_{TL} \mathbf{w}_{TL} \\ \mathbf{r}_{TR} &= \bar{\mathbf{r}}_{TR} + M_{TR} \mathbf{w}_{TR}\end{aligned}\quad (7.15)$$

where the columns of M_S, M_B, M_{TL} , and M_{TR} are the main modes of spring length variation for the stretch, bend, left and right thickness springs, respectively. Associated with each mode is the variance it explains. The details on obtaining the corpus callosum training set can be found in Section 7.5.

For capturing the shape variations at different locations and scales, we study the variations in the rest lengths of the springs in the desired localized region. Furthermore, to decompose the variations into different types of general deformations, each statistical analysis of the spring length in a localized region is restricted to a specific type of deformation springs (Figure 7.10). Accordingly, the PCA becomes a function of the deformation type, location and scale. For example, to analyze the local variation in object length (stretch), we perform a statistical analysis on the lengths of the stretch springs of that local region. In general, for a single deformation/location/scale- specific PCA we obtain

$$\mathbf{r}_{def,loc,scl} = \bar{\mathbf{r}}_{def,loc,scl} + M_{def,loc,scl} \mathbf{w}_{def,loc,scl}\quad (7.16)$$

where def is the deformation type being either, S (for stretch), B (for bend), TL (for left thickness) or TR (for right thickness). The location and scale, determined by the choice of loc and scl respectively, determine which springs are to be included in the analysis according to

$$\mathbf{r}_{def,loc,scl} = [\mathbf{r}_{def}^{loc}, \mathbf{r}_{def}^{loc+1}, \dots, \mathbf{r}_{def}^{loc+scl-1}]. \quad (7.17)$$

For example, for the bending deformation at location ‘five’ with scale ‘three’ ($def, loc, scl = B, 5, 3$) we have

$$\mathbf{r}_{def,loc,scl} = \mathbf{r}_{B,5,3} = [\mathbf{r}_B^5, \mathbf{r}_B^6, \mathbf{r}_B^7]. \quad (7.18)$$

The average values of the spring lengths are calculated according to

$$\bar{\mathbf{r}}_{def,loc,scl} = \frac{1}{N} \sum_{j=1}^N \mathbf{r}(j)_{def,loc,scl} \quad (7.19)$$

where $\mathbf{r}(j)_{def,loc,scl}$ is $\mathbf{r}_{def,loc,scl}$ obtained from the j^{th} training example and N is the number of training examples. The columns of $M_{def,loc,scl}$ are the eigenvectors, $m_{def,loc,scl}$, of the covariance matrix $C_{def,loc,scl}$. That is

$$\{C\mathbf{m} = \lambda\mathbf{m}\}_{def,loc,scl} \quad (7.20)$$

where

$$\left\{ C = \frac{1}{N-1} \sum_{j=1}^N (\mathbf{r}(j) - \bar{\mathbf{r}})(\mathbf{r}(j) - \bar{\mathbf{r}})^T \right\}_{def,loc,scl} \quad (7.21)$$

and where $\{ \}_{def,loc,scl}$ denotes deformation type-, location-, and scale-specific PCA variables. The global PCA can be written as a special case of the HRPCA by specifying $loc = 1$ and $scl = N_S, N_B$, or N_T .

Note that the data set needs to be aligned only with respect to scale. The statistical analysis of spring lengths is independent of orientation and translation. See the different examples in Figure 7.13 to Figure 7.16.

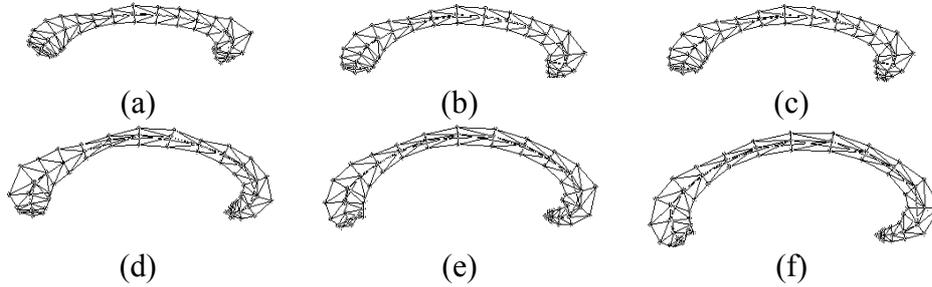


Figure 7.13. Sample corpus callosum mesh model deformations (1st PC for all deformation types over the entire CC) derived from the hierarchical regional PCA.

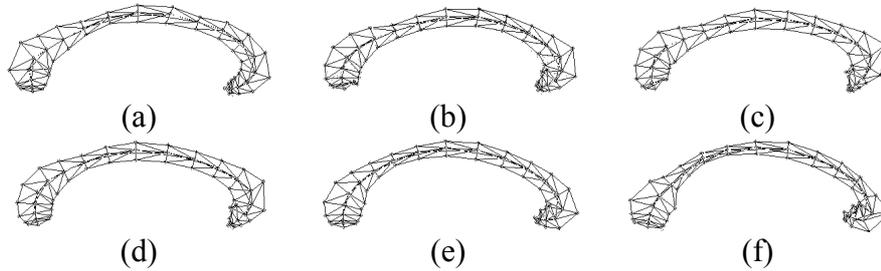


Figure 7.14. Sample CC mesh model deformations (2nd PC for all deformation types over the entire CC) derived from the hierarchical regional PCA.

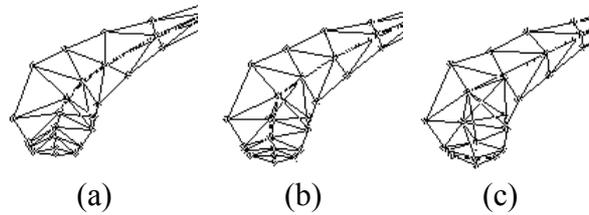


Figure 7.15. Statistical CC mesh model deformations: Stretching the Splenium (see Figure 7.2(a) for anatomy).

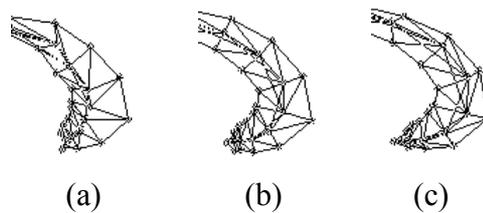


Figure 7.16. Statistical CC mesh model deformations: Bending the Genu (see Figure 7.2(a) for anatomy).

7.5 Mesh Generation from Real Data

From 51 MRI brain volumes, we extracted the mid-sagittal slices from the coronal slices. We then used human expert segmented corpus callosum images (Figure 7.17(a)) to compute the set of spatially ordered boundary coordinates (Figure 7.17(b)). We calculated a pruned skeleton using morphological operations (see Appendix F) to produce a medial axis (Figure 7.17(c-d))

represented by spatially ordered coordinates. We then sampled the medial and boundary coordinates (we experimented with critical point detection algorithm [Zhu1995], fitting line segments [Jain1989], in addition to uniform/equal arc length sampling and non-uniform sampling). We then constructed the mesh by finding the boundary points closest to the line normal to the sampled medial points. Since Delaunay triangulation does not guarantee correspondence between the meshes in different examples and does not generate deformation specific springs (Figure 7.10), we hand crafted the spring connections and applied it to all the training data (Figure 7.17(e-f)).

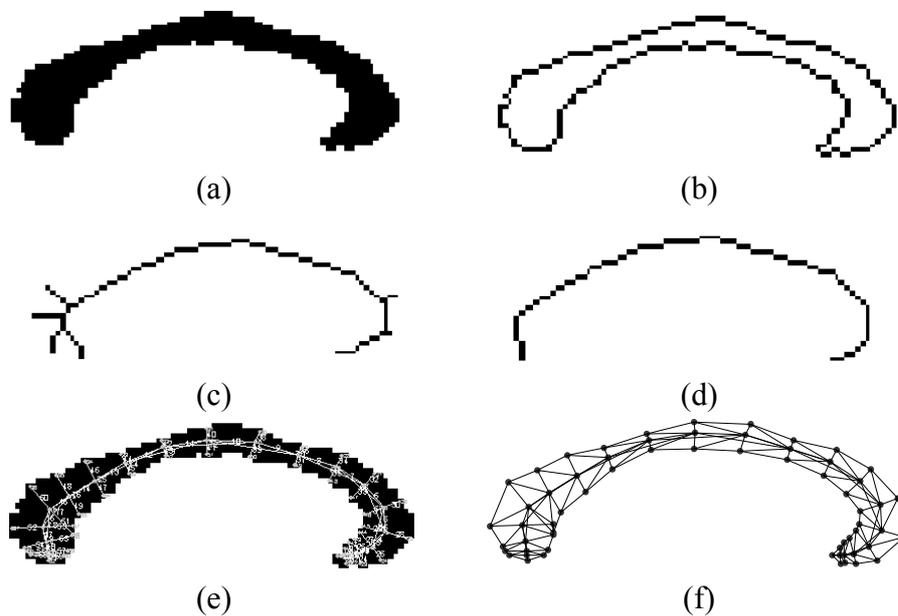


Figure 7.17. CC mesh generation: (a) expert segmented CC images, (b) extracted boundary pixels, (c) skeleton, (d) pruned skeleton/medial axis, (e) spring connections, and (f) final CC mesh model.

7.6 Conclusion

A key requirement of a deformable model-based medical image analysis system is the ability to intelligently schedule and control the type, location, extent, and order of intuitive model deformations during the fitting process. In this chapter we demonstrated the use of a physics-based shape representation and deformation technique to meet such a requirement. This new model can be used (as an alternative to the ‘Medial Profiles’ technique presented in Chapter 6) for the lower layers of the multi-layered model-fitting system (Chapter 8).

Chapter 8. DEFORMABLE ORGANISMS: AN ALIFE APPROACH TO MIA

In this chapter¹, we introduce a new paradigm for medical image analysis that adopts modeling concepts from the field of artificial life. Our approach prescribes deformable organisms, autonomous agents whose objective is the segmentation and labeling of anatomical structures in medical images. A deformable organism is structured as a ‘muscle’-actuated ‘body’ whose behavior is controlled by a ‘brain’ that is capable of making both reactive and deliberate high-level decisions. The brains issue ‘muscle’ actuation commands or ‘growth’ commands, resulting in actively controlled shape deformations of the bodies comprising skeleton and limbs. This intelligent deformable model possesses an ‘awareness’ of the segmentation process, which emerges from a conflux of perceived sensory data, an internal mental state, memorized knowledge, and a pre-stored cognitive plan. The plan, which may involve interaction with other organisms, guides the organism to identify landmark anatomical features during its development. We develop a class of deformable organisms using a medial representation of body morphology that facilitates a variety of controlled Shape deformations and controlled growth at multiple scales and locations. The framework promises to lay the foundation for the construction of robust and automatic medical image analysis tools by combining deformable models and high-level a priori knowledge. We demonstrate how deformable organisms can deal with noise, incomplete edges, anatomical variation, and occlusion in order to segment and label various anatomical structures from medical images.

8.1 Introduction

The automatic segmentation and labeling of anatomical structures in medical images is a persistent problem that continues to defy solution. A substantial amount of knowledge is often available about anatomical structures of interest - shape, position, orientation, symmetry, relationship to neighboring structures, landmarks, etc. - as well as about the associated image intensity characteristics. However, MIA researchers have struggled to develop segmentation techniques that can take full advantage of such knowledge. There is consensus within the medical image analysis research community that the development of general-purpose automatic segmentation algorithms will require not only powerful bottom-up, data-driven processes, but also equally powerful top-down,

¹ This chapter is based primarily on [Hamarneh2001c] (see also [Hamarneh2001b]).

knowledge-driven processes within a robust decision-making framework that operates across multiple levels of abstraction [Duncan2000]. We contend that current frameworks of this sort are inflexible and do not operate at an appropriate level of abstraction. Consequently, not only are they unable to incorporate all available knowledge, but also they are incapable of effectively applying this knowledge (i.e., at the correct time and place during image analysis).

Deformable models, one of the most actively researched model-based segmentation techniques [McInerney1996], feature a potent bottom-up component founded in estimation theory, optimization, and physics-based dynamical systems, but their top-down processes have traditionally relied on interactive initialization and guidance by knowledgeable users (see Section 1.2.2 Deformable Models). Attempts to fully automate deformable model segmentation methods have so far been less than successful at coping with the enormous variation in anatomical structures of interest, the significant variability of image data, the need for intelligent initialization conditions, etc. It is difficult to obtain intelligent, global (i.e., over the whole image) model behavior throughout the segmentation process from fundamentally local decisions. In essence, current deformable models have no explicit awareness of where they are in the image, how their parts are arranged, or what they or any neighboring deformable models are seeking at any time during the optimization process.

The time has come to shift our attention to what promises to be a critical element in any viable, highly automated solution: the decision-making framework itself. Existing decision-making strategies for deformable models are inflexible and do not operate at an appropriate level of abstraction. Hierarchically organized models, which shift their focus from structures associated with stable image features to those associated with less stable features, are a step in the right direction [McInerney1998, Shen2000]. However, high-level contextual knowledge remains largely ineffective because it is intertwined much too tightly with the low-level optimization-based mechanisms.

It is our contention that we must revisit ideas for incorporating knowledge that were explored in earlier systems (e.g., [Tsotsos1980]), and develop new algorithms that focus on top-down reasoning strategies which may best leverage the powerful bottom-up feature detection and integration abilities of deformable models and other modern model-based medical image analysis techniques. We further contend that a layered architecture is appropriate, where the high-level reasoning layer has knowledge about and control over the low-level model (or models) at all times. The reasoning layer should apply an active, explicit search strategy that first looks for the most stable image features before proceeding to less stable image features, and so

on. It should utilize contextual knowledge to resolve regions where there is a deficiency of image feature information.

To achieve these goals, we introduce a new paradigm for automatic medical image analysis that adopts concepts from the emerging field of Artificial Life². In particular, we develop *deformable organisms*, autonomous agents whose objective is the segmentation and analysis of anatomical structures in medical images. A deformable organism is structured as a ‘muscle’-actuated ‘body’ whose behavior is controlled by a ‘brain’ that is capable of making both reactive and deliberate decisions. This intelligent deformable model possesses a non-trivial ‘awareness’ of the segmentation process, which emerges from a conflux of perceived sensory data, an internal mental state, memorized knowledge, and a cognitive plan. By constructing deformable organisms in a layered fashion, we are able to separate the knowledge-driven model-fitting control functionality from the data-driven, local image feature integration functionality, exploiting both for maximal effectiveness. This separation allows us to construct a model-fitting controller, or ‘brain’, from an extensible set of standardized ‘behavior’ subroutines that are defined in terms of high-level anatomical features of an object rather than its low-level image features. The organism’s brain makes decisions at an appropriately high level of abstraction by integrating sensed image features, memorized knowledge, and a pre-stored cognitive plan. It then issues ‘muscle’ commands and ‘growth’ commands to realize shape deformations that correctly match the model to the data.

Our ALife approach provides us with the required flexibility to adhere to an active, explicit search strategy that first directs organisms to look for the most stable anatomical features in images before deforming or growing towards less stable features, and so on. The result is intelligent organisms that are aware of the progress of the segmentation process and of each other, allowing them to effectively and selectively apply prior knowledge of the target objects throughout their development. Our approach combines a common, layered architecture and a set of standard behavior subroutines allowing powerful and flexible ‘custom-tailored’ models to be rapidly constructed, providing general-purpose tools for automated medical image segmentation, object-based registration, and shape variation measurement.

The deformable organisms that we have developed to date are based on medial axis deformable representations, which provide their brains with precise control over the lower-level deformation layer, allowing a variety of controlled deformations and controlled growth at multiple locations and scales. We demonstrate the potential of our approach by releasing several such organisms

² Earlier mentions of the use of ALife in conjunction with segmentation appeared in [Choi1997, Kagawa1999]. These methods still relied on local decision making akin to traditional region growing methods.

into 2D medical images, resulting in the robust, automatic segmentation of various anatomical structures.

8.1.1 ALife for Computer Graphics

In recent years, computer graphics researchers have been exploring the modeling and simulation of living systems by applying concepts from an emerging scientific discipline known as Artificial Life. The Artificial Life (ALife) modeling approach has been applied successfully to produce realistic computer graphics models of plants and animals (ALife) [Terzopoulos1999]. As is characteristic of the deformable models used in MIA, these new graphics models employ geometric and physics-based modeling techniques, but they also aspire to simulate many of the biological processes that uniquely characterize living systems - including birth and death, growth and development, natural selection, evolution, perception, locomotion, manipulation, adaptive behavior, learning, and intelligence. This provocative paradigm subsumes several avenues of research and development, including artificial plants and animals, behavioral modeling and animation, and evolutionary modeling.

To manage their complexity, ALife models of animals are best organized hierarchically (Figure 8.1), where each successive modeling layer augments the more primitive functionalities of underlying layers. At the base of the modeling hierarchy, a geometric modeling layer represents the morphology and appearance of the artificial animal. Next, a physical modeling layer incorporates biomechanical principles to constrain the geometry and simulate biological tissues. Further up the hierarchy is a motor control layer that motivates internal muscle actuators in order to synthesize lifelike locomotion. Behavioral and perceptual modeling layers cooperate to support a reactive behavioral repertoire, e.g. navigation among obstacles, foraging, etc. At the apex of the modeling pyramid is a cognitive modeling layer, which simulates the deliberative behavior of higher animals, governs what an artificial graphical animal knows, how that knowledge is acquired and represented, and how automated reasoning and planning processes can exploit it to achieve high-level goals while producing graphical characters with some level of deliberate intelligence.

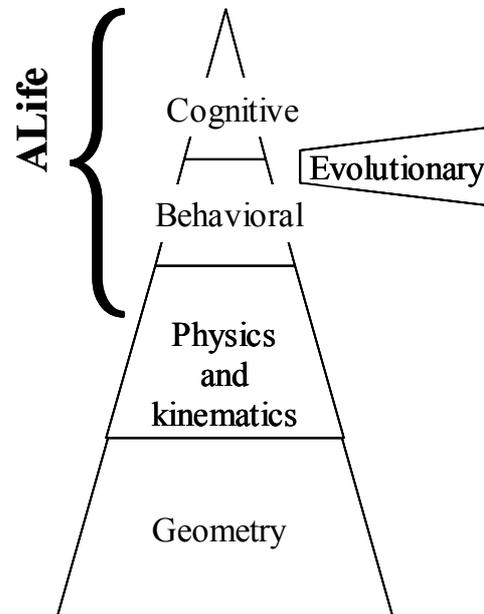


Figure 8.1. The ALife modeling pyramid (adapted from [Terzopoulos1999]).

Autonomous agents known as “artificial fishes” [Terzopoulos1994] (see also [Tu1996]) serve to illustrate the key functional components of artificial animals: bodies that comprise muscle actuators, sensory organs (eyes, etc.) and, most importantly, brains consisting of motor, perception, behavior, learning and cognition centers. Motor controllers (MCs) in the brain’s motor center coordinate muscle actions to carry out specific motor functions, such as locomotion and sensor actuation. For example MCs translate natural control parameters such as forward speed or angle of the turn into detailed muscle actions in order to carry out a specific motor function, such as ‘swim-forward’ or ‘turn-left’. The perception center employs attention mechanisms to interpret sensory information about the dynamic environment³. The perceptual attention mechanism allows the artificial fish to train its sensors at the world in a task-specific way, hence filtering out sensory information superfluous to its current behavioral needs. The behavior center realizes an adaptive sensorimotor system through a repertoire of behavior routines that couple perception to action in meaningful ways. An intention generator (or action-selection mechanism), the fish’s cognitive faculty, combines the fish’s innate characteristics, the mental state, and the incoming stream of sensory information at every simulation time step. It uses this information to activate behavior routines. The behavior

³ For a closely related biomimetic approach to active vision, see [Terzopoulos1997].

routines in turn compute the appropriate motor controller parameters to carry the fish one step closer to fulfilling its current intention. The learning center in the brain enables the artificial animal to learn motor control and behavior through practice and sensory reinforcement. The learning center of the brain comprises a set of optimization-based motor learning algorithms that can discover and perfect motor controllers capable of producing efficient locomotion [Terzopoulos1996]. The cognition center enables it to ‘think’.

8.1.2 An ALife Modeling Paradigm for MIA: Motivation

Viewed in the context of the artificial life modeling hierarchy (Figure 8.1). Current *automatic* deformable model-based approaches to medical image analysis utilize geometric and physical modeling layers only. In *interactive* deformable models, such as snakes, the human operator is relied upon to provide suitable behavioral level and cognitive level support. At the physical level, deformable models interpret image data by simulating dynamics or minimizing energy terms, but the models themselves do not monitor or control this optimization process except in a most primitive way.

At the geometric level, aside from a few notable exceptions (e.g., [Staib1992, Terzopoulos1991]), deformable models are not generally designed with intuitive, multi-scale, multi-location deformation ‘handles’ and are often boundary-based and lack global shape descriptors [McInerney1996]. Their inability to perform global deformations, such as bending, and other global motions such as sliding and backing up makes it difficult to develop reasoning or planning strategies for these models at the correct level of abstraction.

In more sophisticated deformable models prior information in the form of measured statistical variation is used to constrain model shape and appearance [Cootes1995a, Cootes1999, Szekely1996]. However, these models have no explicit awareness of where they or their neighbors are and, consequently, the effectiveness of these constraints is dependent upon model initialization conditions. The lack of awareness also prevents the models from taking proper advantage of neighborhood information via model interaction and prevents them from knowing when to trust the image feature information and ignore the constraint information and vice versa. The constraint information is therefore applied arbitrarily. Furthermore, because there is no active, explicit search for stable image features, the models are prone to latching onto incorrect features [Cootes1999], simply due to their myopic decision-making abilities and the proximity of spurious features. Once this latching occurs, the lack of explicit control of the fitting procedure prevents the model from correcting such missteps. The result is that the local decisions that are made do not add up to intelligent global behavior.

For example, when segmenting the corpus callosum (CC) in 2D mid-sagittal images⁴, the ‘vocabulary’ that one uses should contain words that describe principal anatomical features of the CC, such as the genu, splenium, rostrum, fornix, ‘ribbon’ body (Figure 8.2), rather than pixels and edges. The deformable model should match this natural descriptiveness by grouping intuitive model parameters at different scales and locations within it, rather than providing localized boundary-based parameters only.

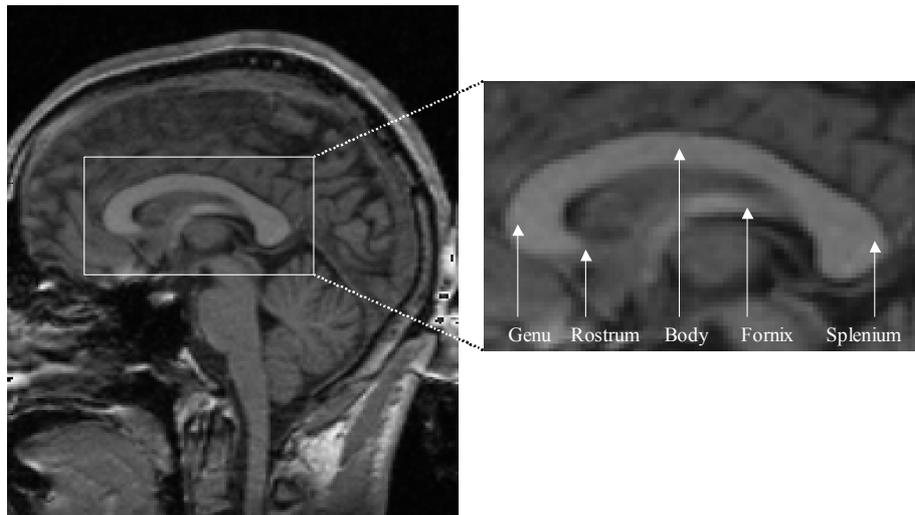


Figure 8.2. CC anatomy overlaid on a mid-sagittal MRI brain slice.

To overcome the aforementioned deficiencies while retaining the core strengths of the deformable model approach, we add high-level controller layers (a ‘brain’) on top of the geometric and physical (or deformation) layers to produce an autonomous deformable organism (Figure 8.3). The planned activation of these lower layers allows us to control the fitting/optimization procedure. The layered architecture approach allows the deformable organism to make deformation decisions at the correct level of abstraction utilizing prior knowledge, memorized information, sensed image features, and inter-organism interaction.

The perception system of the deformable organism comprises a set of sensors that provide information. Any type of sensors can be incorporated, from edge strength and edge direction detectors to snake ‘feelers’. Sensors can be focused or trained for specific image features and image feature variation in a task-specific way; hence, the organism can disregard sensory information superfluous to its current behavioral needs.

⁴ See [Lundervold1999] as an example of previous work on, and motivation for, segmenting the corpus callosum.

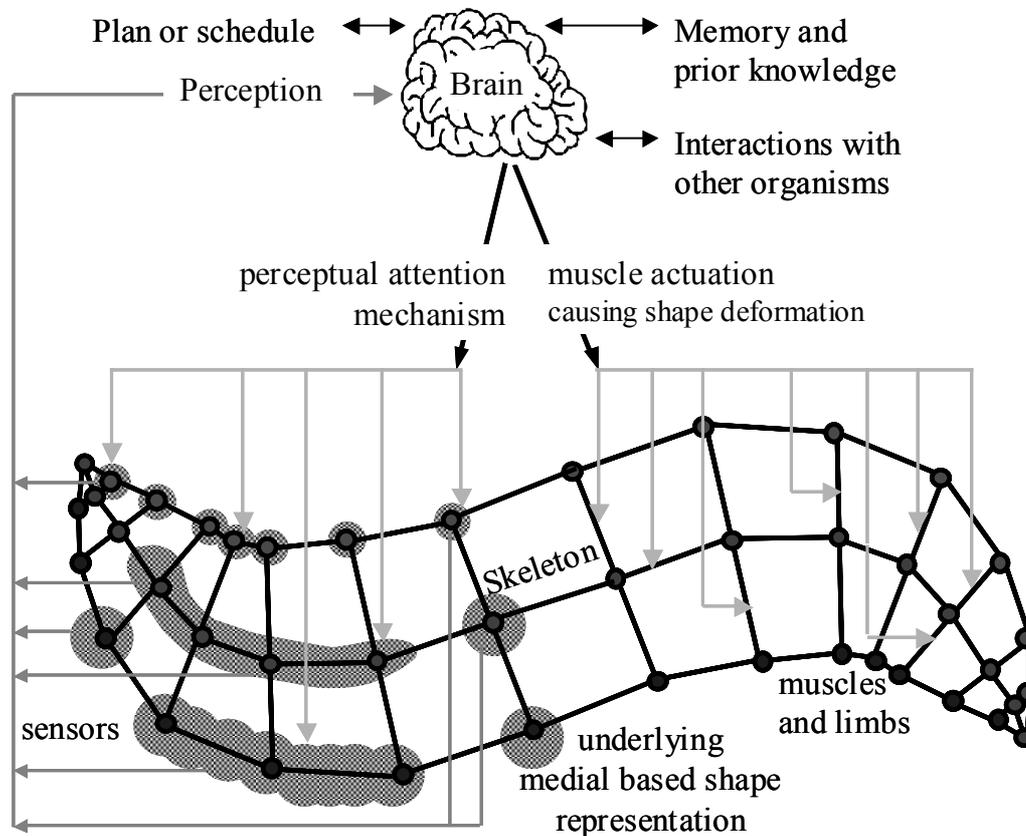


Figure 8.3. A deformable organism: The brain issues ‘muscle’ actuation and perceptual attention commands. The organism deforms and senses image features, whose characteristics are conveyed to the brain. The brain makes decisions based on sensory input, memorized information and prior knowledge, and a pre stored plan, which may involve interaction with other organisms.

Explicit feature search requires powerful, flexible and intuitive model deformation control. We achieve this with a set of ‘motor’ (i.e. deformation) controllers, which are parameterized procedures dedicated to carrying out a complex deformation function, such as successively bending a portion of the organism over some range of angles or stretching part of the organism forward some distance.

We use a natural, intuitive medial description of object shape plus medial-based statistics of (localized) shape variation as our prior shape knowledge representation scheme. The medial-based shape descriptors may easily be mapped onto anatomical features of an object. A primitive cognitive layer activates ‘behavior’ routines (e.g., for a CC organism: find-splenium,

find-genu, find-upper-boundary-of-CC) according to a plan or schedule (Figure 8.4). Customized behavior routines may be quickly constructed from a set of standardized subroutines. The behavior routines subsequently activate ‘motor’ (i.e., deformation) controller routines or growth controller routines, enabling the organism to fulfill its goal of object segmentation. The plan (or plans) can be generated with the aid of a human expert, since the behavior routines are defined using familiar anatomical terminology.

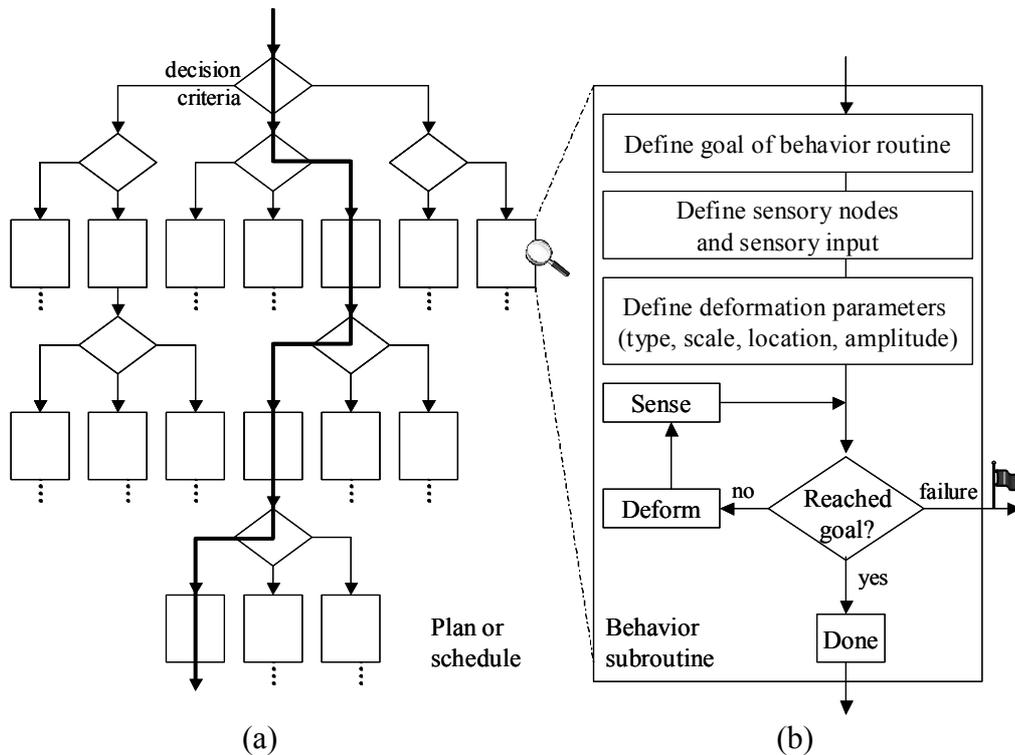


Figure 8.4. (a) A procedural representation of a fragment of a deformable organism's plan or schedule. The organism goes through several behavior subroutines (bold path in (a)). (b) A simple example of a standard behavior subroutine.

An organism is ‘self-aware’ (i.e. knows where it and its parts are and what it is seeking at every stage) and ‘neighbor-aware’ and therefore is able to effectively utilize global contextual knowledge. An organism begins by searching for the most stable anatomical features in the image before proceeding to less stable features. Alternatively, an organism may ‘interact’ with other organisms to determine optimal initial conditions. Once stable features are found and labeled, an organism can selectively use prior knowledge or information from neighbor organisms to determine the object

boundary in regions known to offer little or no feature information. That is, the organism intelligently ‘fills in’ in ways tailored to specific regions of interest in the target structure.

Because an organism carries out active, explicit searches for stable anatomical features, it is not satisfied with the nearest matching feature but looks further within a region to find the best match, thus avoiding local minimum solutions. Furthermore, by carrying out explicit searches for features we ensure correct correspondences between the model and the data. If a feature cannot be found, an organism may ‘flag’ this situation (Figure 8.4(b)). In the future, if multiple plans exist, another plan could potentially be selected and the search for the missing feature postponed until further information is available.

The deformation capabilities are achieved with a set of ‘motor’ (i.e. deformation) controllers and medial-based deformation operators. Deformation controllers are parameterized procedures dedicated to carrying out a complex deformation function, such as successively bending a portion of the organism over some range of angle or stretching part of the organism forward some distance. They translate natural control parameters such as `<bend_angle, location, scale>` or `<stretch_length, location, scale>` into detailed deformations. We currently use medial-based profiles (presented in Chapter 6) for shape representation and deformation. Other shape representation and deformation techniques can be utilized as long as they are able to provide intuitive, controlled, and statistically feasible shape deformations.

Interaction among organisms may be as simple as collision detection and avoidance (and hence prevention of inter-penetration between organisms sharing a common boundary), to one or several organisms supplying intelligent initial conditions to another, to the use of inter-organism statistical shape/image appearance constraint information.

Finally, an organism may begin in an ‘embryonic’ state with a simple proto-shape, and then undergo controlled growth as it develops into an ‘adult’, proceeding from one stable object feature to the next. Alternatively, an organism may begin in a fully developed state and undergo controlled deformations as it carries out its model-fitting plan. The type of organism to use, or whether to use some sort of hybrid organism, is dependent on the image and shape characteristics of the target anatomical structure.

In summary, the ALife modeling paradigm provides a common framework and standard behavior subroutines upon which to build powerful and flexible ‘custom-tailored’ models that offer robustness and generality.

8.2 Deformable Organisms for Medical Image Analysis

In this section we describe the layered-architecture of a deformable organism beginning with the lower layers and progressing upwards (we will make

reference to the detailed construction of the layered-architecture for a corpus callosum organism as an example).

8.2.1 Shape Representation: Geometry

Our organisms are based on a medial representation of body morphology that facilitates a variety of controlled local deformations at multiple spatial scales. In particular, we use the medial-based shape profiles presented in Chapter 6 (see also [Hamarnah2001a]) to describe the body of an organism. In this shape representation and deformation scheme, an anatomical structure (e.g. CC) is described with four shape profiles derived from the primary medial axis of an organism's boundary contour (e.g. CC boundary contour). The medial profiles describe the geometry of the structure in a natural way and provide general, intuitive, and independent shape measures. These profiles are: length, orientation, left (with respect to the medial axis) thickness, and right thickness. The length profile represents the distances between consecutive pairs of medial nodes, and the orientation profile represents the angles of the edges connecting the pairs of nodes. The thickness profiles represent the distances between medial nodes and their corresponding boundary points on both sides of the medial axis.

Other shape representation and shape deformation schemes can be employed in the lower layers of the modeling pyramid as long as they provide sufficient shape description and deformation capabilities to the upper layers. An alternative could be the physics-based shape representation and deformation technique presented in Chapter 7 (see also [Hamarnah2001e]).

8.2.2 Motor System

Shape deformation actuators (motor skills). In addition to the affine transformation abilities (translate, rotate, scale), we control the organism's shape deformations by defining deformation actuators in terms of the medial shape profiles (or alternatively in a physically-based manner as described in Chapter 7). Controlled stretch (or compress), bend, and bulge (or squash) deformations are implemented as deformation operators acting on the length, orientation, or thickness profiles, respectively. Furthermore, by utilizing a hierarchical (multiscale) and regional principal component analysis to capture the shape variation statistics in a training set, we can keep the deformations consistent with prior knowledge of possible shape variations. Whereas general statistically-derived shape models produce global shape variation modes only [Cootes1995a, Cootes1999, Szekely1996], we are able to produce spatially-localized feasible deformations at desired scales, thus supporting our goal of intelligent deformation planning.

Several operators of varying types, amplitudes, scales, and locations can be applied to any of the length, orientation, and thickness shape profiles. Similarly, multiple statistical shape variation modes can be activated, where each mode acts on a specific shape profile with a specific amplitude, location and scale. In general, operator- and statistics-based deformations can be combined into a single expression of shape. More details and examples of medial-based and physics-based shape deformations can be found in Chapter 6 and Chapter 7, respectively.

Deformation (motor) controllers. The organism's low-level motor actuators are controlled by motor controllers. These parameterized procedures carry out complex deformation functions such as sweeping over a range of rigid transformation parameters, sweeping over a range of stretch/bend/thickness amplitudes at a certain location and scale, bending at increasing scales, moving a bulge on the boundary, etc. Other high-level deformation capabilities include smoothing the medial or left or right boundaries, interpolating a missing part of the thickness profile, moving the medial axis (the organism's 'spine') to a position midway between the left and right boundaries, re-sampling the model by including more medial and boundary nodes, and shortening or elongating the organism while maintaining the original thickness and orientations at appropriate arc lengths.

8.2.3 Perception System

The perception system of our organism consists of a set of sensors that provide image information to the deformable organisms. Sensors can be virtually anything - from edge strength and edge direction detectors to snake 'feelers'. Sensors can be focused or trained for specific image feature and image feature variation in a task-specific way and hence the organism is able to disregard sensory information superfluous to its current behavioral needs.

Different parts of the organism are dynamically assigned sensing capabilities and thus act as sensory organs (SO) or receptors. The locations of the SO are typically confined to the organism's body (on-board SO) such as at its medial or boundary nodes, at curves or segments connecting different nodes, or at other inner regions. Nevertheless, off board can be utilized as in the example of segmenting a vessel in an angiogram (Figure 8.14).

In our implementation of the Corpus Callosum organism, for example, the SO are made sensitive to different stimuli such as image intensity, image gradient magnitude and direction, a non-linearly diffused version of the image, an edge detected (using Canny edge detector) image, or even the result of a Hough transform (applied to find the top of the human skull in the MR image). In general, a wide variety of image processing and analysis techniques can be applied to the original image source and thus act as focusers or filters of the

‘outside world’ signals (Figure 8.5). The sensed data are fed to the cognitive center of the brain for processing.

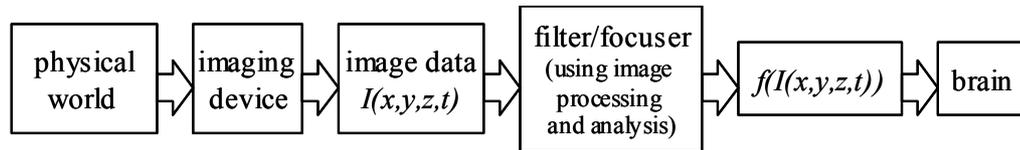


Figure 8.5. Perception System.

8.2.4 Behavioral/Cognitive System

The organism’s cognitive center combines sensory information, memorized information and prior knowledge, and instructions from a pre-stored segmentation plan to carry out active, explicit searches for stable object features by activating ‘behavior’ routines (Figure 8.4). Currently a single fixed plan is used, multiple plans with a plan selection scheme could also be implemented. Behavior routines are designed based on available organism motor skills, perception capabilities, and available anatomical landmarks. For example, the routines implemented for the CC organism include: find-top-of-head, find-upper-boundary-of-CC, find-genu, find-rostrum, find-splenium, latch-to-upper-boundary, latch-to-lower-boundary, find-fornix, thicken-right-side, thicken-left-side, back-up. The behavior routines subsequently activate the deformation or growth controllers to complete a stage in the plan and bring an organism closer to its intention of object segmentation.

The segmentation plan provides a means for human experts to intuitively incorporate global contextual knowledge. It contains instructions on how best to achieve a correct segmentation by optimally prioritizing behaviors. If we know, for example, that the corner-shaped rostrum of the CC is always very clearly defined in an MRI image, then the find-rostrum behavior should be given a very high priority. Adhering to the segmentation plan and defining it at a behavioral level imbues the organism with awareness of the segmentation process. This enables it to make very effective use of prior shape knowledge - it is applied only in anatomical regions of the target object where there is a high level of noise or known gaps in the object boundary edges etc.

8.3 Results

We first present a detailed segmentation plan for the CC organism that serves to illustrate the ability to harness global contextual knowledge. A CC organism is released into a 2D mid-sagittal MRI brain image from an initial default position (Figure 8.6.1). It then goes through different ‘behaviors’ as it progresses towards its goal. As the upper boundary of the CC is very well

defined and can be easily located with respect to the top of the head, the cognitive center of the CC activates behaviors to first locate the top of the head (Figure 8.6.2-3) then move downwards through the gray and white matter in the image space to locate the upper boundary (Figure 8.6.4-7). The organism then bends to latch to the upper boundary (Figure 8.6.8) and activates a find-genu routine (refer to Figure 8.2 for CC anatomy) causing the CC organism to stretch and grow along this boundary towards the genu (Figure 8.6.9-11). It then activates the find-rostrum routine causing the organism to back up, thicken (Figure 8.6.12), and track the lower boundary until reaching the distinctive rostrum (Figure 8.6.13-15). Once the rostrum is located, the find-splenium routine is activated and the organism stretches and grows in the other direction (Figure 8.6.15-16). The genu and splenium are easily detected by looking for a sudden change in direction of the upper boundary towards the middle of the head. At the splenium end of the CC, the organism backs up and finds the center of a circle that approximates the splenium end cap (Figure 8.6.17). The lower boundary is then progressively tracked from the rostrum to the splenium while maintaining parallelism with the organism's medial axis in order to avoid latching to the potentially occluding fornix (Figure 8.6.18-21). Nevertheless, the lower boundary might still dip towards the fornix so a successive step of locating where, if any, the fornix does occlude the CC is performed by activating the find-fornix routine (making use of edge strength along the lower boundary, its parallelism to the medial axis, and statistical thickness values). Thus, prior knowledge is applied only when and where required. If the fornix does indeed occlude the CC, any detected dip in the organism's boundary is repaired by interpolation using neighboring thickness values (Figure 8.9). The thickness of the upper boundary is then adjusted to latch to the corresponding boundary in the image (Figure 8.6.22-26). At this point the boundary of the CC is located (Figure 8.6.26) and the CC organism has almost reached its goal. However, at this stage the medial axis is not in the middle of the CC organism (Figure 8.6.27) so it is re-parameterized until the medial nodes are halfway between the boundary nodes (Figure 8.6.28-30). Finally the upper and lower boundaries, which were reset in the previous step, are re-located (Figure 8.6.31-36) to obtain the final segmentation result (Figure 8.6.36). Other CC segmentation (Figure 8.7), validation results (Figure 8.8), and a demonstration of the organism's self-awareness (Figure 8.10) are presented.

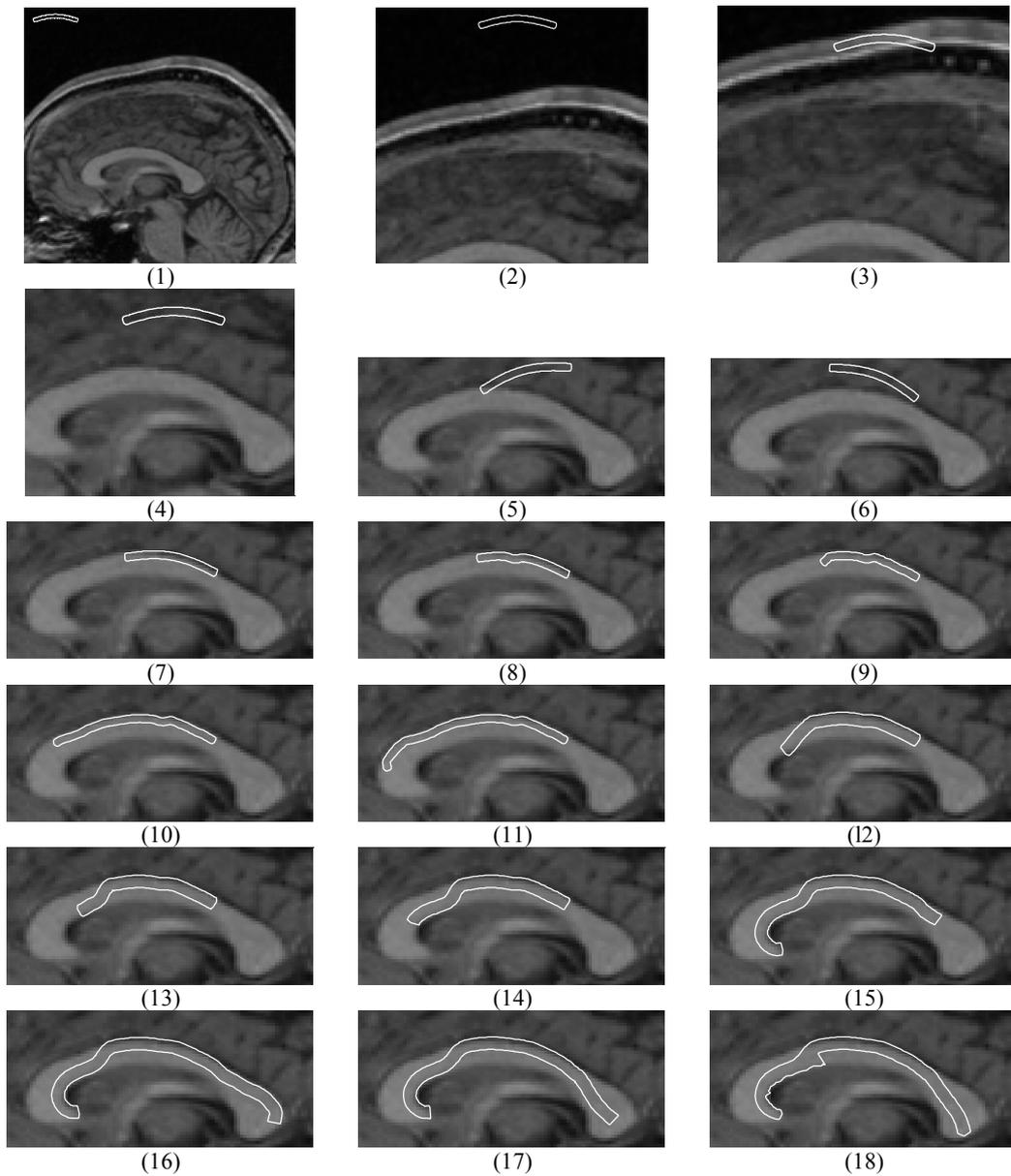
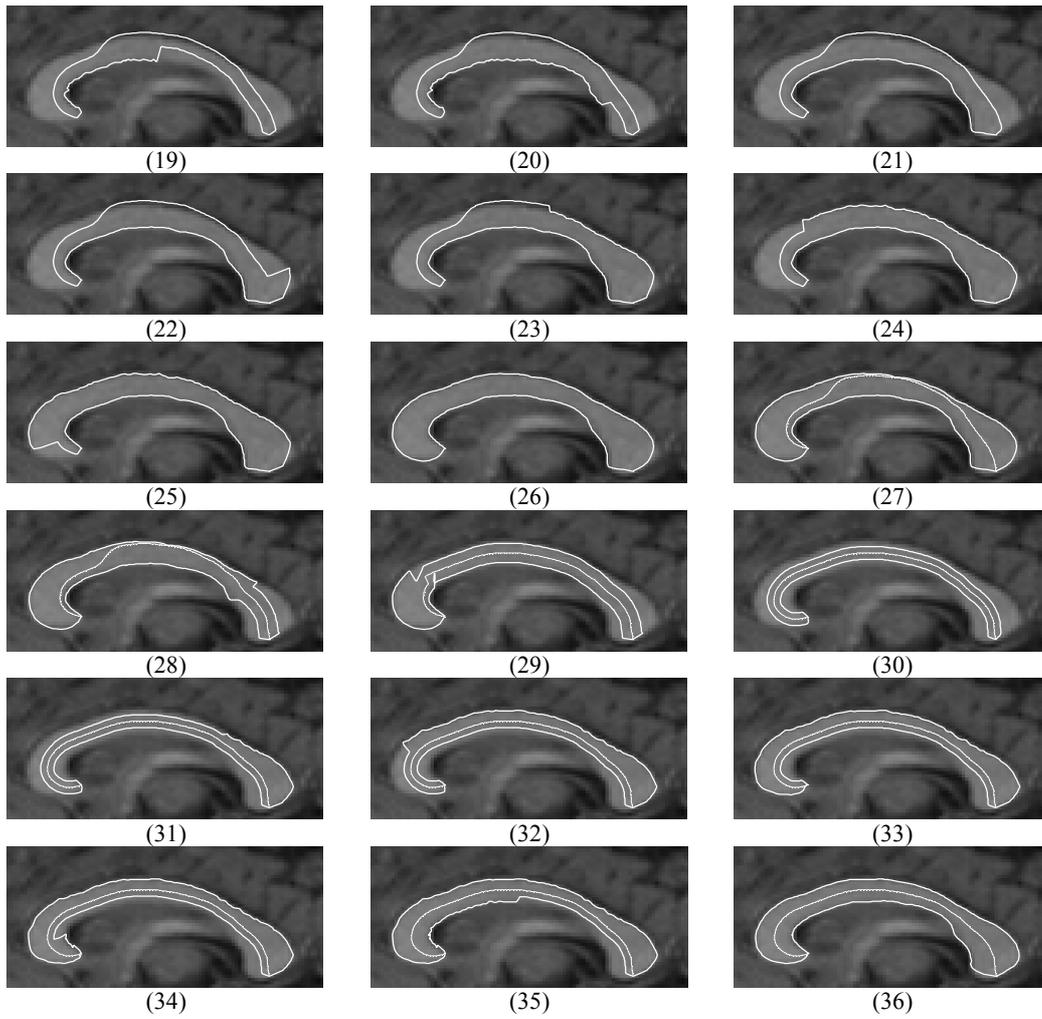


Figure 8.6. Deformable corpus callosum organism progressing through a sequence of behaviors to segment the CC (results continued on next page).



Results continued from previous page.

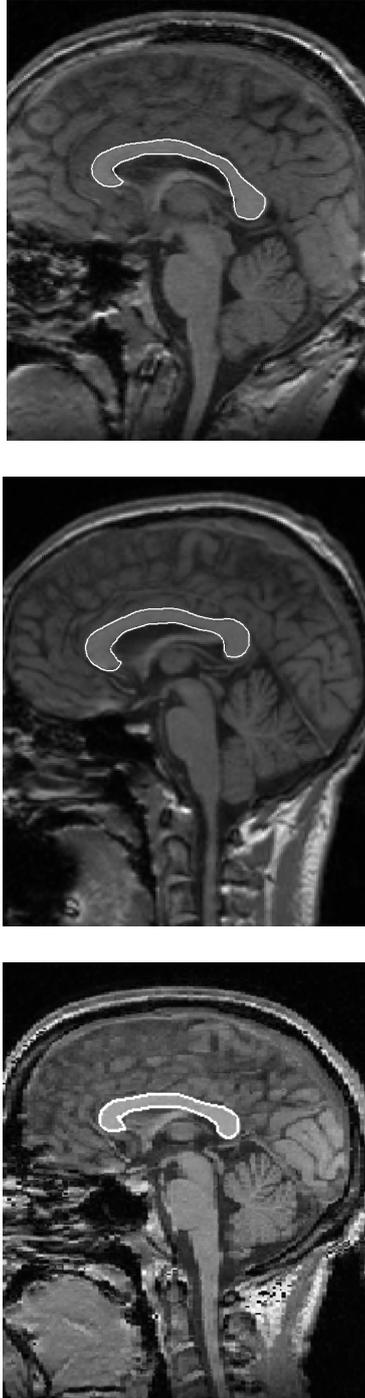


Figure 8.7. Segmentation results

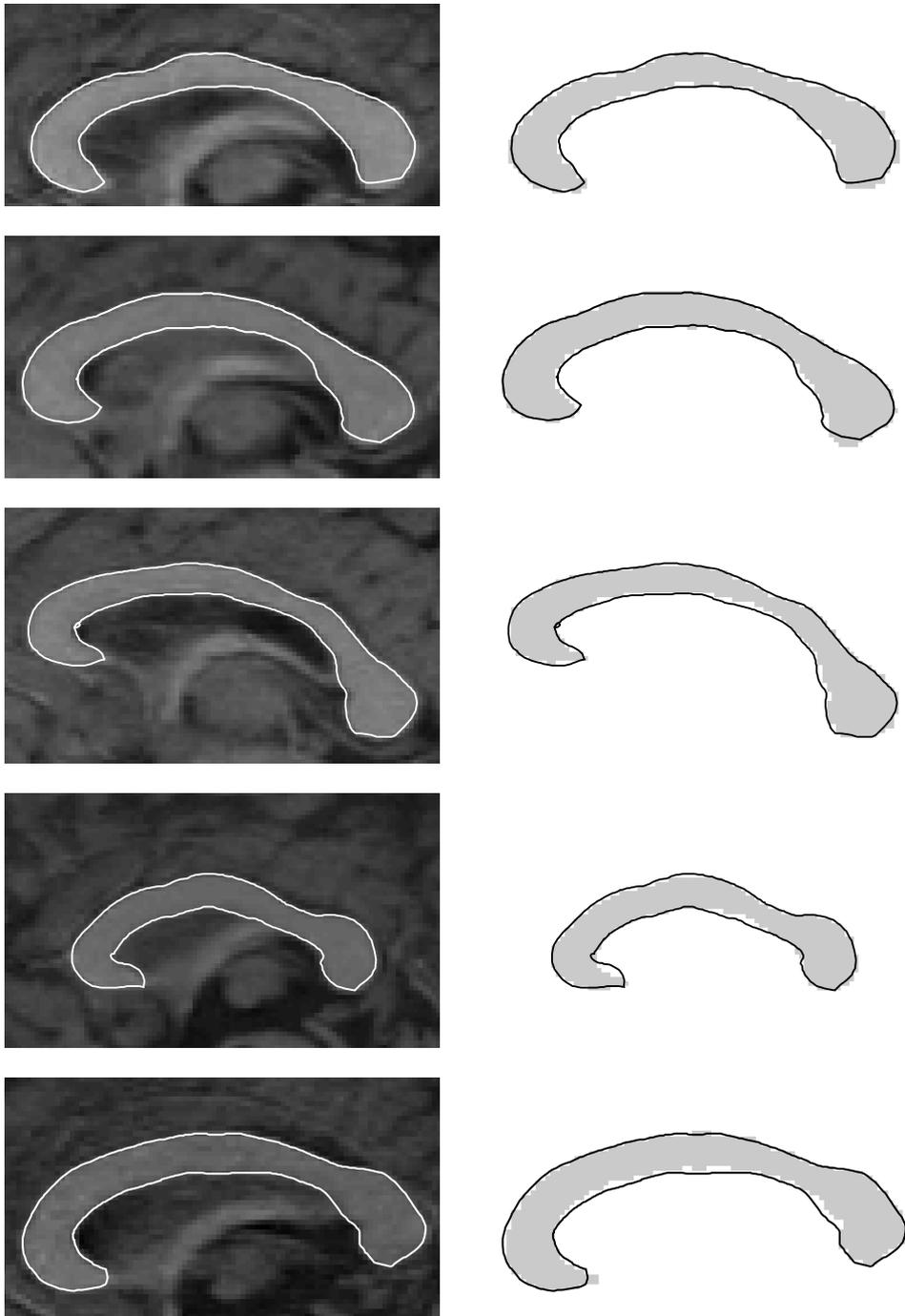


Figure 8.8. Segmentation results (left), also shown (in black) over manually segmented (gray) corpora callosa (right).

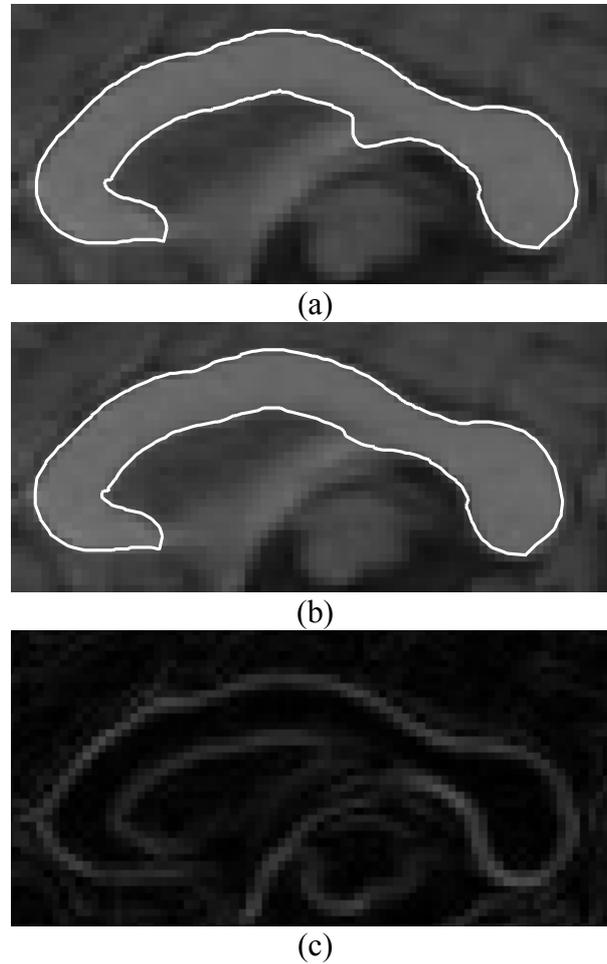


Figure 8.9. Segmentation result (a) before, (b) after detecting and repairing the fornix dip. (c) Note the weak gradient magnitude where the fornix overlaps the CC.

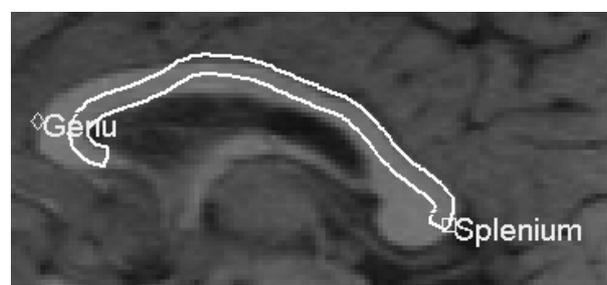


Figure 8.10. The CC organism's self-awareness enables it to identify landmark parts.

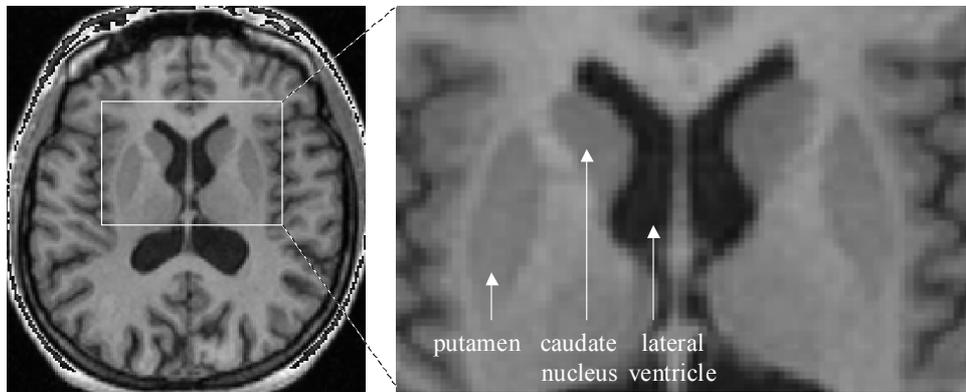


Figure 8.11. The lateral ventricle, caudate nucleus, and putamen shown in transversal brain MRI slice.

Other interacting organisms were also created to locate the lateral ventricles, caudate nuclei, and putamina in the left and right halves of transversal MR brain images (Figure 8.11). Since the ventricles are the most stable of the above structures, a ventricle organism is first released (Figure 8.12.1). It proceeds to locate the top of the ventricle (Figure 8.12.2) and its inner and outer (with respect to the brain) boundaries (Figure 8.12.3-5). Both ends of the ventricle organism are actively stretched to locate both the upper and the lower lobes of the ventricle (Figure 8.12.6). The ventricle organism then passes information about the shape and location of the segmented ventricle (Figure 8.12.7) to the caudate nucleus (CN) organism, which is initialized accordingly in a suitable position (Figure 8.12.8). The CN organism segments the CN by stretching to locate its upper and lower limits (Figure 8.12.9) and thickening to latch to its inner and outer boundaries (Figure 8.12.10). The CN organism passes information about the location of its lowest point (in the image) to the putamen organism, which is initialized accordingly (Figure 8.12.11). The putamen organism moves towards the putamen in the brain image (Figure 8.12.12) and then rotates and bends to latch to the nearer putamen boundary (Figure 8.12.13). It then stretches and grows along the boundary until reaching the upper- and lower-most ends of the putamen (Figure 8.12.14), which identifies the medial axis of the putamen (Figure 8.12.15). Since the edges of the putamen boundary near the gray matter are usually weak, the organism activates an explicit search for an arc (parameterized only by one parameter controlling its curvature) that best fits the weak, sparse edge data in that region (Figure 8.12.16).

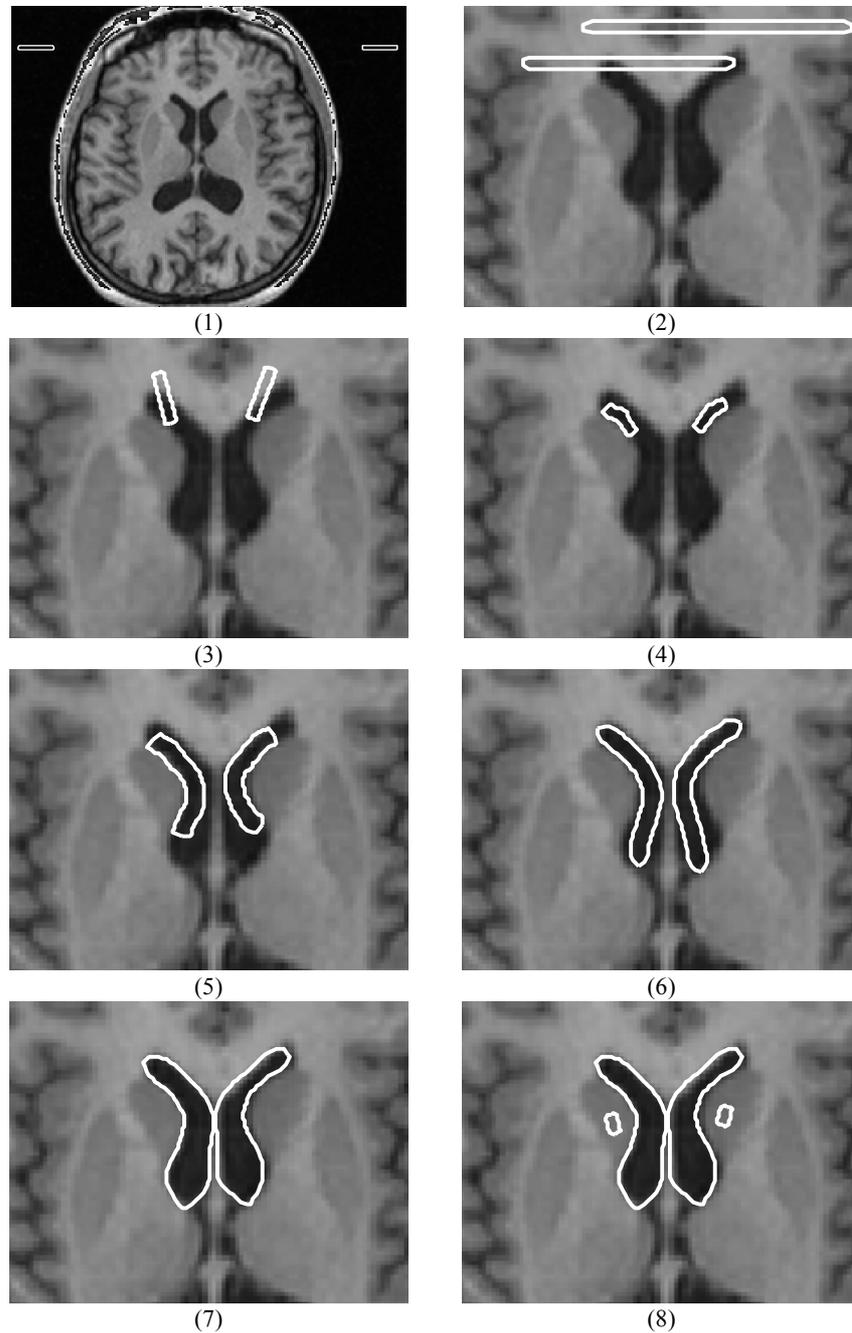
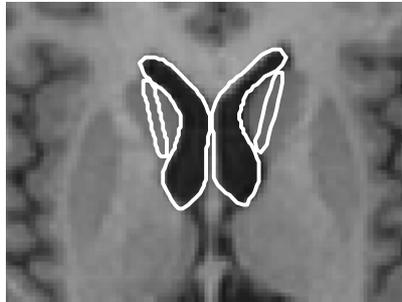
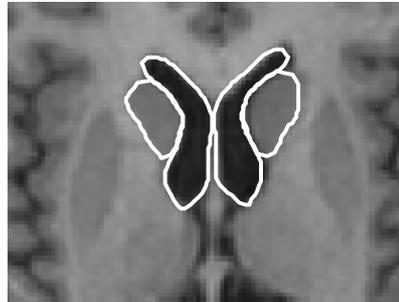


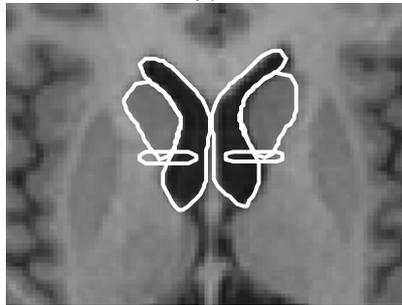
Figure 8.12. Deformable lateral ventricles (1-16), caudate nuclei (CN) (8-16), and putamina (11-16) organisms progressing through a sequence of behaviors to locate the corresponding structures in an MR brain image (results continued on next page).



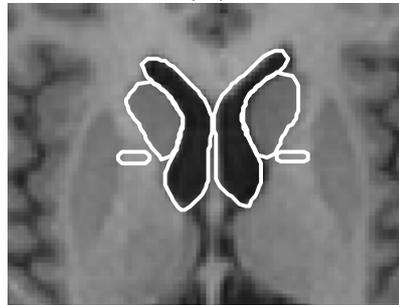
(9)



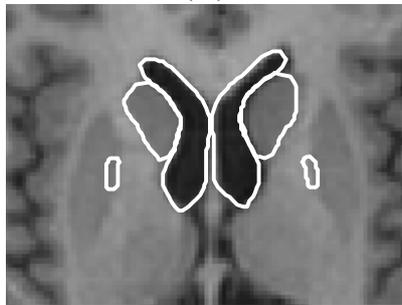
(10)



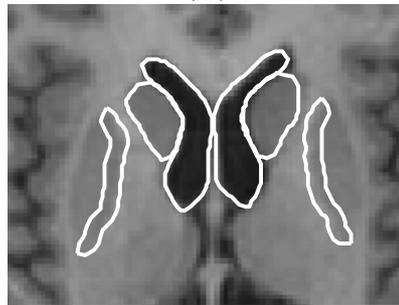
(11)



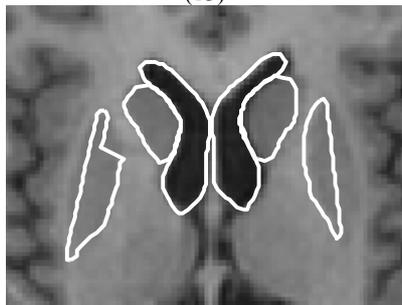
(12)



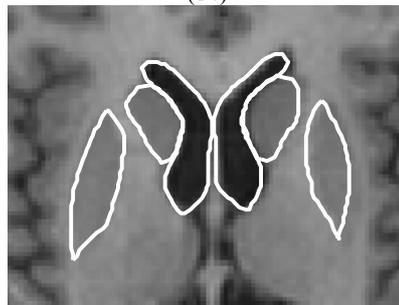
(13)



(14)



(15)



(16)

Results continued from previous page.

We also present the result of segmenting vessels in an angiogram (Figure 8.13) using an ‘artery crawler’. Without proper constraints the vessel organism latches onto the wrong overlapping vessel (Figure 8.13(a)). However, adding additional sensors and high-level constraints enables the organism to distinguish between overlapping vessels and bifurcations (Figure 8.13(b)). When the latter is encountered two new organisms are born from the original main branch organism, one for each branch (Figure 8.13(c)). Figure 8.14 demonstrates how this decision (overlap or bifurcate) strategy is implemented.

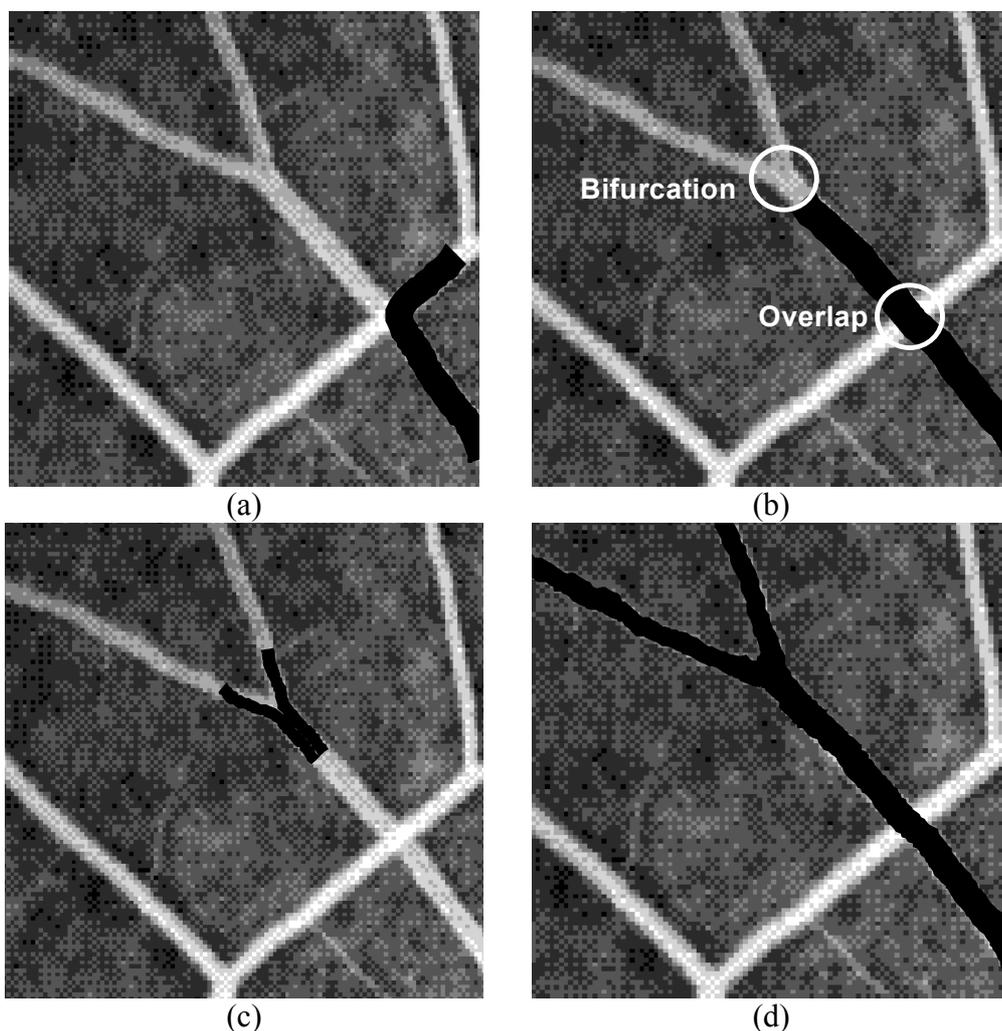


Figure 8.13. Segmenting vessels in an angiogram. (a) A deformable organism turning right and latching onto the wrong overlapping vessel. (b) High-level constraints enable the organism to differentiate between overlapping vessels and bifurcations. (c) Two new organisms are born upon identifying a bifurcation. (d) The segmented main vessel and its two branches.

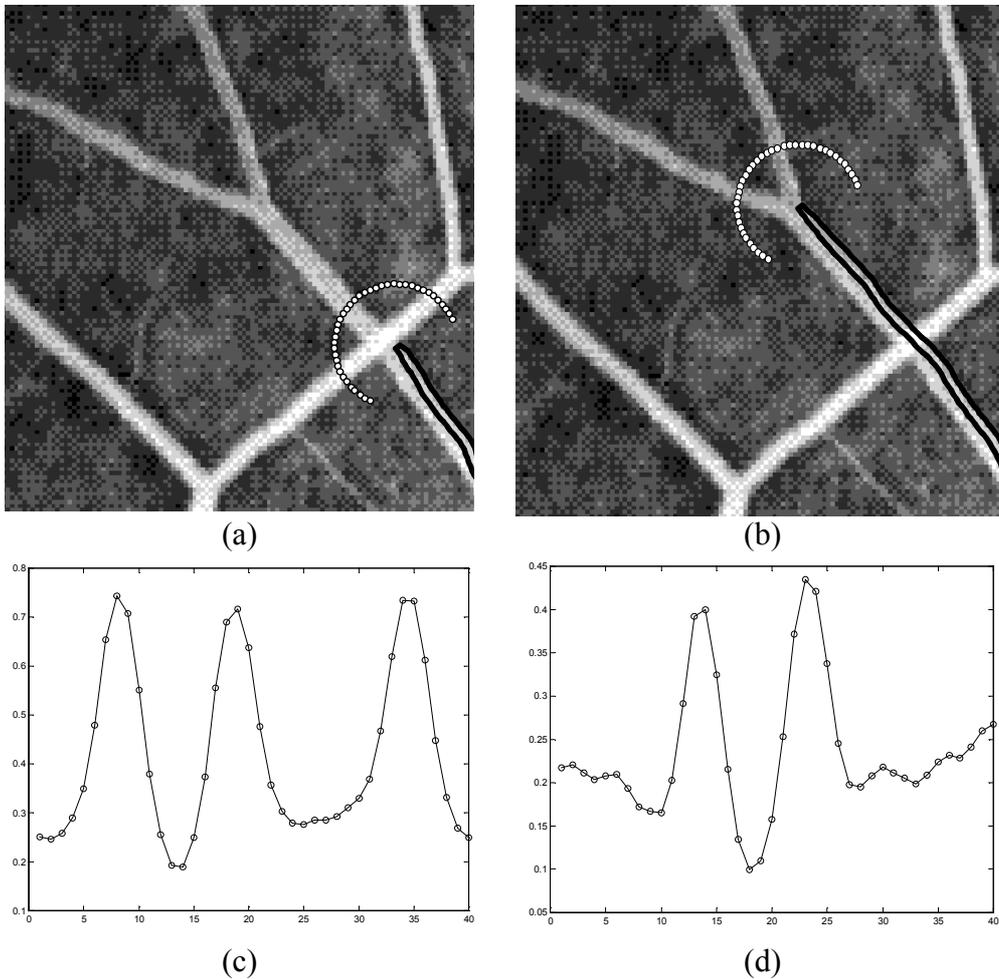


Figure 8.14. Off-board sensors (the arc of white nodes in (a) and (b)) measure image intensity (along the arc). This results in an intensity profile exhibiting three distinct peaks when an overlapping vessel is ahead (c) and only two peaks in the case of a bifurcation (d).

8.4 Conclusion

Robust, automatic medical image analysis requires the incorporation and intelligent utilization of global contextual knowledge. We have introduced a promising new paradigm for medical image analysis that applies concepts from artificial life modeling to meet this requirement. By architecting a deformable model-based framework in a layered fashion, we are able to separate the ‘global’ top-down, model-fitting control functionality from the local, bottom-

up, feature integration functionality. This separation allows us to define a model-fitting controller or 'brain' in terms of the high-level anatomical features of an object rather than low-level image features. The layered-architecture also provides the brain layer with precise control over the lower-level model deformation layer. The result is a deformable organism that is continuously aware of the progress of the segmentation, allowing it to apply effectively prior knowledge of the target object. We have demonstrated the potential of our novel approach by constructing several deformable organisms (corpus callosum worm, artery crawler, interacting lateral ventricle, caudate nucleus, and putamen organisms) and releasing them into several medical images in order to segment and label various structures.

Chapter 9. FUTURE RESEARCH

There's always room for improvement. In this chapter we mention remaining outstanding issues and potential improvements that we believe are interesting and deserve future investigation.

9.1 Specific Improvements

Choosing the weights and parameters of the snake model is an important and often tedious task. When applying snakes to the problem of segmenting oral lesions in color images (Chapter 2), great variability was encountered between the appearances (boundary shape, texture, color) of different lesions. This means that the same snake parameters will not necessarily be optimal for all cases. A potential solution to the optimal estimation of the internal snake parameters, with respect to all images, may benefit from methods that better utilize expert delineated results, such as Reinforcement Learning (RL) [Peng1998]. An alternative approach would be to design different snakes for different medical cases. In particular, we have found it more difficult to detect boundaries of reticular lichenoid reactions, since their shapes are usually more complex. Further potential investigations include the use of vector snakes [Sapiro1996], using parameters that change along the snake contour, and producing more elaborate quantitative results. Generally, we believe that to further automate and improve segmentation, additional or enhanced energy terms and more human knowledge should be incorporated into the snakes design.

Using the 1D Discrete Wavelet Transform (DWT) as opposed to the Discrete Cosine Transform (DCT) (in Chapter 3) for representing the coordinate functions $x(s)$ and $y(s)$ of the contour is worth investigating. Particularly since the DWT coefficients contain more localized information about the spatial behavior of the contour.

In Chapter 4, an additional optical flow-based force is introduced and utilized for tracking the leading edge of injected contrast agent in an echocardiographic image sequence. In our implementation, the optical flow field was calculated over the entire image space, we could reduce computational complexity and gain speed by calculating optical flow only where needed, i.e. around the snake nodes that define the contrast front. Other suggested future work can be found in [Althoff2001].

In our extension of ASM to 2D+time (Chapter 5) we performed elaborate testing using synthetic images. More work is needed to fully evaluate the method's applicability to real-life imagery. Usefulness of multi-resolution search was demonstrated for 2D ASM, we believe a similar extension to 2D+T

would be equally beneficial (possibly including multiple *temporal* resolutions). Extending the current work to include temporal translation and temporal scaling parameters may assist in searching through longer image sequences for target dynamic shapes of varying velocities. In the current segmentation implementation using dynamic programming, equal temporal discontinuity costs were assigned to all landmarks throughout all the frames. A possible improvement may be the utilization of prior information about the velocities of the landmarks in different regions of the object. Another possible direction for investigation is the use of different gray level information models, possibly similar to Active Appearance Models but extended to 2D+T, resulting in a mean 3D (2D+T) intensity template along with the main modes of variation.

Medial-based shape profiles (Chapter 6) provide us with intuitive, controlled, and independent 2D shape deformation controllers, nevertheless, a number of interesting issues remain to be tackled. Currently, the medial profiles are constructed only from the primary medial axis and secondary axes are not considered. This prevents the model from accurately representing highly asymmetrical (with respect to the primary axis) shapes. Thus, the extension of the approach to handle objects with multiple medial axes (i.e. objects with protrusions) is important. The boundary near the terminals/end caps of the model, where the left boundary meets the right boundary, requires special consideration to prevent loss of continuity. We currently perform cubic B-spline interpolation of the boundary nodes. Alternative boundary representation methods could also prove useful. Another enhancement may be the incorporation of boundary-based displacements to accommodate objects with irregular boundaries. Extending the developed 2D scheme that uses 1D medial-based shape profiles to 3D, amounts to using 2D shape profiles or 'medial patches' (replacing the 1D profiles) for describing 3D shape characteristics. Medial patches encompass thickness and two-dimensional elongation and orientation values with respect to the medial surface. A similar yet more involved scheme of medial-based shape patches, operators, and statistical analysis is already underway (see Figure 9.1). Pizer et al [Pizer2000, Pizer1998, Pizer1999] have done extensive work on medial-based representations; a thorough investigation of how to make full use of this considerable body of work is advantageous.

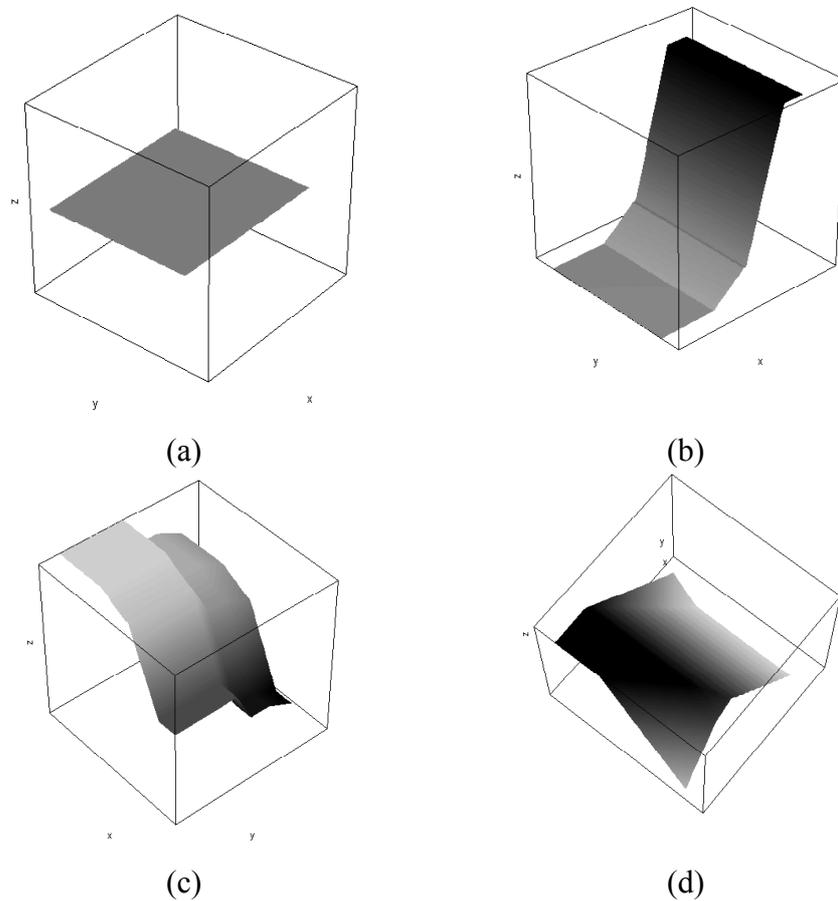


Figure 9.1. Preliminary investigation into ‘medial patches’. (a) A flat medial surface reconstructed from medial patches, (b) bending the surface along the x-axis, (c) multiple bends along the x and y-axis, (d) multiple bends along arbitrary directions.

Several interesting issues regarding the physics-based shape deformation technique (Chapter 7) exist and deserve further exploration. For example, the circular deformation region may be too restrictive for more complex-shaped mesh models. Similar methods for carrying out physics-based deformations to 3D meshes are also needed. We have noticed that sometimes the 2D mesh model may fold on itself or become unstable, particularly upon the application of extreme deformations. To remedy such possible behavior one can resort to the inclusion of additional springs (for example, sheer springs [Provot1995]), or gradually changing the rest lengths, or even the use of springs that have a rest orientation not only a rest length. Another interesting extension is a layered spring-mass system where different layers correspond to different scales or

resolution of deformation. Including topological changes in a spring-mass system is also desirable. For example, springs that come under too much tension break or snap causing a change in topology. Triangulation algorithms to automate the generation of corresponding meshes would also be very beneficial. While mass-spring models do offer some advantages, such as speed and reduced computational complexity, continuum models (such as Finite Element Methods [Fagan1992]) offer higher accuracy and are worth further investigation.

9.2 An ALife Paradigm for Medical Image Analysis

We believe that the new artificial life paradigm for medical image analysis that we have proposed in Chapter 8 opens the way for new and exciting avenues of future research. We summarize a number of interesting aspects of our deformable organisms that deserve further exploration, these include: Extending our model to 3D (which would involve generating 3D medial-based shape representation and deformation techniques, providing intuitive deformation handles capable of producing controlled deformations, as in the ‘medial patches’ discussed in the previous section). Designing a motion tracking plan and releasing an organism into 4D dynamic ‘environments’ (i.e. 4D images). Exploring the use of multiple plans and plan selection schemes for example the application of learning algorithms, such as Genetic Algorithms (GA), to assist human experts in the generation of optimal plans. Developing more sophisticated organism interactions; we only demonstrated simple organism interaction. The layered architecture also provides the option of adding higher levels of cognitive modeling, knowledge representation, reasoning, and planning. On a more specific note, we anticipate further developing the ‘artery crawler’ (Figure 8.13) allowing it to ‘crawl’ along arteries, looking for branches, stenoses and aneurysms, when released into the central artery of angiographic images.

9.3 Towards Intelligent Deformable Models for MIA

Human’s natural cognitive abilities are utilized to provide current deformable models with guidance towards proper medical image analysis results via, for example, model initialization and interaction. Intelligent Deformable Models (IDM), however, attempt to mimic such cognitive abilities by incorporating higher layers of (artificial) intelligence atop the geometry and physics modeling layers (Figure 9.2). IDM would not only provide powerful bottom-up, data-driven processes, but also equally powerful top-down, knowledge-driven processes within a robust decision-making framework that operates across multiple levels of abstraction. The subsequent sections describe some of the issues involved in developing IDM for segmentation and labeling of an anatomical structure (or a set of structures) from medical imagery.

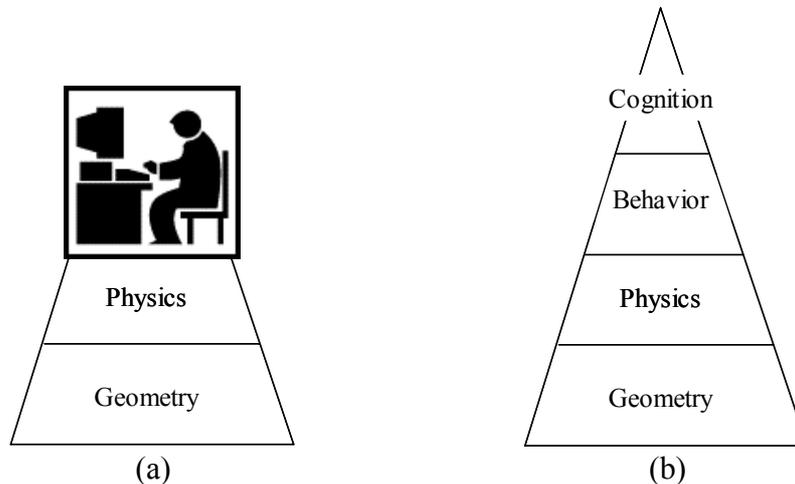


Figure 9.2. (a) Deformable models (incorporating geometry and physics layers) guided by an expert human operator. (b) Intelligent deformable models attempt to model the cognitive abilities of human operators (by including higher cognitive layers).

9.3.1 Knowledge and Knowledge Representation

An initial phase pertaining to the quest for knowledge about the MIA problem in hand may be fundamental. Such knowledge includes: knowledge about structures (physical properties, appearance, etc.), relations between structures (inter-object knowledge may include properties about correlations in deformations, variations in relative positions, physical connections, etc.), and other expert knowledge about the segmentation problem (which features are the most stable, where a tumor is likely to appear, etc.). We believe that the acquisition and appropriate representation of high-level knowledge, about anatomical organs and structures and their interactions, is of high importance for developing IDM for medical image analysis. Knowledge representations should be flexible and intuitive facilitating future additions of information of different natures. Standardized methods for describing the properties of the objects, and how they relate to their appearance in images could be necessary. Mechanisms for converting high-level knowledge (possibly from natural language) to a set of lower level rules or guidelines may also be needed.

9.3.2 Deformable ‘Anatomical’ Models

Another challenging and vital phase would involve creating models of organs and structures that embrace as much of the available relevant knowledge as possible. Once created, such models may define artificial anatomical organs. For example a complete artificial model of the anatomy (not physiology) of the

brain would be made up of different sub-structure models (lateral ventricles, corpora callosa, etc.). Properties of the real structure may need to be embedded in the model, such as geometrical and physical composition and properties, resulting in quantitative description of shape (along with shape variability and deformability), pose, texture, and properties relating to structure interaction. Additionally, landmark anatomical features should be built into these models to ensure proper matching upon segmentation (intrinsic point correspondence).

9.3.3 Use of Knowledge

The deformable organisms we created were responsible for a specific anatomical structure and the plans they followed to locate the structures were fixed. Genetic algorithms, reinforcement learning, or other learning methods may be used, which utilize prior anatomical knowledge and information about the deformation capabilities of the models, in order to find optimal segmentation schedules. Lowering the dimensionality of the model parameter search space can be obtained by providing sets of standardized, natural, and intuitive deformation and image-search behaviors (or subroutines). These behaviors would be described by few intuitive parameters, which in turn activate lower level physical and geometrical shape deformations. Image search may involve matching according to statistical/prior knowledge of shape variation, appearance (intensity-based features), geometrical features (e.g. points, curves such as crest lines, curvature, etc.), anatomical landmarks, and shape (or deformation) features (e.g. a bend at a certain scale in a certain location). The behavior of multiple interacting models may be achieved via modeling physical connections, sending and receiving force activation commands, statistical correlation, etc. A layered architecture provides the option of adding higher levels of cognitive modeling, reasoning, and planning.

Different scientific disciplines may be related to the creation and function of IDM: Geometry- and physics-based modeling may be used for shape representation and deformation. Image processing and scale space theory can be utilized through the IDM focus attention mechanism; the IDM filters out sensory information superfluous to its current behavioral needs by, for example, only considering the image data at a certain level in scale space. Image analysis and pattern recognition may be utilized during image search behaviors. For example, feature detection and classification techniques are used during IDM search for an anatomical landmark described by, say, high edge curvature and bright texture intensity. Optimization methods may be used, for example, for finding the best set of deformation parameters that fit a part of the model to image data. Machine learning may be utilized for constructing the IDM segmentation schedule. For example, human experts are first trained to manipulate the shape of the model via its ‘deformation handles’. The experts’ actions (deformation parameters) during manually segmenting a structure,

along with image features in the vicinity of the model, are input to segmentation schedule learning algorithms (via e.g. GA, RL). Artificial intelligence can be utilized at the IDM cognitive layers. Computer graphics techniques may be used for visualization and producing realistic anatomic structure behaviors. For example, deformable anatomical models (IDM with higher level cognitive layers turned off) are manipulated by hand for training on a medical procedure (surgical simulation).

From another viewpoint, the field of medical imaging is continuously under advancement and possibly new and improved imaging technologies and devices may be developed that use prior knowledge to generate images of (partially) segmented structures, by making use of the physical/chemical properties of the structures under investigation.

APPENDICES

Appendix A. Deformable Models

A.1. Energy-Minimizing Deformable Models - Snakes

Active Contour Models or snakes, the classical deformable model, gained large acceptance as a segmentation tool due to the way snakes consider the boundary as a single, inherently connected, and smooth structure. In short, ACM are energy minimizing parametric contours with smoothness constraints deformed according to image data. ACM are designed to be semi-automatic tools supporting intuitive interactive mechanisms for guiding the segmentation deformations. Some of the problems of the classical snakes are initialization sensitivity and lack of high level automatic control that cause the snakes, for example, to leak or latch to erroneous edges. Without manual intervention one runs the snake and ‘hopes for the best’. In this appendix, the mathematical formulation of snakes, dynamic deformable models, numerical simulation, and probabilistic deformable modes are treated.

In active contour models, a contour is initiated on the image and left to deform in a way that, firstly, moves it toward features of interest in the image and, secondly, maintains a certain degree of smoothness in the contour. In order to favor this type of contour deformation, an energy term is associated with the contour and designed to be inversely proportional to the contour’s smoothness and to the model fit to desired image features. The deformation of the contour in the image plane will change its energy, thus one can imagine an energy (potential) surface on top of which the contour moves (in a way that resembles the slithering of a snake and hence the name) seeking valleys of low energy.

A snake in the continuous spatial domain is represented as a 2D parametric contour curve $\mathbf{v}(s) = (x(s), y(s))$ where $s \in [0,1]$. In order to fit the snake model to the image data we associate energy terms with the snake and aim to deform the snake in a way that minimizes its total energy. The energy of the snake, ξ , depends on both the shape of the contour and the image data $I(x, y)$ reflected via the internal and external energy terms, $\alpha(\mathbf{v})$ and $\beta(\mathbf{v})$, respectively. The total snake energy is written as

$$\xi(\mathbf{v}) = \alpha(\mathbf{v}) + \beta(\mathbf{v}). \quad (\text{A.1})$$

The internal energy term is given as

$$\alpha(\mathbf{v}) = \int_0^1 w_1(s) \left| \frac{\partial \mathbf{v}}{\partial s} \right|^2 + w_2(s) \left| \frac{\partial^2 \mathbf{v}}{\partial s^2} \right|^2 ds. \quad (\text{A.2})$$

The weighting functions w_1 and w_2 control the tension and flexibility of the contour, respectively. The external energy term is given as

$$\beta(\mathbf{v}) = \int_0^1 P(\mathbf{v}(s)) ds. \quad (\text{A.3})$$

For the contour to be attracted to image features, the function $P(x, y)$ is designed such that it has minima where the features have maxima. For example, for the contour to be attracted to high intensity changes (high gradients) we can choose

$$P(x, y) = -c \|\nabla [G_\sigma * I(x, y)]\| \quad (\text{A.4})$$

where $G_\sigma * I$ denotes the image convolved with a smoothing (e.g. Gaussian) filter with a parameter σ controlling the extent of the smoothing (e.g. variance of Gaussian).

The contour $\mathbf{v}(s)$ that minimizes the energy ξ must, according to the calculus of variations¹, satisfy the vector-valued partial differential (Euler-Lagrange) equation

$$-\frac{\partial}{\partial s} \left(w_1 \frac{\partial \mathbf{v}}{\partial s} \right) + \frac{\partial^2}{\partial s^2} \left(w_2 \frac{\partial^2 \mathbf{v}}{\partial s^2} \right) + \nabla P(\mathbf{v}(s)) = \mathbf{0}. \quad (\text{A.5})$$

A.2. Dynamic Deformable Models

In order to attack the problem of tracking non-rigid time-varying objects, deformable models were extended to dynamic deformable models. These describe the shape changes (with time) in a single model that evolves through time to reach a state of equilibrium where internal forces representing constraints on shape smoothness balance the external image forces and the contour comes to rest. In this case the time-varying (dynamic) contour is written as $\mathbf{v}(s, t) = (x(s, t), y(s, t))$ where $s \in [0, 1]$ and the corresponding constraint equation becomes

$$\mu(s) \frac{\partial^2 \mathbf{v}}{\partial t^2} + \gamma(s) \frac{\partial \mathbf{v}}{\partial t} - \frac{\partial}{\partial s} \left(w_1 \frac{\partial \mathbf{v}}{\partial s} \right) + \frac{\partial^2}{\partial s^2} \left(w_2 \frac{\partial^2 \mathbf{v}}{\partial s^2} \right) + \nabla P(\mathbf{v}(s, t)) = \mathbf{0} \quad (\text{A.6})$$

where $\mu(s)$ and $\gamma(s)$ are mass and damping densities, respectively.

¹ Calculus of variations is a branch of mathematics concerned with the problem of finding a function for which the value of a certain integral is either the largest or the smallest possible. Many problems of this kind are easy to state, but their solutions commonly involve difficult procedures related to differential calculus and differential equations [EB].

A.3. Discretization and Numerical Simulation

Polygonal snake. For a polygonal snake contour the discrete version of (A.6) can be written as

$$\mu_i \ddot{\mathbf{v}}_i(t) + \gamma_i \dot{\mathbf{v}}_i(t) - w_1 \mathbf{v}''(t) + w_2 \mathbf{v}''''(t) + \nabla P(\mathbf{v}_i(t)) = \mathbf{0} \quad (\text{A.7})$$

where $\{\mathbf{v}_i(t) = (x_i(t), y_i(t))\}_{i=1,2,\dots,N}$ are the nodes of the snake polygon.

t is used here as the discrete time variable and i is the snake node index. $\dot{\mathbf{v}}$ and $\ddot{\mathbf{v}}$ are the first and second derivatives of \mathbf{v} with respect to t . \mathbf{v}'' and \mathbf{v}'''' are second and fourth derivatives of \mathbf{v} with respect to i . Setting the mass density to zero² ($\mu_i = \mu = 0$) and the damping density to a constant ($\gamma_i = \gamma$), we rewrite equation (A.7) for simulating the deformations of the discrete snake as

$$\gamma \dot{\mathbf{v}}_i + \alpha \mathbf{F}_i^{\text{tensile}}(t) + \beta \mathbf{F}_i^{\text{flexural}}(t) = \mathbf{F}_i^{\text{external}}(t) \quad (\text{A.8})$$

where α and β are weighting factors. $\mathbf{F}_i^{\text{tensile}}(t)$ is a tensile force (resisting stretching) acting on node i at time t and is given by

$$\mathbf{F}_i^{\text{tensile}}(t) = 2\mathbf{v}_i(t) - \mathbf{v}_{i-1}(t) - \mathbf{v}_{i+1}(t) \quad (\text{A.9})$$

$\mathbf{F}_i^{\text{flexural}}(t)$ is a flexural force (resisting bending) and is given by

$$\mathbf{F}_i^{\text{flexural}}(t) = 2\mathbf{F}_i^{\text{tensile}}(t) - \mathbf{F}_{i-1}^{\text{tensile}}(t) - \mathbf{F}_{i+1}^{\text{tensile}}(t) \quad (\text{A.10})$$

$\mathbf{F}_i^{\text{external}}(t)$ is an external (image-derived) force. It is derived in a way that causes the snake node to move towards regions of higher intensity gradient in the image and is given by

$$\mathbf{F}_i^{\text{external}}(t) = \nabla P(x_i(t), y_i(t)) \quad (\text{A.11})$$

where $P(x_i(t), y_i(t))$ is given in (A.4).

In addition to the above forces, an inflation force, $\mathbf{F}_i^{\text{inflation}}(t)$, is commonly utilized allowing, in many cases, the initialization of the snake farther away from the target boundary. $\mathbf{F}_i^{\text{inflation}}(t)$ is given by

$$\mathbf{F}_i^{\text{inflation}}(t) = F(I_s(x_i, y_i)) \mathbf{n}_i(t) \quad (\text{A.12})$$

where $\mathbf{n}_i(t)$ is the unit vector in the direction normal to the contour at node i and the binary function

$$F(I(x, y)) = \begin{cases} +1 & \text{if } I(x, y) \geq T \\ -1 & \text{otherwise} \end{cases} \quad (\text{A.13})$$

links the inflation force to the image data, and T is an image intensity threshold. Consequently equation (A.8) becomes

² In static shape recovery problems not involving time-varying data, the mass density is often set to zero, resulting in simplified equations of motion and a snake that comes to rest as soon as the internal forces balance the external forces [McInerney2000].

$$\gamma \dot{\mathbf{v}}_i + \alpha \mathbf{F}_i^{tensile}(t) + \beta \mathbf{F}_i^{flexural}(t) = \mathbf{F}_i^{external}(t) + \mathbf{F}_i^{inflation}(t). \quad (\text{A.14})$$

The equation used for updating the position of any snake node i can be obtained from (A.14) by using a finite difference derivative approximation $\dot{\mathbf{v}}_i = (\mathbf{v}_i(t + \Delta t) - \mathbf{v}_i(t))/\Delta t$, where Δt is a finite time step, yielding

$$\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) - \frac{\Delta t}{\gamma} (\alpha \mathbf{F}_i^{tensile}(t) + \beta \mathbf{F}_i^{flexural}(t) - \mathbf{F}_i^{external}(t) - \mathbf{F}_i^{inflation}(t)). \quad (\text{A.15})$$

Moreover, certain forces can be designed (or derived from energy terms) in a way that the resulting contour deformations will reduce its energy, thus yielding a smooth contour located along desired nearby image features such as edges. For example, another way to derive the tensile snake force above is by first defining an energy term that increases with increased stretching and then seeking the force that causes the snake node to change its position to a location that minimizes this energy. For example, the tensile energy is chosen to be

$$E^{tensile}(t) = \sum_{n=1}^N |\mathbf{v}_n(t) - \mathbf{v}_{n-1}(t)|^2 = \sum_{n=1}^N (x_n(t) - x_{n-1}(t))^2 + (y_n(t) - y_{n-1}(t))^2 \quad (\text{A.16})$$

and the tensile force $\mathbf{F}_i^{tensile}(t)$ acting on $\mathbf{v}_i(t-1)$ according to (A.15) and updating its position to $\mathbf{v}_i(t)$ is defined as in

$$\mathbf{v}_i(t) = \mathbf{v}_i(t-1) + k \mathbf{F}_i^{tensile}(t) \quad (\text{A.17})$$

where k is a constant. To find $\mathbf{F}_i^{tensile}(t)$, we find $\mathbf{v}_i(t) = (x_i(t), y_i(t))$ that minimizes $E^{tensile}(t)$. To find $x_i(t)$ we write $\partial E^{tensile}(t)/\partial x_i(t) = 0$, which gives

$$\frac{\partial \sum_{n=i}^{i+1} (x_n(t) - x_{n-1}(t))^2}{\partial x_i(t)} = \frac{\partial (x_i(t) - x_{i-1}(t))^2 + (x_{i+1}(t) - x_i(t))^2}{\partial x_i(t)} = 0, \quad (\text{A.18})$$

which can be rewritten as

$$\frac{\partial (x_i^2(t) - 2x_i(t)x_{i-1}(t) + x_{i-1}^2(t) + x_{i+1}^2(t) - 2x_{i+1}(t)x_i(t) + x_i^2(t))}{\partial x_i} = 0 \quad (\text{A.19})$$

yielding $4x_i(t) - 2x_{i-1}(t) - 2x_{i+1}(t) = 0$ and hence $x_i(t) = (x_{i-1}(t) + x_{i+1}(t))/2$ and similarly for $y_i(t)$ we get $y_i(t) = (y_{i-1}(t) + y_{i+1}(t))/2$. This finally gives

$$\mathbf{v}_i(t) = (\mathbf{v}_{i-1}(t) + \mathbf{v}_{i+1}(t))/2. \quad (\text{A.20})$$

Substituting (A.20) into (A.17) and solving for $\mathbf{F}_i^{tensile}(t)$ gives

$$\mathbf{F}_i^{tensile}(t) \propto 2\mathbf{v}_i(t) - \mathbf{v}_{i-1}(t) - \mathbf{v}_{i+1}(t). \quad (\text{A.21})$$

In a similar manner other energy terms can be defined and forces derived (or designed).

Shape parameters. A vector of shape parameters \mathbf{u} can also be used to represent the continuous geometric contour model $\mathbf{v}(s)$. These shape parameters are generally associated with some local-support basis functions (such as splines and finite elements) or global-support basis functions (such as Fourier bases). In this case the discrete form of $\xi(\mathbf{v})$ may be written as

$$E(\mathbf{u}) = \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} + P(\mathbf{u}) \quad (\text{A.22})$$

where \mathbf{K} is called the stiffness matrix and $P(\mathbf{u})$ is the discrete version of the external potential $P(\mathbf{v}(s))$. The contour parameters that minimize the energy function can now be obtained by solving the set of algebraic equations

$$\mathbf{K} \mathbf{u} = -\nabla P. \quad (\text{A.23})$$

The motion equation for the contour (represented by \mathbf{u}) can be written as

$$\mathbf{M} \ddot{\mathbf{u}} + \mathbf{N} \dot{\mathbf{u}} + \mathbf{K} \mathbf{u} = -\nabla P \quad (\text{A.24})$$

where \mathbf{M} and \mathbf{N} are mass and damping matrices, respectively.

A.4. Probabilistic Deformable Models

A probabilistic formulation is an alternative view of deformable models. The deformable model can be fitted to the image data by finding the model shape parameters \mathbf{u} that maximize the posterior probability (maximum a posteriori or MAP solution) expressed using Bayes' theorem as

$$p(\mathbf{u} | I) = \frac{p(I | \mathbf{u})p(\mathbf{u})}{p(I)}. \quad (\text{A.25})$$

where $p(\mathbf{u})$ is the prior probability density of the model shape parameters: a mechanism for probabilistic regularization. $p(I | \mathbf{u})$ is the probability of producing an image I given the parameters \mathbf{u} : an imaging (sensor) model. $p(\mathbf{u})$ and $p(I | \mathbf{u})$ can be written (in the form of Gibbs distribution) as

$$p(\mathbf{u}) = k_1 \exp(-A(\mathbf{u})) \quad (\text{A.26})$$

$$p(I | \mathbf{u}) = k_2 \exp(-P(\mathbf{u})) \quad (\text{A.27})$$

where k_1 and k_2 are normalizing constants and $A(\mathbf{u})$ is the discrete version of the internal energy $\alpha(\mathbf{v})$ and $P(\mathbf{u})$ is the discrete version of the external potential $P(\mathbf{v}(s))$.

Snakes segmentation examples (using adaptive inflation reversal and damping, see Section 2.2.1) are shown in Figure A.1-4.

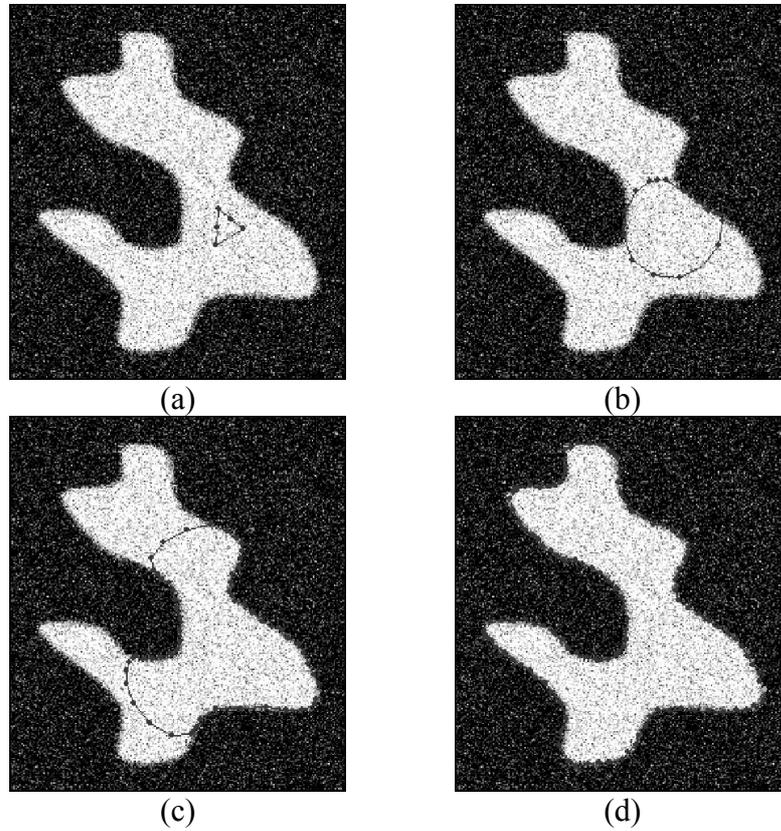


Figure A.1. Snake segmentation progress from (a) to (d), of a synthetic image.

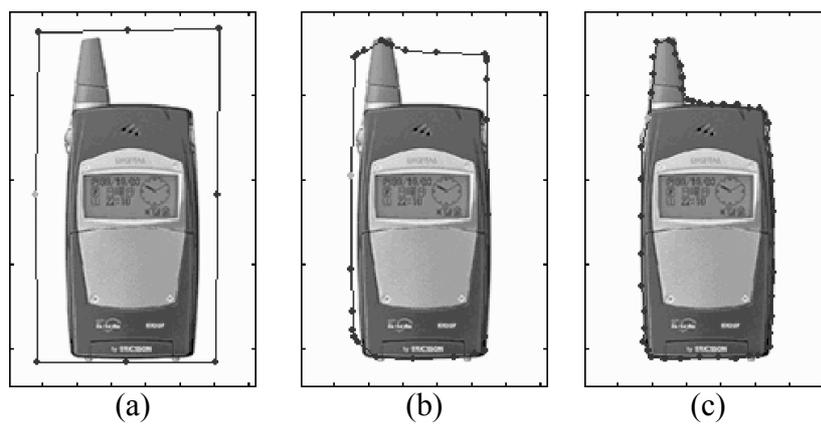


Figure A.2. Snake segmentation progress from (a) to (c), of a man made object.

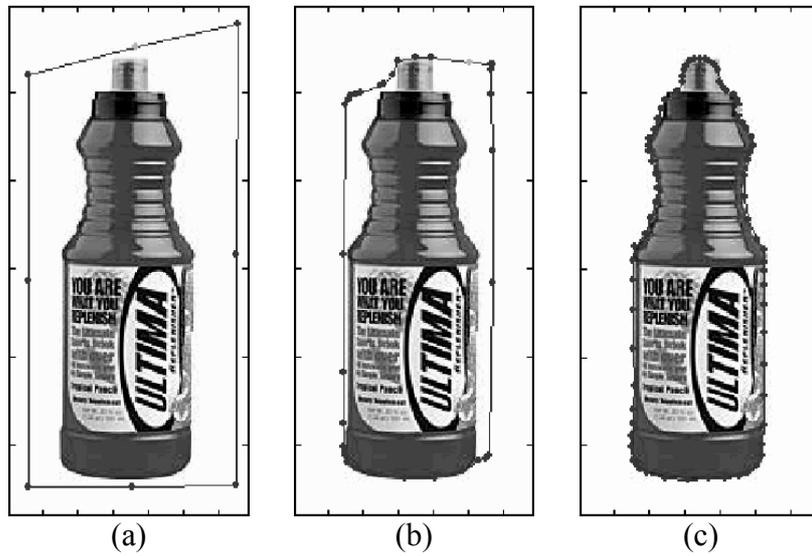


Figure A.3. Snake segmentation progress from (a) to (c), of a man made object.

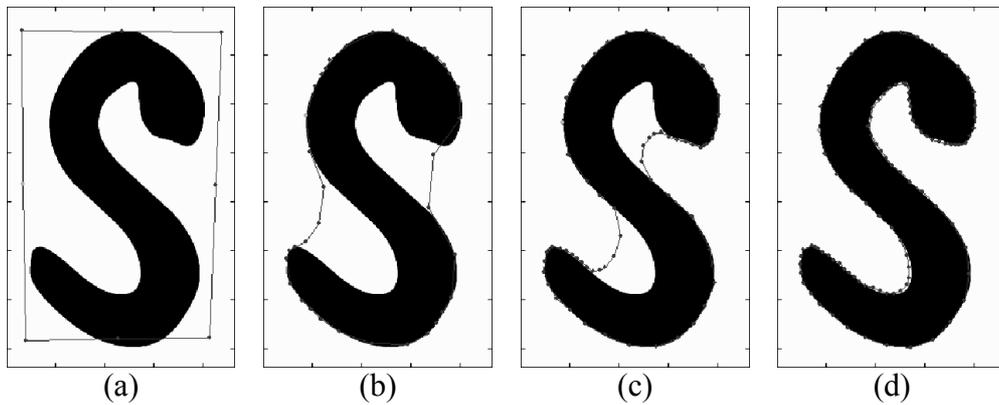


Figure A.4. Snake segmentation progress from (a) to (d) of a synthetic image.

Appendix B. Active Shape Models

This appendix presents an overview of the steps involved in generating a Point Distribution Model (PDM) and the use of Active Shape Models (ASM) for image search.

A representative training set of images that convey the shape variations we wish to study is collected. A number of landmarks, say L , are chosen to describe the training shapes [Cootes1992]. The driving principals governing the behavior of the variations of the training shapes can be found using Principal Component Analysis (Appendix C). An object shape is then represented by the sum of a mean shape and a linear combination of, say t , principal components, i.e.,

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b} \quad (\text{B.1})$$

where \mathbf{x} is the vector of landmark coordinates, $\bar{\mathbf{x}}$ is the mean shape, \mathbf{P} is the matrix of principal components, \mathbf{b} is a vector of weighting parameters, also called shape parameters. \mathbf{x} and $\bar{\mathbf{x}}$ are each of length $2L$. \mathbf{P} is $2L \times t$ and \mathbf{b} is a vector of length t .

Constraints are put on these weighting parameters to insure that only allowable shapes are produced, belonging to the Allowable Shape Domain (ASD). This model of the distribution of the landmarks points (or variation of the shapes) is referred to as the Point Distribution Model (PDM).

PDM are an essential component of Active Shape Models (ASM)[Cootes1995a], an image segmentation technique. In ASM, it is required to find the proposed movement of the landmarks of a current shape estimate to new and better locations. The gray level information (image data) is essential to finding such suggested movements. This implies that we need to model the gray level information and make such a model available during image search. This may be done by examining the intensity profiles at each landmark and normal to the boundary created by the landmark and its neighbors. Then the intensity profiles are used to derive a normalized intensity difference (gradient, or derivative) profile giving invariance to the offsets and uniform scaling of the gray levels [Cootes1993]. With L landmarks representing each shape, N training shapes, and N training images, we derive N profiles for each landmark, one from each image, and calculate the mean profile for each landmark using

$$\bar{\mathbf{y}}_j = \frac{1}{N} \sum_{i=1}^N \mathbf{y}_{ij} \quad (\text{B.2})$$

where \mathbf{y}_{ij} is the normalized derivative profile for the j^{th} landmark in the i^{th} image and $\bar{\mathbf{y}}_j$ is the mean normalized derivative profile for the j^{th} landmark.

Given a new image, the basic idea is to start with an initial estimate, then examine the neighborhood of the landmarks aiming at finding better locations for the landmarks. The shape and the pose of the current estimate are hence changed to better fit the new locations of the landmarks while producing in the process a new acceptable or allowable shape.

The pose parameters are first found by aligning the current estimate to the new proposed shape (as explained in Appendix D). The remaining landmark position modifications generally span $2L$ -dimensions, whereas the shape variations obtained from the model are only t -dimensional. A least-squares solution can be used to solve the following equation for the changes in shape parameters $d\mathbf{b}$ (with orthonormal column of \mathbf{P} we have $\mathbf{P}^T\mathbf{P} = \mathbf{I}$)

$$d\mathbf{x} = \mathbf{P} d\mathbf{b} \Rightarrow d\mathbf{b} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T d\mathbf{x} = \mathbf{P}^T d\mathbf{x} \quad (\text{B.3})$$

where $d\mathbf{x}$ is a vector containing the remaining landmark position modifications, $d\mathbf{b}$ is a vector of changes in the shape parameters, and \mathbf{P} is the matrix of principal components.

Finally, the shape variations are limited to obtain an acceptable or allowable shape within the ASD by applying the constraints on the shape parameters. With this step we complete a single iteration. In a similar fashion, we obtain new estimates and re-iterate until we assume convergence (when the parameter changes are insignificant).

Appendix C. Principal Component Analysis

Given a set of multivariate observations, applying Principal Component Analysis (PCA) generates a new set of variables called the principal components. Each principal component is a linear combination of the original variables (referred to as a Standardized Linear Combination). All the principal components are orthogonal to each other so there is no redundant information. The principal components as a whole form an orthogonal basis for the space of data. The first principal component is a single axis in space. When projecting each of the observations on this axis, the resulting values form a new variable. The variance of this variable is the maximum along all possible choices of this axis. The second component is another axis in space, perpendicular to the first. Projecting the observations on this axis generates another new variable. The variance of this variable is the maximum among all possible choices of this second axis. The dimension of both the full set of principal components and the original set of variables is the same.

In many applications it can be assumed that the first few principal components describe a high percentage of the total variance of the original data. Hence, the dimension of the model can be reduced and the variations described by a smaller number of variables; that is performing what is referred to as “parsimonious summarization”. For more on PCA the reader is referred to [Mardia1995, Bradley1997, Jackson1991].

Given an observation matrix $\mathbf{X} = [\mathbf{x}_1 \ \mathbf{x}_2 \ \dots \ \mathbf{x}_m]$ of size $n \ m$, where m is the number of observations and n is the number of variables in each observation, *i.e.* each row of the observation matrix is a variable and each column is an observation. The principal components are the eigenvectors of the covariance matrix of the observations. The covariance matrix is given by

$$\mathbf{S} = \frac{1}{m-1} \sum_{i=1}^m (\mathbf{x}_i - \bar{\mathbf{x}})^T (\mathbf{x}_i - \bar{\mathbf{x}}) \quad (\text{C.1})$$

where

$$\bar{\mathbf{x}} = \frac{1}{m} \sum_{i=1}^m \mathbf{x}_i. \quad (\text{C.2})$$

The unit eigenvectors \mathbf{p}_k , $1 \leq k \leq n$, of \mathbf{S} are such that

$$\mathbf{S}\mathbf{p}_k = \lambda_k \mathbf{p}_k \quad (\text{C.3})$$

where λ_k is the k^{th} eigenvalue of \mathbf{S} , $\lambda_k \geq \lambda_{k+1}$, and $\mathbf{p}_k^T \mathbf{p}_k = 1$.

Assuming that the observations form a hyper-ellipsoid in n dimensions, then the eigenvectors of the covariance matrix corresponding to the largest eigenvalues describe the longest axes of the ellipsoid, and thus the most significant modes of variations in the variables used to derive the covariance

matrix. The variance explained by each eigenvector is equal to the corresponding eigenvalue. We can approximate the n dimensional ellipsoid by a t dimensional ellipsoid, and thus take the first t eigenvectors as the first t principal components.

Another method to obtain the principal components is by performing a Singular Value Decomposition (SVD) of the observation matrix centered around its mean [Jackson1991]. First, we center the observation matrix \mathbf{X} around its mean and obtain

$$\mathbf{X}_c = \mathbf{X} - \bar{\mathbf{x}} \cdot \mathbf{1}^T \quad (\text{C.4})$$

where $\mathbf{1}^T = [1 \ 1 \ \dots \ 1]$ and is of length m . Then we perform an ‘economy size’ SVD of $\frac{1}{\sqrt{m-1}} \mathbf{X}_c$ and obtain $\mathbf{U}, \mathbf{S}, \mathbf{V}$

$$[\mathbf{U}, \mathbf{S}, \mathbf{V}] = \text{svd}\left(\frac{1}{\sqrt{m-1}} \mathbf{X}_c, 0\right) \quad (\text{C.5})$$

such that

$$\mathbf{X}_c = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T \quad (\text{C.6})$$

where the principal components are the columns of the \mathbf{V} matrix, and the eigenvalues of the covariance are contained in \mathbf{S} .

Appendix D. Aligning 2D Shapes

Given two shapes, \mathbf{x}_1 and $\hat{\mathbf{x}}_1$, represented by landmarks, i.e.

$$\mathbf{x}_1 = [x_{11} \ y_{11} \ x_{12} \ y_{12} \ \dots \ x_{1L} \ y_{1L}]^T \text{ and}$$

$$\mathbf{x}_2 = [x_{21} \ y_{21} \ x_{22} \ y_{22} \ \dots \ x_{2L} \ y_{2L}]^T$$

we want to find the rotation angle, θ , the scaling factor, s , and the value of the translation in both directions, (t_x, t_y) , that will align \mathbf{x}_2 to \mathbf{x}_1 . This is done

by mapping \mathbf{x}_2 to $\hat{\mathbf{x}}_2 = [\hat{x}_{21} \ \hat{y}_{21} \ \hat{x}_{22} \ \hat{y}_{22} \ \dots \ \hat{x}_{2L} \ \hat{y}_{2L}]^T$ using

$$\hat{\mathbf{x}}_2 = M(s, \theta)[\mathbf{x}_2] + \mathbf{t} \quad (\text{D.1})$$

where $M(s, \theta)[\mathbf{x}_2]$ is a rotated and scaled version of \mathbf{x}_2 (by θ and s respectively) and $\mathbf{t} = [t_x \ t_y \ t_x \ t_y \ \dots \ t_x \ t_y]^T$ is the translation vector of length $2L$. The weighted distance between the two shapes \mathbf{x}_1 and $\hat{\mathbf{x}}_2$, is given by:

$$d_{1\hat{2}}^2 = (\hat{\mathbf{x}}_2 - \mathbf{x}_1)^T \mathbf{W}^T \mathbf{W} (\hat{\mathbf{x}}_2 - \mathbf{x}_1) \quad (\text{D.2})$$

where \mathbf{W} is an appropriate weighting matrix. Substituting (D.1) in (D.2) we have

$$d_{1\hat{2}}^2 = (M(s, \theta)[\mathbf{x}_2] + \mathbf{t} - \mathbf{x}_1)^T \mathbf{W}^T \mathbf{W} (M(s, \theta)[\mathbf{x}_2] + \mathbf{t} - \mathbf{x}_1). \quad (\text{D.3})$$

For the rotation, scaling and translation of a single coordinate (x_{2k}, y_{2k}) we have

$$\begin{bmatrix} \hat{x}_{2k} \\ \hat{y}_{2k} \end{bmatrix} = M(s, \theta) \begin{bmatrix} x_{2k} \\ y_{2k} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} s \cos(\theta) & -s \sin(\theta) \\ s \sin(\theta) & s \cos(\theta) \end{bmatrix} \cdot \begin{bmatrix} x_{2k} \\ y_{2k} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \quad (\text{D.4})$$

By denoting

$$\left. \begin{aligned} a_x &= s \cos(\theta) \\ a_y &= s \sin(\theta) \end{aligned} \right\} \quad (\text{D.5})$$

we have

$$\begin{bmatrix} \hat{x}_{2k} \\ \hat{y}_{2k} \end{bmatrix} = \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \cdot \begin{bmatrix} x_{2k} \\ y_{2k} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \quad (\text{D.6})$$

Substituting (D.6) for all coordinates of $\hat{\mathbf{x}}_2$ in (D.1) we obtain

$$\hat{\mathbf{x}}_2 = \begin{bmatrix} \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \cdot \begin{bmatrix} x_{21} \\ y_{21} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \cdot \begin{bmatrix} x_{22} \\ y_{22} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \vdots \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \cdot \begin{bmatrix} x_{2L} \\ y_{2L} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \end{bmatrix} = \begin{bmatrix} a_x x_{21} - a_y y_{21} + t_x \\ a_y x_{21} + a_x y_{21} + t_y \\ a_x x_{22} - a_y y_{22} + t_x \\ a_y x_{22} + a_x y_{22} + t_y \\ \vdots \\ a_x x_{2L} - a_y y_{2L} + t_x \\ a_y x_{2L} + a_x y_{2L} + t_y \end{bmatrix} = \begin{bmatrix} x_{21} & -y_{21} & 1 & 0 \\ y_{21} & x_{21} & 0 & 1 \\ x_{22} & -y_{22} & 1 & 0 \\ y_{22} & x_{22} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{2L} & -y_{2L} & 1 & 0 \\ y_{2L} & x_{2L} & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} a_x \\ a_y \\ t_x \\ t_y \end{bmatrix}, \quad (\text{D.7})$$

which can be put in the form

$$\hat{\mathbf{x}}_2 = \mathbf{A}\mathbf{z} \quad (\text{D.8})$$

where

$$\mathbf{A} = \begin{bmatrix} x_{21} & -y_{21} & 1 & 0 \\ y_{21} & x_{21} & 0 & 1 \\ x_{22} & -y_{22} & 1 & 0 \\ y_{22} & x_{22} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{2L} & -y_{2L} & 1 & 0 \\ y_{2L} & x_{2L} & 0 & 1 \end{bmatrix} \quad (\text{D.9})$$

and

$$\mathbf{z} = [a_x \quad a_y \quad t_x \quad t_y]^T. \quad (\text{D.10})$$

Substituting (D.8) in (D.2) we get

$$d_{12}^2 = (\mathbf{A}\mathbf{z} - \mathbf{x}_1)^T \mathbf{W}^T \mathbf{W} (\mathbf{A}\mathbf{z} - \mathbf{x}_1). \quad (\text{D.11})$$

Solving for \mathbf{z} that minimizes d_{12}^2 , which is the least squares solution to

$\mathbf{W}\mathbf{A}\mathbf{z} = \mathbf{W}\mathbf{x}_1$, [Strang1988], we get

$$\mathbf{z} = ((\mathbf{W}\mathbf{A})^T (\mathbf{W}\mathbf{A}))^{-1} (\mathbf{W}\mathbf{A})^T \mathbf{W}\mathbf{x}_1 = (\mathbf{A}^T \mathbf{W}^T \mathbf{W} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{W}^T \mathbf{W}\mathbf{x}_1. \quad (\text{D.12})$$

Once \mathbf{z} is known, s and θ can be found since (refer to (D.5) and (D.10))

$$\frac{a_y}{a_x} = \frac{s \sin(\theta)}{s \cos(\theta)} = \tan(\theta).$$

This gives

$$\theta = \arctan\left(\frac{a_y}{a_x}\right) \quad (\text{D.13})$$

and with $a_x = s \cos(\theta)$ we have

$$s = \frac{a_x}{\cos(\arctan(\frac{a_y}{a_x}))}. \quad (\text{D.14})$$

In [Cootes1995a] a different route is followed for aligning two shapes \mathbf{x}_1 and \mathbf{x}_2 . Here, (D.3) is differentiated with respect to the variables a_x , a_y , t_x , and t_y and then arriving at the following set of four linear equations, which can be solved for the same variables using standard matrix methods,

$$\begin{bmatrix} X_2 & -Y_2 & W & 0 \\ Y_2 & X_2 & 0 & W \\ Z & 0 & X_2 & Y_2 \\ 0 & Z & -Y_2 & X_2 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} X_1 \\ Y_1 \\ C_1 \\ C_2 \end{bmatrix} \quad (\text{D.15})$$

where

$$X_i = \sum_{k=1}^L w_k x_{ik} \quad (\text{D.16})$$

$$Y_i = \sum_{k=1}^L w_k y_{ik} \quad (\text{D.17})$$

$$Z = \sum_{k=1}^L w_k (x_{2k}^2 + y_{2k}^2) \quad (\text{D.18})$$

$$W = \sum_{k=1}^L w_k \quad (\text{D.19})$$

$$C_1 = \sum_{k=1}^L w_k (x_{1k}x_{2k} + y_{1k}y_{2k}) \quad (\text{D.20})$$

$$C_2 = \sum_{k=1}^L w_k (y_{1k}x_{2k} - x_{1k}y_{2k}) \quad (\text{D.21})$$

and

$$\mathbf{W} = \text{diag}(w_1 \quad w_1 \quad w_2 \quad w_2 \quad \dots \quad w_L \quad w_L). \quad (\text{D.22})$$

Another approach [Cootes1997] for aligning two shapes, \mathbf{x}_1 and \mathbf{x}_2 , assuming they are centered at the origin (i.e. $\mathbf{1}^T \mathbf{x}_1 = \mathbf{1}^T \mathbf{x}_2 = 0$) is by finding only the scaling factor and the rotation angle to minimize $|M(s, \theta)[\mathbf{x}_2] - \mathbf{x}_1|$. This yields

$$s^2 = a^2 + b^2 \quad (\text{D.23})$$

and

$$\theta = \arctan(b/a) \quad (\text{D.24})$$

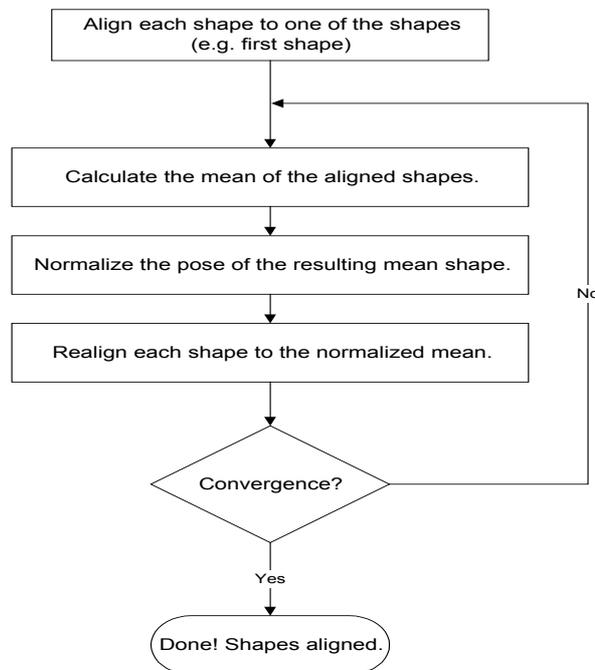
where

$$a = \frac{(\mathbf{x}_1^T \mathbf{x}_2)}{|\mathbf{x}_2|^2} \quad (\text{D.25})$$

and

$$b = \frac{\sum_{k=1}^L (x_{2k}y_{1k} - y_{2k}x_{1k})}{|\mathbf{x}_2|^2}. \quad (\text{D.26})$$

The previous discussion dealt only with the problem of aligning one shape to another. However, the training stage of ASM requires aligning all the set of shape examples. The following algorithm is used for aligning the set of N shapes to each other [Cootes1995a]:



The pose of a shape is described by its scaling, rotation, and translation, with respect to a known reference. Normalization of the pose implies (a) *scaling* of the shape so that the distance between two points becomes a certain constant, (b) *rotating* the shape so that the line joining two pre-specified landmarks is directed in a certain direction, and (c) *translating* the shape so that it becomes centered at a certain coordinate. Normalization is carried out in order to force the process to converge. Otherwise the mean shape may translate or expand (or shrink) indefinitely. Convergence is established if the shapes are not changing more than a pre-defined threshold.

Appendix E. Aligning Spatio-Temporal Shapes

Aligning ST-shapes amounts to rotating, scaling and translating the shape in each frame of the ST-shape by an amount that is fixed within one ST-shape. Scaling of time could also be performed, but here we assume that the frame-rate and the total sequence acquisition time is the same for different ST-shapes, thus there is no need for aligning in the temporal dimension. We first solve the problem of aligning only two ST-shapes then we will explain how to align all the ST-shapes.

Given two vectors, S_1 and S_2 , representing two ST-shapes we would like to find the rotation angle: θ , the scaling factor: s , and the value of the translation (t_x, t_y) that will align S_2 to S_1 . The two ST-shapes are written as (from Section 5.2.1). This gives

$$S_1 = [x_{111}, y_{111}, \dots, x_{11L}, y_{11L}, x_{121}, y_{121}, \dots, x_{12L}, y_{12L}, \dots, x_{1F1}, y_{1F1}, \dots, x_{1FL}, y_{1FL}] \quad (\text{E.1})$$

$$S_2 = [x_{211}, y_{211}, \dots, x_{21L}, y_{21L}, x_{221}, y_{221}, \dots, x_{22L}, y_{22L}, \dots, x_{2F1}, y_{2F1}, \dots, x_{2FL}, y_{2FL}] \quad (\text{E.2})$$

where each vector contains the (x, y) coordinates of L landmarks in F frames. For aligning S_2 to S_1 , S_2 is mapped to

$$S_2 = [\hat{x}_{211}, \hat{y}_{211}, \dots, \hat{x}_{21L}, \hat{y}_{21L}, \hat{x}_{221}, \hat{y}_{221}, \dots, \hat{x}_{22L}, \hat{y}_{22L}, \dots, \hat{x}_{2F1}, \hat{y}_{2F1}, \dots, \hat{x}_{2FL}, \hat{y}_{2FL}] \quad (\text{E.3})$$

Using

$$\hat{S}_2 = M(s, \theta)[S_2] + \mathbf{t} \quad (\text{E.4})$$

where

$M(s, \theta)[S_2]$ is a rotated then scaled version of each coordinate of S_2 (by θ and s respectively) and $\mathbf{t} = [t_x \ t_y \ t_x \ t_y \ \dots \ t_x \ t_y]^T$ is the translation vector and is of length $2FL$. The weighted distance between the two ST-shapes S_1 and \hat{S}_2 in the $2FL$ (x and y coordinates of L landmarks in F frames) dimensional space is given by

$$d_{1\hat{2}}^2 = (\hat{S}_2 - S_1)^T \mathbf{W}^T \mathbf{W} (\hat{S}_2 - S_1) \quad (\text{E.5})$$

where

$$\mathbf{W} = \text{diag}(w_{11x}, w_{11y}, \dots, w_{1Lx}, w_{1Ly}, w_{21x}, w_{21y}, \dots, w_{2Lx}, w_{2Ly}, \dots, w_{F1x}, w_{F1y}, \dots, w_{FLx}, w_{FLy}) \quad (\text{E.6})$$

The elements of \mathbf{W} can be chosen to reflect our trust in each coordinate. The identity matrix can be used for equal weighting of all coordinates. Substituting (E.4) in (E.5) we have

$$d_{1\hat{2}}^2 = (M(s, \theta)[S_2] + \mathbf{t} - S_1)^T \mathbf{W}^T \mathbf{W} (M(s, \theta)[S_2] + \mathbf{t} - S_1). \quad (\text{E.7})$$

For the rotation (around the origin), scaling and translation of a single coordinate (x_{2kl}, y_{2kl}) we have

$$\begin{bmatrix} \hat{x}_{2kl} \\ \hat{y}_{2kl} \end{bmatrix} = M(s, \theta) \begin{bmatrix} x_{2kl} \\ y_{2kl} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \begin{bmatrix} s \cos(\theta) & -s \sin(\theta) \\ s \sin(\theta) & s \cos(\theta) \end{bmatrix} \begin{bmatrix} x_{2kl} \\ y_{2kl} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (\text{E.8})$$

and denoting

$$\left. \begin{aligned} a_x &= s \cos(\theta) \\ a_y &= s \sin(\theta) \end{aligned} \right\} \quad (\text{E.9})$$

we have

$$\begin{bmatrix} \hat{x}_{2kl} \\ \hat{y}_{2kl} \end{bmatrix} = \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{2kl} \\ y_{2kl} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}. \quad (\text{E.10})$$

Now, we can rewrite \hat{S}_2 as

$$\hat{S}_2 = \begin{bmatrix} \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{211} \\ y_{211} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \vdots \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{21L} \\ y_{21L} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{221} \\ y_{221} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \vdots \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{22L} \\ y_{22L} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \vdots \\ \vdots \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{2F1} \\ y_{2F1} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \\ \vdots \\ \begin{bmatrix} a_x & -a_y \\ a_y & a_x \end{bmatrix} \begin{bmatrix} x_{2FL} \\ y_{2FL} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \end{bmatrix} = \begin{bmatrix} a_x x_{211} - a_y y_{211} + t_x \\ a_y x_{211} + a_x y_{211} + t_y \\ \vdots \\ a_x x_{21L} - a_y y_{21L} + t_x \\ a_y x_{21L} + a_x y_{21L} + t_y \\ \vdots \\ a_x x_{221} - a_y y_{221} + t_x \\ a_y x_{221} + a_x y_{221} + t_y \\ \vdots \\ a_x x_{22L} - a_y y_{22L} + t_x \\ a_y x_{22L} + a_x y_{22L} + t_y \\ \vdots \\ \vdots \\ a_x x_{2F1} - a_y y_{2F1} + t_x \\ a_y x_{2F1} + a_x y_{2F1} + t_y \\ \vdots \\ a_x x_{2FL} - a_y y_{2FL} + t_x \\ a_y x_{2FL} + a_x y_{2FL} + t_y \end{bmatrix} = \begin{bmatrix} x_{211} & -y_{211} & 1 & 0 \\ y_{211} & x_{211} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{21L} & -y_{21L} & 1 & 0 \\ y_{21L} & x_{21L} & 0 & 1 \\ x_{221} & -y_{221} & 1 & 0 \\ y_{221} & x_{221} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{22L} & -y_{22L} & 1 & 0 \\ y_{22L} & x_{22L} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ x_{2F1} & -y_{2F1} & 1 & 0 \\ y_{2F1} & x_{2F1} & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{2FL} & -y_{2FL} & 1 & 0 \\ y_{2FL} & x_{2FL} & 0 & 1 \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ t_x \\ t_y \end{bmatrix} = \mathbf{A} \mathbf{z}$$

or

$$\hat{S}_2 = \mathbf{A} \mathbf{z} \quad (\text{E.11})$$

with

$$\mathbf{A}^T = \begin{bmatrix} x_{211} & y_{211} & \cdots & x_{21L} & y_{21L} & x_{221} & y_{221} & \cdots & x_{22L} & y_{22L} & \cdots & x_{2F1} & y_{2F1} & \cdots & x_{2FL} & y_{2FL} \\ -y_{211} & x_{211} & \cdots & -y_{21L} & x_{21L} & -y_{221} & x_{221} & \cdots & -y_{22L} & x_{22L} & \cdots & -y_{2F1} & x_{2F1} & \cdots & -y_{2FL} & x_{2FL} \\ 1 & 0 & \cdots & 1 & 0 & 1 & 0 & \cdots & 1 & 0 & \cdots & 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 & 0 & 1 & \cdots & 0 & 1 & \cdots & 0 & 1 & \cdots & 0 & 1 \end{bmatrix} \quad (\text{E.12})$$

and

$$\mathbf{z} = [a_x, a_y, t_x, t_y]^T. \quad (\text{E.13})$$

Substituting (E.11) in (E.5) we obtain

$$d_{1\hat{2}}^2 = (\mathbf{A}\mathbf{z} - S_1)^T \mathbf{W}^T \mathbf{W} (\mathbf{A}\mathbf{z} - S_1). \quad (\text{E.14})$$

Now, we can solve for \mathbf{z} that minimizes $d_{1\hat{2}}^2$, which is the least-squares solution to $\mathbf{W}\mathbf{A}\mathbf{z} = \mathbf{W}S_1$, [Strang1988],

$$\mathbf{z} = ((\mathbf{W}\mathbf{A})^T (\mathbf{W}\mathbf{A}))^{-1} (\mathbf{W}\mathbf{A})^T \mathbf{W}S_1 = (\mathbf{A}^T \mathbf{W}^T \mathbf{W}\mathbf{A})^{-1} \mathbf{A}^T \mathbf{W}^T \mathbf{W}S_1 \quad (\text{E.15})$$

Once \mathbf{z} is known, s and θ can be found since ((E.9) and (E.13))

$$\frac{a_y}{a_x} = \frac{s \sin(\theta)}{s \cos(\theta)} = \tan(\theta) \quad (\text{E.16})$$

which gives

$$\theta = \arctan\left(\frac{a_y}{a_x}\right) \quad (\text{E.17})$$

and with $a_x = s \cos(\theta)$ we have

$$s = \frac{a_x}{\cos\left(\arctan\left(\frac{a_y}{a_x}\right)\right)}. \quad (\text{E.18})$$

The following algorithm is used for aligning the set of N ST-shapes:

1. Choose an arbitrary ST-shape and align all the others to it.
2. Calculate the mean of the aligned ST-shapes.
3. Normalize the resulting mean ST-shape.
4. Re-align each ST-shape to the normalized mean.
5. Check for convergence. If convergence is not reached go to step 2.
6. Done! ST-shapes aligned.

Notes:

- (a) The pose of an ST-shape is described by the scaling, rotation, and translation of all its shapes with respect to a known reference.
- (b) One way to normalize the mean ST-shape is by aligning it to the first ST-shape.
- (c) Normalization is carried out in order to force the process to converge, otherwise the mean ST-shape may translate or expand (or shrink) indefinitely.
- (d) Convergence is established if the ST-shapes are not changing more than a pre-defined threshold.

- (e) The mean of a set of ST-shapes is found by $\bar{S} = \frac{1}{N} \sum_{i=1}^N S_i$.

Appendix F. Skeleton Pruning for Medial Axis Extraction

This appendix describes the method we developed and performed to extract the primary medial axis from expert segmented images (binary images) of the corpus callosum (CC). Extracting the medial axis is necessary for the representation and the statistical analysis of the CC shapes for both the medial-based shape profiles (Chapter 6) and for the physics-based (spring-mass) mesh model (Chapter 7).

Using the binary expert segmented image as input, we first extract the skeleton using a standard technique (morphological operations to remove pixels on the boundaries of objects without allowing objects to break apart). Then, we find all branches by looking for terminal points or terminal pixels. We find the terminal pixels by utilizing six 3×3 terminal detection masks (see Table F.1 and first row of Figure F.1) at each image pixel of the binary image containing the skeleton.

Table F.1. Six terminal detection masks.

$\begin{pmatrix} 0 & 0 & 0 \\ 0 & (1) & 0 \\ 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & (1) & 0 \\ 0 & 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & (1) & 0 \\ 1 & 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & (1) & 0 \\ 1 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & (1) & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ 0 & (1) & 1 \\ 0 & 1 & 1 \end{pmatrix}$
(a)	(b)	(c)	(d)	(e)	(f)

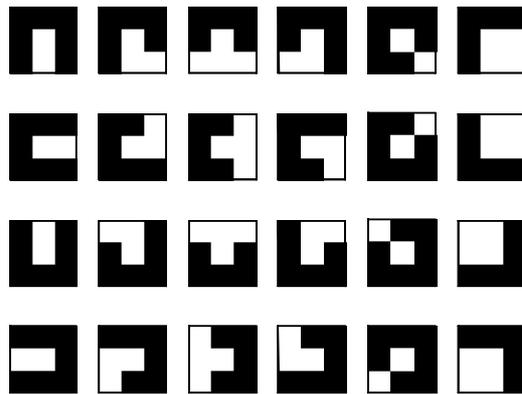


Figure F.1. The 24 terminal detection masks. Each column shows the four rotated versions of a single mask.

For example in the mask in Table F.1(a), the mid-point corresponds to a terminal point pointing upwards. Three other masks are generated from each of the six masks by applying 90 degrees rotations totaling 24 masks (Figure F.1).

For example, rotating mask (a) by 90 degrees counterclockwise will give a similar terminal-detection mask used for detecting terminal points that are

directed to the left: $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$.

Each terminal point detected on the skeleton is considered to be the tip of one branch. Now, each branch is pruned *with all other original branches still existing* until the branch is so short that no terminal point is found. Pruning one branch after removing a previous branch gives unacceptable results since the location where the pruning of a single branch stops depends on the where the others are. All locations of the pruned pixels are collected in a matrix and then removed from the original skeleton image when all branches are pruned. In some structures (unlike the corpus callosum case) we may need to specify a minimum primary medial axis length of the (minimum length of the main branch).

Appendix G. Physics-Based Shape Deformation Tool

This appendix complements Chapter 7 by presenting details about the implementation of a tool for physics-based shape deformations in MATLAB.

In addition to the properties of the nodes (mass, position, velocity, and acceleration) and springs (terminal nodes, Hook's spring constant, damping constant, and rest length) mentioned in Section 7.2, other options or properties are implemented for example: nodes can be *im/movable*, *boundary/medial* (inner) node, *un/freeze-able*, and springs can be *un/accept-able*.

The *un/freeze-able* property of a node determines whether or not the node is affected by the *accept* action: an action that sets the velocity, acceleration, and forces of a node to zero. The *un/accept-able* property of a spring determines whether or not the spring is affected by the *freeze* action: an action that sets the rest length of a spring to the value of its current length. The setting of the values of these two properties was found to be useful for "sculpting" the shape of different mesh models (for manual segmentation, for example) by simulating a heated plastic like behavior of the mesh or parts of it.

Other application dependent forces can be incorporated, for example gravity (weight) and wall (or image boundary) reflection (with the choice of the restitution coefficient) were implemented. Others such as universal gravitation, electrostatic forces, etc. can also be easily added. Rotation and translation forces can also be set to decay at a certain rate and last for a number of iterations.

The physics-based shape deformation tool was equipped with other features such as: the choice of explicit or semi-implicit [Tu1996, pp. 41-52] Euler dynamic simulation with a settable time step, pausing/running the time integration simulation, random and rectangular grid generation of mesh nodes, wiggle action and auto-wiggle that generate random forces at the mesh nodes, capturing movie sessions, loading and restoring node positions or mesh models.

The tool was also equipped with a facility to run pre-scheduled deformations by reading them from a script file. We ran simple synthetic image segmentation examples where nodes sense image searching for certain features and resulting mismatch values evaluated during the scheduled deformations, followed by a decision-making phase. The schedules (script files) can include both learned (e.g. main modes of variation) and unlearned deformations commands (e.g. bend, stretch, etc.) applied at different scales and locations. A list of possible physics-based deformations scripting language commands is given in Table G.1. An example use of the commands in a script file is shown in Table G.2. Figure G.1 shows the graphical user interface of the tool developed. Figure G.2 shows sample iteration results during a scheduled segmentation of a synthetic object.

Table G.1. Physics-based deformations, scripting language commands.

STRETCH from (x,y)	
<code><tttt> stretch xy--- <prm> <x> <y> <dx> <dy> <rad> <mag></code>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 6.
x,y	the center of deformation (influence region measured w.r.t. this point)
dx,dy	specify stretching direction
rad	radius of influence
mag	magnitude of stretching
STRETCH from node	
<code><tttt> stretch_nd--- <prm> <node> <dx> <dy> <rad> <mag></code>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 5.
node	this node will be the center of deformation (influence region measured w.r.t. this point)
dx,dy	specify stretching direction
rad	radius of influence
mag	magnitude of stretching
STRETCH from node in direction of nodes	
<code><tttt> stretch_ndnd- <prm> <node> <dirnd1> <dirnd2> <rad> <mag></code>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 5.
node	this node will be the center of deformation (influence region measured w.r.t. this point)
dirnd1/2	specify stretching direction, i.e. dirnd1 --> dirnd2
rad	radius of influence
mag	magnitude of stretching
BULGE from (x,y)	
<code><tttt> bulge_xy----- <prm> <x> <y> <rad> <mag></code>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 5.
x,y	the center of deformation (influence region measured w.r.t. this point)
node	this node will be the center of deformation (influence region measured w.r.t. this point)
rad	radius of influence
mag	magnitude of stretching
BULGE from node	
<code><tttt> bulge_xy----- <prm> <node> <rad> <mag></code>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 3.
node	this node will be the center of deformation (influence region measured w.r.t. this point)
rad	radius of influence
mag	magnitude of stretching
SCALE	
<code><tttt> scale----- <prm> <amount></code>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 1.
amount	amount of scaling

ROTATE	
<code><tttt> rotate----- <prm> <amount> <decay> <duration></code>	
<code>tttt</code>	timer value, after how many iterations this deformation starts
<code>prm</code>	number of parameters to follow, should be set to --> 3.
<code>amount</code>	amount of scaling
<code>decay</code>	rotation force decay factor (0-1)
<code>durttion</code>	how many iterations should the force stay active
TAPER from xys	
<code><tttt> taper----- <prm> <x1> <y1> <x2> <y2> <x3> <y3> <x4> <y4> <stretch 1> <stretch 2></code>	
<code>tttt</code>	timer value, after how many iterations this deformation starts
<code>prm</code>	number of parameters to follow, should be set to --> 10.
<code>x1-x4</code>	springs with nodes in the quadrilateral defined by (x1,y1)-(x4,y4) will
<code>y1-y4</code>	be shrunk or extended according to their distance from the side 1-2
<code>stretch1</code>	amount of stretching (or shrinking) for springs close to side 1-2
<code>stretch2</code>	amount of stretching (or shrinking) for springs far away from side 1-2
TAPER from nodes	
<code><tttt> taper----- <prm> <n1> <n2> <n3> <n4> <stretch 1> <stretch 2></code>	
<code>tttt</code>	timer value, after how many iterations this deformation starts
<code>prm</code>	number of parameters to follow, should be set to --> 6.
<code>n1-n4</code>	springs with nodes in the quadrilateral defined by the nodes n1-n4 will be shrunk or extended according to their distance from the side 1-2
<code>stretch1</code>	amount of stretching (or shrinking) for springs close to side 1-2
<code>stretch2</code>	amount of stretching (or shrinking) for springs far away from side 1-2
BENDING from xys	
<code><tttt> taper----- <prm> <x1> <y1> <x2> <y2> <x3> <y3> <x4> <y4> <x5> <y5> <x6> <y6> <amount></code>	
<code>tttt:</code>	timer value, after how many iterations this deformation starts
<code>prm</code>	number of parameters to follow, should be set to --> 13.
<code>x1-x6</code>	springs with nodes in the quadrilateral 1234(xy's) will shrunk and those
<code>y1-y6</code>	in 3456 will be elongated
<code>amount</code>	amount of bending
BENDING from nodes	
<code><tttt> taper----- <prm> <n1> <n2> <n3> <n4> <n5> <n6> <amount></code>	
<code>tttt</code>	timer value, after how many iterations this deformation starts
<code>prm</code>	number of parameters to follow, should be set to --> 6.
<code>n1-n6</code>	springs with nodes in the quadrilateral 1234(nodes) will be shrunk and those in 3456 will be elongated
<code>amount</code>	amount of bending
FREEZING (F,A,V set to zeros)	
<code><tttt> freeze----- <prm></code>	
<code>tttt</code>	timer value, after how many iterations this command runs
<code>prm</code>	number of parameters to follow, should be set to --> 0.

STORING (store nodesXY)	
<tttt> storeXY----- <prm>	
tttt	timer value, after how many iterations this command runs
prm	number of parameters to follow, should be set to --> 0.
RETRIEVING (load nodesXY)	
<tttt> retrieveXY--- <prm>	
tttt	timer value, after how many iterations this command runs
prm	number of parameters to follow, should be set to --> 0.
FIXING NODES (or releasing)	
<tttt> fixNodes----- <prm> <fix/rls> <nodes>	
tttt	timer value, after how many iterations this command runs
prm	number of parameters to follow, should be set to --> 2.
fix/rls	fix or release nodes, 1:fix, 0: rls
nodes	vector of nodes to fix or release.
MATLAB COMMAND	
<tttt> MATLABcmd---- <prm> <cmd>	
tttt	timer value, after how many iterations this command runs
prm	number of parameters to follow, should be set to --> 1.
cmd	MATLAB command.
SENSE	
<tttt> sense----- <prm> <info>	
tttt	timer value, after how many iterations this command runs
prm	number of parameters to follow, should be set to --> 1.
info	info is a cell array that contains {filter,sensor nodes,source} source can be: 'image' or 'grad'
DECIDE	
<tttt> decide----- <prm> <func>	
tttt	timer value, after how many iterations this command runs
prm	number of parameters to follow, should be set to --> 1.
info	info is a cell array that contains the decision maker function name {'dm_fname'} that takes sensed data as input and returns the best deformation: function cmdNum=dm_fname(sensData)
ACTUATE SPRINGS	
<tttt> actsprings--- <prm> <sprnums> <fracts>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 2.
sprnums	vector of spring numbers
fracts	vector of amount of change in rest lengths e.g. 1 means no change, 2 doubles and 0.5 halves the rest lengths
ACTUATE SPRINGS RELATIVE	
<tttt> actspringsrl- <prm> <sprnums> <refspr> <fracts>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 3.
sprnums	vector of spring numbers to be actuated
refspr	vector of reference spring numbers
fracts	vector of amount of change in rest lengths as a fraction of the rest length of reference springs
ACTUATE SPRINGS ABSOLUTE	
<tttt> actspringsab- <prm> <sprnums> <lengths>	
tttt	timer value, after how many iterations this deformation starts
prm	number of parameters to follow, should be set to --> 2.
sprnums	vector of spring numbers to be actuated
lengths	vector of length values

SETTING FREEZABLE NODES (or unfreezable)	
<code><tttt> frznodes----- <prm> <frz/unfrz> <nodes></code>	
<code>tttt</code>	timer value, after how many iterations this command runs
<code>prm</code>	number of parameters to follow, should be set to --> 2.
<code>frz/unfrz</code>	freezable or unfreezable nodes, 1:frz, 0: unfrz
<code>nodes</code>	vector of nodes to set the freezable property.
SETTING ACCEPT-ABLE SPRINGS (or un-accept-able)	
<code><tttt> accsprings--- <prm> <acc/unacc> <springs></code>	
<code>tttt</code>	timer value, after how many iterations this command runs
<code>prm</code>	number of parameters to follow, should be set to --> 2.
<code>acc/unacc</code>	accept-able or un-accept-able nodes, 1:acc, 0: unacc
<code>nodes</code>	vector of springs to set the accept-able property

Table G.2. Physics-based shape deformation, example schedule (script file).

```

0000 bend_nd----- 06 1 2 3 4 5 6 1.3
0040 bend_xy----- 13 100 50 200 50 200 150 100 150 100 200 200 200 1.3
0040 taper_xy----- 10 50 200 50 50 200 50 200 200 1.2 0.8
0030 taper_nd----- 06 1 2 3 4 1.2 0.8
0030 scale----- 01 1.8
0030 rotate----- 03 80 0.9 30
0030 translate---- 04 50 80 0.9 30
0030 bulge_xy----- 04 100 100 30 2
0030 bulge_nd----- 03 4 30 2
0030 stretch_xy--- 06 100 100 40 40 30 3
0030 stretch_nd--- 05 4 40 40 30 .2
0010 stretch_ndnd- 05 3 6 1 30 3
0000 storeXY----- 00
0000 freeze----- 00
0030 retrieveXY--- 00
0000 fixNodes----- 02 1 [1 2 3 10 13]
0000 MATLABcmd---- 01 a=3;a+5,clear a;
0000 sense----- 01 {fspecial('log',5),[1 2 6 8 10], 'image'};
0010 decide----- 01 {'dm_synth2'}
0010 actsprings--- 02 [1 2 3 6 7 8] [1.1 1.2 1.1 .8 .8 .7]
0010 actspringsrl- 03 [53 58 63] [51 56 61] [1.0 1.0 1.0]
0010 actspringsab- 02 [53 58 63] [10 12 10]
0010 accsprings--- 02 1 [2 4 5]
0010 frznodes----- 02 1 [1 2 3 9 11]

```

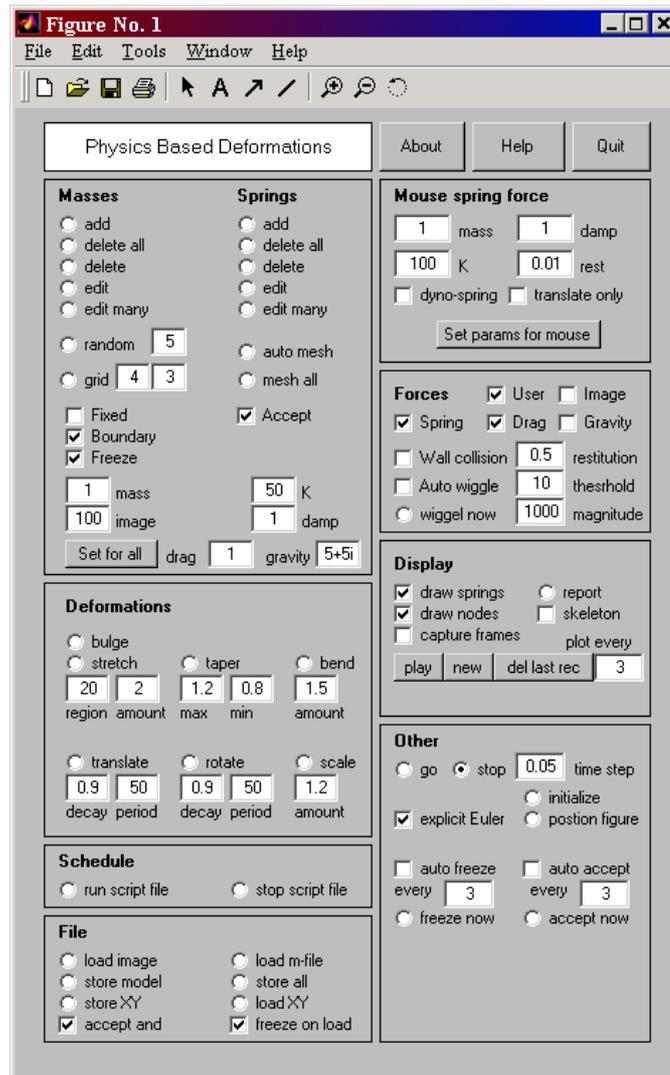


Figure G.1. Graphical user interface of the physics-based deformation tool.

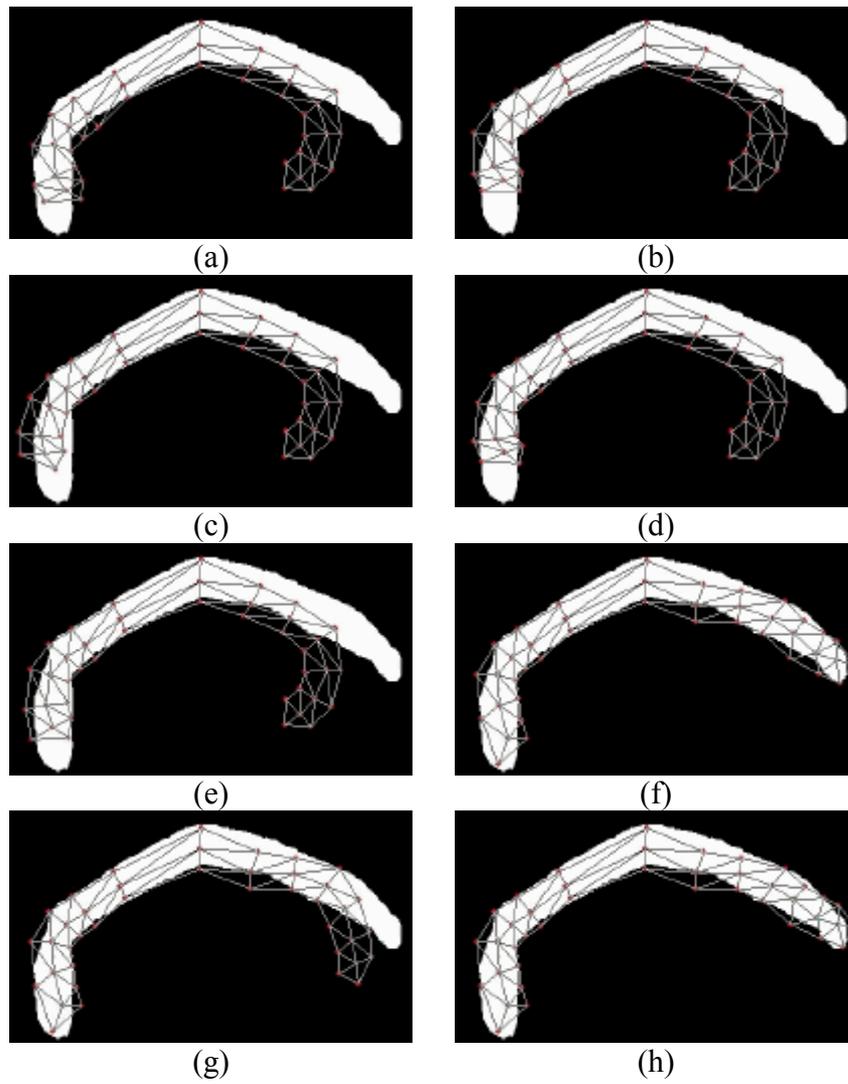


Figure G.2. Sample iteration results during a scheduled segmentation of a synthetic object (from (a) to (h)).

BIBLIOGRAPHY

- [Abu-Gharbieh1998] R. Abu-Gharbieh, G. Hamarneh, T. Gustavsson. "Review - Active Shape Models - Part II: Image Search and Classification". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 1998, pp. 129-132, Uppsala, Sweden, March 16-17, 1998.
- [Abu-Gharbieh2001a] R. Abu-Gharbieh, C. Kaminski, T. Gustavsson, G. Hamarneh. "Flame Front Matching and Tracking in PLIF Images Using Geodesic Paths and Level Sets". *IEEE Proceeding of the workshop on Variational and Level Set Methods in Computer Vision*. pp. 112-118, Vancouver, Canada, July 13, 2001.
- [Abu-Gharbieh2001b] R. Abu-Gharbieh, G. Hamarneh, T. Gustavsson, C. Kaminski. "Flame front tracking by laser induced fluorescence spectroscopy and advanced image analysis". *Journal of Optics Express*. Vol. 8(5), pp. 278-287, 2001.
- [ACS] ACS, American Cancer Society,
<http://www.cancer.org>.
- [Akgul1998] Y. S. Akgul, C. Kambhamettu, M. Stone. "Extraction and tracking of the tongue surface from ultrasound image sequences". *IEEE Proceedings on Computer Vision and Pattern Recognition*, pp. 298 -303. 1998.
- [Althoff2000] K. Althoff, G. Hamarneh, T. Gustavsson. "Tracking Contrast in Echocardiography by a Combined Snake and Optical Flow Technique". *IEEE Proceedings on Computers in Cardiology*. Vol. 27, pp. 29-32, Cambridge, USA, September 24-27, 2000.
- [Althoff2001] K. Althoff, *Flow Measurements in Contrast Echocardiographic Image Sequences*. Licentiate Thesis, Technical report 386L, Department of Signals and Systems, Chalmers University of Technology, 2001.
- [Amini1990] A. Amini, T. Weymouth, R. Jain. "Using dynamic programming for solving variational problems in vision". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 12(9), pp. 855 -867, 1990.
- [Antoine1998] J. Antoine, M. Viergever. "A Survey of Medical Image Registration". *Medical Image Analysis*. Vol. 2(1), pp. 1-37, 1998.

- [Audette2000] M. Audette, F. Ferrie, T. Peters. "An algorithmic overview of surface registration techniques for medical imaging". *Medical Image Analysis*. Vol. 4(3), pp. 201-217, 2000.
- [Ayache1995] N. Ayache. "Medical computer vision, virtual reality and robotics". *Image and Vision Computing*. Vol. 13(4), pp. 295-313, May 1995.
- [Ballerini1998] L. Ballerini. "Genetic Snakes for Medical Images Segmentation". *SPIE Proceedings on Mathematical Modeling and Estimation Techniques in Computer Vision*. Preteux, Davison and Dougherty ed., Vol. 3457, pp. 284-295, 1998.
- [Barr1984] A. Barr. "Global and Local Deformations of Solid Primitives". *Proceedings of SIGGRAPH'84, Computer Graphics*. Vol. 18(3), pp. 21-30, 1984.
- [Baumberg1994] A. Baumberg, D. Hogg. "An efficient method for contour tracking using active shape models". *IEEE Workshop Proceedings on Motion of Non-Rigid and Articulated Objects*, pp. 194 -199, 1994.
- [Bill1995] J. Bill, S. Lodha. "Sculpting Polygonal Models using Virtual Tools". *Proceedings of Graphics Interface*. GI 1995, pp. 272-279, 1995.
- [Black1997] M. Black, Y. Yacoob. "Recognizing Facial Expressions in Image Sequences using Local Parametrized Models of Image Motion". *International Journal of Computer Vision*. Vol. 25(1), pp. 23-48, 1997.
- [Blake1998] A. Blake, Isard M. *Active Contours*. Springer-Verlag, 1998.
- [Bonciu1998] C. Bonciu, C. Léger, J. Thiel. "A Fourier-Shannon approach to closed contour modeling". *Bioimaging*. Vol. 6, pp. 111-125, 1998.
- [Bookstein1989] F. Bookstein. "Principal Warps: Thin-Plate Splines and the Decomposition of Deformations". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol 11(6), pp. 567-585, June 1989.
- [Borgefors1984] G. Borgefors. "Distance transformations in arbitrary dimension". *Computer Vision, Graphics and Image processing: Image understanding*, Vol. 27, pp. 321-345, 1984.

-
- [Bowden1997] R. Bowden, T. Mitchell, M. Sarhadi. "Cluster based nonlinear principle component analysis". *IEEE Electronics Letters*. Vol. 33(22), pp. 1858-1859, October 1997.
- [Bowden2000] R. Bowden, T. Mitchell, M. Sarhadi. "Non-linear Statistical Models for the 3D Reconstruction of Human Pose and Motion from Monocular Image Sequences". *Image And Vision Computing*. Vol. 18(9), pp. 729-737, 2000.
- [Bradley1997] J. Bradley. "MATLAB Statistics Toolbox - User's Guide". *Publications of the The MathWorks Inc.*, Tutorial, pp. 77-87, 1997.
- [Caselles1995] V. Caselles, R. Kimmel, G. Sapiro. "Geodesic active contours". *IEEE Proceedings of the Fifth International Conference on Computer Vision, ICCV 1995*, pp. 694 -699, 1995.
- [Chennubhotla2001] C. Chennubhotla, A. Jepson, "Sparse PCA: Extracting Multi-scale Structure from Data". *IEEE Proceedings of the International Conference on Computer Vision*. Vancouver, Canada, July 2001.
- [Chiou1996] G. I. Chiou, J-N. Hwang, "Lipreading from color motion video". *IEEE conference proceedings on Acoustics, Speech, and Signal Processing*. Vol. 4, pp. 2156-2159, 1996.
- [Chodorowski1999] A. Chodorowski, U. Mattsson, T. Gustavsson. "Oral lesion classification using true-color images". *SPIE Proceedings on Medical Imaging: Image Processing*, Editor: M. Hansson, Vol. 3661, pp. 1127-1138, 1999.
- [Chodorowski2000] A. Chodorowski, *Pattern Recognition Methods for Oral Lesion Classification using Digital Color Images*. Ph.D. thesis, Technical report 396, Department of Signals and Systems, Chalmers University of Technology, December 2000.
- [Choi1997] C. Choi, M. Wirth, A. Jennings. "Segmentation: artificial life and watershed transform approach". *IEEE Sixth International Conference on Image Processing and Its Applications*. Vol. 1, pp. 371-375, 1997.
- [Christensen1996] G. Christensen, R. Rabbitt, M. Miller. "Deformable templates using large deformation kinematics". *IEEE Transactions on Image Processing*, Vol. 5(10), pp. 1435-1447, October 1996.

- [Cohen1991] L. D. Cohen. "On active contour models and balloons". *Computer Vision, Graphics and Image processing: Image understanding*, Vol. 53(2), pp. 211-218, 1991.
- [Cohen1993] L. Cohen, I. Cohen. "Finite-element methods for active contour models and balloons for 2-D and 3-D images". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 15(11), pp. 1131 -1147, November 1993.
- [Cohen1996] L. Cohen, R. Kimmel. "Global minimum for active contour models: a minimal path approach". *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. CVPR 1996, pp. 666 -673, 1996.
- [Cootes1992] T. Cootes, C. Taylor, D. Cooper, J. Graham. "Training Models of Shape from Sets of Examples". *Proceedings of the British Machine Vision Conference*. Springer-Verlag, pp. 9-18, 1992.
- [Cootes1993] T. Cootes, C. Taylor, A. Hill, J. Halsam. "The Use of Active Shape Models for Locating Structures in Medical Images". *Proceedings of the 13th International Conference on Information Processing in Medical Imaging*, Editors: H.H.Barrett, A.F.Gmitro, Springer-Verlag, pp. 33-47, 1993.
- [Cootes1994a] T. Cootes, C. Taylor, A. Lanitis. "Active Shape Models: Evaluation of a Multi-Resolution Method for Improving Image Search". *Proceedings of the British Machine Vision Conference*. BMVC1994, pp. 327-336, 1994.
- [Cootes1994b] A. Parker, A. Hill, C. Taylor, T. Cootes, X. Jin, D. Gibson. "Application of point distribution models to the automated analysis of echocardiograms". *IEEE Proceedings on Computers in Cardiology*. pp. 25 -28, 1994.
- [Cootes1995a] T. Cootes, C. Taylor, D. Cooper, J. Graham. "Active Shape Models - Their Training and Application". *Computer Vision and Image Understanding*. Vol. 61(1), pp.38-59, 1995.
- [Cootes1995b] T. Cootes, C. Taylor. "Combining point distribution models with shape models based on finite element analysis". *Image and Vision Computing*. Vol. 13(5), pp. 403-409, June 1995.

-
- [Cootes1995c] T. Cootes, A. Hill, C. Taylor. "Medical Image Interpretation Using Active Shape Models: Recent Advances". *Information Processing in Medical Imaging*. Kluwer Academic Publishers, Editor: Y. Bizais, pp. 371-372, 1995.
- [Cootes1997] T. Cootes, C. Taylor. "A Mixture Model for Representing Shape Variation". *Proceedings of the 8th British Machine Vision Conference*. Vol. 1, Editor: A.F.Clark. British Machine Vision Association Press, pp. 110-119, 1997.
- [Cootes1998] T. Cootes, G. Edwards, C. Taylor. "Active Appearance Models". *Proceedings of the European Conference on Computer Vision*. Editors: H.Burkhardt & B.Neumann. Vol. 2, pp. 484-498, Springer, 1998.
- [Cootes1999] T. Cootes, C. Beeston, G. Edwards, C. Taylor. "A Unified Framework for Atlas Matching using Active Appearance Models". *Proceedings of the International Conference on Image Processing in Medical Imaging*, pp 322-333, 1999.
- [Cootes2000a] T. Cootes, K. Walker, C. Taylor. "View-Based Active Appearance Models". *Proceedings of the International Conference on Face and Gesture Recognition*, pp. 227-232, 2000.
- [Cootes2000b] T. Cootes, C. Taylor. "Combining Elastic and Statistical Models of Appearance Variation". *Proceedings of the European Conference on Computer Vision*. Vol.1, pp.149-163, 2000.
- [Coquillart1990] S. Coquillart. "Extended Free Form Deformations: A Sculpting Tool for 3D Geometric Modeling". *Proceedings of SIGGRAPH'90*. Vol. 24(2), pp. 187-196, 1990.
- [Curwen1994] R. Curwen, A. Amini, J. Duncan, F. Lee. "Tracking vascular motion in X-ray image sequences with Kalman snakes". *IEEE Proceedings on Computers in Cardiology*, pp.109 -112, 1994.
- [Delingette1999] H. Delingette. "General Object Reconstruction Based on Simplex Meshes". *International Journal of Computer Vision*. Vol. 32, pp. 111-142, 1999.

- [DhIngra1996] J. K. DhIngra, D. F. Jr. Perrault, K. McMillan, E. E. Rebeiz, S. M. Shapshay, R. Manoharan, I. Itzkan, M. S. Feld, S. Kabani, "Spectroscopic analysis of precancerous and cancerous lesions of the oral cavity". *IEEE Lasers and Electro-Optics Society Annual Meeting*, Vol. 1, pp. 332-333, 1996.
- [Duncan2000] J. Duncan, N. Ayache. "Medical Image Analysis: Progress Over Two Decades and the Challenges Ahead". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 22(1), 85 -106, 2000.
- [EB] Encyclopedia Britannica,
<http://www.britannica.com/>.
- [Edwards1998] G. Edwards, C. Taylor, T. Cootes. "Interpreting face images using active appearance models". *IEEE Proceedings of the Third International Conference on Automatic Face and Gesture Recognition*, pp. 300-305, 1998.
- [Ercal1993] F. Ercal, M. Moganti, W. V. Stoecker, R. H. Moss. "Detection of skin tumor boundaries in color images". *IEEE Transactions on Medical Imaging*. Vol. 123, pp. 624-626, 1993.
- [Fagan1992] M. Fagan. *Finite Element Analysis: Theory and Practice*, Longman Scientific and Technical, ISBN: 0-58-202247-9, 1992.
- [Fischler1973] M. Fischler, R. Elschlager. "The representation and matching of pictorial structures". *IEEE Transactions on Computers*. Vol. 22(1), 67-92, 1973.
- [Fritsch1997] D. Fritsch, S. Pizer, L. Yu, V. Johnson, E. Chaney. "Segmentation of medical image objects using deformable shape loci". *Proceedings of the 15th International Conference on Information Processing in Medical Imaging*, Lecture Notes in Computer Science. Vol. 1230, pp. 127-140, Springer-Verlag, 1997.
- [Gibson1997] S. Gibson, B. Mirtich. "A survey of deformable modeling in computer graphics". Technical Report TR-97-19, Mitsubishi Electric Research Laboratory, November 1997.
- [Grzeszczuk1997] R. Grzeszczuk, D. Levin. "Brownian strings: Segmenting images with stochastically deformable contours". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 19(10), pp. 1100-1114, 1997.

-
- [Gunn1994] S. Gunn, M. Nixon. "A dual active contour for head boundary extraction". *IEE Colloquium on Image Processing for Biometric Measurement*, Vol 6, pp. 1-4, 1994.
- [Gunn1996] S. Gunn, M. Nixon. "Snake head boundary extraction using global and local energy minimization". *Proceedings of the 13th International Conference on Pattern Recognition*. Vol. 2, pp. 581-585, 1996.
- [Gunn1997] S. Gunn, M. Nixon. "A robust snake implementation; a dual active contour". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19(1), pp. 63-68, 1997.
- [Gustavsson1997] T. Gustavsson, R. Abu-Gharbieh, G. Hamarneh, Q. Liang. "Implementation and Comparison of Four Different Boundary Detection Algorithms for Quantitative Ultrasonic Measurements of the Human Carotid Artery". *IEEE Proceedings on Computers in Cardiology*. Vol. 24, pp. 69-72, Lund, Sweden, September 1997.
- [Hamarneh1996] G. Hamarneh. "Implementation and Comparison of Four Different Boundary Detection Algorithms for Quantitative Ultrasonic Measurements of the Human Carotid Artery". Masters thesis, Technical report EX0NA/1996, Department of Applied Electronics, Chalmers University of Technology, December 1996.
- [Hamarneh1998a] G. Hamarneh, R. Abu-Gharbieh, T. Gustavsson. "Review - Active Shape Models - Part I: Modeling Shape and Gray Level Variation". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 1998, pp. 125-128, Uppsala, Sweden, March 16-17, 1998.
- [Hamarneh1998b] G. Hamarneh. "Active Shape Models, Modeling Shape Variations and Gray Level Information and an Application to Image Search and Classification". Technical Report R005/1998 (S2-IAG-98-1), Department of Signals and Systems, Chalmers University of Technology, Sweden, September 1998.
- [Hamarneh1999a] G. Hamarneh, T. Gustavsson. "A Method for Modeling and Segmentation of Spatio-Temporal Shapes". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 1999, pp. 49-52, Göteborg, Sweden, March 9-10, 1999.

- [Hamarneh1999b] G. Hamarneh. *Deformable Spatio-Temporal Shape Modeling*. Technical report 311L, Licentiate Thesis, Department of Signals and Systems, Chalmers University of Technology, 1999.
- [Hamarneh1999c] G. Hamarneh. "Digital Image Analysis of Fundus Photographs on the WWW". Technical Report R002/1999 (S2-IAG-99-1), Department of Signals and Systems, Chalmers University of Technology, Sweden, February 1999.
- [Hamarneh2000a] G. Hamarneh, K. Althoff, T. Gustavsson. "Snake Deformations Based on Optical Flow Forces for Contrast Agent Tracking in Echocardiography". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 2000, pp. 45-48, Halmstad, Sweden, March 7-8, 2000.
- [Hamarneh2000b] G. Hamarneh, T. Gustavsson. "Constraining Contour Deformations Using Statistical A Priori Knowledge of Shape Without Requiring Point-to-Point Correspondence". *Proceedings of the Swedish Symposium on Image Analysis*. SSAB 2000, pp. 33-36, Halmstad, Sweden, March 7-8, 2000.
- [Hamarneh2000c] G. Hamarneh, T. Gustavsson. "Combining Snakes and Active Shape Models for Segmenting the Human Left Ventricle in Echocardiographic Images". *IEEE Proceedings on Computers in Cardiology*. Vol. 27, pp.115-118, Cambridge, USA, September 24-27, 2000.
- [Hamarneh2000d] G. Hamarneh, A. Chodorowski, T. Gustavsson. "Active Contour Models: Application to Oral Lesion Detection in Color Images". *IEEE Proceedings of the International Conference on Systems, Man, and Cybernetics*. Vol. 4, pp. 2458-2463, Nashville, Tennessee, USA, October 8-11, 2000.
- [Hamarneh2000e] G. Hamarneh, T. Gustavsson. "Statistically Constrained Snake Deformations". *IEEE Proceedings of the International Conference on Systems, Man, and Cybernetics*. Vol. 3, pp. 1610-1615, Nashville, Tennessee, USA, October 8-11, 2000.
- [Hamarneh2000f] G. Hamarneh. "Image Segmentation with Constrained Snakes". *Swedish society for image analysis newsletter*. SSABlaskan Nr. 8, pp. 5, 2000.
- [Hamarneh2001a] G. Hamarneh, T. McInerney. "Controlled Shape Deformations via Medial Profiles". *Proceedings of Vision Interface*. VI 2001, pp. 252-258, Ottawa, Canada, June 7-9, 2001.

-
- [Hamarneh2001b] G. Hamarneh, T. McInerney, D. Terzopoulos. "Intelligent Deformable Organisms: An Artificial Life Approach to Medical Image Analysis". Technical report CSRG-432, Department of Computer Science, University of Toronto, 2001, <ftp://ftp.cs.toronto.edu/cs/ftp/csrsg-technical-reports/>.
- [Hamarneh2001c] G. Hamarneh, T. McInerney, D. Terzopoulos. "Deformable Organisms for Automatic Medical Image Analysis". *Proceedings of the Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention*. MICCAI 2001, Utrecht, The Netherlands, October 14-17, 2001.
- [Hamarneh2001d] G. Hamarneh, T. Gustavsson. "Deformable Spatio-Temporal Shape Models: Extending ASM to 2D+Time". *Proceedings of the British Machine Vision Conference*. BMVC 2001, Vol. 1, pp. 13-22, Manchester, UK, September 10-13, 2001.
- [Hamarneh2001e] G. Hamarneh, T. McInerney. "Physics-Based Shape Deformations for Medical Image Analysis". Technical report CSRG-436, Department of Computer Science, University of Toronto, 2001, <ftp://ftp.cs.toronto.edu/cs/ftp/csrsg-technical-reports/>.
- [Herlin1992] I. Herlin, C. Nguyen, C. Graffigne, "A deformable region model using stochastic processes applied to echocardiographic images". *Proceedings of Computer Vision and Pattern Recognition*. CVPR 1992, pp. 534-539, 1992.
- [Hill1992] A. Hill, C. Taylor, "Model-Based Image Interpretation Using Genetic Algorithms". *Image and Vision Computing*. Vol. 10(5), pp. 295-300, 1992.
- [Hill1993] A. Hill, A. Thornham, C. Taylor. "Model-Based Interpretation of 3D Medical Images". *Proceedings of the Fourth British machine Vision Conference*. BMVC 1993, pp. 339-348, Guilford, England, September 1993.
- [Hill1994] A. Hill, C. Taylor. "Automatic Landmark Generation for Point Distribution Models". *Proceedings of the British Machine Vision Conference*. Vol. 2, pp. 429-438, 1994.

- [Hill1997] A. Hill, A. Brett, C. Taylor. "Automatic landmark identification using a new method of non-rigid correspondence". *In Proceedings of the 15th International Conference on Information Processing in Medical Imaging*. IPMI 1997, pp. 483-488, June 1997.
- [Hill2000] A. Hill, C. Taylor, A. Brett. "A framework for automatic landmark identification using a new method of nonrigid correspondence". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 22(3), pp. 241-251, March 2000.
- [Horn1981] B. Horn, B. Schunk. "Determining Optical Flow". *Artificial Intelligence*. Vol. 17, pp. 185-204, 1981.
- [Hug1999] J. Hug, C. Brechbühler, G. Székely. "Tamed Snake: A Particle System for Robust Semi-automatic Segmentation". *Proceedings of the Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention*. MICCAI 1999, pp. 106-115, Cambridge, England, September 1999.
- [Hunter1993] I. A. Hunter, J. J. Soraghan, J. Christie, J. S. Durrani, "Detection of echocardiographic left ventricle boundaries using neural networks", *IEEE Proceedings of Computers in Cardiology*, pp. 201-204, 1993.
- [Jackson1991] E. A. Jackson. *User's Guide to Principal Components*. John Wiley & Sons, Inc., pp. 1-25, B. Jones 3-17-94, 1991.
- [Jain1989] A. Jain. *Fundamentals of Digital Image Processing*. Prentice Hall 1989.
- [Ji1999] L. Ji, H. Yan. "Loop-free snakes for image segmentation". *IEEE Proceedings of the International Conference on Image Processing*. Vol. 3, pp. 193-196, 1999.
- [Joshi2001] S. Joshi, S. Pizer, P. Fletcher, A. Thall, G. Tracton. "Multi-scale 3D deformable model segmentation based on medial description". *Proceedings of the International Conference on Information Processing in Medical Imaging*. IPMI 2001, Lecture Notes in Computer Science 2082 Springer 2001, ISBN 3-540-42245-5, Editors: M. Insana, R. Leahy, pp. 64-77, Davis USA, 2001.

-
- [Kagawa1999] H. Kagawa, M. Kinouchi, M. Hagiwara. "Image segmentation by artificial life approach using autonomous agents". *International Joint Conference on Neural Networks*. Vol. 6, pp. 4413-4418, 1999.
- [Kass1987] M. Kass, A. Witkin, D. Terzopoulos. "Snakes: Active Contour Models". *International Journal of Computer Vision*. Vol. 1(4), pp.321-331, 1987.
- [Kelemen1999] A. Kelemen, G. Szekely, G. Gerig. "Elastic model-based segmentation of 3-D neuroradiological data sets". *IEEE Transactions on Medical Imaging*, Vol. 18(10), pp. 828-839, October 1999.
- [Lachaud1999] J. Lachaud, A. Montanvert, "Deformable meshes with automated topology changes for coarse-to-fine three-dimensional surface extraction". *Medical Image Analysis*, Vol. 3(1), pp. 1-21, 1999.
- [Lanitis1994a] A. Lanitis, C. Taylor, T. Cootes. "Recognising Human Faces Using Shape and Grey-Level Information". *Proceedings of the Third International Conference on Automation, Robotics and Computer Vision*. Vol. 2, pp 1153-1157, Singapore, 1994.
- [Lanitis1994b] A. Lanitis, C. Taylor, T. Cootes. "Automatic Tracking, Coding and Reconstruction of Human Faces Using Flexible Appearance Models". *Electronics Letters*. Vol. 30(9), pp. 1578-1579, 1994.
- [Lanitis1995] A. Lanitis, C. Taylor, T. Cootes. "Automatic face identification system using flexible appearance models". *Image and Vision Computing*. Vol. 13(5), pp.393-401, 1995.
- [Lanitis1997] A. Lanitis, C. Taylor, T. Cootes. "Automatic Interpretation and Coding of Face Images Using Flexible Models". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 19(7), pp. 743-756, July 1997.
- [Ledley1990] R. Ledley, M. Buas, T. J. Golab. "Fundamentals of true-color image processing". *IEEE Proceedings of the Conference on Pattern Recognition*, Vol. 1, pp. 791-795, 1990.
- [Leventon2000] M. Leventon, W. Grimson, O. Faugeras, "Statistical shape influence in geodesic active contours". *IEEE Conference Proceedings on Computer Vision and Pattern Recognition*, Vol 1. pp. 316-323, 2000.

- [Leymarie1993] F. Leymarie, M. Levine, Tracking deformable objects in the plane using an active contour model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 15(6), pp. 617-634, 1993.
- [Liang1999a] J. Liang, T. McInerney, D. Terzopoulos. "United Snakes". *IEEE Proceedings on Computer Vision*. Vol. 2, pp. 933-940, 1999.
- [Liang1999b] J. Liang, T. McInerney, D. Terzopoulos. "Interactive Medical Image Segmentation with United Snakes". *Proceedings of the Fourth International Conference on Medical Image Computing and Computer-Assisted Intervention*. MICCAI 1999, pp. 116-127, Cambridge, England, September 1999.
- [Little1996] J. Little, D. Hill, D. Hawkes. "Deformations incorporating rigid structures". *Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis*. MMBIA 1996, pp. 104-113, 1996.
- [Lobregt1995] S. Lobregt, M. Viergever. "A discrete dynamic contour model". *IEEE Transactions on Medical Imaging*. Vol. 14(1), pp.12 -24. 1995.
- [Lundervold1999] A. Lundervold, N. Duta, T. Taxt, A. Jain. "Model-guided segmentation of corpus callosum in MR images". *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. Vol. 1, pp. 231-237, 1999.
- [MacCracken1996] R. MacCracken, K. Joy. "Free-Form Deformations with Lattices of Arbitrary Topology". *Proceedings of SIGGRAPH'96*, pp. 181-188, 1996.
- [MacEachern1998] L. MacEachern, T. Manku. "Genetic algorithms for active contour optimization". *IEEE Proceedings of the International Symposium on Circuits and Systems*. Vol. 4, pp. 229 -232, 1998.
- [Malassiotis1999] S. Malassiotis, M. G. Strintzis. "Tracking the left ventricle in echocardiographic images by learning heart dynamics". *IEEE Transactions on Medical Imaging*. Vol. 18(3), pp. 282-290, 1999.

-
- [Mandal1998] C. Mandal, B. Vemuri, H. Qin. "A New Dynamic FEM-based Subdivision Surface Model for Shape Recovery and Tracking in Medical Images". *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*. MICCAI 1998. Vol. 1496, pp. 753-760, 1998.
- [Mardia1995] K. Mardia, J. Kent, J. Bibby. *Multivariate Analysis*. Academic Press, 1995.
- [McInerney1995] T. McInerney, D. Terzopoulos. "A dynamic finite element surface model for segmentation and tracking in multidimensional medical images with application to cardiac 4D image analysis". *Computerized Medical Imaging and Graphics*. Vol. 19(1), pp. 69-83, 1995.
- [McInerney1996] T. McInerney, D. Terzopoulos. "Deformable Models in Medical Image Analysis: A Survey". *Medical Image Analysis*. Vol. 1(2), pp. 91-108, 1996.
- [McInerney1998] T. McInerney, R. Kikinis. "An Object-based Volumetric Deformable Atlas for the Improved Localization of Neuroanatomy in MR Images". *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*. MICCAI 1998. Vol. 1496, pp. 861-869, 1998.
- [McInerney1999] T. McInerney, D. Terzopoulos. "Topology adaptive deformable surfaces for medical image volume segmentation". *IEEE Transactions on Medical Imaging*. Vol. 18(10), pp. 840-850, October, 1999.
- [McInerney2000] T. McInerney, D. Terzopoulos. "T-Snakes: Topology adaptive snakes". *Medical Image Analysis*. Vol. 4(2), pp. 73-91, June 2000.
- [McKenna1994] W.J. McKenna, G. Thiene, A. Nava, F. Fontaliran, C. Blomström-Lundqvist, G. Fontaine, F. Camerini. "Diagnosis of arrhythmogenic right ventricular dysplasia/ cardiomyopathy". *British Heart Journal*. Vol. 71, pp. 215-218, 1994.
- [Menet1990] S. Menet, P. Saint-Marc, G. Medioni. "Active contour models: overview, implementation and applications". *IEEE International Conference Proceedings Systems, Man and Cybernetics*. pp. 194 -199, 1990.

- [Metaxas1991] D. Metaxas, D. Terzopoulos. "Constrained deformable superquadrics and nonrigid motion tracking". *IEEE Proceedings of Computer Vision and Pattern Recognition*. CVPR 1991, pp. 337-343, 1991.
- [Mikic1998] I. Mikic, S. Krucinski, J. D. Thomas. "Segmentation and tracking in echocardiographic sequences: active contours guided by optical flow estimates". *IEEE Transactions on Medical Imaging*. Vol. 17(2), pp.274-284, 1998.
- [Miller1991] J. Miller, D. Breen, W. Lorensen, R. O'Bara, M.J. Wozny. "Geometrically Deformed Models: A Method for Extracting Closed Geometric Models from Volume Data". *Proceedings of SIGGRAPH'91*, Vol. 25(4), pp. 217-226, 1991.
- [Moccozet1997] L. Moccozet, M. Thalmann, "Dirichlet free-form deformations and their application to hand simulation". *Computer Animation*. pp. 93-102, 1997.
- [Molloy2000] D. Molloy and P. Whelan. "Active-meshes". *Pattern Recognition Letters*, Vol. 21, pp. 1071-1080, 2000.
- [Montagnat1997] J. Montagnat, H. Delingette. "Volumetric medical image segmentation using shape constrained deformable models". *Proceedings of the Second International Conference on Computer Vision, Virtual Reality and Robotics in Medicine*, pp. 13-22, 1997.
- [Mortenson1997] M. Mortenson. *Geometric Modeling*. Wiley Computer Publishing 1997.
- [Mortenssen1992] E. Mortenssen, B. Morse, W. Barret, J. Udupa. "Adaptive Boundary Detection Using "Live-Wire" Two-Dimensional Dynamic Programming". *IEEE Proceedings of Computers in Cardiology*. pp. 635-638, 1992.
- [Ohta1980] Y. Ohta, T. Kanade, T. Sakai. "Color Information for Region Segmentation". *Computer Graphics and Image Processing*. Vol. 13, pp. 222-241, 1980.
- [Papadopoulos1995] I. Papadopoulos, M.G. Strintzis. "Bayesian contour estimation of the left ventricle in ultrasound images of the heart". *IEEE Proceedings of the Conference on Engineering in Medicine and Biology Society*. Vol. 1, pp. 591-592, 1995.

-
- [Parker1994] A. Parker, A. Hill, C. Taylor, T. Cootes, X. Jin, D. Gibson. "Application of Point Distribution Models to the Automated Analysis of Echocardiograms". *IEEE Proceedings of Computers in Cardiology*. IEEE Computer Society Press, pp. 25-28, 1994.
- [Peng1998] J. Peng, B. Bhanu, "Closed-loop object recognition using reinforcement learning". *IEEE transactions on Pattern Analysis and Machine Intelligence*, Vol. 20(2), pp. 139-154, 1998.
- [Pentland1991] A. Pentland, S. Sclaroff. "Closed-form solutions for physically based shape modeling and recognition". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 13(7), pp. 715-729, July 1991.
- [Perona1990] P. Perona, J. Malik. "Scale-space and edge detection using anisotropic diffusion". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 12(7), pp. 629-639, 1990.
- [Peterfreund1999] N. Peterfreund. "Robust tracking of position and velocity with Kalman snakes". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 21(6), pp. 564 -569, June 1999.
- [Pindborg1992] J. Pindborg. *Atlas of diseases of the oral mucosa*. Fifth edition, ISBN 87-16-10585-0, 1992.
- [Pizer1998] S. Pizer, D. Fritsch, K. Low, J. Furst. "2D & 3D Figural Models of Anatomic Objects from Medical Images". *Mathematical Morphology and Its Applications to Image Processing*. Kluwer Computational Imaging and Vision Series, Editors: H.J.A.M. Heijmans, J.B.T.M. Roerdink, Amsterdam, The Netherlands, pp. 139-150, Kluwer, 1998.
- [Pizer1999] S. Pizer, D. Fritsch. "Segmentation, Registration, and Measurement of Shape Variation via Image Object Shape". *IEEE Transactions on Medical Imaging*, Vol. 18(10), pp. 851-865, October 1999.
- [Pizer2000] S. Pizer, P. Fletcher, Y. Fridman, D. Fritsch, A. Gash, J. Glotzer, S. Joshi, A. Thall, G. Tracton, P. Yushkevich, E. Chaney. "Deformable M-Reps for 3D Medical Image Segmentation". Submitted to Medical Image Analysis 2000.

- [Pluim2000] J. Pluim, J. Maintz, M. Viergever. "Image registration by maximization of combined mutual information and gradient information". *IEEE Transactions on Medical Imaging*. Vol. 19(8), pp. 809-814, August 2000.
- [Provot1995] X. Provot. "Deformation Constraints in a Mass-Spring Model to Describe Rigid Cloth Behavior". *Proceedings of Graphics Interface*. GI 1995, pp. 147-154, Canada, May 1995.
- [Pun1993] T. Pun, G. Gerig, O. Ratib. "Image analysis and computer vision in medicine". *Computerized Medical Imaging and Graphics*. Special Issue: Multimedia Techniques in the Medical Environment, Vol. 18,(2), pp. 85-96, 1994.
- [Rasmussen1998] C. Rasmussen, G. Hager. "Joint probabilistic techniques for tracking objects using multiple visual cues". *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vol. 1, pp. 191-196, 1998.
- [Round1997] A. Round, A. Duller, P. Fish. "Colour segmentation for lesion classification". *IEEE Proceedings of the Engineering in Medicine and Biology Society*. Vol. 2, pp. 582-585, 1997.
- [Rueckert1995] D. Rueckert, P. Burger. "Contour fitting using an adaptive spline model". *Proceedings of the British Machine Vision Conference*. British Machine Vision Association Press, pp. 207-216, Editor: D. Pycock, Birmingham, UK, 1995.
- [Sapiro1996] G. Sapiro. "Vector (self) snakes: a geometric framework for color, texture, and multiscale image segmentation". *International Conference on Image Processing*. Vol. 1, pp. 817-820, 1996.
- [Scarloff1998] S. Sclaroff, J. Isidoro. "Active Blobs". *Proceedings of the Sixth International Conference on Computer Vision*. pp. 1146-1153, 1998.
- [Sederberg1986] T. Sederberg, S. Parry. "Free-Form Deformation of Solid Geometric Models". *Proceedings of SIGGRAPH'86*, Vol. 4(20), pp. 151-160, 1986.
- [Shen2000] D. Shen, C. Davatzikos. "An Adaptive-Focus Deformable Model Using Statistical and Geometric Information". *IEEE Transactions Pattern Analysis and Machine Intelligence*. Vol. 22(8). pp. 906 -913, 2000.

-
- [Singh1993] A. Singh, L. Von Kurowski, M. Chiu. "Cardiac MR image segmentation using deformable models". *SPIE Proceedings on Biomedical Image Processing and Biomedical Visualization*. Vol. 1905, pp. 8-28, Bellingham, WA, 1993.
- [Singh1998a] A. Singh, D. Goldgof, D. Terzopoulos. *Deformable Models in Medical Image Analysis*. IEEE Computer Society; ISBN: 0818685212, 1998.
- [Singh1998b] K. Singh, E. Fiume. "Wires: A Geometric Deformation Technique". *Proceedings of SIGGRAPH'98*, Vol. 99(1), pp. 405-414, 1998.
- [Sobottka1996] K. Sobottka, I. Pitas. "Segmentation and tracking of faces in color images". *Conference Proceedings on Automatic Face and Gesture Recognition*. pp. 236-241, 1996.
- [Sozou1994] P. Sozou, T. Cootes, C. Taylor, E. Di-Mauro. "A Non-linear Generalisation of PDMs using Polynomial Regression". *Proceedings of Fifth British Machine Vision Conference*. British Machine Vision Association Press, Vol. 1, pp. 397-406, Editor: Edwin Hancock, York, UK, 1994.
- [Sozou1995] P. Sozou, T. Cootes, C. Taylor, E. Di-Mauro. "Non-Linear Point Distribution Modelling using a Multi-Layer Perceptron". *Proceedings of the British Machine Vision Conference*. British Machine Vision Association Press, pp. 107-116, Editor: D. Pycock, Birmingham, UK, 1995.
- [Staib1992] L. Staib, J. Duncan. "Boundary Finding with Parametrically Deformable Models". *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14(11), pp. 1061 –1075, 1992.
- [Stark1996] K. Stark, S. Fuchs. "A method for tracking the pose of known 3-D objects based on an active contour model". *Proceedings of the 13th International Conference on Pattern Recognition*. Vol. 1, pp. 905 -909, 1996.
- [Storvik1994] G. Storvik. "A Bayesian approach to dynamic contours through stochastic sampling and simulated annealing". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 16(10), pp. 976 -986, 1994.
- [Strang1988] G. Strang. *Linear Algebra and Its Applications*. Saunders 1988.

- [Suurkula1997] M. Suurkula, T. Gustavsson. "Contrast-echocardiography and videodensitometry for assessing right ventricular hemodynamics". *IEEE Proceedings of Computers in Cardiology*, pp. 355 -358, 1997.
- [Székely1996] G. Székely, A. Kelemen, C. Brechbühler, G. Gerig. "Segmentation of 3D objects from MRI volume data using constrained elastic deformations of flexible Fourier surface models". *Medical Image Analysis*. Vol. 1(1), pp. 19-34, 1996.
- [Taine1994] M. Taine, A. Herment, B. Diebold, P. Peronneau. "Segmentation of cardiac and vascular ultrasound images with extension to border kinetics". *IEEE Proceedings of the Ultrasonics Symposium*. Vol. 3, pp. 1773-1776, 1994.
- [Terzopoulos1987] D. Terzopoulos, "On matching deformable models to images". Technical Report 60, Schlumberger Palo Alto research, 1986. Reprinted in *Topical Meeting on Machine Vision*, Technical Digest Series, Vol. 12, pp. 160-167, 1987.
- [Terzopoulos1988] D. Terzopoulos, K. Fleischer. "Deformable Models". *The Visual computer*. Vol. 4, pp. 306-331, 1988.
- [Terzopoulos1991] D. Terzopoulos, D. Metaxas. "Dynamic 3D models with local and global deformations: deformable superquadrics". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 13(7), pp. 703-714, July 1991.
- [Terzopoulos1992] D. Terzopoulos, R. Szeliski. "Tracking with Kalman snakes". *Active Vision*. Editors: A. Blake and A. Yuille, MIT Press, Cambridge, MA, Ch.1, pp.3-20, 1992.
- [Terzopoulos1994] D. Terzopoulos, X. Tu, R. Grzeszczuk. "Artificial Fishes: Autonomous Locomotion, Perception, Behavior, and Learning in a Simulated Physical World". *Artificial Life*. Vol. 1(4), 327-351, 1994.
- [Terzopoulos1996] D. Terzopoulos, T. Rabie, R. Grzeszczuk. "Perception and learning in artificial animals". *Artificial Life V: Proceedings of the Fifth International Conference on the Synthesis and Simulation of Living Systems*, pp. 313-320, Nara, Japan, May 1996.
- [Terzopoulos1997] D. Terzopoulos, T. Rabie. "Animat Vision: Active Vision in Artificial Animals". *Videre: Journal of Computer Vision Research*. Vol. 1(1), pp. 2-19, 1997.

-
- [Terzopoulos1999] D. Terzopoulos. "Artificial Life for Computer Graphics". *Communications of the ACM*. Vol. 42(8), pp. 32-42, 1999.
- [Thacker1996] N. Thacker, T. Cootes, "Vision Through Optimization". *British Machine Vision Conference Tutorial Notes*, Edinburgh, September 1996.
- [Tsotsos1980] J. Tsotsos, J. Mylopoulos, H. Covvey, S. Zucker. "A Framework for Visual Motion Understanding". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 2(6), pp. 563-573, 1980.
- [Tu1996] X. Tu. "Artificial Animals for Computer Animation: Biomechanics, Locomotion, Perception, and Behavior". Ph.D. Thesis, Department of Computer Science, University of Toronto, 1996. Also published in *Lecture Notes in Computer Science, ACM Distinguished Theses*, 2000. Vol. 1635, ISBN: 3-540-66939-6.
- [Ullman1984] S. Ullman. "Visual routines". *Cognition*. Vol. 18, pp. 97-159, 1984.
- [Vandenbroucke1997] N. Vandenbroucke, L. Macaire, C. Vieren, J. Postaire. "Contribution of a color classification to soccer players tracking with snakes". *IEEE Proceedings of the International Conference on Systems, Man, and Cybernetics-Computational Cybernetics and Simulation*. Vol. 4, pp. 3660-3665, 1997.
- [Viola1997] P. Viola and W. Wells. "Alignment by maximization of mutual information". *International Journal of Computer Vision*. Vol. 24(2), pp. 137-154, 1997.
- [VIS] Visible Human Project,
<http://www.nlm.nih.gov/research/visible/>.
- [Wang1998] Y. Wang, L. Staib. "Boundary finding with correspondence using statistical shape models". *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 1998*, pp. 338-345, 1998.
- [Wang2000] Y. Wang, L. Staib. "Physical model-based non-rigid registration incorporating statistical shape information". *Medical Image Analysis*, Vol. 4, pp. 7-20, 2000.
- [Widrow1973] B. Widrow. "The rubber mask technique, Part I". *Pattern Recognition*. Vol. 5(3), pp. 175-211, 1973.

- [Xu1998] C. Xu, J. Prince. "Snakes, shapes, and gradient vector flow". *IEEE Transactions on Image Processing*. Vol. 7(3), pp. 359 - 369, March 1998.
- [Xu2000] C. Xu, D. Pham, J. Prince, "Image Segmentation Using Deformable Models". *SPIE Handbook of Medical Imaging. Medical Image Processing and Analysis*. Editors: M. Sonka, J. Fitzpatrick, Vol. 2, Chapter 3, ISBN 0-8184-3622-4, SPIE Press, Vol. PM80, June 2000.
- [Yoshida1997] H. Yoshida, S. Katsuragawa, Y. Amit, K. Doi. "Wavelet-based deformable contour and its application to detection of pulmonary nodules on chest radiographs". *SPIE Proceedings on Wavelet Applications in Signal and Image Processing V*. Editors: A. Aldroubi, A. Laine, M. Unser, ISBN: 0-8194-2591-5, Vol. 3169, pp.328-336, San Diego, USA, 1997.
- [Zhu1995] P. Zhu, P. Chirlian. "On Critical Point Detection of Digital Shapes". *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 17(8), pp. 737-748, 1995.
- [Zhu1996] S. Zhu, A. Yuille. "Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18(9), pp. 884-900, 1996.