SCALE-BASED DECOMPOSABLE SHAPE REPRESENTATIONS FOR MEDICAL IMAGE SEGMENTATION AND SHAPE ANALYSIS

A Dissertation Presented to The Academic Faculty

by

Delphine Nain

In Partial Fulfillment of the Requirements for the Degree Doctor of Philosophy in the College of Computing

Georgia Institute of Technology December 2006

Copyright \bigodot 2006 by Delphine Nain

SCALE-BASED DECOMPOSABLE SHAPE REPRESENTATIONS FOR MEDICAL IMAGE SEGMENTATION AND SHAPE ANALYSIS

Approved by:

Professor Aaron Bobick, Advisor College of Computing Georgia Institute of Technology

Professor Allen Tannenbaum, Advisor School of Electrical and Computer Engineering Georgia Institute of Technology

Professor Greg Turk College of Computing Georgia Institute of Technology Professor Steven Haker Department of Radiology Harvard Medical School

Professor W. Eric. L. Grimson Department of Electrical Engineering and Computer Science *MIT*

Date Approved: 18 October 2006

To my parents,

Marie-Pierre and Philippe.

ACKNOWLEDGEMENTS

Many people need to be thanked for their support, help and enthusiasm for my work throughout my doctorate years. Without you, this thesis would have not been possible. In particular, I would like to thank:

- My advisors, Aaron Bobick and Allen Tannenbaum. Thank you for giving me great academic freedom while always guiding me in the right direction. It has been a privilege to work with you and to learn from you. Your enthusiasm for science, engineering and much more has been inspiring and contagious.
- Prof. Steven Haker, who suggested looking into spherical wavelets and has always been a great source of technical and moral support throughout both my master's and doctorate theses. Thank you for your friendship.
- Profs. Greg Turk and Tony Yezzi, who advised me througout the vessel segmentation work presented in this thesis and other projects. It was a pleasure to work with you, thank you for always being supportive and teaching me the beauty of mathematics and computer graphics.
- Prof. Eric Grimson, my undergraduate and master's advisor, who remained a great mentor throughout my Ph.D. What I learned from you and the medical vision group at MIT has been a tremendous inspiration and help throughout my doctorate years.
- The Biomedical Imaging Lab: Eric Pichon and Marc Niethammer, my co-jedimasters (I will miss our philosophical discussions whether it is about politics or shape analysis and many fun trips), Amanda Wake (thanks for helping us break down cubicle barriers, and being a great friend and presence in lab), Patricio (I

have benefited much from your energy and enthusiasm for the many facets of life), Lei, Yan, Gallagher, Jimi, Eli, Oleg, Yogesh, Ramsey, Sam, Julien, Xavier, John, Tauseef, Shawn, Yi and friends of the lab from whom I have learned so much and always bring a smile to my face.

- The CPL, geometry and GVU labs. You have given me great examples to follow and made these last four years a lot of fun.
- The women@cc group. It has been a pleasure to get to know all of you, thank you for all the advice and support.
- NA-MIC collaborators. It has been inspiring and rewarding to work with all of you and always learn more about algorithms, engineering and medicine.
- The Sam Nunn/MacArthur Security Program.
- Gozde Unal and Lauren O'Donnell who has been great mentors throughout my academic years and dear friends.
- My Pioux, MIT/Boston, Georgia Tech/Atlanta friends, thank you for always being there for me.
- My family: you are the best one could hope for. To my grand-parents, my parents and my brother: thank you for your unconditional love and support.
- Omar Zurkiya: thank you for sharing this journey with me. I look forward to the many more adventures to come.

TABLE OF CONTENTS

DED	ICAT	FION .		iii
ACK	NOW	VLEDG	EMENTS	iv
LIST	OF	TABLE	ES	х
LIST	OF	FIGUR	RES	xi
SUM	IMAF	RY		xvii
Ι	INT	RODU	CTION	3
	1.1	Shape	Representation and Segmentation Framework	6
		1.1.1	Shape Representation	7
		1.1.2	Probabilistic Segmentation using Deformable Models	7
		1.1.3	Adding prior shape information to deformable models \ldots	9
	1.2	Challe	enges addressed by this Thesis	10
	1.3	Contri	ibution of this Thesis	13
II	BAC Mei	CKGRC DICAL	OUND: SHAPE REPRESENTATION AND SHAPE PRIORS FO IMAGE SEGMENTATION)R 15
	2.1	Overv	iew	15
	2.2	Shape	Characterization	15
		2.2.1	Two Dimensional Parametric Shape Representation	16
		2.2.2	Three Dimensional Parametric Shape Representation	19
		2.2.3	Adding Data-Driven Shape Priors to Parametric Models	25
		2.2.4	2D and 3D Implicit Shape Representation	29
		2.2.5	Shape Priors for Implicit Representations	30
	2.3	Deform	mable Models for Probabilistic Segmentation	31
		2.3.1	Parametric Deformable Models	32
		2.3.2	Geometric Deformable Models	38
		2.3.3	Numerical Implementation: level set method	44
		2.3.4	Discussion	46

III	SOF DRI	T SHA VEN F	PE PRIORS: VESSEL SEGMENTATION USING A SHAPE LOW	49
	3.1	Blood	Vessel Segmentation	49
	3.2	Relate	d Work	50
	3.3	Shape	Driven Flow	52
		3.3.1	Region Based Flow	52
		3.3.2	Shape Filters	54
		3.3.3	Curve Evolution using Local Filters	56
	3.4	Impler	nentation	57
		3.4.1	Numerical Methods	57
		3.4.2	Optimizations	59
	3.5	Result	S	59
		3.5.1	2D Images	61
		3.5.2	3D Images	63
	3.6	Conclu	usion	64
IV	MUI	LTI-SC.	ALE 3D SHAPE ANALYSIS USING SPHERICAL WAVELETS	66
	4.1	Motiva	ation for a multiscale parametric shape representation \ldots	69
		4.1.1	Implicit versus Parametric representation	69
		4.1.2	Statistical Modeling of a population of shapes with paramet- ric representation	70
		4.1.3	Statistical Modeling using PCA and its limitations	71
		4.1.4	Related Work	81
		4.1.5	Our Contributions	82
	4.2	Spheri	cal wavelets	83
		4.2.1	Grid Structure for Spherical Wavelet Construction	86
		4.2.2	Scaling Function	87
		4.2.3	Wavelet Function	88
	4.3	Shape	Representation using spherical wavelets	90
		4.3.1	Data Description	91

		4.3.2 Shape Remeshing and Registration
		4.3.3 Spherical Wavelets on the Mean Shape
		4.3.4 Encoding the Shape Signal with Spherical Wavelets 99
		4.3.5 Filtering the Shape Signal by Projection onto a Reduced Set of Basis functions
	4.4	Scale-Space Spherical Wavelet Prior for Statistical Population Analysis101
		4.4.1 Coefficient Truncation via Power Analysis
		4.4.2 Multiscale Decomposition
	4.5	Experiments
	4.6	Conclusion
V	3D S PRI	SEGMENTATION USING THE MULTI-SCALE SPHERICAL WAVELETIOR 12^4
	5.1	Related Work $\ldots \ldots 12^{4}$
	5.2	Pose Parameters
	5.3	Shape Parameters
	5.4	Segmentation Energy
	5.5	Parameter Optimization via Multiresolution Gradient Descent 128
	5.6	Results of Segmentation
	5.7	Conclusion
VI	STA ICA	ATISTICAL SHAPE ANALYSIS OF CAUDATE USING THE SPHER- L WAVELET SHAPE REPRESENTATION
	6.1	Related Work
	6.2	Methods
		6.2.1 SPHARM-PDM
		6.2.2 Spherical Wavelet Shape Representation (SWC) 140
		6.2.3 Shape Analysis $\ldots \ldots 140$
		6.2.4 Significance Map Visualization
	6.3	Results
		6.3.1 Left Caudate Nucleus

		6.3.2	Right Caudate Nucleus	144
		6.3.3	Left Hippocampus	144
		6.3.4	Right Hippocampus	145
	6.4	Conclu	sion	145
VII	CON	ICLUS	ION	154
APP	END	IX A	APPENDIX	157
REF	EREI	NCES .		162
VITA	A.			168

LIST OF TABLES

1	Properties of parametric and geometric active contours. Adapted from $\left[1\right.$	5] 47
2	Number of truncated basis functions and the average maximum error between original shapes and their filtered version, using a filter matrix specific to the shape population, for both the hippocampus and caudate	
	dataset.	105
3	H, H_f distance for 5 test shapes for Mscale and ASM	135

LIST OF FIGURES

1	Three standard orientations shown with MRI brain slices and a 3D transparent model of the human head	3
2	To visualize the three-dimensional information in an MRI volume, the physician can visualize an ordered sequence of slices. In this figure, sagittal slices are displayed	4
3	Segmented White Matter from a MRI Brain Scan. The top window shows the three-dimensional model superimposed on a coronal slice. The output of a segmentation is also labelmap where voxels that belong to the interior or the boundary of structure are labeled in pink in the bottom three windows (respectively axial, sagittal and coronal views)	4
4	Framework for Shape Representation and Probabilistic Segmentation with a shape prior. Solid boxes represent tasks, dashed lines represent categories of techniques to accomplish task.	6
5	Framework for Shape Representation and Probabilistic Segmentation without a shape prior. Solid boxes represent tasks, dashed lines represent categories of techniques to accomplish task.	8
6	Framework for Shape Representation and Probabilistic Segmentation with a shape prior. Solid boxes represent tasks, dashed lines represent categories of techniques to accomplish task.	16
7	Visualization of the spherical harmonic functions on the sphere. A darker color shows where the function has support. The figure shows the real parts of the spherical harmonic function Y_l^m , with l growing from 0 (top) to 5 (bottom), and m ranging from 0 (left) to l in each row (from [23])	22
8	Visualization of the spherical harmonic correspondence. A first order ellipsoid and six left ventricles are displayed. The surface net shows the (θ_i, ϕ_i) parametrization (same parameters = same homologous points). The ridges on the first order ellipsoid are the equator and $0, \pi/2, \pi, 3\pi/2$ meridian lines in all objects. The equator and meridian lines are emphasized in different colors. The poles are at the crossing of the meridian lines (from [23])	23
9	Visualization of a lateral ventricle (side view) at different spherical harmonics degrees; $m=1,4,8,12$ top to bottom (from [23])	23
10	Outline of several brain structures in a single model labeled with land- marks (from [11])	27

11	In ASM, the learned shape variation can be shown by selecting one of the eigenvectors (modes of variation) and varying its magnitude from $-n\sqrt{\sigma}$ to $n\sqrt{\sigma}$ with $n = 2$ and adding the result to the mean shape to display the variation(from [11])	27
12	Level Sets of an embedding function ϕ for a closed curve in \mathbb{R}^2	29
13	An example of a snake active contour (a) Intensity CT image of the heart left ventricle (b) edge detected image (c) initial contour (d) deformable contour moving toward the left ventricle boundary, driven by an inflating pressure force (from [41])	33
14	Ventricle segmentation in MRI heart image via shrinking conformal active contour.	41
15	Bone segmentation in CT image with splitting shrinking conformal active contour.	42
16	Four steps in the segmentation of two different corpora callosa. The last image in each case shows the final segmentation in red. The cyan contour is the standard evolution without the shape influence (from [36])	45
17	A Region Based Geometric Active Contour is prone to leak if the image term is not reliable.	54
18	(a) $\epsilon_1(\mathbf{x}, r)$ is the intersection of the ball of radius r centered at \mathbf{x} and the region R inside the contour. (b) Points inside the widening region have a higher ϵ_1 measure but most points on the contour have the same measure. (c)Contour points close to the widening of the contour have a higher ϵ_2 measure $\ldots \ldots \ldots$	55
19	Dilation Flow without Shape Prior	60
20	Dilation Flow with Shape Prior	60
21	Flow with Shape Prior on 2D Synthetic Images	61
22	Vessel Flow on Angiogram Images	62
23	Vessel Flow on Angiogram Images	62
24	Different Flows on the first CT Coronary Data	63
25	Base Flow with Shape Prior on the second CT coronary dataset	64
26	(a-b) Coronal and Sagittal view of the left caudate nucleus (c-e) Example of 3 shapes from the left caudate nucleus dataset	67
27	(a-b) Coronal and Sagittal view of the left hippocampus (c-e) Example of 3 shapes from the left hippocampus dataset	68

28	Shape Model	71
29	Synthetic Shape Global Deformations	74
30	Synthetic Shape Local Deformations	75
31	Synthetic Shapes	76
32	Visualization of the deformation of the mean shape along the eigenvectors found by PCA. The transparent shape is the reference mean shape with no deformation. For each figure, the mean shape is deformed along a particular eigenvector (or mode) according to a certain magnitude.	77
33	Reconstruction experiment for Test Shape 1 using PCA as a shape space compared to WDM (technique presented in this chapter)	78
34	Reconstruction experiment for Test Shape 2, using PCA as a shape space compared to WDM (technique presented in this chapter)	79
35	A wavelet transform is applied to the parametric functions represent- ing a deformable contour. The resulting wavelet coefficients are then grouped into 64 bands, and the joint probability function of each band is estimated from the available training samples, via its mean and co- variance matrix. This effectively transforms the covariance matrix of the full joint probability function into a matrix that is close to, but not necessarily exactly, a block diagonal matrix. The submatrix corre- sponding to band B reflects global shape characteristics, whereas the submatrix corresponding to B reflects local shape characteristics of a neigh- boring segment	80
36	Recursive Partitioning of an icosahedron: successive levels of the tri- angulation are obtained by subdividing triangles into 4 children triangles	83
37	The wavelet transform of a random scalar signal defined on spherical mesh. The mesh is an icosahedron subdivided 3 times. The original signal is decomposed into a series of high-pass (HP) and low-pass (LP) coefficients.	84
38	Visualization of Scaling and Wavelet Basis Functions. The color corre- sponds to the value of the function at each point on the sphere. Notice that the support of each basis gets smaller as the resolution increases	86
39	Refinement of the surface grid by subdivision of the edges	87
40	Example of 5 shapes from the Prostate dataset	91
41	Illustration of the remeshing step for two left hippocampus shapes. See Section 4.3.2 for details.	92

42	Example of principal axes and 3 points \mathbf{x}_N , \mathbf{x}_S and \mathbf{x}_E found on three left hippocampus and left caudate shapes.	93
43	A new point \mathbf{x}_S is found such that it is the farthest point on the mesh from \mathbf{x}_N (in geodesic distance) for the three left hippocampus and left caudate shapes shown in Figure 42	94
44	After remeshing, shapes are aligned with Procrustes alignment and the Mean shape is computed for (a-c) left hippocampus dataset (d-f) left caudate dataset	98
45	(a-c) Icosahedron mesh shown with the Mean shape(d-f) Visualization of wavelet basis functions constructed on the Mean shape at various levels. The color corresponds to the value of the functions $\ldots \ldots$	99
46	Example of the filtering operation for Shape 1 of Left Caudate (a-d) and Left Hippocampus (e-h) dataset.	102
47	Result of the Filtering Operation to create a reduced basis set. (a,c) original shapes (b,d) filtered shapes with mean squared error from the original shape as colormap	105
48	Shape Representation from Sections 4.3-4.4.2	106
49	Illustration of Band Creation Algorithm	108
50	Coefficient Clustering and selected Band Variation Visualization for the left hippocampus data	111
51	Coefficient Clustering and selected Band Variation Visualization for the left caudate data	112
52	Band Decomposition: various bands $B_{j,i}$, where j is the resolution and i is the band number, shown in Anterior view (A) and Posterior view (P), see text for color	113
53	Band Decomposition, Caudate Nucleus dataset: various bands shown for scale 2 (shown in Anterior view and Posterior view), see text for colo	r114
54	Max Squared Reconstruction Error (averaged over testing shapes) for various training set sizes	118
55	Reconstruction Task for a test shape using 10 training shapes (first 2 columns) and a test shape using 20 training shapes (last 2 columns) .	119
56	Prostate dataset: Ground Truth Test shape, Test shape with noise, and reconstruction with PDM and WDM with scale and bands with 5 training samples . Color is error from blue (lowest) to red	120

57	Caudate dataset: Ground Truth Test shape, Test shape with noise, and reconstruction with PDM and WDM with scale and bands shape priors with 25 training samples . Color is error from blue (lowest) to red	121
58	Surface Evolution using the Ground Truth label-map as the image force for the ASM algorithm. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale).	130
59	Surface Evolution using the Ground Truth label-map as the image force for the Mscale algorithms. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale).	131
60	Surface Evolution using the density estimation as the image force for ASM. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale)	132
61	Surface Evolution using the density estimation as the image force for Mscale (bottom row) algorithms. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale)	133
62	Visualization of spherical wavelet functions and associated membership regions at three levels (columns). Top row : Values of single spherical Wavelet Basis Function shown on the sphere at scales 1 through 3. Middle and Bottom row : Membership regions of spherical wavelet basis functions shown on the sphere and on the original surface, color- ing is random	141
63	Left Caudate Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color	146
64	Left Caudate Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color	147
65	Right Caudate Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color	148
66	Right Caudate Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color	149
67	Left Hippocampus Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color.	150
68	Left Hippocampus Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color.	151

69	Right Hippocampus Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color.	152
70	Right Hippocampus Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color.	153

SUMMARY

In this thesis, we propose and evaluate two novel scale-based decomposable representations of shape for the segmentation and shape analysis of anatomical structures in medical imaging. We propose two representations that are adapted to a particular class of anatomical structures and allow for a richer description and a more finegrained control over the deformation of models based on these representations, when compared to previous techniques. In particular, the decomposition of these shape representations can be localized both in space and in scale, enabling the construction of more descriptive, non-global, non-uniform shape priors to be included in the segmentation framework. For each representation, we derive a segmentation algorithm using the parameters of the shape representation to easily include and benefit from the prior in the optimization framework.

We first review the two main classes of shape representation, parametric and implicit models, and discuss the impact of existing representations on the construction of data-driven or knowledge-driven shape priors and segmentation algorithms. In particular, we note that most shape priors fall either under the category of global priors, constraining the full shape model to a predefined shape space, or very local priors, constraining the smoothness of the model on a very local level. This thesis addresses the gap between the two categories by proposing and evaluating two novel multi-scale shape representations and shape probability priors adapted to particular classes of anatomical shapes.

The first novel shape probability prior proposed by this thesis is a knowledgedriven semi-local prior using local shape filters that measure shape properties for

xvii

implicit shape representations. We use these filters for the segmentation of blood vessels, and introduce the notion of segmentation with a *soft* shape prior, where the segmented model is not globally constrained to a predefined shape space, but is penalized locally if it deviates strongly from a tubular structure. We introduce the concept of a scale-space shape filter that measures the deviation from a tubular shape in a local neighborhood of points, given a particular scale of analysis. Using this filter, we derive a region-based active contour segmentation algorithm for tubular structures that penalizes leakages. We present results on synthetic and real 2D and 3D datasets.

The second novel shape representation proposed by this thesis is a multi-scale parametric shape representation using spherical wavelets. This work is motivated by the need to compactly and accurately encode variations at multiple scales in the shape representation in order to drive the segmentation and shape analysis of deep brain structures, such as the caudate nucleus or the hippocampus. Our proposed shape representation can be optimized to compactly encode shape variations in a population at the needed scale and spatial locations, enabling the construction of more descriptive, non-global, non-uniform shape probability priors to be included in the segmentation and shape analysis framework. We apply this representation to two important medical imaging tasks, segmentation and shape analysis. For segmentation, we derive a parametric active surface evolution using the multiscale prior coefficients as parameters for our optimization procedure to naturally include the prior for segmentation. Additionally, the optimization method can be applied in a coarse-to-fine manner. Our results on synthetic and real data show that our algorithm is computationally efficient and outperforms the Active Shape Model (ASM) algorithm, by capturing finer shape details. For shape analysis, we use the coefficients of our shape representation as features to describe a population of shapes and perform hypothesis testing using an existing non-parametric permutation testing technique. The technique tests for shape differences in the caudate brain structure among two population of patients with and without schizo-typal disorder. Our results show that significant differences are found among the group of patients, and in particular the spherical wavelet representation finds a greater number of locations on the shapes that exhibit a statistical shape difference among the shapes, when compared to a point distribution representation of the shapes.

Roadmap

In chapter 1, we present an overview of shape representation, segmentation and shape analysis for medical imaging. We discuss the challenges addressed by this thesis and the main contributions of this thesis.

In chapter 2 we present in greater detail existing work for the topics of shape representation, shape priors and deformable models for medical image segmentation. We present two main shape representations: parametric models and geometric models, for both curves in \mathbb{R}^2 and surfaces in \mathbb{R}^3 . For parametric models, we focus in particular on shape representations using basis functions. We then discuss the types of knowledge-driven and data-driven shape priors that can be learned from existing shape representations. Finally, we give an overview of the theory of segmentation using active contours and active surfaces, both for the parametric and geometric models.

In chapter 3, we present a segmentation method for blood vessels using an implicit deformable model with a knowledge-driven localized shape prior. We combine image statistics and scale-space shape information in a variational framework to derive a region-based active contour that segments tubular structures and penalizes leakages. We present results on synthetic and real 2D and 3D datasets.

In chapter 4, we present a novel algorithm that learns the shape variation at *mul*tiple scales and locations from a training set. Our technique uses spherical wavelets to generate a multi-scale description of surfaces and spectral graph partitioning to adaptively discover independent shape variations at multiple scales. We compare our technique to the standard Point Distribution Model (PDM) shape prior and present results on synthetic datasets, as well as 3D brain and prostate datasets.

In chapter 5, we present a parametric active surface evolution using the multiscale shape prior presented in chapter 4 as parameters for our optimization procedure to naturally include the prior for segmentation. We compare our technique to the standard Active Shape Model (ASM) algorithm and present results on synthetic datasets, as well as 3D brain and prostate datasets.

In chapter 6, we present a statistical shape analysis application of the spherical wavelet shape representation to detect shape differences in the caudate nucleus between healthy patients and patients with schizotypal personality disorder. We compare the test results using the spherical wavelet representation (SWC) to the test results using a point distribution model (PDM) representation and show that the SWC verifies previous findings by the PDM technique, as well as finds more areas of the shape that appear to differ among groups.

In the Conclusion, we summarize and discuss the contributions and results of this thesis and in particular focus on challenges that remain to be addressed. We also suggest future research directions based on the techniques and frameworks laid out in this thesis.

CHAPTER I

INTRODUCTION

Medical Imaging has rapidly become a useful tool for physicians to peek noninvasively into the human body and diagnose, treat and track the progress of a disease. The goal of the medical imaging community is to assist physicians by extracting, with the assistance of computers, clinically useful information about anatomical structures digitally imaged through CT, MRI and other modalities in an efficient, repeatable and accurate manner [4, 41].

A medical scanner often outputs a three-dimensional volume that represents a portion of the human body. The volume can then be visualized as a sequence of 2D images, by "slicing" the volume along a chosen direction. Standard medical directions are Axial, Sagittal and Coronal, shown in Figure 1. Figure 2 shows a typical sequence through an MRI volume of a human head in a sagittal direction. The size of the volume is $256 \times 256 \times 124$ voxels. By scrolling through the slices, physicians can create a mental 3D reconstruction of the body part being imaged. Often, however, it is useful to be able to visualize particular structures directly in three dimensions, to



(a) Axial Slice



(b) Sagittal Slice



(c) Coronal Slice

Figure 1: Three standard orientations shown with MRI brain slices and a 3D transparent model of the human head



Figure 2: To visualize the three-dimensional information in an MRI volume, the physician can visualize an ordered sequence of slices. In this figure, sagittal slices are displayed



Figure 3: Segmented White Matter from a MRI Brain Scan. The top window shows the three-dimensional model superimposed on a coronal slice. The output of a segmentation is also labelmap where voxels that belong to the interior or the boundary of structure are labeled in pink in the bottom three windows (respectively axial, sagittal and coronal views)

directly observe the shape of the structure, as shown in Figure 3, where the segmented white matter from an MRI Brain Scan is displayed in the top window. In this thesis, we will describe in detail two important medical imaging tasks, segmentation and shape analysis [2, 26]. Segmentation is a process to detect anatomical structures in medical scans. Shape Analysis is a process to locate morphological changes between healthy and pathological anatomical structures. Segmentation is therefore a precursor to shape analysis since structures have to first be extracted before they can be further analyzed and compared.

For segmentation and shape analysis to be carried out, a fundamental task is to establish a **shape representation** in order to characterize useful properties of the structure under study. To characterize a **class** of anatomical structures, a probability function can be learned over such a representation using a collection of representative structures. The probability function can then be included as a **probability prior** in a probabilistic segmentation framework to overcome imaging limitations, such as noise. A representation that is both expressive and specific to a structure is also useful to increase the accuracy of shape analysis.

In this thesis, we propose and evaluate two novel scale-based decomposable representations of shape for the segmentation of blood vessels and the segmentation and shape analysis of brain structures. We propose two representations that are adapted to the class of anatomical structures under study. For blood vessel characterization, we present a novel scale-space shape filter that measures the deviation from a tubular shape in a local neighborhood of points, given a particular scale of analysis. For deep brain structure characterization, we propose a multi-scale parametric shape representation using spherical wavelets that can be optimized to compactly encode shape variations in a population at the needed scale and spatial locations. We demonstrate that learning a prior over those representations allow for a richer description and a more fine-grained control in segmentation and shape analysis tasks, when compared



Figure 4: Framework for Shape Representation and Probabilistic Segmentation with a shape prior. Solid boxes represent tasks, dashed lines represent categories of techniques to accomplish task.

to previous techniques. In particular, the decomposition of these shape representations can be localized both in space and in scale, enabling the construction of more descriptive, non-global, non-uniform shape priors. For each representation, we derive a segmentation algorithm using the parameters of the shape representation to easily include and benefit from the prior in the optimization framework. We also conduct shape analysis using the spherical wavelet representation to detect morphological differences between patients diagnosed with schizophrenia and healthy controls.

1.1 Shape Representation and Segmentation Framework

Figure 4 presents a general framework and categorizes techniques used in the medical imaging community for shape representation and probabilistic deformable model segmentation. In the remainder of this introduction we give a high level overview of each task presented in this framework and detail the specific contributions of this thesis for each task. Chapter 2 further explains the framework from Figure 4 and presents related work in the medical imaging literature. Chapters 3-5 present our novel contributions for shape representation and segmentation. Chapter 6 presents an application of our multiscale shape representation to shape analysis.

1.1.1 Shape Representation

Shape representation is a key first step in many medical image processing tasks. The chosen parameters to represent shape create a mathematic model that is used for subsequent tasks. In Figure 4, the shape is denoted by S and is described by a set of parameters $\alpha, \beta, \gamma, \dots$ The researcher has to carefully pick the parameters to balance expressiveness (how well can all the shapes in a population be represented with the shape parameters), complexity (the number of parameters) and specificity (whether the parameters are specific enough to the shape population being studied). Parametric representations are models that describe a shape with a set of parameters. For example, a circle can be represented parametrically with two independent parameters: the position of its center and its radius. **Implicit representations** are models that describe a shape with a scalar function, where the shape boundary is represented implicitly as a level set of the function. For example, a circle in two-dimensions can be represented implicitly by the equation $(x - p_x)^2 + (y - p_y)^2 - r^2 = 0$, where the center of the circle is at position (p_x, p_y) , its radius is r and the circle boundary is the set of (x, y) points in \mathbb{R}^2 that verify the previous equation. In Chapter 2, we present various representations from each category.

1.1.2 Probabilistic Segmentation using Deformable Models

Traditional segmentation techniques, such as edge detection and thresholding techniques, often fail to locate the object boundary or generate disconnected boundaries or invalid boundaries that must be refined in a post-processing step. This is due to imaging limitations, such as noise and sampling artifacts, or due to the fact that many structures do not have a distinctive intensity range or distinctive connected



Figure 5: Framework for Shape Representation and Probabilistic Segmentation without a shape prior. Solid boxes represent tasks, dashed lines represent categories of techniques to accomplish task.

edges [41]. To address these difficulties, deformable models have been extensively studied and widely used in medical image segmentation, with good results [71, 2]. In deformable models, the boundary of the structure to be segmented is explicitly encoded in a model. The model is deformed to fit the image data with constraints on the topology of the boundary (usually that the boundary remains closed), overcoming previous limitations.

There are two types of deformable models, based on the object representation used for the model: parametric deformable models and geometric deformable models. **Parametric** models are compact representations that are computationally efficient. However changes in model topology during the deformation, such as splitting and merging, need to be detected and require special handling. **Geometric** deformable models that use an implicit shape representation have a higher computational cost, but topological changes during the evolution do not need to be detected and are automatically handled. The segmentation techniques described in this thesis are part of a probabilistic segmentation framework using deformable models. If S is a shape representation and D is the image data, the segmented shape is the maximum a posteriori estimate $S^* = \arg \max_S P(S|D)$. Assuming that P(D) is uniform, the posterior can be expressed as the product of the likelihood P(D|S) and the model prior P(S): $P(S|D) \propto P(D|S)P(S)$. As will be further explained in Chapter 2, the maximum a posteriori formulation can be translated into an energy minimization formulation where the shape parameters are evolved via a gradient descent to fit the image data. A simplified view of probabilistic segmentation is given in Figure 5, where the prior probability of shape is uniform. The case with non-uniform probability is explained in the next section.

1.1.3 Adding prior shape information to deformable models

In medical images, the general topology, shape, location and orientation of anatomical structures to be segmented are either known, or can be learned prior to the segmentation task. For this reason, object segmentation with deformable models and statistical shape modeling are often combined to obtain a more robust and accurate segmentation [11, 36, 64, 71, 41]. Statistical shape modeling is the process of estimating a priori the probability function of the shape to be segmented, P(S), from a collection of representative shapes, called a training set. This additional step is shown in Figure 4. Statistical shape information can guide the model fitting procedure in the presence of noise since only shapes that have been previously seen in a training set have a high probability (or low energy) during the optimization procedure.

There exists two types of shape priors: knowledge-driven and data-driven.

Knowledge-driven shape priors are derived from human knowledge of what a shape should "look like" and translate this knowledge to a constraint on the parameters of the model. One of the most commonly used knowledge-driven priors is the local smoothness constraint applied to curves or surfaces. This smoothness prior can be applied by minimizing the curvature, or second derivative, of the curve by expressing curvature in terms of the model parameters. Unless the parameters of the shape representation have a particular meaning for the shape of the object (such as the radius parameter for a sphere), it is often hard to know a priori how to constrain the parameters of the shape representation for a particular class of objects. Therefore in practice, the smoothness prior is a common knowledge-driven prior used both for parametric and geometric deformable models. We also note that choosing a shape representation for deformable models is itself a knowledge-driven prior since the type of representation chosen (parametric or implicit), as well as the type and number of parameters, affects the initial topology, possible topological changes and expressiveness of the shape model.

A **data-driven** shape prior can be used to represent a class (or population) of shapes by learning the distribution of the shape parameters from a training set. The prior represents the generic shapes of the object belonging to the class, as well as the possible shape variations within the class, while excluding variations not previously seen in the training set. The most common approach is to assume a multivariate Gaussian probability function of the model parameters and learn the mean and major modes of variation of the training set by using principal component analysis. This prior is fully global, constraining all shape parameters.

1.2 Challenges addressed by this Thesis

In this thesis, we develop targeted shape representations for two important classes of anatomical structures in medical imaging and derive probabilistic segmentation and shape analysis algorithms based on these representations. The first class of shapes that we study are two-dimensional or three-dimensional tubular anatomical structures with branches, such as blood vessels. Blood vessel segmentation and visualization of blood vessels are important for clinical tasks such as diagnosis of vascular diseases, surgery planning and blood flow simulation. The second class of shapes are threedimensional anatomical structures with genus zero ¹ and high levels of detail, such as deep brain structures. Segmentation and shape analysis of deep brain structures are important for the study and diagnosis of neurological disorders.

The particular challenges we face with such structures are:

- Vessel structures do not have a fixed topology. Vessels have many branches, and the number and location of the branches are patient specific. Therefore a geometric active contour for segmentation is desirable since the topology is not known in advance. However, using image intensity alone to deform the contour often results in leakages at areas where the image information is ambiguous, motivating the need to use a shape prior. The issue is that to date only strictly local or fully global shape priors have been developed for geometric active contours. A global data-driven shape prior is not desirable since there does not exist a known a priori shape. A very local shape prior that imposes smoothness of the contour is not constraining enough to prevent leaks.
- Deep brain structures have fixed topology, therefore a parametrized active contour is desirable due to its efficiency. Again, image information alone is not reliable for brain segmentation motivating the need to incorporate a shape prior in the segmentation. Given the complexity of the surface, a data-driven prior is preferable over a knowledge-driven one. The surface contain high frequency information that varies from patient to patient therefore the shape prior needs to encode high frequency variations at specific locations. Existing data-driven shape priors favor the encoding of global (low frequency) information, particularly if they have been trained on a limited number of samples. This means that

¹shapes that do not have holes or handles, shapes that are equivalent to a sphere topologically

high frequency information on the surface of the shapes (such as small bumps or sharp edges) are not encoded in the prior representation.

Most shape priors used for deformable models fall either under the category of global priors, constraining the full shape model to a predefined shape space, or very local priors, constraining the smoothness of the model on a very local level. This thesis addresses the gap between the two categories by proposing and evaluating multi-scale shape representations that enable the estimation of shape priors that span the range of scales, from local to global.

The notion of multi-scale has been developed by the computer vision community to decompose image data in order to control the scale of observation, using for example quadtrees or image pyramids [63]. Similarly an object boundary can be encoded as a 2D or 3D signal that can be decomposed and analyzed at different scales. The notion of scale is coupled with the location and spatial support of a filter used to analyze the signal. At the coarsest scale of analysis, the filter has global support and is applied to the whole object boundary. As the scale becomes finer, the spatial support of the filter is decreased so that the filter is only applied to a segment of the object boundary. This decomposition reveals different information content of the shape signal depending on the scale, information not readily available from looking at the original signal. Filters at a coarse scale usually analyze the global, low-frequency content of the signal (global features) while filters at the finest scale analyze the local, high frequency content (local features). This decomposition enables shape representations and shape priors to encode information at various scales.

In this thesis, we propose to use operators and basis functions that can decompose shape information both in space and in scale, enabling the construction of more descriptive, non-global, non-uniform shape priors to be included in the segmentation framework.

1.3 Contribution of this Thesis

The contributions of this thesis are:

- A novel knowledge-driven prior based on local shape filters for implicit representations of shape. We use these filters for the segmentation of blood vessels, and introduce the notion of segmentation with a **soft** shape prior, where the segmented model is not globally constrained to a predefined shape space, but is penalized locally if it deviates strongly from a tubular structure. We introduce the concept of a scale-specific shape filter that measures the deviation from a tubular shape in a local neighborhood of points, given a particular scale of analysis. Using this filter, we derive a region-based active contour segmentation algorithm for tubular structures that penalizes leakages. We present results on synthetic and real 2D and 3D datasets where we show that our technique can prevent leaks in the segmentation when compared to a traditional region-based active contour segmentation technique.
- A novel multi-scale shape representation using spherical wavelets for parametric models from which a data-driven prior can be learned to segment shapes with fine variations, such as deep brain structures. This novel representation is motivated by the observation that the classical active shape models prior (ASM) technique cannot accurately encode fine, localized shape variations when the training set is of moderate size, since the relatively few eigenvectors will represent the most global modes of variation in the shapes. To address this issue, we present a novel algorithm that learns shape variations from data at *multiple scales and locations* using the spherical wavelet representation and spectral graph partitioning. Our results show that for a given training set size, our algorithm significantly improves the approximation of shapes in a testing set over ASM, even in the presence of noise.

- The derivation of a parametric active surface evolution equation using the multiscale prior coefficients as parameters for the optimization procedure to naturally include the prior for segmentation. The advantage of such a segmentation framework is that the spherical wavelet shape prior can directly be used to constrain the parameters during the surface evolution. Additionally, the optimization method can be applied in a coarse-to-fine manner. Our validation on caudate datasets in brain MRI shows that our algorithm is computationally efficient, qualitatively captures finer shape details and quantitatively outperforms the Active Shape Model (ASM) algorithm when measuring the distance of the segmentation to the ground truth using the Hausdorff distance.
- Shape analysis using spherical wavelets. We use the spherical wavelet coefficients of our shape representation to locate morphological changes in the caudate nucleus brain structure between healthy control patients and patients diagnosed with schizo-typal personality disorder (SPD). The hypothesis testing is conducted using a non-parametric permutation testing technique with correction for multiple comparisons. Our results show that statistically significant differences are found among the group of patients (p<0.05), and in particular the spherical wavelet representation finds a greater number of locations on the shapes that exhibit a statistical shape difference among the shapes, when compared to a point distribution model (PDM) that represents shape at a single scale of analysis

Our conclusion discusses remaining challenges and extensions of the representations and algorithms presented in this thesis, and discusses future work directions.

CHAPTER II

BACKGROUND: SHAPE REPRESENTATION AND SHAPE PRIORS FOR MEDICAL IMAGE SEGMENTATION

2.1 Overview

This chapter gives an overview of existing techniques for shape representation and segmentation used in the framework shown in Figure 6.

We focus in particular on reviewing existing shape representations and their impact on the construction of data-driven or knowledge-driven shape priors in Section 2.2. The description of shape priors is of particular importance as a background to our work.

We then present deformable model segmentation algorithms. We present two types of deformable models based on the chosen shape representation: parametric models in Section 2.3.1 and geometric models in Section 2.3.2. Geometric models will be used in in our vessel segmentation framework presented in Chapter 3 and parametric models will be used in our multiscale shape representation presented in Chapter 4 and segmentation using the multiscale shape representation in Chapter 5.

2.2 Shape Characterization

In a model-driven segmentation framework, it is necessary to have a mathematical description of the boundary of an object. Two main approaches exist: to represent the boundary explicitly in a parametric form, or implicitly as the level set of a higherdimensional function. We describe both approaches in this section. We note that this thesis only focuses on closed curves for 2D boundaries and simply-connected



Figure 6: Framework for Shape Representation and Probabilistic Segmentation with a shape prior. Solid boxes represent tasks, dashed lines represent categories of techniques to accomplish task.

surfaces (surfaces with no holes) for 3D boundaries. See [45] for a treatment of open curves/open surfaces.

2.2.1 Two Dimensional Parametric Shape Representation

Parametric models represent curves explicitly with a set of ordered control points located on the boundary of the curve. The points are identified with a parameter pvarying from 0 to 1. Mathematically, the curve is represented as a collection of control points $\overrightarrow{C}(p): [0,1] \to \mathbb{R}^2$ such that $\overrightarrow{C}(p) = (x(p), y(p))$. If one assumes that it takes constant time to travel from one control point to another, a special parametrization is arc-length parametrization (denoted s in this thesis), where the distance traveled between points is equal (hence the curve is traversed at constant speed). In this case, the parameter s varies from 0 to L, where L is the total length of the curve. However with parametrized models, it is often the case that the distance between each control point is not equal, and we denote the arbitrary parametrization with p that varies





(a) A parametrized polygonal boundary (b) The snake model is a parametrized closed boundary: it is a spline parametrized by a set of node points in \mathbb{R}^2

between 0 and 1 (see Figure 7(a)).

To represent a closed curve with control points, the simplest technique is to use a polygonal approximation, where control points are vertices connected by straight lines (segments), as shown in Figure 7(a).

A more sophisticated and visually pleasing representation is to use a piecewise polynomial interpolation between control points (see Figure 7(b)). B-splines representation is an example of using a linear combination of local, continuous bases functions to represent a piecewise polynomial curve. For a n^{th} order B-spline, the bases functions are n^{th} order polynomials (with the first n - 1 derivatives continuous). Let $\mathbf{x}_i = (x(i), y(i)), i = 1, ..., n$ be control points of a curve $\overrightarrow{\mathcal{C}}(p)$, and pbe a linearly increasing parameter that has integer value at control points, that is $\mathbf{x}_i = \overrightarrow{\mathcal{C}}(i)$. The curve $\overrightarrow{\mathcal{C}}(p)$ is given by:

$$\overrightarrow{\mathcal{C}}(p) = \sum_{i=0}^{n+1} \mathbf{v}_i B_i(p) \tag{1}$$

where \mathbf{v}_i are coefficients and B_i are bases functions whose shape is given by the spline order. Third order splines are most common because they are the lowest order with a continuous second derivative, used to calculate curvature. The bases functions are non-negative and have local support. Each basis function $B_i(p)$ is non-zero only for
$p \in (i-2, i+2)$. This means that if a control point changes its position, a resulting change of the curve will only occur in a small neighborhood around the control point. The position of points of the curve between control points can be calculated directly from equation 1, using any positive real number for p.

Using B-splines, the curve can be *decomposed* spatially in terms of local polynomial bases functions. The scale of the decomposition however (the size of the support of the bases function) is predetermined by the spacing of the control points.

To represent a curve at multiple scales, global Fourier bases functions can be used. Using arc-length parametrization, a continuous curve $\overrightarrow{\mathcal{C}}(s) : [0, L] \to \mathbb{R}^2$ can be represented as:

$$\overrightarrow{\mathcal{C}}(s) = \sum_{n} T_n e^{i(2\pi/L)ns} \tag{2}$$

The coefficients T_n of the series are called the Fourier descriptors and they uniquely represent the curve. They are given by:

$$T_n = \frac{1}{L} \int_0^L \overrightarrow{\mathcal{C}}(s) e^{-i(2\pi/L)ns} \,\mathrm{d}s \tag{3}$$

In practice $\overrightarrow{C}(s)$ is only evaluated at discrete control points, so the T_n are calculated from the discrete Fourier transform. The main advantage of using Fourier descriptors is that *n* represents the frequency of the Fourier basis, the higher the *n*, the higher the frequency. Therefore, the curve can be represented at a particular scale by filtering out (zeroing) Fourier descriptors that fall outside of the frequency content of that scale in equation 2. A spectrum analysis of the Fourier descriptors can also reveal which scale contains significant content, allowing for a compact representation of the curve by using only the Fourier descriptors with significant power.

2.2.2 Three Dimensional Parametric Shape Representation

Just like for the 2D case, the continuous three-dimensional surface $\vec{S} : [0, 1] \times [0, 1] \rightarrow \mathbb{R}^3$ is represented as discrete points and linear combination of basis functions evaluated at those points.

The simplest representation for the shape is a triangular mesh (piecewise linear surface) and the use of finite difference methods or finite element methods to discretize derivatives. In the finite difference case, one might choose the space of Dirac delta functions as bases functions. This means that the shape can be represented as a finite linear combination of K Dirac functions (where K is the number of vertices) defined at each vertex of the triangulation. In the finite element case, one uses the space of piecewise linear functions. This means that the shape can be represented as a finite linear combination of K functions $v_k : [0,1] \times [0,1] \rightarrow \mathbb{R}^3$, $k \in [0,K]$ (where K is the number of vertices). Each basis function v_k is defined to be 1 at vertex x_k and linearly decreases to 0 at every other vertex.

One disadvantage of such techniques is the local support of the bases functions (a function has support where it evaluates to nonzero). In the finite difference case for example, the movement of one vertex is completely independent of the movement of other vertices due to the fact that the Dirac delta function only has support at that vertex. As the surface evolves, vertices of triangles can move independently of each other and triangles can cross or overlap. As a result, frequent re-parametrization are needed to ensure that the surface does not develop holes, handles or foldings. In [10, 40], the surface is represented in the form of weighted sums of local polynomial basis functions. The use of higher order polynomials ensures a larger support for each basis function. However frequent re-parametrization remain an issue with these techniques.

For this reason, smooth, complete bases functions with global support are used in practice. The global support ensures a smoothness constraint between vertices of the triangulation and therefore require less frequent re-parametrization in practice [41]. The requirement of completeness means that every function $f \in L^2(\mathbb{R}^3)$ may be expanded in the basis as a linear combination of coefficients and basis functions [70]. We present such shape representation that are relevant to our work.

2.2.2.1 Fourier Surfaces

In the work of Staib et al. [57], a Fourier parametrization decomposes the surface into a weighted sum of sinusoidal basis functions. The surfaces are represented explicitly by three functions of two surface parameters:

$$\overrightarrow{\mathcal{S}}(u,v) = (x(u,v), y(u,v), z(u,v))$$
(4)

The function is then represented by:

$$\overrightarrow{\mathcal{S}}(u,v) = \sum_{m=0}^{2K} \sum_{l=0}^{2K} \lambda_{m,l} \frac{[a_{m,l}\cos 2\pi mu\cos 2\pi lv + bm, l\sin 2\pi mu\cos 2\pi lv + c_{m,l}\cos 2\pi mu\sin 2\pi lv + dm, l\sin 2\pi mu\sin 2\pi lv]}{c_{m,l}\cos 2\pi mu\sin 2\pi lv + dm, l\sin 2\pi mu\sin 2\pi lv]}$$

where

$$\lambda_{m,l} = \begin{cases} 1 & \text{for} & m = 0, l = 0\\ 2 & \text{for} & m > 0, l = 0 \text{ or } m = 0, l > 0\\ 3 & \text{for} & m > 0, l > 0 \end{cases}$$

The parameters of the model are $p = [a_x, b_x, c_x, d_x, a_y, b_y, c_y, d_y, a_z, b_z, c_z, d_z]$, where a_x represents all the $a_{m,l}$ for the function x(u, v). The main issue with this technique, as with any 3D parametrized active surface, is the choice of surface parametrization (u, v). Spheres or cylinders can be easily parametrized, but more complex surfaces are not. The torus is the easiest surface to represent because surface parameters are forced to be periodic. Other types of surfaces can be described using subsets of the Fourier bases which flatten out or constrain the torus in different ways. Since parameters representing open surfaces result in discontinuities at the boundaries, the discontinuities are avoided by having the two surface parameters start at one side of

the surface, trace along the surface to the other end, and then retrace the surface in the opposite direction to create a closed path and a periodic parametrization. Closed surfaces are the most difficult to represent because they are the most dissimilar to the torus. They are represented as open surfaces (tubes) whose ends close up to a point at both ends.

Despite this complexity, one nice feature of the Fourier representation is the inherent multi-resolution properties. The coefficients associated with the basis functions with higher frequency represent higher spatial variations on the surface. These coefficients can thus be truncated and the series will still represent relatively smooth objects accurately, using only a few coefficients associated with the lowest frequencies. Another advantage of the Fourier representation is the geometric interpretation of the coefficients. Low index coefficients describe global shape deformations and higher indexed coefficients represent more local deformations.

More recent work on using Fourier bases has avoided the parametrization problem by first mapping the surface to the sphere, and decomposing the shape signal using bases functions defined on the sphere.

2.2.2.2 Spherical Harmonics

In the work of Brechbuhler et al. [5], a continuous, one-to-one mapping from the surface of an original object to the sphere is defined. The mapping is a nonlinear optimization constrained by two requirements: minimization of distortions and preservation of area. The basic idea is to start with an initial parametrization of the shape. Then the initial parametrization is optimized so that every surface patch gets assigned an area in parameter space that is proportional to its area in object space, while the distortion is minimized. This global parametrization allows for the systematical scanning of the object surface by the variation of two parameters θ and ϕ , overcoming previous limitations of expressing object surfaces in polar-coordinates,



Figure 7: Visualization of the spherical harmonic functions on the sphere. A darker color shows where the function has support. The figure shows the real parts of the spherical harmonic function Y_l^m , with l growing from 0 (top) to 5 (bottom), and m ranging from 0 (left) to l in each row (from [23])

which restrict such descriptions to star-shaped objects. The object surface can then be expanded into a complete set of spherical harmonics basis functions.

The basis functions are defined as:

$$Y_l^m(\theta,\phi) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) e^{im\phi}$$

$$Y_l^{-m}(\theta,\phi) = (-1)^m Y_l^{m*}(\theta,\phi)$$
(5)

where Y_l^{m*} denotes the complex conjugate of Y_l^m and P_l^m describes the associated Legendre polynomials

$$P_l^m(w) = \frac{(-1)^m}{2^l l!} (1 - w^2)^{\frac{m}{2}} \frac{d^{m+l}}{dw^{m+l}} (w^2 - 1)^l$$
(6)



Figure 8: Visualization of the spherical harmonic correspondence. A first order ellipsoid and six left ventricles are displayed. The surface net shows the (θ_i, ϕ_i) parametrization (same parameters = same homologous points). The ridges on the first order ellipsoid are the equator and $0, \pi/2, \pi, 3\pi/2$ meridian lines in all objects. The equator and meridian lines are emphasized in different colors. The poles are at the crossing of the meridian lines (from [23])



Figure 9: Visualization of a lateral ventricle (side view) at different spherical harmonics degrees; m=1,4,8,12 top to bottom (from [23])

Figure 7 shows a visualization of the spherical harmonic functions on the sphere. A darker color shows where the function has support. The figure shows the real parts of the spherical harmonic function Y_l^m , with l growing from 0 (top) to 5 (bottom), and m ranging from 0 (left) to l in each row.

To express the surface $\overrightarrow{S}(\theta, \phi) = (x(\theta, \phi), y(\theta, \phi), z(\theta, \phi))$ using spherical harmonics:

$$\vec{\mathcal{S}}(\theta,\phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} c_l^m Y_l^m(\theta,\phi)$$
(7)

where the SPHARM coefficients c_l^m are three-dimensional vectors due to the three coordinate functions x, yandz. The coefficients c_l^m are obtained by solving a least squares problem, see [5] for the derivation.

Just like for the Fourier surfaces, the coefficients of the spherical harmonic functions of different degrees provide a measure of the spatial frequency constituents that comprise the structure. Partial sums of the series in Equation 7 can be used to represent selected frequencies of the object. As higher frequency components are included, more detailed features of the object appear. This process is shown in Figure 9.

The parametrization of surfaces is invariant to object scaling since the whole surface is mapped to the unit sphere. It is also invariant to translation if the object is centered at the origin. However the coefficients obtained still depend on the orientation of the object in space. This means for example that if two objects with identical shape differ by a rotation, different points of the two objects will be mapped to the north pole. To get rid of these dependencies, it is necessary to rotate the object to a standard position, so that in our previous example, the two same points on the object map to the north pole. This can be done by representing the object using only the first order ellipsoid (using the first three coefficients) and applying a rotation in parameter space so that for all shapes the poles of the first order ellipsoid correspond to the poles of the sphere, and one end of the short axis of the ellipsoid corresponds to the point where the Greenwich meridian meets the equator (there is a mirroring ambiguity depending on which end of the axis is mapped to the Greenwichequator point. The authors propose to use information from higher degree coefficient to disambiguate that case). Figure 8 shows this process.

Spherical harmonics have been used in medical imaging for shape analysis [23]. For shape analysis, a distance between two objects is calculated directly from their spherical harmonics coefficients via a difference calculation. The authors conducted a statistical shape analysis of the lateral ventricles, a fluid filled structure in the center of the human brain, that aimed to distinguish monozygotic twins from dizogotic twins and from unrelated pairs. They took pairwise differences in volume and SPHARM coefficient for each pair of twins. Significant differences between MZ and DZ pairs could not be found by volume measurements but only by SPHARM measurements. The authors identified that one weakness of shape analysis by SPHARM is the nonintuitive and non-localized nature of the set of coefficients. If a particular coefficient is found to be statistically significant in shape difference findings, the coefficient itself does not reveal the localization of the effect, only its "frequency".

In chapter 4, we propose to use a set of basis functions, spherical wavelets, that address this shortcoming.

2.2.3 Adding Data-Driven Shape Priors to Parametric Models

Parametric models can be restricted to a particular shape space by constraining the values of the coefficients of the bases functions used the represent the shapes. The main challenge is to learn the probability function of the coefficients from example shapes, called a training set. These shapes are said to belong to the same "population", meaning that they represent the same type of object (for example brain ventricles). The most common assumption is to assume a joint Gaussian probability function for the shape coefficients, as in the active shape model.

2.2.3.1 Point Distribution Models

In active shape models (ASM) [11], the authors represent shapes as linear combinations of landmark points. The landmark points must be placed in the same way on each object boundary in a training set. Figure 10 shows an example of landmarks placed on several brain structures in a single model.

Each shape can be described by three coordinate functions, $x, y, z \in R$ such that the position of the n^{th} landmark is $(x(n), y(n), z(n))^T, n \in [1, N]$.

The k^{th} shape in a population of K shapes is a column vector of size $3N \times 1$:

$$S_k = [x_k(1), ..., x_k(N), y_k(1), ..., y_k(N), z_k(1), ..., z_k(N)]^T, k \in [1, K]$$
(8)

Since all landmarks are registered, we can interpret each entry in S_k as a random variable so that each shape is a realization from a multivariate probability function, P(S).

A mean shape can then be calculated as $\overline{S} = \frac{1}{K} (\sum_{k=1}^{K} S_k).$

The main assumption of ASM is that the set of K shapes in the training set gives a cloud of K points in 3N dimensional space that is approximately *ellipsoidal* and has a multivariate normal probability function in \mathbb{R}^{3N} :

$$P(\mathbf{S} = \mathbf{s}_{i}) = \frac{1}{\mathbf{z}} \cdot \exp\{-(\mathbf{s}_{i} - \overline{\mathbf{S}})^{\mathrm{T}} \mathbf{C}^{-1}(\mathbf{s}_{i} - \overline{\mathbf{S}})\}$$
(9)

where s_i is a shape realization, z is a normalization constant, \overline{S} is the mean shape and C is the covariance matrix that encodes the axes of the ellipsoid cloud of data.

To find the major axes, the covariance matrix C is estimated by building a covariance matrix SS^T from the data and diagonalizing it via singular value decomposition, such that $C \simeq SS^T = U\Sigma U^T$. The eigenvectors U are the major axes of the ellipsoidal cloud and form an orthonormal L^2 basis for the shapes. Hence each major axis represents a *mode of variation* of the data. The corresponding eigenvalue σ is the magnitude of the variation.



Figure 10: Outline of several brain structures in a single model labeled with landmarks (from [11])



(a) Varying the most significant mode of vari- (b) Varying the second most significant mode ation $(\pm 2 \text{ std})$ of variation $(\pm 2 \text{ std})$

Figure 11: In ASM, the learned shape variation can be shown by selecting one of the eigenvectors (modes of variation) and varying its magnitude from $-n\sqrt{\sigma}$ to $n\sqrt{\sigma}$ with n = 2 and adding the result to the mean shape to display the variation(from [11])

With this interpretation, it is common practice in the literature to show shape variation by selecting one of the eigenvectors and varying its magnitude from $-n\sqrt{\sigma}$ to $n\sqrt{\sigma}$ where n = 2 for example and adding the result to the mean shape to display the variation, as shown in Figure 11.

2.2.3.2 Using the Covariance Matrix for Shape Prior

To use the preceding L^2 basis as a shape prior, one could directly apply Equation (9) to a new shape s_n to test whether it has a high probability of belonging to the shape class. Alternatively, to restrict a new shape to belong to the population, we can project the shape unto the most significant eigenvectors U_t (the ones who cumulative eigenvalues account for a high percentage of the total eigenvalues) and translate the coordinates of the new shape to the closest point that lies inside or at a reasonable distance of the training data point cloud. A shape s_n is corrected to s'_n by applying equations 10-12:

$$b_n = U_t^T (s_n - \overline{S}) \tag{10}$$

$$b'_{n}(m) = \left\{ \begin{array}{ccc} b_{n}(m) & if \quad -3\sqrt{(\sigma_{m})} \leq b_{n}(m) \leq 3\sqrt{(\sigma_{m})} \\ -3\sqrt{(\sigma_{m})} & if \quad b_{n} < -3\sqrt{(\sigma_{m})} \\ 3\sqrt{(\sigma_{m})} & if \quad b_{n} > 3\sqrt{(\sigma_{m})} \end{array} \right\}$$
(11)

$$s_n' = \overline{S} + U_t b_n' \tag{12}$$

2.2.3.3 Limitation of PCA in Representing Finer Shape Details

One problem with this technique is that PCA favors the discovery of global variations over local variations. For a training set size of K shapes with N vertices, where $N \gg K$ and the size of S is $3N \times K$, the rank of the covariance matrix SS^T will be at most K. It can be shown that the eigenvectors of the covariance matrix describe the longest axes of the ellipsoid, and thus the most significant modes of variation in the vertices [11]. If there exist M shape "processes", such that $M \gg K$, then only the first K processes that account for the most variation in the data (the most global processes) will be represented. This problem is aggravated if the training sample is very small since the number of independent processes discovered to build a shape prior will be at most K, the size of the training sample, as will be shown in Chapter 4.



Figure 12: Level Sets of an embedding function ϕ for a closed curve in \mathbb{R}^2

2.2.4 2D and 3D Implicit Shape Representation

With this representation, curves are represented implicitly as the zero level set of a higher-dimensional scalar function in \mathbb{R}^2 and surfaces are the zero level set of a scalar function in \mathbb{R}^3 . The scalar function used is usually a signed distance map. In 2D, a curve $\overrightarrow{\mathcal{C}}(p)$: $[0,1] \to \mathbb{R}^2$ to be represented is embedded as the zero level set of a surface, ϕ , in \mathbb{R}^3 whose height is sampled at regular intervals on the x, y grid as shown in Figure 12(a). Each height value $z = \phi(x, y)$ encodes the distance to the nearest point on the curve, with negative values inside the curve. Similarly, a surface $\overrightarrow{\mathcal{S}}(u, v) : [0, 1] \times [0, 1] \to \mathbb{R}^3$ to be represented is embedded as the zero level set of a hyper-surface, ϕ , in \mathbb{R}^4 . To visualize the curve, only the set of grid points $\phi(x, y) = 0$ are shown. For a surface, only the isosurface $\phi(x, y, z) = 0$ is shown.

2.2.5 Shape Priors for Implicit Representations

Shape priors in the geometric active contours were introduced by Leventon et al [36]. They compute a statistical shape model over a training set of curves.

To build the shape model, they choose a representation of curves, and then define a probability density function over the parameters of the representation, in similar spirit to active shape models [11]. The major difference however is in the curve representation. Each curve in the training dataset is embedded as the zero level set of a higher dimensional surface, u, whose height is sampled at regular intervals (say N^d samples, where d is the number of dimensions). The embedding function chosen is the commonly used signed distance function, where each sample encodes the distance to the nearest point on the curve, with negative values inside the curve. Each such surface (distance map) can be considered a point in a high dimensional space $(u \in \mathbb{R}^{N^d})$. The training set, T, then consists of a set of n surfaces $T = u_1, u_2, ..., u_n$.

The cloud of points corresponding to the training set is approximated to have a Gaussian probability function, where most of the dimensions of the Gaussian collapse, leaving the principal modes of shape variation. The mean surface, μ , is computed by taking the mean of the signed distance functions,

$$\mu = \frac{1}{n} \sum u_i.$$

The variance in shape is computed using Principal Component Analysis (PCA). The mean shape, μ , is subtracted from each u_i to create an mean-offset map, \tilde{u}_i . Each such map, \tilde{u}_i , is placed as a column vector in an $N^d \times n$ -dimensional matrix M. Using Singular Value Decomposition (SVD), the covariance matrix

$$\frac{1}{n}MM^{T}$$

is decomposed as:

$$U\Sigma U^T = \frac{1}{n} M M^T \tag{13}$$

where U is a matrix whose column vectors represent the set of orthogonal modes of shape variation and Σ is a diagonal matrix of corresponding singular values. An estimate of a novel shape, u, of the same class of object can be represented by kprincipal components in a k-dimensional vector of coefficients, α .

$$\alpha = U_k^T (u - \mu) \tag{14}$$

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

$$u' = U_k \alpha + \mu \tag{15}$$

The authors note that in general u' will not be a true distance function, since convex linear combinations of distance maps do not produce distance maps. However, in practice the surfaces generally still have advantageous properties of smoothness, local dependence, and zero level sets consistent with the combination of original curves.

2.3 Deformable Models for Probabilistic Segmentation

As explained in Chapter 1, the segmentation techniques described in this thesis are part of a probabilistic segmentation framework using deformable models. If S is a shape representation and D is the image data, the segmented shape is the maximum a posteriori estimate

$$S^* = \arg\max_{S} P(S|D) \tag{16}$$

Assuming that P(D) is uniform, the posterior can be expressed as the product of the likelihood P(D|S) and the model prior P(S)

$$P(D,S) = P(S|D)P(D)$$
(17)

$$P(S|D) = \frac{P(D|S).P(S)}{P(D)}$$
(18)

$$P(S|D) \propto P(D|S)P(S) \tag{19}$$

To derive an evolution equation for deformable models, an energy formulation is given:

$$E(D,S) \equiv -log(p(S|D)) \tag{20}$$

$$E(D,S) = -log(p(D|S)) - log(p(S))$$

$$(21)$$

$$E(D,S) = E_{data}(D,S) + E_{prior}(S)$$
(22)

Segmentation can then be formulated as an energy minimization problem:

$$S* = \arg\min_{S} E(I,S) \tag{23}$$

This minimization is often solved via gradient descent optimization. The shape representation is augmented with an artificial time parameter t and an initial shape at time t = 0 of the optimization procedure is given by $S_{|t=0} = S_0$. The shape representation is then updated at each iteration in the gradient direction so as to minimize the overall energy by using the forward Euler equation:

$$S_{t+1} = S_t + \delta \frac{\partial S}{\partial t} \frac{\partial S}{\partial t} = -\nabla_S E \ (D, S)$$
(24)

This formulation applies both to parametric and implicit deformable models described in the next sections.

2.3.1 Parametric Deformable Models

The classical parametric model is the snake formulation. We also present interesting extensions of parametric models relevant to our work.



Figure 13: An example of a snake active contour (a) Intensity CT image of the heart left ventricle (b) edge detected image (c) initial contour (d) deformable contour moving toward the left ventricle boundary, driven by an inflating pressure force (from [41])

2.3.1.1 Snakes

Mathematically, a snake is a parametrized contour embedded in the image plane $I(x, y) : \mathbb{R}^2 \to \mathbb{R}$. The contour is represented by $\overrightarrow{\mathcal{C}}(p) = (x(p), y(p))^T$ where $p \in [0, 1]$ is an arbitrary parametrization of the curve. The shape of the contour is defined by the energy functional:

$$E(\overrightarrow{\mathcal{C}}) = \mathcal{S}(\overrightarrow{\mathcal{C}}) + \mathcal{P}(\overrightarrow{\mathcal{C}})$$
(25)

The final shape of the contour will correspond to the minimum of this energy functional. The first term,

$$\mathcal{S}(\overrightarrow{\mathcal{C}}) = \int_0^1 \alpha(p) |\overrightarrow{\mathcal{C}}_p|^2 + \beta(p) |\overrightarrow{\mathcal{C}}_{pp}|^2 \, \mathrm{d}p \tag{26}$$

is the internal deformation energy. $\alpha(p)$ controls the "tension" of the contour while $\beta(p)$ controls its "rigidity". Since $\overrightarrow{C}_p(p)$ is equivalent to the local length of the contour at p, a high value of α places a penalty on extraneous ripples of the curve. Similarly a high value of β places a penalty on extraneous bending. The second term,

$$\mathcal{P}(\overrightarrow{\mathcal{C}}) = \int_0^1 \Psi(\overrightarrow{\mathcal{C}}(p)) \, \mathrm{d}p \tag{27}$$

is the potential energy that couples the snake to the image. $P(x, y) : \mathbb{R}^2 \to \mathbb{R}$ is a scalar function designed to have local minima to coincide with strong edges in the image that correspond to structure boundaries. A classical choice is:

$$\Psi(x,y) = \frac{1}{1 + \|(\nabla(G_{\sigma}(x,y) * I(x,y))\|^2}$$
(28)

where $G_{\sigma}(x, y)$ is the two-dimensional Gaussian function with standard deviation σ , ∇ is the gradient operator, and * is the 2D image convolution operator. Minimizing $\mathcal{P}(\overrightarrow{\mathcal{C}})$ will therefore attract $\overrightarrow{\mathcal{C}}$ toward the edges.

Finding a curve $\overrightarrow{\mathcal{C}}$ that minimizes the overall energy function E is a variational problem [13]. To minimize Equation 25, $\overrightarrow{\mathcal{C}}$ must satisfy the following Euler-Lagrange

equation[13]:

$$\frac{\partial E}{\partial \overrightarrow{\mathcal{C}}} = -\frac{\partial}{\partial p} \left(\alpha \frac{\partial \overrightarrow{\mathcal{C}}}{\partial p} \right) + \frac{\partial^2}{\partial p^2} \left(\beta \frac{\partial^2 \overrightarrow{\mathcal{C}}}{\partial p^2} \right) + \nabla P(\overrightarrow{\mathcal{C}}) = 0$$
(29)

To solve Equation 29 given an initial curve $\vec{\mathcal{C}}_0$, the curve is made dynamic by augmenting it with a time parameter, $\vec{\mathcal{C}}(p,t)$, with $\vec{\mathcal{C}}(p,0) = \vec{\mathcal{C}}_0(p)$.

By the chain rule,

$$<\frac{\partial E}{\partial \overrightarrow{\mathcal{C}}}, \frac{\partial \overrightarrow{\mathcal{C}}}{\partial t}>=\frac{\partial E}{\partial t}$$
(30)

where $\langle ., . \rangle$ represents an inner product between two functions (infinite dimensional vectors). Therefore the energy is minimized in time when $\frac{\partial E}{\partial \vec{c}} = -\frac{\partial \vec{c}}{\partial t}$ (the inner product between two vectors $\langle u, v \rangle$ is minimum when u = -v).

The curve is then deformed at each time step according to the following gradient descent equation:

$$\gamma \frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = \frac{\partial}{\partial p} \left(\alpha \frac{\partial \overrightarrow{\mathcal{C}}}{\partial p} \right) - \frac{\partial^2}{\partial p^2} \left(\beta \frac{\partial^2 \overrightarrow{\mathcal{C}}}{\partial p^2} \right) - \nabla \Psi(\overrightarrow{\mathcal{C}})$$
(31)

The coefficient γ is introduced to make units consistent. Thus the minimization is solved by placing an initial contour on the image domain and allowing it to deform at each time step according to equation 31. This can be done with Forward-Euler Equation [53]:

$$\overrightarrow{\mathcal{C}}(p,t+1) = \overrightarrow{\mathcal{C}}(p,t) + \Delta t * \frac{\partial \overrightarrow{\mathcal{C}}(p,t)}{\partial t}$$
(32)

where Δt is chosen based on stability conditions (the CFL condition [53]).

One limitation of the classical snake model is that it needs to be initialized very close to the object boundary in order to be within the attraction range of the potential force [41]. To address this limitation, Cohen [9] proposes to increase the attraction range by using a pressure force in addition to the potential energy defined in Equation 27. The pressure force can either inflate or deflate the contour, removing the requirement that the model is initialized near the desired object boundary. The pressure force is defined as:

$$P(\overrightarrow{\mathcal{C}}) = w_p \mathcal{N}(\overrightarrow{\mathcal{C}}) \tag{33}$$

where $\mathcal{N}(\vec{C})$ is the inward unit normal and w_p is a constant weighting parameter. $P(\vec{C})$ is added to as an additional term in Equation 31 and has the effect of moving the control points of the model either along their inward normal (if w_p is positive) or along their outward normal (if w_p is negative). An example of using a deformable contour using an inflationary ($w_p < 0$) pressure force is shown in Figure 13. See [41, 71] for a detailed survey of snakes and their use in medical image analysis.

2.3.1.2 Snakes Implementation

Various numerical implementations of snakes exist in the literature [41, 71]. In order to numerically compute the snake evolution, it is necessary to discretize the energy $E(\vec{C})$. The approach is to represent the continuous contour \vec{C} in terms of discrete points and linear combinations of basis functions expressed at those sample points. Finite differences [32], finite elements [47] and geometric splines [19] are local basis functions. Fourier Bases have also been used as global representation methods [56]. The trade-off is that local representations have more degrees of freedom to represent complex shapes with high local variations, but require special care to guarantee the smoothness of the curve. Global representations have built-in smoothness constraints, but fewer degrees of freedom.

The points are parametrized by the parameter p and the spacing between the points evolves dynamically as the curve evolves. The spacing between the points can be represented mathematically as the length of the tangent at each sample point, $||\frac{\partial \vec{C}}{\partial p}||$. If the spacing between two points becomes too small or too large (relative to the rest of the points), a re-parametrization is necessary to keep the numerical implementation well behaved.

2.3.1.3 Snakes Limitations

Parametric deformable models such as snakes have been applied successfully in a wide range of applications [71]. However two limitations are often cited in the literature:

- In situations where the initial contour and the final solution differ greatly, it is necessary to re-parametrize the curve to keep the sample points equally spaced to more faithfully recover the object boundary and keep sample points from crossing and create "loops" in the contour. Methods for re-parametrization in 2D are straightforward, but in 3D re-parametrization is more complex and requires computationally expensive methods.
- The contour has a fixed topology and cannot automatically merge or split. If a topology change is needed, a new parametrization needs to be constructed which requires sophisticated schemes for detection and re-parametrization [42].

On the other hand, if the desired topology of the contour is known a priori and a shape prior is applied to the contour to regularize the parametrization, parametrized models are still a method of choice due to their fast running time. In addition, the shape prior probability function is a function of a smaller number of shape parameters when compared to implicit models, which is an advantage to estimate the function more accurately. We will discuss the pros and cons of parametric models versus geometric models further at the end of this chapter.

2.3.1.4 Parametric Deformable Surfaces

Since most medical structures are three-dimensional, the use of a true 3D deformable surface to segment a structure in a volume ensures a globally smooth and coherent segmentation. The theory presented in section 2.3.1.1 can easily be extended to surfaces by using a parametrized surface embedded in the volume $I(x, y, z) : \mathbb{R}^3 \to \mathbb{R}$. The surface is represented by $\overrightarrow{S}(u, v) = (x(u, v), y(u, v), z(u, v))^T$ where $u \in [0, 1], v \in$ [0, 1] parametrize the surface.

As an example of surface evolution, Staib et al. [57] use the Fourier surface representation presented in 2.2.2.1 for segmentation of 3D medical images. In order to fit one of these models to the image data, a measure of fit is optimized by varying the model parameters. They assume that the surface is distinguishable by a measure of boundary strength $b(\vec{S})$ computed from the image (using a term similar to Ψ defined in Equation 28). A measure of fit for curves can then be written as:

$$M(b,\mathbf{p}) = \int \int |b(x(\mathbf{p},u,v), y(\mathbf{p},u,v), z(\mathbf{p},u,v)| dA$$
(34)

where **p** is a vector consisting of the Fourier coefficients. Equation 34 can be evaluated by numerical integration by taking the derivative of the function M(b, p) with respect to each parameter in **p**. They show results for smooth spherical shapes, such as the heart's endocardium.

2.3.1.5 Discussion

Parametric Deformable Models can be a very fast and powerful technique, so long as topological changes are not expected to happen during the segmentation. If topological changes happen, more complex bookkeeping is needed. In addition to their fast running time, another main advantage of parametric models is the fact that shape priors can easily be included in the formulation by learning a prior probability function of the model parameters and limiting the evolution within a particular shape space by keeping the model parameters within a certain standard deviation of the values of the training set. One main issue is the problem of aligning training samples in order to learn a prior probability function. In 3D, techniques that use spherical harmonics to align shapes address this issue.

2.3.2 Geometric Deformable Models

Geometric deformable models, proposed independently by Caselles et al. [6] and Malladi et al. [48], provide an elegant solution to address the topological change limitation of parametric deformable models. Curves and surfaces are evolved using only geometric measures that can easily be represented by an implicit shape representation. By representing evolving curves and surfaces implicitly as a level set of a higherdimensional function [44, 53], the evolution no longer depends on parametrization. As a result, topology changes can be handled automatically.

The purpose of geometric curve evolution theory is to study the deformation of curves using only geometric measures such as the unit normal, the arc length parameter and curvature as opposed to using quantities that depend on an arbitrary parametrization of the curve. Let us consider a moving curve $\vec{\mathcal{C}}(s,t) =$ (x(s,t),y(s,t)) where s is an arbitrary parametrization and t is an artificial time parameter. If we denote the inward normal at point p at time t as $\mathcal{N}(s,t)$ and the curvature at point s at time t as $\kappa(s,t)$, then the evolution of the curve along its normal direction can be characterized by the following partial differential equation:

$$\frac{\partial \overline{\mathcal{C}}(s,t)}{\partial t} = V(\kappa(s,t))\mathcal{N}(s,t)$$
(35)

where $V(\kappa)$ is called speed function, since it determines the speed of the curve evolution. Only the normal direction is needed since the tangent deformation affects only the curves parameterization, not its shape and geometry [44]. The partial differential equation in 35 is derived by defining an energy functional that depends on the geometry of the curve and possibly other external quantities, such as image intensities, and finding the curve that maximizes or minimizes this energy using calculus of variations.

There are two types of energies: *edge-based*, where quantities used in the energy definition are a function only of the *curve boundary*, or *area-based*, where quantities used in the energy are a function both of the *curve boundary* and the *region inside the curve*.

In the next sections, we present the most extensively studies curve deformations using various energy functionals.

2.3.2.1 Curvature Flow

The Curvature deformation evolution equation is derived by setting up an energy that represents the total length of the curve:

$$E(\overrightarrow{\mathcal{C}}) = \int_0^L \,\mathrm{d}s \tag{36}$$

where s corresponds to the arc-length parameter. The solution to minimize such an energy (and hence to minimize arc-length) is the following gradient flow, also called the *geometric heat equation*:

$$\frac{\partial \overrightarrow{\mathcal{C}}(p,t)}{\partial t} = \alpha \kappa(p,t) \mathcal{N}(p,t)$$
(37)

where α is a positive constant. This equation will smooth a curve, eventually shrinking it to a circular point [27]. The use of the curvature deformation has a smoothing effect similar to the use of the elastic internal force in parametric deformable models.

2.3.2.2 Dilation/Erosion Flow

To derive the constant deformation flow, we write down an energy that represents the enclosed area of the contour:

$$E(\overrightarrow{\mathcal{C}}) = \int \int_{R} \, \mathrm{d}A \tag{38}$$

where R is the region enclosed by $\overrightarrow{\mathcal{C}}$ and A is the area element. The solution to minimize/maximize such an energy (and hence to minimize/maximize total area) is:

$$\frac{\partial \, \overrightarrow{\mathcal{C}} \left(p, t \right)}{\partial t} = V_0 \mathcal{N}(p, t) \tag{39}$$

where V_0 is either +1 for an erosion flow (recall that \mathcal{N} is the inward normal) and -1 is the erosion flow. Constant deformation plays the same role as the pressure force in parametric deformable models.



(a) Initial active contour. (b) Evolving active contour. (c) Steady state.

Figure 14: Ventricle segmentation in MRI heart image via shrinking conformal active contour.

The properties of curvature deformation and constant deformation are complementary to each other. Curvature deformation removes singularities by smoothing the curve, while constant deformation can create singularities from an initially smooth curve. The basic idea of the geometric deformable model is to couple the speed of deformation (using curvature and/or constant deformation) with the image data, so that the evolution of the curve stops at object boundaries. The evolution is implemented using the level set method [44, 53]. Most of the research in geometric deformable models has been focused in the design of speed functions. We present the most popular in the next sections.

2.3.2.3 Conformal (Geodesic) Active Contours

Caselles et al. [7] and Kichenassamy et al. [33] both used an energy minimization formulation that includes image data in the speed function. It is based on finding a geodesic in a Riemannian space where the metric is derived from the image content, i.e., a curve of minimal weighted length. The metric used is Ψ , previously defined in Equation 28. This will result in finding a curve that has minimum energy when it is located on an edge (when Ψ is the smallest).



(d) Splitting of active contour and steady state.

Figure 15: Bone segmentation in CT image with splitting shrinking conformal active contour.

$$E(\overrightarrow{\mathcal{C}}) = \int_0^L \Psi(\overrightarrow{\mathcal{C}}) \, \mathrm{d}s \tag{40}$$

This leads to the gradient flow:

$$\frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = \{ (\nabla \Psi \cdot \mathcal{N}) - \Psi \kappa \} \mathcal{N}$$
(41)

The term $\nabla \Psi \cdot \mathcal{N}$ is large when the gradient of the edge detector Ψ coincides with the normal \mathcal{N} of the evolving curve and leads the curve into the boundary and eventually forces it to stay there [7]. An example of the geodesic active contour is shown in Figure 14. Note that in the case where Ψ would be uniform, Eq. (42) reduces to the geometric heat equation [27]. Figure 15 shows a bone segmentation in CT image with splitting shrinking conformal active contour.

2.3.2.4 Conformal (Geodesic) Active Contours with Inflationary Term

The geodesic active contour of Equation 42 is strictly deflationary, meaning that in the presence of no edges, the contour will just shrink to minimize its length (due to the curvature term). Other contours have been devised to allow for inflation as well, using an inflationary term c [71].

$$\frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = \{ (\nabla \Psi \cdot \mathcal{N}) - (\kappa + c) \Psi \} \mathcal{N}$$
(42)

2.3.2.5 Region-Based Active Contours

One issue with edge-based active contours is that they are not robust to noise in the image and the gradient terms can stop the curve evolution at spurious edges. Recently there has been a considerable amount of work on image segmentation using region-based curve evolution techniques. In those technique, the force that influences the evolution of the curve depends on a region statistics, inspired by the region competition work of Zhu and Yuille [75] and more recently the work of Chan and Vese [8] and Yezzi [73]. For example in [73], the authors developed a segmentation technique that evolves a curve in order to maximizes the difference between the mean of the pixels that lie inside the curve and the mean of the pixels that lie outside the curve.

The basic underlying mathematical idea is to write down an energy that minimizes a particular function f inside a region R, enclosed inside the curve $\overrightarrow{\mathcal{C}}$. The goal is then to find a gradient flow of $\overrightarrow{\mathcal{C}}$ that minimizes the following energy:

$$E(\overrightarrow{\mathcal{C}}) = \int_{R} f(\mathbf{x}) \, \mathrm{d}\mathbf{x} \tag{43}$$

The gradient flow for this energy is derived in the appendix, and we give its final form:

$$\frac{\partial \vec{\mathcal{C}}}{\partial t} = -f\vec{\mathcal{N}} \tag{44}$$

This type of flow can be very interesting for medical imaging applications where strong edges are not necessarily present, but image statistics can be used reliably to segment particular structures. For example, if an organ has a know mean and variance for the pixel values that lie inside and on its boundary, then this type of energy can be used:

$$E(\vec{\mathcal{C}}) = \int_{R} \phi(\mathbf{x}) \, \mathrm{d}\mathbf{x} \tag{45}$$

where ϕ is negative if it lies within the range and positive otherwise (where the magnitude depends on how close it is in the range). This type of curve evolution will be used in Chapter 3.

2.3.3 Numerical Implementation: level set method

The level set method is used to account for automatic topology adaptation, and it also provides the basis for a numerical scheme that is used by geometric deformable models. The level set method for evolving curves is due to Osher and Sethian [44, 53].



Figure 16: Four steps in the segmentation of two different corpora callosa. The last image in each case shows the final segmentation in red. The cyan contour is the standard evolution without the shape influence (from [36])

In the level set method, the curve or surface is represented implicitly as a level set of a scalar function which is usually defined on the same domain as the 2D or 3D image that is to be segmented. The level set is defined as the set of points that have the same function value. Figure 12 shows an example of embedding a curve as a zero level set. The sole purpose of the level set function is to provide an implicit representation of the evolving curve and this is made possible by the fact that the evolution equation is defined in terms of geometrical or external quantities. A detailed description of the level set scheme will be given in Section 3.4.1.

2.3.3.1 Shape Priors for Geometric Active Contours

Shape priors in the geometric active contours were introduced by Leventon et al [36]. Their approach to object segmentation extends geodesic active contours by incorporating shape information into the evolution process. They first compute a statistical shape model as described in section 2.2.5 over a training set of curves. To segment a structure from an image, they then evolve an active contour both locally, based on image gradients and curvature, and globally to a maximum a posteriori estimate of shape and pose.

Overall the authors use the learned α shape parameters, as well as rigid pose parameters p (rotation, translation) to parametrize the model. To segment the curve, their technique uses a two step approach: they evolve an initial contour according to the geodesic curve evolution defined in Equation 42. Then, they seek to estimate the shape parameters, α , and the rigid pose parameters, p, of the final curve using a maximum a posteriori approach:

$$<\alpha_{MAP}, p_{MAP}> = argmax_{\alpha,p}P(\alpha, p|\overrightarrow{\mathcal{C}}(t), \nabla I)$$
(46)

In this equation, $\vec{\mathcal{C}}(t)$ is the evolving curve (or surface) at some point in time t. The term ∇I is the gradient of the image containing the object to be segmented. In calculating Equation 46, the authors use a shape prior $P(\alpha)$ to be the Gaussian model over the shape parameters, α , with shape variance Σ_k :

$$P(\alpha) = \frac{1}{sqrt2\pi^k |\Sigma_k|} exp(-\frac{1}{2}\alpha^T \Sigma_k^{-1} \alpha)$$
(47)

Once the optimal shape and pose parameters are found, the optimal curve $\overrightarrow{\mathcal{C}}^*$ is calculated from α_{MAP} and p_{MAP} and the prior is balanced with the image prior according to the following curve evolution equation:

$$\frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = \lambda_1 \{ (\nabla \Psi . \mathcal{N}) - \Psi \kappa \} \mathcal{N} + \lambda_2 \{ \overrightarrow{\mathcal{C}}^* - \overrightarrow{\mathcal{C}} \}$$
(48)

This technique gives convincing results on imagery where the classic geodesic active contour leaks, as shown in Figure 16.

2.3.4 Discussion

The parametric representations are Lagrangian in nature, meaning that coordinates representing the shape (either the coordinate of the control points, or the coefficients of the basis functions) move as the shape deforms. In contrast, implicit representations are Eulerian representations where the underlying coordinate system remains fixed as the shapes change (it is the value of each grid point that changes). One advantage of using an implicit representation is that topological changes in the shape during the evolution are automatically handled, as shown in Figure 12. Figure 12(a) shows the shape at time t = 0 that consists of two disjoint circles. As the value of the function ϕ changes over time, the shape becomes a single blob at time t = 1. No special handling is needed during merging or splitting of the shape. In contrast, changes in model topology for parametric models during the deformation, such as splitting and merging, need to be detected and require special handling. For example when two disjoint (individually parametrized) boundaries merge, the single object needs to be re-parametrized to reflect the new topology. One drawback of the implicit representation however is the higher dimensionality of the model as well as the loss of meaningful shape representation since the shape of the object is only represented as a collection of unordered pixels. Further analysis is required to link pixels in a particular order to represent the notion of a parametrized curve.

Table 1 summarizes this chapter by comparing parametric and geometric active contours.

	Parametric	Geometric
Efficiency	***	*
Ease of Implementation (2D)	***	**
Ease of Implementation (3D)	**	**
Topology Change	No	Yes
Inclusion of Smoothness Shape Prior	Yes	Yes
Inclusion of Global Shape Prior	Yes	Yes

 Table 1: Properties of parametric and geometric active contours. Adapted from [15]

We note therefore that the need for topology change vs. efficiency are big factors in deciding whether one should choose parametric or geometric active contours for shape segmentation.

Shape priors, both for parametric or geometric models, generally fall into two categories:

• Very local Priors that impose a smoothness constraint (usually based on second

derivatives)

• Fully Global Priors that constrain the *whole* shape of the model to a predefined shape space, learned from a training sample.

These priors are good to segment structures that have a consistent shape and topology across the population, for example the brain's ventricles that are consistently "butterfly-shaped" objects. However some structures, such as vessels, do not always have a consistent shape across patients. Each patient can have a different number of branches, located at different areas down from the root of the vessel. In another situation, when the training set is very small, the global prior can be very constraining since it only allows new shapes to resemble shapes seen in the training set. While it might be true that the shapes "overall" look the same (at a low resolution), it is usually the case that higher frequency information differs among shapes. We address these issues in our work.

CHAPTER III

SOFT SHAPE PRIORS: VESSEL SEGMENTATION USING A SHAPE DRIVEN FLOW

In this chapter, we present a segmentation method for vessels using an implicit deformable model with a knowledge-driven scale-specific shape prior. Blood vessels are challenging structures to segment due to their branching and thinning geometry as well as the decrease in image contrast from the root of the vessel to its thin branches. Using image intensity alone to deform an active contour for the task of segmentation often results in leakages at areas where the image information is ambiguous. To address this problem, we combine image statistics and a semi-local shape prior in a variational framework to derive a region-based active contour that segments tubular structures and penalizes leakages. We present results on synthetic and real 2D and 3D datasets.

3.1 Blood Vessel Segmentation

Blood vessel segmentation and visualization of blood vessels are important for clinical tasks such as diagnosis of vascular diseases, surgery planning and blood flow simulation. A number of methods have been developed for vessel segmentation, however most of those techniques do not use a shape prior, or use a strong global shape prior.

Using image intensity alone for the task of segmentation often results in leakages perpendicular to the vessel walls at areas where the image information is ambiguous. Leakages cause the segmented model to expand into areas that are not part of the vessel structure, and result in incorrect segmentations.

To remedy this problem, strong knowledge-driven global shape priors can be used,

such as tubular shape templates to constrain the segmentation to a particular shape space. However since diseased vessels can have abnormal shapes, a strict shape template may result in incorrect segmentation that misses important anatomical information.

Data-driven shape templates, where a shape prior is learned from a training set, are not typically used for vessel structures given that the topology of vessels (the number and location of branches) is highly variable based on the patient.

In this work, we introduce the notion of segmentation with a *soft* knowledgedriven shape prior, where the segmented model is not constrained to a predefined shape space with a predefined shape template, but is penalized if it deviates strongly from a tubular structure, since those deviations have a high probability of being leaks. Our method uses a shape prior localized in space in addition to image statistics to deform an active contour for the task of blood vessel segmentation.

3.2 Related Work

Many geometric methods exist for vessel segmentation that range from using no shape priors to strong shape priors. Tubular structures can be identified by the response of derivative and Gaussian filters convolved with the image. Sato et al use the second derivatives of a set of multi-scale Gaussian filters to detect curvilinear structures and penalize high intensity (bumps) on the vessel walls [50]. The filter response can be used to visualize the vessels through Maximum Intensity Projections (MIP) or isosurface extraction. Since these methods rely on the intensity of the image, a noisy intensity map may result in incorrect filter response and additional shape information might be needed for a correct segmentation. Skeletons [21, 16] can be used as a basis of graph analysis of vessels, and further processing is needed to extract the 3D shape of the vessel. Krissian et al. use multi-scale filtering, based on a set of Gaussian kernels and their derivatives to extract a skeleton of vasculature [34]. The local maxima of the filter response is used to find centerpoints and radius information in order to fit a cylindrical model to the data. The restriction of the shapes can be a limitation since diseased vessels can have cross-sections that deviate from an elliptical shape.

As presented in chapter 2, deformable models are a powerful technique for flexible automatic 3D segmentation. Deformable models are based on an initial contour or surface deformed by a combination of internal (shape) and external (image and user defined) forces to segment the object of interest. In particular, the addition of a shape prior as an internal force can greatly increase the robustness of deformable models when applied to the task of vessel segmentation. Snakes are parametrized models and shape templates can easily be incorporated in this framework [22, 42]. However those methods have a limitation since the surface cannot handle topological changes as easily as the level set methods and re-parameterization is often necessary and complex in 3D.

Wang et al [66] did an analysis of different active contour models and reported that geometric active contours are the best choice for vessel segmentation due to their topological adaptability. As presented in section 2.3.2, geometric deformable models represent a surface implicitly by the zero level set of a scalar-valued function. The evolution is carried out within this framework without the need for parameterization or explicit handling of topological changes [44, 53].

However, since there is no explicit parameterization, incorporating a knowledgedriven shape prior in the geometric active contour framework is more difficult and ad-hoc. For example, to address the issue of leaks that form at the root of vessels during a fast marching segmentation, Deschamps et al. [62] freeze a percentage of the points that are closer to the starting seed (since it is assumed that they have segmented the structure of interest) while allowing the fast evolving points to evolve normally. However this technique does not prevent leaks that form far away from the root of the vessels, close to the fast evolving part of the front. Strong knowledgedriven shape priors in combination with level set techniques for vessel segmentation was used by Lorigo et al. [38]. They evolve 1D curves in a 3D volume and then estimate the radius of the vessels locally using the inverse of principal curvature. However, this technique makes a strong assumption about the shape of the vessels since they are modeled as tubes with varying width.

To segment vessels with a knowledge-driven shape prior, it would be useful to have a shape prior that constrains the shape of the model locally to have a tubular shape, without over-constraining the shape globally. We call this concept a *soft* shape prior. The scale of the soft prior should be adapted to the vessel radius, so that deviations from a tubular shape on a small scale are acceptable, to account for fine details in vessel structure, but that on the scale of the vessel radius, the structure keeps a tubular structure and does not exhibit leaks. To our knowledge, no existing geometric active contour uses such a prior for vessel segmentation.

3.3 Shape Driven Flow

We first present a classical region-based geometric active contour that is prone to leakage problems. We then present our method that adds a soft shape prior for a better behaved flow.

3.3.1 Region Based Flow

The classical region-based geometric active contour flow deforms the curve of interest according to a smoothing term and an image term ϕ . In this work, we use image intensity statistics to determine the optimal image term ϕ . We learn the mean μ and standard deviation σ of a sample of pixels located inside the vessel structure to be segmented (the pixels are chosen interactively by the user). The ϕ term evaluated at pixel \mathbf{x} is then an adaptive threshold:

$$\phi(\mathbf{x}) = 1 - \left| \frac{I(\mathbf{x}) - \mu}{\sigma} \right|,$$

where $I(\mathbf{x})$ is the image intensity at point \mathbf{x} . The term $\phi(\mathbf{x})$ is positive if $I(\mathbf{x})$ falls within one standard deviation of the learned mean μ and negative otherwise. More sophisticated schemes can be used to determine ϕ , such as using a Bayesian classifier [72].

We define the following energy in the region R inside the curve $\overrightarrow{\mathcal{C}}$ parameterized by arc-length s:

$$E(\vec{\mathcal{C}}) = -\int_{R} \phi \, \mathrm{d}\mathbf{x} + \int_{\vec{\mathcal{C}}} \, \mathrm{d}s.$$
(49)

The first term is a region based active contour (see Section 2.3.2.5) that moves the curve towards the region of interest (the energy is minimum when the region enclosed inside the curve is as large as possible, with ϕ of positive value). The second term is an edge-based active contour that regularizes the curve as shown in Section 2.3.2.1. As shown in section 2.3.2.5 and the Appendix, we minimize Equation 49 by computing its first variation and solving the obtained Euler-Lagrange equation by means of gradient descent. We find the following curve evolution equation:

$$\frac{\partial \vec{\mathcal{C}}(\mathbf{x})}{\mathrm{d}t} = (-\phi(\mathbf{x}) + \kappa(\mathbf{x}))\vec{\mathcal{N}},\tag{50}$$

where ϕ is a speed determined by the underlying image, $\kappa(\mathbf{x})$ is the curvature of the curve at point \mathbf{x} and $\overrightarrow{\mathcal{N}}$ is the unit inward normal to the curve. We evolve this active contour using Level Set Techniques [44, 53].

Image statistics alone might lead to unsatisfactory segmentations. Figure 17 shows an example of very noisy image data where areas of pixels close to the vessel have very similar image statistics. This results in a leak when segmented with the type of flow in Equation 50. More sophisticated algorithms can be devised based on image


Figure 17: A Region Based Geometric Active Contour is prone to leak if the image term is not reliable.

statistics or prior knowledge such as multi-scale filter responses tuned to detect vessels [73, 22, 65], but these algorithms will be very specific to the type of data and image acquisition. This leakage problem can be addressed in a more general way by adding a soft shape constraint to the flow so that the algorithm penalizes obvious leaks.

3.3.2 Shape Filters

We would like to locally determine the shape of a contour, and particularly areas where it is widening and potentially leaking. The information from derivatives of the curve, such as curvature, is too local since widening of the contour cannot be discriminated from small noise and bumps. We propose to use a local filter at a scale larger than the derivative scale. We define a local neighborhood $B(\mathbf{x}, r)$ in the shape of a ball (disk in R^2 , solid sphere in R^3) centered at the point \mathbf{x} and of radius r, see Figure 18(a). For every point \mathbf{x} inside and on the contour (region R), we define a filtering operation that calculates a measure ϵ_1 in the neighborhood $B(\mathbf{x}, r)$. The measure ϵ_1 is the percentage of points that fall both within the ball centered at \mathbf{x} and the region R inside the contour¹:

¹If $R \to 0$, then $\epsilon_1(\mathbf{x})$ is just the curvature at \mathbf{x} .



Figure 18: (a) $\epsilon_1(\mathbf{x}, r)$ is the intersection of the ball of radius r centered at \mathbf{x} and the region R inside the contour. (b) Points inside the widening region have a higher ϵ_1 measure but most points on the contour have the same measure. (c)Contour points close to the widening of the contour have a higher ϵ_2 measure

$$\epsilon_1(\mathbf{x}, r) = \int_R \mathcal{X}(\mathbf{x}, r, \mathbf{y}) d\mathbf{y} \quad \text{where} \quad \mathcal{X}(\mathbf{x}, r, \mathbf{y}) = \begin{cases} 1 & if \quad \mathbf{y} \in B(\mathbf{x}, r) \\ 0 & if \quad \mathbf{y} \notin B(\mathbf{x}, r) \end{cases}$$
(51)

The parameter r defines the scale of analysis. It must be chosen by the user and be an upper bound to the expected radius of the vessels. In our simulations, the user picked the width of the largest vessel with the mouse to define r.

The filter response ϵ_1 for a synthetic shape that represents a potential leak is shown in Figure 18(b). Given a radius that is the width of the tube, the points inside the widening region will have a higher measure than the points inside the tube. We will formalize this observation in the next Section by defining an energy minimization that uses the measure ϵ_1 to penalize regions inside the contour that deviate from a tubular shape.

3.3.3 Curve Evolution using Local Filters

We design a flow that uses the shape filter to penalize leaks by using the following energy:

$$E(\overrightarrow{\mathcal{C}}) = -\int_{R} \phi \, \mathrm{d}\mathbf{x} + \int_{\overrightarrow{\mathcal{C}}} \, \mathrm{d}s + \alpha \int_{R} \epsilon_{1}^{p}(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$
(52)

The first and second term are the same as for the region flow previously introduced. The third term is a constraint on shape. When we minimize $E(\vec{C})$, the third term will force deviations from tubular shapes to be penalized. In order to obtain a curve evolution formula, we need take the first variation of Equation 52.

The first variation for such an equation with nested integrals was first derived in [3]. We give the full derivation in Appendix A.2 and state here its result (from Equation 133):

$$\frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = \left(-\phi(\overrightarrow{\mathcal{C}}) + \kappa(\overrightarrow{\mathcal{C}}) + \alpha\epsilon_2(\overrightarrow{\mathcal{C}}, p, r))\right)\mathcal{N},\tag{53}$$

where

$$\epsilon_2(\overrightarrow{\mathcal{C}}, p, r) = \epsilon_1^p(\overrightarrow{\mathcal{C}}, r) + p \int_R \epsilon_1^{p-1}(\overrightarrow{\mathcal{C}}) \mathcal{X}(\mathbf{x}, r, \overrightarrow{\mathcal{C}}) \, \mathrm{d}\mathbf{x}.$$
(54)

We note that ϵ_2 that comes out of the energy minimization is an interesting term. The measure ϵ_2 is again the output of a local ball filter. For a point, the response is its ϵ_1 measure plus the sum of the ϵ_1 measure of its neighboring points that are inside the contour. For a radius r similar to the vessel width, most points on the contour have the same ϵ_1 measure since locally the same percentage of neighbors fall within the filter radius. This can be seen in Figure 18(b), on the left. To see if the contour point lies near a leak region, it is necessary to look at the ϵ_1 measure of its neighbors inside the contour since their measure is high for points inside widening regions. This is what ϵ_2 measures, as shown in Figure 18(c), on the right. We observe that contour points close to the widening of the contour have a higher measure than contour points on the tube ².

Since ϵ_2 is always positive, the third part of Equation 53 is an erosion term (flow along the inward normal) that is proportional to the ϵ_2 measure. At a point with a high measure, the contour shape deviates from a tubular shape and therefore the flow is penalized. The parameter α is chosen according to the amount of penalty desired.

3.4 Implementation

In the section, we describe the level set technique used to implement numerically the proposed flow.

3.4.1 Numerical Methods

We use the level set technique to evolve the curves described in Equations 50 and 53. The level set implementation of Equation 50 has this form:

$$\psi_t = -\frac{d\vec{\mathcal{C}}}{dt} \cdot \nabla \psi = \left(\phi + \kappa + \alpha(\epsilon_2)\right) \mathcal{N} \cdot \nabla \psi$$
(55)

where $\psi : \mathbb{R}^2 \to \mathbb{R}$ is a signed distance function. If $\overrightarrow{\mathcal{C}}$ at time t is the set of points where $\psi = 0$ and we let $\psi_t < 0$ inside $\overrightarrow{\mathcal{C}}$ and $\psi_t > 0$ outside $\overrightarrow{\mathcal{C}}$, and use the fact that $\overrightarrow{\mathcal{N}} = \nabla \psi / ||\nabla \psi||$ and $\kappa = \nabla \cdot (\nabla \psi / ||\nabla \psi||)$, we can write:

$$\psi_t = \left(\phi + \nabla \cdot \frac{\nabla \psi}{||\nabla \psi||} + \alpha(\epsilon_1^p + (p-1)\epsilon_2^{p-1})\right) ||\nabla \psi||$$
(56)

We then use a Forward-Euler Equation to update the value of the level set function ψ at every time step:

$$\psi^{t+1} = \psi^t + \Delta t * \psi_t \tag{57}$$

 $^{^{2}\}mathrm{In}$ our implementation, we scale both measures to lie between 0 and 1 so that all three terms in Equation 54 have similar scaling.

We choose a Δt that satisfies the CFL stability condition (ψ^{t+1} cannot be greater than $\psi^t + 1$).

3.4.1.1 Extensions

The image term ϕ should only depend upon the values of the image along the curve (the zero level set). For other level sets, we have to use the value of the image from nearby points on the zero level set. We therefore use the *extension* technique described in [53]. Similarly the value of ϵ_1 and ϵ_2 is extended from the zero level set to all other level sets.

The second term is a curvature term that depend on the second and first derivatives of the curve C. We calculate those derivatives using central differences (for more details on numerical schemes for curvature, see [53]).

When the sum of the first, third and fourth term is negative, the flow is a dilation since we are going in the negative direction of the inward normal. Conversely, when the sum is positive, the flow is an erosion. We use a different scheme to calculate derivatives depending on whether the flow at a point is an erosion or dilation, as described in [53]. This is to handle to creation of singularities called "shocks" with these types of flow that propagate information in one direction. To still obtain a solution, an *entropy condition* is used. This condition states that "once a particle is burnt, it stays burnt", meaning that once a corner has developed in the curve, the corner is propagated during the evolution of the curve with a special scheme to stop the curve from self-intersecting (or in other terms, to stop the solution from being multiple-valued). The scheme we use in an upwinding scheme that correctly calculates derivatives in the presence of shocks. The scheme is further described in [53], chapter 6.

3.4.2 Optimizations

The filter operations are time consuming. At every time step, we need to calculate ϵ_1 for every point inside the curve and ϵ_2 for every point on the zero level set.

We devised a scheme to only recompute those values for the set of grid points that changes from being positive to negative (points that were outside the contour and are now inside the contour).

We define a *valid neighborhood* of a point P as its neighboring points that fall both inside the local filter centered at P and inside the curve. We store for every point inside the curve their valid neighborhood size.

At every time step, we do the following:

- When the curve evolves, we detect all the grid points that were positive and became negative and put them in the set S⁺. We then update the valid neighborhood of points ∈ S⁺ by adding +1 to their valid neighborhood count. Finally, we calculate the measure for points ∈ S⁺.
- 2. We then detect all the points that were removed from inside the contour (points that were negative and became positive) and put them in the set S^- . We then update the valid neighborhood of points $\in S^-$ by adding -1 to their valid neighborhood count.

3.5 Results

It is interesting to evolve a curve using Equation 53 on a blank image so that $\phi = 1$ just to see the effect that our flow has on the evolution of a small near-circle. Without the second and third term of the equation, the curve evolution would be a simple dilation of the circle, as shown in Figure 19. We then ran the full curve evolution using all terms of Equation 53, with 2 different radii ($r = 20, r = 40, \alpha = 0.75$). As seen in Figures 20(a) and 20(b), the curve evolves keeping a "tubular" shape at all times. The



Figure 19: Dilation Flow without Shape Prior





(b) Evolution in time, $p{=}2$, $r{=}40$, $\alpha{=}0.75$

Figure 20: Dilation Flow with Shape Prior

width of the tube depends on the radius of the local filter. This demonstrates that our shape prior (second and third terms of Equation 53) has the desired effect since the optimal curve is one that keeps a tubular shape, with a width proportional to the radius value. We also verified that if we ran the full evolution on a perfect circle, then the evolving shape does not become tubular since all points on the curve have the same values for the second and third terms of Equation 53, therefore all points evolve at the same speed and the perfect circle shape remains circular. However, with a near-circle, some points that do not lie on the perfectly circular portion of the curve have a slightly different speed due to their non-uniform neighborhood.

We now test our flow on both 2D and 3D datasets for synthetic and real data. For all flows presented, the user specified the expected biggest radius by clicking two points on the image.

3.5.1 2D Images

In 2D, we first used our flow evolution on a synthetic model of a branching vessel. The radius was chosen to be the largest width of the vessel. We observed that the value of the chosen α influences the penalty on deviations from tubular structures. In Figure 21(a), for $\alpha > 0.65$, we observe an erosion around the branching area since points in that region have a higher measure. Figure 21(b), for $\alpha <= 0.65$, the penalty is softened and the vessel is correctly segmented. This value is used for subsequent segmentations and erosion in branching areas was not observed.

We used our flow on a noisy projection of an angiogram and compared it to the base flow without a shape constraint. We show details of the segmentation where leaks were detected. When the neck of the leak is thin, the leak disconnects from the



(a) A high penalty α =0.75 causes ero-(b) α =0.65. Erosion around the branch sion around the branch. is not observed.

Figure 21: Flow with Shape Prior on 2D Synthetic Images



(a) Base Flow, no Shape Prior, 100 iterations



(b) Flow with Shape Prior, at t=50(left), 100(middle) and 200(right) iterations.

Figure 22: Vessel Flow on Angiogram Images

main vessel as shown in Figure 22. This is because points on the contour near the leak region have a high ϵ_2 measure (as can be seen in Figure 18(c)) that causes the contour to erode and eventually pinch off. Once the leak pinches off, the user can then click on the disconnected contour to eliminate it, or the algorithm can detect the change of topology [29] and automatically remove the leakage. The contour points at the neck of the leak are frozen so that the leak does not re-appear while the rest of the contour evolves.

In Figure 23 we observe that the flow with a shape prior is able to prevent many leaks and produce a flow that is much better behaved than the same flow without a shape prior. We notice that without a shape prior, the flow becomes "chaotic" and leaky regions merge with vessel regions (see Figure 23(a)). The repair of



(a) Base Flow, no Shape Prior



(b) Flow with Shape Prior

Figure 23: Vessel Flow on Angiogram Images

such leaks with user interaction would almost amount to a manual segmentation. Figure 23(b), shows that the flow is much better behaved and produces a better segmentation for most of the image. In the left part of the image, we notice an interesting behavior of the shape-constrained flow. This part of the image is very noisy and image statistics are almost identical inside and outside the vessel, so the base flow without a shape prior completely leaks out of the vessel area. The flow with a shape prior also leaks into the background since the image statistics no longer discriminates between foreground and background, but the flow maintains a "vessellike" shape due to the shape constraint and the leak is mostly contained. This is important since we do not want the leak to expand to areas of the image that are correctly segmented.

3.5.2 3D Images

In this section we demonstrate our flow on two 3D CT datasets of a coronary artery. The mathematics derived for the 2D case extend to the 3D case and the level set numerical scheme is easily extended to the 3D case as well. In all Figures, the color on the surface represents the measure ϵ_2 . We see that this measure is closely related to the thickness of a vessel.



(a) Base Flow with no (b) Flow with Shape Prior, (c) Flow with Shape Prior, Shape Prior 400 iterations 800 iterations

Figure 24: Different Flows on the first CT Coronary Data



(a) Initial Leak into an adjacent (b) Leaking structure (c) Final Segmentation, from a different viewpoint

Figure 25: Base Flow with Shape Prior on the second CT coronary dataset

For the first CT coronary dataset, if we use a flow based on image statistics alone, the artery "leaks" into the heart cage (Figure 24(a)). As we can see, the connecting area between the coronary and the heart has the highest measure. However, when we use the flow with the shape constraint, the leak "pinches" off from the main vessel artery (Figure 24(b) and 24(c)). The second CT coronary dataset leaks into an adjacent structure (Figure 25, left). Again, running the vessel flow separates the coronary from the leak. The user can then click on the isolated leak to remove it. The full segmentation is shown on the right.

We note that in the future, it would be interesting to fully validate our algorithm by using ground truth vessel segmentations by physicians. In order to do so, it would be nice to include an automatic way to remove parts of the vessel that have split away from the main vessel contour, in order to reduce the amount of user interaction during the segmentation.

3.6 Conclusion

We have presented a soft shape prior that can be combined with any other image force to deform an active contour and penalize leaks. We found that the shape prior successfully penalizes leak regions and either disconnects the leak from the vessel, or contains the leak. We find these results encouraging since in the presence of noise, the shape driven flow is better behaved than the flow based on image statistics alone. This flow can be combined with minimal user interaction to repair leak effects that were not prevented by the algorithm.

CHAPTER IV

MULTI-SCALE 3D SHAPE ANALYSIS USING SPHERICAL WAVELETS

The characterization of local variations specific to a shape population is an important problem in medical imaging since a given disease often only affects a portion of the surface of an organ. One motivation for our work is the study of schizophrenia, a multi-faceted illness affecting 1% of the US population and consuming a significant portion of the health-care budget (estimates of yearly costs are 60 billion dollars) [67]. One important contribution of medical imaging in the study of schizophrenia is the segmentation and shape analysis of selected brain structures, such as the hippocampus or the caudate nucleus, in order to find differences between groups of healthy and diseased patients and localize those differences to better understand the effect of the disease.

Currently such segmentations are typically carried out by hand. An automatic tool would be a great advance if it were reliably and reproducibly segment cortical structures for multiple patients, across multiple time points. After the shapes are segmented the geometrical differences between brain structures of patients with schizophrenia and patients without can be studied.

Figure 26 shows a rendering of left caudate nucleus along with an MRI slice in the coronal and sagittal view, as well as three typical surfaces from our dataset. The caudate nucleus is located in the basil ganglia, a group of nuclei in the brain associated with motor and learning functions [68]. Figure 27 shows the same information for the left hippocampus. The hippocampus is a part of the brain located inside the temporal lobe. It forms a part of the limbic system and plays a part in memory



Figure 26: (a-b) Coronal and Sagittal view of the left caudate nucleus (c-e) Example of 3 shapes from the left caudate nucleus dataset

and navigation [69]. As can be seen on Figure 26 and Figures 27, those structures contain sharp features that could be important in shape analysis [61]. An automated segmentation of such structures must therefore be highly accurate and include high frequency variations in the surface. Since shape representation is a key component of the segmentation, it must be descriptive enough to express shape variations at various frequency levels, from low harmonics to sharp edges. Additionally, a shape representation that encodes variations at multiple scales can be useful in itself as a rich feature set for shape analysis and classification.

As discussed in Section 2, medical object segmentation with deformable models and statistical shape modeling are often combined to obtain a more robust and accurate segmentation [71, 11, 36, 64, 49]. In that framework, a joint prior probability over shape parameters is learned using a training set in order to learn the existing shape variations in the population to better constrain the parameter values during the segmentation process.



Figure 27: (a-b) Coronal and Sagittal view of the left hippocampus (c-e) Example of 3 shapes from the left hippocampus dataset

To encode relevant variations in a population of shapes at multiple scales, a decomposable shape representation targeted to the population seems natural, where the shape parameters are separated into groups that describe independent global and/or local biological variations in the population, and a prior induced over each group explicitly encodes these variations.

Wavelet basis functions are useful for such a representation since they range from functions with global support to functions localized both in frequency and space, so that their coefficients can be used both as global and local shape descriptors, unlike Fourier basis functions or principal components over landmarks which are global shape descriptors. The use of spherical wavelet basis in medical imaging has been investigated for statistical shape analysis and segmentation in 2D imagery [14] but not yet for surfaces in 3D imagery. This work addresses this gap.

In the remaining sections of this chapter, we first motivate the need for a multiscale representation by showing the limitations of principal component analysis (PCA) as a shape representation in Section 4.1. We then describe the spherical wavelet transform in Section 4.2 and our shape representation based on the spherical wavelet transform in Section 4.3. In Section 4.4, we detail the construction of a scale-space prior over the wavelet coefficients for a population of shapes and evaluate this prior in a reconstruction task both for a synthetic dataset and 3D brain structures in Section 4.5.

4.1 Motivation for a multiscale parametric shape representation

4.1.1 Implicit versus Parametric representation

In this chapter, we consider 3D shapes with a spherical topology, meaning that those shapes have no holes or handles. We chose to represent shapes using a parametrized surface as opposed to distance maps used in the level set framework since we do not expect topological changes during the segmentation process. This representation can be more accurately and efficiently manipulated compared to a distance map representation of the shape that is higher-dimensional and increases in complexity if the volume is supersampled. Surface meshes can be represented at any level of accuracy desired (subvoxel accuracy), while still maintaining a fixed set of parameters (typically a set of sample points).

Additionally, as long as topology changes are not expected during the segmentation evolution, a parametrized active contour can be used in this framework which is more efficient than a geometric active contour implemented via the level sets technique due to the lower number of parameters to evolve.

4.1.2 Statistical Modeling of a population of shapes with parametric representation

As discussed in Section 2.2.3, a popular parametric representation for a population of shapes is the Point Distribution Model (PDM). Each shape is represented by a set of N discrete 3D points connected that form a mesh (piecewise linear surface). The shapes are assumed to be registered, meaning that the shapes are all represented by N 3D points, where the k^{th} points across shapes are assumed to correspond to the same anatomical point.

Each shape can be described by its three coordinate functions, $x, y, z \in \mathbb{R}$ such that the k^{th} shape S_k is a column vector of size 3N:

$$S_k = [x_k(1), ..., x_k(N), y_k(1), ..., y_k(N), z_k(1), ..., z_k(N)]^T.$$

Since all vertices in the shape population are registered, we interpret each entry of S_k as a random variable and each shape as a realization from a multivariate probability density function. We assume that a population of K shapes can be described by a mean shape $\overline{S} = \frac{1}{K} (\sum_{k=1}^{K} S_k)$ and a set of M transformations $T = [T_1, ..., T_M]$ that describe the variability observed in the population. Each transformation vector T_m is of size 3N where the i^{th} entry is a transformation applied to the i^{th} entry of the mean shape with a corresponding magnitude $\beta_m \in \mathbb{R}$. This representation is shown



Figure 28: Shape Model

in Figure 28.

Each transformation vector (also called variation mode) can be characterized by :

- 1. scale: the transformation vectors T_k can be global and apply to all vertices (all entries of T_k are non-zero) or local (T_k is a sparse vector).
- 2. spatial location: the effect of the transformation vector T_k can be localized in space (non-zero entries of Tk are close spatially), or not.
- 3. magnitude: the value of each β_k

We postulate that characterization of local variations could be important for shape analysis since a disease, such as cancer, could affect only a portion of an organ's surface. Therefore descriptive shape priors should discern shape variations at different scales and spatial location.

4.1.3 Statistical Modeling using PCA and its limitations

As we have previously described in Section 2.2.3.3, in the Point Distribution Model (PDM) of Cootes and Taylor [12], the joint shape probability function is assumed to be a multivariate Gaussian probability function. It is estimated over a set of landmarks on the shapes using principal component analysis (PCA). A covariance

matrix is built from the data, and a diagonalization of the covariance matrix provides eigenvectors that are the principal axes of the function (also called principal modes). The eigenvectors span a space of shapes and the eigenvalues provide the coordinate of the shapes seen in the training set in that space. Using the notation introduced previously, each eigenvector is a transformation vector T_m and each corresponding eigenvalue is a magnitude β_m and therefore each shape in the population should be expressed as a linear combination of the eigenvectors found by PCA, with a reasonable bound on the eigenvalues (usually within certain standard deviation of observed eigenvalues in a representative set of the population).

However as previously described, using PCA on PDM to represent shape variations in a population has two major limitations. First, it often restricts deformable shape too much, particularly if it has been trained on a relatively small number of samples since the number of principal components extracted from diagonalizing the covariance matrix is bound by the number of training shapes. Indeed, for a training set of Kshapes with N vertices, typically $N \gg K$ and therefore the rank of the data covariance matrix will be at most K-1. Therefore if there exist M shape variation modes, such that $M \gg K - 1$, then only the first K - 1 processes that account for the most variation in the data (the most global processes) will be represented. Second, PCA favors the discovery of *qlobal* variations over *local* variations since it can be shown that the eigenvectors of the covariance matrix describe the longest axes of the ellipsoid, and thus the most significant modes of variation in the vertices [11]. This means that for a limited training set size, only the first K-1 processes that account for the most variation in the data (the most global processes) will be represented, hence finer, more local variations of shapes are often not encoded given a limited training set.

To illustrate this limitation, we present an experiment on synthetic shapes. We simulate a simple shape present in a population with a combination of local and global transformations. To this end, we produce a base shape that is an ellipsoid shown in the top left of Figure 29. We then apply 7 independent global non-linear transformations to the base shape (with its vertices denoted by x_0, y_0, z_0), varying the magnitude according to a normal probability function with a fixed mean and variance for each transformation:

$$x = x_0 + \beta_1 * y_0^2 + \beta_2 (y_0/4)^3,$$
(58)

$$y = y_0 + \beta_3 * (x_0/4)^3 + \beta_4 * (z_0/4)^3,$$
(59)

$$z = z0 + \beta_5 (x_0/4)^3 + \beta_6 * y_0^2 + \beta_7 (y_0/4)^3,$$
(60)

The effect of each transformation is shown in Figures 29.

We then add 2 smaller processes that vary in scale and spatial location. The first process is a bump on negative z where $|y_0| < 4 \& |x_0| < 4$ with magnitude β_8 , the second process is a bump on positive z where |y| < 4 & |x| < 4 with magnitude β_9 . The effect of each transformation is shown in Figures 30.

We produce 15 shapes by sampling the values of β from a normal probability function with a fixed mean and standard deviation for each β . The first 10 resulting shapes are shown in Figure 31.

Using a procedure that will be described in Section 4.3.2, we register the shapes so that each shape is described by N vertices. We then perform a PCA analysis on a training set of size 8 and obtain a set of 7 eigenvectors and eigenvalues.

As expected the first few eigenvectors describe the most global variations, and no single eigenvector describes only the local variations, as seen in Figures 32. We see that the local variations are included in most eigenvectors, as part of more global deformations.

To test whether the preceding eigenvector basis is an expressive shape prior, we can project a test shape not included in the training set unto the new eigenvector basis and see how much the projected shape differs from the original test shape.



Figure 29: Synthetic Shape Global Deformations



Figure 30: Synthetic Shape Local Deformations

We show the results for two training shapes in Figures 33(a-d)-34(a-d). In both figures, subfigures (a) and (b) show two views of a test shape. Subfigures (c) and (d) show the projection of the test shape onto the PCA space for the same two views. We see that the prior is not descriptive enough to model the small scale variations in the test shape. For test shape 1, the bottom bump does not appear in the reconstruction. This is due to the fact that no single eigenvector models just the variation of the bottom bump since all eigenvectors model the more global deformations in the data. For test shape 2, the bottom bump appears when it should not, due to the fact that the two bumps are modeled by the same eigenvector.



(j) Shape 10

Figure 31: Synthetic Shapes



(a) Mode 1, magnitude (b) Mode 1, magni- (c) Mode 2, magnitude (d) Mode 2, magni-2 tude 2 -2 tude 2



(e) Mode 3, magnitude (f) Mode 3, magnitude (g) Mode 4, magnitude (h) Mode 4, magni-2 2 -2 tude 2



(i) Mode 5, magnitude (j) Mode 5, magnitude (k) Mode 6, magnitude (l) Mode 6, magnitude -2 2 -2 2



(m) Mode 7, magni- (n) Mode 7, magni- tude -2 tude 2

Figure 32: Visualization of the deformation of the mean shape along the eigenvectors found by PCA. The transparent shape is the reference mean shape with no deformation. For each figure, the mean shape is deformed along a particular eigenvector (or mode) according to a certain magnitude.



(e) WDM Reconstructed Shape, (f) WDM Reconstructed Shape, lateral view bottom view

Figure 33: Reconstruction experiment for Test Shape 1 using PCA as a shape space compared to WDM (technique presented in this chapter)



(e) WDM Reconstructed Shape, (f) WDM Reconstructed Shape, lateral view bottom view

Figure 34: Reconstruction experiment for Test Shape 2, using PCA as a shape space compared to WDM (technique presented in this chapter)



Figure 35: A wavelet transform is applied to the parametric functions representing a deformable contour. The resulting wavelet coefficients are then grouped into 64 bands, and the joint probability function of each band is estimated from the available training samples, via its mean and covariance matrix. This effectively transforms the covariance matrix of the full joint probability function into a matrix that is close to, but not necessarily exactly, a block diagonal matrix. The submatrix corresponding to band B reflects global shape characteristics, whereas the submatrix corresponding to B reflects local shape characteristics at a particular segment. B reflects local shape characteristics of a neighboring segment

To address this issue, a decomposable shape representation targeted to the population seems natural, where the shape parameters are separated into groups that describe independent global and/or local biological variations in the population, and a prior induced over each group explicitly encodes these variations. We describe such a prior in the remainder of this chapter. As a preview, we show in subfigures 33(e,f))-34(e,f) the result of projecting the test shape on the Wavelet Distribution Model (WDM) prior described in the remainder of this chapter. As we can see, local variations are better captured by the WDM prior.

4.1.4 Related Work

As previously stated, we would like a decomposable shape representation where the shape parameters are separated into groups that describe independent global and/or local biological variations in the population. As discussed in 2.2.2.2, previous techniques have expressed functions representing a surface into a complete set of spherical harmonics basis functions (SPHARM). The advantage of this representation is that the coefficients of the spherical harmonic functions of different degrees provide a measure of the spatial frequency constituents that comprise the structure. However, due to the global support of the spherical harmonic functions, the coefficients cannot be used to identify *where* on the surface the frequency content appears. Spherical wavelet functions can address this shortcoming since they have local support at various resolutions. In [74], the authors showed that a spherical wavelet basis can capture shape changes with fewer coefficients than a spherical harmonic basis.

The authors in Davatzikos *et al.* [14] have proposed a hierarchical active shape model framework for contours in 2D medical imagery using standard 1D wavelets, with convincing results. The basic idea is to use the wavelet transform [39] to produce a scale space decomposition of the signal. The authors apply a wavelet transform to the parametric functions (x, y, z) representing a deformable contour. One nice property of the wavelet transform is that it is often a *decorrelator* of real-world signals [39] and therefore the covariance matrix of the wavelet coefficients is sparse. Using this property, the authors approximate the full covariance matrix of the wavelet coefficients as a matrix that is block diagonal, when rearranging the coefficients in the right order, as shown in Figure 35. Coefficients that belong to the same *band* make up a diagonal block of the covariance matrix. Coefficients are grouped into bands using a logarithm tree to divide the space-frequency domain. This groups coefficients of the same scale and nearby spatial location in the same band, following the assumption that only coefficients close in space and scale are closely correlated. Each diagonal block can then be statistically modeled independently of the rest of the matrix and eigenvectors are extracted for each diagonal block, bringing the total number of eigenvectors to (approximately) B(K-1) if there are B blocks and Ktraining shapes. The eigenvectors corresponding to bands at coarse scales reflects global shape characteristics, whereas the eigenvectors corresponding to bands at finer scales reflect local shape characteristics at a particular segment of the curve. Using this technique, the authors show that a segmentation using the wavelet shape prior is more accurate than a segmentation with traditional active shape models.

4.1.5 Our Contributions

In this chapter, we propose to extend the framework proposed by Davatzikos et al in two novel ways. First we describe a multi-scale representation of surfaces in 3D medical imagery using conformal mapping and spherical wavelets. Spherical wavelets have been used primarily by the computer graphics community to generate multiresolution description of 3D shapes but have not yet been widely used by the medical imaging community[52].

Further, we present a novel algorithm to discover optimal multi-scale bands from the data. Our technique is different from Davatzikos et al [14] where the authors cluster coefficients of spatially adjacent bases into bands in each frequency plane. In this work, we cluster highly *correlated* coefficients into a band, with the constraint that coefficients across bands have minimum cross-correlation. Such a decomposition and its visualization can in itself be interesting for shape analysis.

We first present the theory of Spherical Wavelets in Section 4.2. We then explain our shape representation using spherical wavelets in Section 4.3. In section 4.4, we explain our framework to conduct statistical population analysis using spherical wavelets. We derive and compare a number of shape priors based on spherical wavelets to pick the optimal scale-space band decomposition when building the prior,



Figure 36: Recursive Partitioning of an icosahedron: successive levels of the triangulation are obtained by subdividing triangles into 4 children triangles

and present our results.

4.2 Spherical wavelets

A spherical wavelet basis is an L^2 basis composed of functions defined on the sphere that are localized in space and characteristic scales and therefore match a wide range of signal characteristics, from high frequency edges to slowly varying harmonics [39]. The basis is constructed of scaling functions defined at the coarsest scale and wavelet functions defined at subsequent scales.

In this work, we use the discrete biorthogonal spherical wavelets functions defined on a 3D mesh proposed by Schröder and Sweldens [52, 51]. These are secondgeneration wavelets adapted to manifolds with non-regular grids. The main difference with the classical wavelet is that the filter coefficients of second generation wavelets are not the same throughout, but can change locally to reflect the changing (non translation invariant) nature of the surface and its measure. This means that wavelet functions defined on a mesh are not scaled and shifted versions of the function on a coarser grid, although they are similar in shape, in order to account for the varying shape of mesh triangles. The second-generation spherical wavelets that we use are defined on surfaces which are topologically equivalent to the unit sphere (S) and



Figure 37: The wavelet transform of a random scalar signal defined on spherical mesh. The mesh is an icosahedron subdivided 3 times. The original signal is decomposed into a series of high-pass (HP) and low-pass (LP) coefficients.

equipped with a multiresolution mesh. A multiresolution mesh is created by recursively subdividing an initial polyhedral mesh so that each triangle is split into 4 "child" triangles at each new subdivision (resolution) level. This is done by adding a new midpoint at each edge, and connecting midpoints together.

This process is shown on Figures 36(a)-Figures 36(e). The starting shape is an icosahedron with 12 vertices and 20 faces, and at the fourth subdivision level, it contains 5120 faces and 2562 vertices. Any shape (not necessarily a sphere) that is equipped with such a multiresolution mesh can be used to create a spherical wavelet basis and perform the spherical transform of a signal defined on that mesh.

The transform is shown pictorially in Figure 37. We start with a random scalar

signal defined on a spherical mesh produced by subdividing an icosahedron 3 times (therefore with 642 points) in the first row of the Figure. In the second row, the signal is decomposed into a high pass component by projecting it onto a set of spherical wavelet basis functions that represent high pass filters (note the local support of each function). We only show 2 of those functions, but in total there are 480 basis functions for that high frequency. These basis functions are said to be of resolution 3 because they are associated with points on the mesh that were introduced at the third subdivision of the icosahedron (480 new points). At that resolution, what differs for each function is the location of the "center" of the function, that is the point on the sphere where the function has its highest value. In the third row, the remaining signal is again decomposed into a high pass component by projection onto another set of spherical wavelet functions that represent band pass filters (note that the support of each function on the sphere is larger than the previous functions in the row above). Here again we only show 2 functions but there are in total 120 functions for that medium frequency of resolution 2 (the 120 new points added after the second subdivision). Finally in the fourth row, the remaining signal is again decomposed into a high pass component by projection onto band pass spherical wavelet functions of resolution 1 (30 of them) and a low pass component by projection onto scaling functions of resolution 0 (12 of them). Scaling functions also exist at other resolution levels but only the scaling functions of resolution 0 are in the basis, the other scaling function are only used to build wavelet functions, as will be described next. At every row, the coefficients denoted $\gamma_{j,k}$ shown on the Figure represent the value of the projection of the signal onto the spherical wavelet function at resolution i and centered at point k. Therefore the set of all coefficients represent the initial function in the spherical wavelet domain.

We now sketch the construction of the spherical wavelet and scaling functions, the analysis step that transforms a signal on the sphere to the wavelet domain and the



(f) wavelet function, (g) wavelet function, (h) wavelet function, resolution 2, position 2 resolution 3, position 1 resolution 3, position 2

Figure 38: Visualization of Scaling and Wavelet Basis Functions. The color corresponds to the value of the function at each point on the sphere. Notice that the support of each basis gets smaller as the resolution increases

synthesis step that transforms a signal in the wavelet domain back to a signal defined on the sphere.

4.2.1 Grid Structure for Spherical Wavelet Construction

Spherical scaling functions and spherical wavelets functions are defined on a multiresolution mesh as shown in Figure 36. On this mesh, we denote the set of all vertices obtained after j subdivisions with an index set K(j) (see Figure 39). The $j + 1^{st}$ resolution mesh is obtained by introducing new nodes, identified by an index set M(j) which subdivide existing edges (typically at their midpoint, but this is not a requirement). The complete set of nodes in the $j + 1^{th}$ resolution mesh is given by $K(j + 1) = K(j) \bigcup M(j)$. Figure 39 represents a portion of a triangular surface mesh at resolution j + 1, here $K(j) = [k_0, ..., k_6]$, $M(j) = [m_7, ..., m_{12}]$ and $K(j + 1) = [k_0, ..., k_6, m_7, ..., m_{12}]$. Therefore, for the icosahedron case shown in Figures 36(a)-Figures 36(e), K(0) = 12, M(0) = 30, K(1) = 42, M(1) = 120, K(2) =162, M(2) = 480, K(3) = 642, M(3) = 1920, K(4) = 2562.



Figure 39: Refinement of the surface grid by subdivision of the edges

4.2.2 Scaling Function

A scaling function is a scalar function defined on a spherical mesh of a chosen resolution r (icosahedron subdivided r times) denoted by $\varphi_{j,k} : \mathbb{S} \to \mathbb{R}$ where j is the scale (or resolution) of the function and $k \in K(j)$ is a spatial index that indicates where on the surface the function is centered. The initial functions are constructed at the highest resolution (r) and there is one function centered at every point in the set K(r). The typical function is delta function defined to be 1 at its center and 0 everywhere else.

We use the notation \mathbf{x} to denote a point (vertex) on the surface. As in the onedimensional case, the scaling function on the *jth* resolution mesh can be expressed as a combination of scaling functions on the j + 1st resolution mesh:

$$\varphi_{j,k}(\mathbf{x}) = \varphi_{j+1,k}(\mathbf{x}) + \sum_{m \in n(j,k)} h_0^j[k,m]\varphi_{j+1,m}(\mathbf{x})$$
(61)

where n(j, k) is the set of neighboring subdivision points at the j + 1 level that share an edge with the vertex \mathbf{x}_k at level j (in the Figure $39, n(j, k) = [m_7, ..., m_{12}]$). At every resolution these functions are "hat-shaped" and vary linearly from the value of 1 at their center vertex to 0 at at the vertices that share an edge with the center point at the same resolution (in the Figure 39, the hat function would be 1 at x_{k_0} and 0 at $x_{k_1} - x_{k_6}$). Therefore as the resolution j decreases, the spatial support of the scaling function on the spherical mesh increases. The top left in Figure 38 shows two scaling functions centered at different vertices for resolution 0. The color on the sphere indicates the value of the function $\varphi_{0,k}$ at every point on the sphere.

4.2.3 Wavelet Function

At resolution level j, a wavelet function is defined for every vertex $m \in M(j)$: $\psi_{j,m}$: $\mathbb{S} \to \mathbb{R}$.

At a particular scale j, wavelet functions are combinations of resolution j and (j + 1) scaling functions:

$$\psi_{j,m}(\mathbf{x}) = \varphi_{j+1,m}(\mathbf{x}) - \sum_{k \in A(j,m)} h_1^j[k,m]\varphi_{j,k}(\mathbf{x})$$
(62)

where A(j, m) is a neighborhood of node m. The simplest choice for A(j, m) is to use the two immediate neighbors, e.g. the parents of the subdivision point m. Wavelet functions capture finer features since they are composed of higher resolution (j + 1)scaling functions. Figures 38 show scaling and wavelet functions for different values of j and m. Note that the support of the functions becomes smaller as the resolution increases.

4.2.3.1 Spherical Wavelet transform of functions:

Together, the coarsest level scaling function and all wavelet scaling functions construct a basis for the function space L^2 (all functions of finite energy):

$$L^{2} = \{\varphi_{0}, k | k \in N_{0}\} \bigcup \{\psi_{j}, m | j \ge 0, m \in N_{j+1}\}.$$
(63)

A given scalar data function defined on the sphere $f : \mathbb{S} \to \mathbb{R}$ can be expressed in the basis as a linear combination of basis functions and coefficients:

$$f(\mathbf{x}) = \sum_{k} \lambda_{0,k} \varphi_{0,k}(\mathbf{x}) + \sum_{0 \le j} \sum_{m} \gamma_{j,m} \psi_{j,m}(\mathbf{x}).$$
(64)

This is the Inverse Wavelet Transform. For the Forward Wavelet Transform, the coefficients $\gamma_{j,m}$ are calculated by inner product between the data function f and a dual wavelet function $\psi_{j,m}^{-1}(\mathbf{x})$ such that the inner product $\langle \psi_{j,m}, \psi_{l,n}^{-1} \rangle = 1$ when j = l and n = m and $\langle \psi_{j,m}, \psi_{l,n}^{-1} \rangle = 0$ otherwise. This dual function is needed since the wavelet basis is not orthogonal.

As shown pictorially in Figure 37, scaling coefficients $\lambda_{0,k}$ represent the low pass content of the signal f, localized where the associated scaling function has support, whereas wavelet coefficients $\gamma_{j,m}$ represent localized band pass content of the signal, where the band pass frequency depends on the resolution of the associated wavelet function and the localization depends on the support of the function.

Why are scaling functions of resolution 0 low pass filters and wavelet functions band pass filters? At the lowest resolution level j = 0, there are only K(0) = 12scaling functions and projecting a function onto each scaling function $\varphi_{0,k}$ results in an averaging of the function f in a large neighborhood. Projecting a data function f onto a wavelet function of resolution $j \psi_{j,m}$ defined in Equation 62 is a difference between a projection onto scaling functions of resolution j + 1 and a projection onto scaling functions of resolution j. That difference is the content lost from the smoothing operation when projecting onto scaling functions of resolution j + 1. Therefore the wavelet function can be viewed as a band pass filter.

4.2.3.2 Discrete Spherical Wavelet Transform

The algorithm for the fast discrete spherical wavelet transform (FSWT) is given in [51]. Here, we sketch the transform algorithm in matrix form which gives a more compact and intuitive notation for the rest of this thesis. In practice, we use the FSWT in our implementation.

If there exist N vertices on the highest resolution mesh (j subdivisions), a total of N basis functions are created, composed of K(0) scaling functions and
$M(0) \bigcup M(1) \dots \bigcup M(j)$ wavelet functions. In the remainder of this thesis, we refer to all basis functions as wavelet basis functions as a shorthand.

In matrix form, the set of basis functions can be stacked as columns of a matrix Φ of size $N \times N$ where each column is a basis function evaluated at each of the N vertices. The basis functions are arranged by increasing resolution (subscript j) and within each resolution level, by increasing spatial index (subscript k). Since the spherical wavelet functions are biorthogonal, $\Phi^T \Phi \neq Id$ (the identity matrix), so the inverse basis Φ^{-1} is used for perfect reconstruction, since $\Phi^{-1}\Phi = Id$.

Any finite energy scalar function evaluated at N vertices, denoted by the vector F of size $N \times 1$, can be transformed into a vector of basis coefficients Γ_F of size $N \times 1$ using the matrix version of the Forward Wavelet Transform:

$$\Gamma_F = \Phi^{-1}F,\tag{65}$$

and recovered using the matrix version of the Inverse Wavelet Transform:

$$F = \Phi \Gamma_F. \tag{66}$$

The vector of coefficients Γ is composed of coefficients associated with each basis function in Φ . It contains scaling coefficients as its first N_0 entries, then wavelet coefficients associated with wavelet functions of resolution 0 for the next N_1 entries, and so forth, all the way to wavelet coefficients of resolution R for the last N_{R+1} entries. Therefore in total the vector contains N entries. Next, we describe how to represent shapes using spherical wavelets.

4.3 Shape Representation using spherical wavelets

In this section, we explain how to equip a set of anatomical shapes with the correct multiresolution mesh structure in order to build wavelet functions directly on a mean shape that is representative of the population. We then explain how to encode a shape signal into wavelet coefficients.



Figure 40: Example of 5 shapes from the Prostate dataset

4.3.1 Data Description

We will use three real datasets to illustrate our technique in the remainder of this chapter. The first dataset of prostates was chosen as a test dataset due to the existence of some interesting high frequency content on the surface (for example on the superior side of the surface). We use a dataset of 39 prostates obtained from pre-operative 1.5T MRI scans (axial, T2-weighted, 120mm field of view, matrix size 256×256 , 3.0mm thickness, 0.5mm gap, 1.5T, using both endorectal and pelvic coil arrays) using endorectal coil imaging provided to us by the Surgical Planning Lab of Brigham and Women's Hospital. In these images the prostate capsule is visible and was manually segmented by a radiologist. Each manual segmentation defined a 3D surface which was extracted as a triangulated surface using the Marching Cubes algorithm, with examples shown in Figure 40.

The other two datasets are two key brain structures. We use a dataset of 29 left caudate nucleus structures and a dataset of 25 left hippocampus structures from a 1.5 Tesla GE Echospeed MR system, coronal SPGR images, 124 slices of 1.5 mm thickness, voxel dimensions $0.9375 \times 0.9375 \times 1.5$ mm. The MRI scans were hand-segmented by an expert neuroanatomist to provide ground truth segmentations for each structure. Each manual segmentation defined a 3D surface for each structure extracted by a standard isosurface algorithm with examples shown in Figures 26 and 27.



Figure 41: Illustration of the remeshing step for two left hippocampus shapes. See Section 4.3.2 for details.

4.3.2 Shape Remeshing and Registration

Before we can perform our wavelet analysis, we need to re-triangulate and register all the surfaces in the dataset, so that they have the same mutiresolution mesh and mesh nodes at corresponding anatomical locations. One approach to surface registration is based on the theory of conformal (angle-preserving) mappings of surfaces with spherical topology [1]. Regardless of the degree of surface variation, such as variations in convexity to concavity, the method efficiently *unfolds* each surface, yielding an analytic one-to-one (conformal) mapping of each surface onto the sphere, and between each pair of surfaces by composition of mappings. Although we have had success with the conformal mapping approach of [1] presented next, we note that the wavelet analysis presented here does not require this particular method of spherical mapping. Indeed, other techniques such as inflation [20], harmonic mapping with rectangular grids [5], circle packing [30], least squares mapping [31], and conformal mapping



Figure 42: Example of principal axes and 3 points \mathbf{x}_N , \mathbf{x}_S and \mathbf{x}_E found on three left hippocampus and left caudate shapes.

with parabolic equations [28] could also be used. The steps of our registration and re-meshing technique are:

4.3.2.1 Conformal Mapping

Let Σ be a surface of spherical topology we wish to register and remesh. As noted above, our registration method is based on complex variables and the conformal mapping of Riemann surfaces. The core of the algorithm requires the solution of a pair of sparse linear systems of equations, and uses finite element techniques to solve an elliptic partial differential equation of the form:

$$\Delta f = \frac{\partial}{\partial w} \delta_p,\tag{67}$$

where Δ denotes the Laplace-Beltrami operator on Σ , p is an arbitrary point on Σ , f is the desired conformal mapping to the sphere \mathbb{S} , δ_p is the Dirac delta function at p, and w denotes a complex conformal coordinate around p. See [1] for details.

The resulting mapping f to the sphere can be made unique by specifying three points on Σ to be mapped respectively to the north pole, south pole, and an equatorial



Figure 43: A new point \mathbf{x}_S is found such that it is the farthest point on the mesh from \mathbf{x}_N (in geodesic distance) for the three left hippocampus and left caudate shapes shown in Figure 42

point on the sphere. Choosing these points consistently helps insure that corresponding surface locations are well registered. We now show how to consistently pick three corresponding points (\mathbf{x}_N , \mathbf{x}_S , \mathbf{x}_E) on each shape of the prostate, hippocampus and caudate population in an automatic way. These three points will be used as boundary conditions of the conformal mapping so that \mathbf{x}_N is mapped to the North Pole of the sphere, \mathbf{x}_S is mapped to the South Pole and \mathbf{x}_E to the point on the equator that intersects the Greenwich meridian. The steps are as follows:

- 1. Let $\Sigma \ (\mathbb{R} \to \mathbb{R}^3)$ be a mesh in RAS patient coordinate space, with N points denoted $\mathbf{x}_1...\mathbf{x}_N$. The RAS coordinate space signifies that the first axis is the patient's left-right (LR) where left is negative and right is positive, the second axis is the patient's posterior-anterior (PA) and the third axis is the patient's inferior-superior (SI).
- 2. For each point \mathbf{x}_i on the mesh, let $a(\mathbf{x}_i)$ be the sum of 1/3 of the triangles area that have \mathbf{x}_i at its vertex. Hence $a(\mathbf{x}_i)$ measures the portion of surface area

attributed to vertex \mathbf{x}_i .

3. Calculate the weighted mean of the surface

$$m = \frac{\sum_{i=1}^{N} a(\mathbf{x}_i) \mathbf{x}_i}{\sum_{i=1}^{N} a(\mathbf{x}_i)}$$
(68)

4. Center shape Σ around its mean:

$$\Sigma' = \Sigma - m \tag{69}$$

5. Extract the principal axes of the shape by finding the eigenvectors V of the weighted moment tensor $(\mathbf{x}_i(k)$ means we are taking the k^{th} coordinate of \mathbf{x}_i):

$$T = \begin{bmatrix} \sum \mathbf{x}_{i}(1)\mathbf{x}_{i}(1) & \sum \mathbf{x}_{i}(1)\mathbf{x}_{i}(2) & \sum \mathbf{x}_{i}(1)\mathbf{x}_{i}(3) \\ \sum \mathbf{x}_{i}(2)\mathbf{x}_{i}(1) & \sum \mathbf{x}_{i}(2)\mathbf{x}_{i}(2) & \sum \mathbf{x}_{i}(2)\mathbf{x}_{i}(3) \\ \sum \mathbf{x}_{i}(3)\mathbf{x}_{i}(1) & \sum \mathbf{x}_{i}(3)\mathbf{x}_{i}(2) & \sum \mathbf{x}_{i}(3)\mathbf{x}_{i}(3) \end{bmatrix}$$
(70)
$$T = T \begin{bmatrix} a(\mathbf{x}_{i}) \\ a(\mathbf{x}_{i}) \\ a(\mathbf{x}_{i}) \end{bmatrix}$$
(71)
$$T = VDV^{T}$$
(72)

- 6. reorder the eigenvectors to most L → R, P → A, I → S. The axes are shown in Figure 42 for three hippocampus and three caudate shapes. The magenta axis is LR, the blue axis is PA and the green axis is IS.
- 7. Define \mathbf{x}_N to be the farthest surface point from the center of the shape in the A direction of the PA axis (shown in blue in Figure 42), \mathbf{x}_S to be the surface point in the P direction of the PA axis (shown in yellow in Figure 42) and \mathbf{x}_E to be the surface point in the L direction of the LR axis (shown in magenta in Figure 42).

8. Since the \mathbf{x}_N and \mathbf{x}_S should be as far as possible, we change the \mathbf{x}_S location to be the farthest point (in geodesic distance) from \mathbf{x}_N . The distance is calculated on the mesh, using Dijkstra's algorithm as an approximation of the geodesic distance. We show the result of this step on Figure 43 for the same six shapes.

This first step of the registration is illustrated in the first two columns of Figure 41, where each row of the first column represents a different initial left hippocampus surface Σ with 3 automatically chosen control points. The color represents the z coordinate of Σ for reference. The second column shows the result of mapping each point on Σ to the sphere using the conformal mapping f. The sphere S then has the same triangulation as Σ . The color shown at the vertices of S is still the z coordinate of the corresponding vertex on Σ .

4.3.2.2 Area Correction

As pointed out in the early spherical mapping work of [5], conformal mappings may result in extreme distortion of area which needs to be corrected for certain applications. In particular, when remeshing a surface using a standard multi-resolution mesh, large distortions in area can result in a non-uniform distribution of mesh nodes on the original surface and a loss of fine detail. To prevent this, we have implemented a simple method to adjust the conformal mapping to have better area-preservation properties. The method works as follows. Let $q_n = \left((1-z_n^2)^{\frac{1}{2}}\cos(\theta_n), (1-z_n^2)^{\frac{1}{2}}\sin(\theta_n), z_n\right)$ for $n = 1, 2, \ldots, N$ represent the mesh points on the sphere, indexed so that $z_1 = -1 \leq z_2 \leq \cdots \leq z_N = 1$. Then z_1 is the south pole (0, 0, -1) and z_N is the north pole (0, 0, 1). For each z_n , the area of the region of the sphere south of the latitude through z_n is given analytically by $2\pi(z_n + 1)$. This region corresponds to a region on the original surface Σ of area A_n , which can be calculated using the triangulation of Σ . In particular, $A_1 = 0$ and $A_N = A$, where A is the total area of Σ . If $\frac{A_n}{A} = \frac{2\pi(z_n+1)}{4\pi}$ for all n, then these areas on Σ and the sphere are spread out proportionally from south to north pole. This is unlikely to be the case in practice, so we adjust each point p_n to get a new point \tilde{p}_n by setting $\tilde{z}_n = \frac{2A_n}{A} - 1$ and setting $\tilde{p}_n = \left((1 - \tilde{z}_n^2)^{\frac{1}{2}}\cos(\theta_n), (1 - \tilde{z}_n^2)^{\frac{1}{2}}\sin(\theta_n), \tilde{z}_n\right)$. This effectively spreads out the areas on the sphere in proportion to their original surface areas. The advantage of this approach is that the algorithm involved does not require iteration, as a functional minimization or flow technique would, and guarantees that the adjusted spherical mapping remains bijective. In the third column of Figure 41, each row shows an adjusted spherical mapping. The color coding is again the z coordinate of the original surface shown in the first column. Comparison of the second and third columns to the first clearly shows that the third has a better distribution of area than the second. We note that although this method has produced satisfactory results for the surfaces we analyze here, more irregular surfaces may require additional area adjustments.

4.3.2.3 Remeshing

We retriangulate S with the vertices of a subdivided icosahedron shown in Figure 36(e). This yields a sphere with a new triangulation denoted by S^R . If we apply the inverse mapping to the vertices of S^R , $f^{-1}(S^R) = \Sigma^R$, we then obtain a retriangulated version of the original surface Σ . After this transformation, the retriangulated shape Σ^R has two nice properties: 1) it has the required mesh for spherical wavelet analysis; 2) it has a one-to-one mapping with a canonical spherical mesh, therefore providing one-to-one correspondence with other shapes that have this property.

4.3.2.4 Registration

After remeshing, all shapes have the same mesh with N vertices. A Procrustes transformation [17] can be applied to all shapes to register them in Euclidean space. This



(d) Remeshed shapes be- (e) Remeshed shapes af- (f) Mean Shape fore alignment ter alignment

Figure 44: After remeshing, shapes are aligned with Procrustes alignment and the Mean shape is computed for (a-c) left hippocampus dataset (d-f) left caudate dataset

result of the Procrustes alignment is shown in Figure 44(a)-44(b) for the left hippocampus dataset and in Figure 44(d)-44(e) for the left caudate dataset. After Procrustes alignment, we denote shape i by a vector Σ_i^P of size $3N \times 1$ (the first N entries are the x coordinates of the vertices, the next N entries are the y coordinates and the last N entries are the z coordinates). After all K shapes are registered, the Mean shape is found with the following equation:

$$\overline{\Sigma^P} = \frac{1}{K} \sum_{i=1}^{K} \Sigma_i^P \tag{73}$$

The mean shape for the left hippocampus dataset is shown in Figure 44(c) and for the left caudate dataset in Figure 44(f).

4.3.3 Spherical Wavelets on the Mean Shape

After the spherical mapping and registration, all shapes in the population are equipped with the same multiresolution mesh, where each vertex of the mesh corresponds to the same anatomical location across shapes. Since the spherical wavelet functions used in



Figure 45: (a-c) Icosahedron mesh shown with the Mean shape(d-f) Visualization of wavelet basis functions constructed on the Mean shape at various levels. The color corresponds to the value of the functions

this work can be defined on any surface of spherical topology with a multiresolution mesh, we can build the basis functions directly on the mesh of a mean shape for a population. This creates a set of basis functions adapted to the geometry of the mean shape since the spherical wavelet transform in [51] takes triangle area into account when building the functions. Therefore the basis on the mean shape is more specific to each shape in the population than if we had used a basis built on a sphere. For each shape, we denote the matrix of basis functions built on the mean shape mesh as Φ_m . A scaling function for resolution 0 and a wavelet function for resolution 2 and 4 are shown in Figure 45(a)- 45(c) on the mean shape of the left hippocampus population.

4.3.4 Encoding the Shape Signal with Spherical Wavelets

We represent each shape in the population by encoding the deviation from the mean using the spherical wavelet transform. We first encode the variation from the mean for the i^{th} shape with the signal v_i of size $3N \times 1$:

$$v_i = \Sigma_i^P - \overline{\Sigma^P} \tag{74}$$

We then transform v_i into a matrix of spherical wavelet basis coefficients Γ_{v_i} with the forward spherical wavelet transform:

$$\Gamma_{v_i} = \underbrace{\begin{bmatrix} \Phi_m^{-1} & 0 & 0\\ 0 & \Phi_m^{-1} & 0\\ 0 & 0 & \Phi_m^{-1} \end{bmatrix}}_{\Pi^{-1}} v_i,$$
(75)

where Φ_m is the wavelet basis functions evaluated on the mean shape for that population. Therefore a shape is transformed into wavelet coefficients by taking the forward wavelet transform of the x, y and z variation from the mean signal. The resulting vector of coefficients Γ_{v_i} contains as the first N entries the wavelet encoding of the x coordinates of the shape, ranked from coarse resolution to high resolution, and similarly for the next N entries that encode the y coordinates, and the last N entries that encode the z coordinate.

4.3.5 Filtering the Shape Signal by Projection onto a Reduced Set of Basis functions

This representation is already useful in itself by allowing us to represent the shape at various resolution levels, by a filtering operation that projects the shape onto a limited set of basis functions. This can be done by creating a filter matrix P of size $M \times N$, where M is the number of basis functions to keep. Each row of the matrix P keeps a particular basis function of index j by having entries of value 0, except for the j^{th} entry that is set to 1. The filtering (for all coefficients) is then performed with the following equation:

$$\Gamma_{v_i}^* = \underbrace{\begin{bmatrix} p & 0 & 0 \\ 0 & p & 0 \\ 0 & 0 & p \end{bmatrix}}_{\mathcal{P}^*} \Gamma_{v_i}.$$
(76)

The resulting filtered coefficients $\Gamma_{v_i}^*$ are of size $3M \times 1$. To display the filtered shape, $\Gamma_{v_i}^*$ is first premultiplied by the transpose of \mathcal{P} to be of size $3N \times 1$ (the coefficients that correspond to the eliminated basis functions are zero), the inverse wavelet transform is applied and the mean shape is added:

$$\Sigma_i^P = \overline{\Sigma^P} + \Pi \mathcal{P}^{*T} \Gamma_{v_i}^*. \tag{77}$$

One example of this projection process is shown in Figure 46 for the left caudate shape 1 and left hippocampus shape 1. Figure 46(a) shows the mean shape for the left caudate population. The following three subfigures show the mean caudate shape plus filtered variations from the mean for shape 1. If only low resolution basis functions are used (Figure 46(b)), the resulting shape is coarse with low frequency variations from the mean shape. If basis functions of higher resolution are added to the projection set, the resulting shape contains additional high frequency variations (Figure 46(c)-Figure 46(d)). The same information is shown for the hippocampus in Figures 46(e)-46(h). This type of filtering (or truncation) operation where a whole resolution level is suppressed is commonly used with Fourier functions, such as spherical harmonics. However, one advantage of spherical wavelets is that due to the local support of its basis functions, a more granular truncation can be done, where only certain basis functions at a resolution level are suppressed, instead of all functions for that level. This allows for a targeted truncation, by selecting only those functions that represent important information in the signal, at all resolution levels. The next section explains how we do this in a principled way for a whole population of shapes.

4.4 Scale-Space Spherical Wavelet Prior for Statistical Population Analysis

To build a prior that captures both global and local variations in the population, we first reduce the dimensionality of the Γ_i coefficients and keep only the coefficients that encode relevant variations in the training data. This process is described in Section 4.4.1. After truncation, we wish to decompose the set of remaining coefficients such that *highly correlated* coefficients in the population are grouped together



Figure 46: Example of the filtering operation for Shape 1 of Left Caudate (a-d) and Left Hippocampus (e-h) dataset.

in a band, with the constraint that coefficients across bands have minimum crosscorrelation. This allows us to model the joint probability function of the coefficients by a product of smaller independent probability functions over each band, assumed to model uncorrelated shape variations. This process is described in Section 4.4.2.

4.4.1 Coefficient Truncation via Power Analysis

Given the total power $||p||^2$ of the shape signal for a population, we would like to remove the basis functions that do not contribute significantly to that power. We define the population shape signal p (size $N \times 1$) by:

$$p(n) := \left(\sum_{i=1}^{K} v_i(n)^2 + v_i(N+n)^2 + v_i(2N+n)^2\right)^{\frac{1}{2}}$$
(78)

where $v_i(n)$ selects the variation from the mean of vertex n of shape i along the x axis, and $v_i(N+n)$ and $v_i(2N+n)$ along the y and z axis respectively.

Since the wavelet basis functions are not orthonormal, we cannot directly apply Parseval's theorem for spectrum analysis. Indeed:

$$\langle p, p \rangle \neq \langle \Gamma_p, \Gamma_p \rangle$$
 (79)

where $\Gamma_p = \Phi_m^{-1} p$ are the coefficients of the spherical wavelet transform of p.

In order to still perform a power analysis, we wish to see the contribution of each wavelet basis function to the total power for the population $||p||^2$. We know that for any signal p sampled at N vertices of the mean shape

$$\langle p, p \rangle = p^T \Phi_m \Phi_m^{-1} p, \tag{80}$$

since $\Phi_m \Phi_m^{-1} = Id$. Then:

$$\langle p, p \rangle = p^T \Phi_m \Gamma_p$$
 (81)

The contribution of the k^{th} wavelet basis function to the sum in (81) is therefore

$$c(k) = p^T \Phi_m(:,k) \Gamma_p(k), \qquad (82)$$

where $\Phi_m(:,k)$ is the k^{th} column of the basis matrix Φ_m and $\Gamma_p(k)$ is the k^{th} element of the coefficients vector Γ_p .

Since we wish to remove basis functions that have no effect on $||p||^2$, we remove those whose contribution c(k) is close to 0. Noting that c(k) can be both positive and negative, we rank the contribution of each basis by their absolute value from highest to lowest, and truncate the lowest basis functions whose cumulative contribution is lower than 0.01% of the total power. Based on this analysis, we build a filter matrix \mathcal{P}^* that indicates which basis functions to keep to represent that population of shapes.

This truncation step leads to a nice compression property since the reduced set of basis functions match variations specific to a shape population, without introducing large error between the filtered shape and the non-filtered shape. Table 2 shows the number of truncated basis functions and the average maximum error between original shapes and their filtered version, using a filter matrix \mathcal{P}^* specific to the shape population, for both the hippocampus and caudate dataset. The error is shown in percentage of the shape's bounding box (smallest axis). We show the amount of truncation and error for varying population size from 5 to 20. For the caudate dataset, between 958 and 1383 basis functions out of 2562 are truncated, depending on the number of shapes used to find \mathcal{P}^* . This represents a compression level between 45% and 54%. With the truncation, the filtered shapes differ less than 2% of their bounding box from the non-filtered shapes. Figure 47(a) shows the original caudate shape 1 and Figure 47(b) shows the filtered shape (based on a 54% truncation level). The colormap shows the distance between the filtered and non-filtered shapes. As can be seen in the figure, the truncation filtering does not seem to effect the shape significantly and all high resolution variations are still present. For the hippocampus dataset, between 780 and 888 basis functions are truncated, representing a compression level between 30%and 35%. Again, the filtered shapes differ less than 2% of their bounding box from the non-filtered shapes. Figure 47(c) shows the original hippocampus shape 1 and Figure 47(d) shows the filtered shape (based on a 35% truncation level). Again, the truncation filtering does not affect the shape significantly and all the high resolution variation is still present.

Figure 48 summarizes the steps from Sections 4.3-4.4.1 to transform a shape into its wavelet representation. Next, we detail the steps to learn the multivariate probability function of the truncated wavelet coefficients for a shape population.

4.4.2 Multiscale Decomposition

After finding a truncated set of basis functions for a population of shapes, we wish to estimate the joint multivariate probability function $P(\Gamma^*)$ of the wavelet coefficients

Dataset	\setminus Num Training	N=5	N=10	N=15	N=20
Caudate					
	# initial fn.	2562	2562	2562	2562
	# truncated fn.	1125	958	1050	1383
	(% truncation)	(44)	(37)	(41)	(54)
	Avg. Max. Error				
	(%bounding box)	1.69	0.99	1.33	1.33
Hipp.					
	# initial fn.	2562	2562	2562	2562
	# truncated fn.	888	820	780	888
	(% truncation)	(35)	(32)	(30)	(35)
	Avg. Max. Error				
	(%bounding box)	0.05	1.07	0.68	0.81

Table 2: Number of truncated basis functions and the average maximum error between original shapes and their filtered version, using a filter matrix specific to the shape population, for both the hippocampus and caudate dataset.



truncation), Shape 1

Figure 47: Result of the Filtering Operation to create a reduced basis set. (a,c) original shapes (b,d) filtered shapes with mean squared error from the original shape as colormap

1



Figure 48: Shape Representation from Sections 4.3-4.4.2

for that population. Each shape of that population is then a random realization from $P(\Gamma^*)$.

4.4.2.1 Motivation

To model the variation in the data, we take advantage of the natural multiresolution decomposition of the wavelet transform and learn variations in the population at every resolution level. This means that small scale variations in the data will not be overpowered by large scale variations, which would be the case if we were to apply PCA directly to all the vertices or to all the wavelet coefficients since PCA is a leastsquares fit that finds the first K - 1 major (large scale) variations in a dataset of Kshapes. By finding variations at separate scales, we find K - 1 variations for each scale of analysis.

One additional benefit of the wavelet transform is that it often acts as a decorrelator of many real-world signals [39]. We computed the correlation matrix of wavelet basis coefficients for each scale, and observed that the matrices were indeed sparse. As a comparison, the correlation matrix of the coordinates of the vertices (used in PDM) is dense.

For a given scale, we can refine our model by taking advantage of this decorrelation property and cluster correlated basis coefficients together, with the constraint that coefficients across clusters have minimum correlation. Coefficients in the same cluster then represent areas of the shape that have correlated variations in the population, for a given scale. Coefficients that do not belong to the same cluster do not tend to be correlated in the population. This further decomposition, learned from the data, allows us to model variations for *each* cluster at a given scale, hence increasing the number of variation modes learned from the data.

We note that this hierarchical decomposition is inspired by the previous work of Davatzikos *et al.* [14] who used 1D wavelet basis functions to analyze shape contours in 2D imagery and performed a scale-space decomposition of the wavelet coefficients. However, their work assumes that coefficients associated with wavelet functions of the same resolution that are also close in space are correlated to each other. In this work, we relax the assumption that only spatial proximity would dictate correlation, and find clusters directly based on the correlation that exists in the data.

In the next section 4.4.2.2 we show how to discover the clusters from the data. In section 4.4.2.4 we show how to learn variations over every cluster and how to combine them into a multiscale prior.



Figure 49: Illustration of Band Creation Algorithm

4.4.2.2 Coefficient Clustering via Spectral Graph Partitioning

Figure 49 shows a simple example used throughout this section to illustrate our clustering algorithm. We used 20 left hippocampus training shapes to create the graph.

To cluster correlated wavelet coefficients, we use a spectral graph partitioning technique [54]. We use a fully connected undirected graph $G = (V_j, E)$ where each node indexed by $n \in V_j$ is a random variable that represents the coefficients associated with the n^{th} wavelet basis function of resolution j. Each wavelet basis function n has 3 associated coefficients per shape i that represent the x, y and z variation, we denote those coefficients by $\gamma_i^{x^n}$, $\gamma_i^{y^n}$ and $\gamma_i^{z^n}$. For each shape i, we combine those three coefficient values associated with basis function n into one variable $u_i^n =$ $\gamma_i^{x^n} + \gamma_i^{y^n} + \gamma_i^{z^n}$. Then the random variable at node n is represented by its K realizations (K is the number of training shapes): $U^n = [u_1^n, ..., u_K^n]$. The weight on the edge that connects node n and m is a function of similarity between those nodes and is denoted w(n,m). To define w(n,m), we first find the sample correlation and p-value between the random variables U^n and U^m :

$$r_{n,m} = \frac{\sum_{j=1}^{K} (U^{n}(j) - \overline{U^{n}}) (U^{m}(j) - \overline{U^{m}})}{(K-1)\sigma_{U^{n}}\sigma_{U^{m}}},$$
(83)

where $\overline{U^n}$ is the sample mean of U^n , σ_{U^n} is the sample standard deviations of U^n and K is the total number of samples (number of shapes in the population).

With the correlation we compute an associated *p*-value that is the probability of getting a correlation as large as the observed value by random chance, when the true correlation is zero. If $p_{n,m}$ is small then the correlation $r_{n,m}$ is significant. We pick a significance threshold of 0.05.

We then define the weight on the edge that connect nodes n and m to be:

$$w(n,m) = \begin{cases} r_{n,m} & \text{if } p(n,m) < 0.05, \\ 0 & \text{otherwise.} \end{cases}$$
(84)

This similarity between the nodes can be represented as a matrix where each entry n, m is the value w(n, m). In Figure 49 the similarity matrix for the hippocampus population for all nodes of resolution 0, is shown at the top left. The lighter the entry (n, m), the more similarity between coefficients of basis n and m.

Using the normalized cuts technique [54], we find the optimal partitioning of the nodes of resolution j in the set V_j^0 into two disjoint sets V_j^1 and V_j^2 such that nodes within a partition have the highest similarity and nodes across partitions have the lowest similarity. For example in Figure 49, nodes 2, 3, 5, 6, 7, 8, 9, 10 and 11 are put in the subgraph V_0^1 and nodes 1, 4 and 12 are put in the subgraph V_0^2 . We show the new similarity matrix where the node indices are re-ordered such that the first contiguous nodes belong to V_0^1 and the next contiguous nodes belong to V_0^2 . This effectively transforms the similarity matrix into a block diagonal matrix, where entries outside of the diagonal blocks have minimum correlation.

For each subgraph, we recursively iterate the normalized cuts until we reach a stopping criterion. The stopping criterion is based on the quality of the decomposition of each graph, validating whether the total correlation between the coefficients separated in two subgraphs V_j^{k+1} and V_j^{k+2} is less than a percentage p of the total correlation between coefficients in the combined set V_j^k . So if V_j^k is partitioned into sets V_j^{k+1} and V_j^{k+2} , we ensure that

$$\sum_{u \in V_j^{k+1}} \sum_{v \in V_j^{k+2}} w(u, v)
(85)$$

In practice, we use $p \leq 0.1$. For example in Figure 49, subgraph V_0^2 is further subdivided into subgraphs V_0^3 and V_0^4 .

After the recursion, each subgraph represents a set of wavelet basis functions whose coefficients are correlated at that scale. We group these wavelet basis functions into a *band*, encoded by an index set $B_{j,b}$ where j is the resolution level of the band and b is the band index. For example in Figure 49, a total of 3 bands were discovered for resolution 0: Band $B_{0,1}$ corresponding to the nodes in subgraph V_0^1 , Band $B_{0,2}$ corresponding to the nodes in subgraph V_0^3 and Band $B_{0,3}$ corresponding to the nodes in subgraph V_0^4 .

4.4.2.3 Visualizing Clustering Results

The visualization of resulting bands on the mean shape can in itself be interesting for shape analysis by indicating which surface patches co-vary across the training set.

To visualize the band $B_{j,b}$, we visualize the cumulative support of all wavelet basis functions in band $B_{j,b}$ on the surface of the mean shape. This can be done by using as a colormap the sum of wavelet basis functions indexed by $B_{j,b}$. The higher (lighter) values of the colormap then indicate where the wavelet basis functions in the band cumulatively have support.

This is shown in Figure 49 at the bottom right for all three bands of resolution



Figure 50: Coefficient Clustering and selected Band Variation Visualization for the left hippocampus data



Figure 51: Coefficient Clustering and selected Band Variation Visualization for the left caudate data

0. Band $B_{0,1}$ indicates correlated variation at the anterior/lateral side of the hippocampus (the wider portion of the shape) for that population. Band $B_{0,2}$ indicates correlated variation at the posterior/lateral side (the thinner portion) and Band $B_{0,3}$ indicates variation on the medial side (the portion that appears at the bottom in the Figure). These variations make sense anatomically. It is also interesting that each band has contiguous spatial support, though this is not a constraint of our technique.

Further visualizations of the bands at various resolutions of the left hippocampus dataset with 20 training shapes are shown in the left columns of Figure 50. Each row corresponds to a resolution level. The first column shows the initial similarity matrix for each level. The second column shows the resulting partitioned similarity matrix. As expected, the off-diagonal inter-band covariance is minimal. There are various band sizes, due to the fact that new bands are only recursively divided if condition (85) is met. The last three columns show the location of a selected band



Figure 52: Band Decomposition: various bands $B_{j,i}$, where j is the resolution and i is the band number, shown in Anterior view (A) and Posterior view (P), see text for color

and the variation found within that band, as will be explained next.

Figures 52(a)- 52(b) show the first band for the lowest scale for the prostate data. The second band is the complement of the first. As expected each band has a large spatial extent and indicate two uncorrelated shape processes on the prostate data: the variation of the anterior wall of the prostate (typically rounded) and the variation of the posterior wall of the prostate (typically flatter). Figures 52(c)- 52(h) show three bands for scale 3. These bands are more localized. These are uncorrelated variations of the superior and inferior walls of the shape, as well as an uncorrelated variations of the anterior wall at that scale. Bands have contiguous support, though this is not a constraint of our technique. The symmetry in bands 2 and 3 is also interesting, showing that both the right and left side tend to co-vary similarly. This symmetry of variation is plausible for the prostate, and we plan to investigate this further. Notably a diseased organ could possibly be detected if there is a lack of symmetry. Figures 53 show the band decomposition of the caudate population for scale 2. Again, the bands seem to be well localized in space.



(g) Scale 2, Band 6, Anterior View (h) Scale 2, Band 6, Posterior View

Figure 53: Band Decomposition, Caudate Nucleus dataset: various bands shown for scale 2 (shown in Anterior view and Posterior view), see text for color

4.4.2.4 Building the Prior

The final step for building the prior consists in finding variations within each band of wavelet coefficients. We call this approach the wavelet distribution model (WDM).

Given a band index set $B_{j,b}$, we create a filter matrix $P_{j,b}$ that selects the wavelet basis functions associated with the coefficients in the band $B_{j,b}$. We then select the wavelet coefficients corresponding to band $B_{j,b}$ for shape *i* using the equation:

$$\Gamma_{v_i}^{*_{j,b}} = \underbrace{\begin{bmatrix} P_{j,b} & 0 & 0\\ 0 & P_{j,b} & 0\\ 0 & 0 & P_{j,b} \end{bmatrix}}_{\mathcal{P}_{j,b}} \Gamma_i^*.$$
(86)

The size of $\Gamma_{v_i}^{*_{j,b}}$ is $3|B_{j,b}| \times 1$, where $|B_{j,b}|$ is the number of basis functions in $B_{j,b}$.

We learn the major modes of variations in a band just like in PDM, by calculating the mean:

$$\overline{\Gamma}^{*_{j,b}} = \frac{1}{K} \sum_{i=1}^{K} \Gamma_{v_i}^{*_{j,b}},\tag{87}$$

forming a shape matrix

$$\Gamma^{*_{j,b}} = \left[\begin{array}{ccc} \Gamma^{*_{j,b}}_{v_1} & \dots & \Gamma^{*_{j,b}}_{v_K} \end{array} \right], \tag{88}$$

and covariance matrix

$$C^{j,b} = (\Gamma^{*_{j,b}} - \overline{\Gamma}^{*_{j,b}})(\Gamma^{*_{j,b}} - \overline{\Gamma}^{*_{j,b}})^T,$$
(89)

and then diagonalizing the covariance matrix to find the eigenvectors (major modes of variation) $U^{j,b}$. Each column of $U^{j,b}$ is an eigenvector of size $3|B_{j,b}| \times 1$ that represents an axis of variation for the coefficients $\Gamma^{*_{j,b}}$. In total for that band, we find $3|B_{j,b}|$ or K-1 eigenvectors, whichever number is smaller.

To create the shape prior, we transform the eigenvectors back into the right dimensions:

$$U^{*j,b} = \mathcal{P}_{j,b}^T U^{j,b},\tag{90}$$

so that the only non-zero entries of $U^{*j,b}$ are at the indices corresponding to band $B_{j,b}$.

We can visualize the effect of the k^{th} eigenvector for band b and resolution j, $U^{*j,b}(k)$, by varying the shape wavelet coefficients along that eigenvector by an amount $\alpha^{*j,b}(k)$:

$$\Gamma_{v_i}^* = \overline{\Gamma}^* + U^{*j,b}(k)\alpha^{*j,b}(k), \qquad (91)$$

and then recovering the shape with equation (77).

This process is shown for the hippocampus dataset in Figure 50 and the caudate dataset in Figure 51 for a selected band for 4 different resolution levels. The eigenvectors of lower scale bands represent relatively global aspects of shape variability, whereas bands at higher scales represent higher frequency and more localized aspects of shape variability. Hence, our technique discovers shape variations at every scale, where the variations are all the eigenvectors of all the bands, and does not favor the discovery of global variations over local variations. Additionally, our prior accurately encodes finer details even with small training sets, since if there are a total of B bands, there exists on the order of L = B(K - 1) eigenvectors, as opposed to just K - 1 eigenvectors when using PDM.

The full prior contains all the eigenvectors for all bands and all resolutions in a matrix U^* of size $3M \times L$ if there are L eigenvectors in total.

A shape i can then be represented with the full prior:

$$\Gamma_{v_i}^* = \overline{\Gamma}^* + \underbrace{\left[\begin{array}{ccc} U^{*0,1} & \dots & U^{*r,k} \end{array}\right]}_{U^*} \alpha_{v_i} \tag{92}$$

where α_{v_i} (size $L \times 1$) represents the coordinates of the wavelet coefficients of that shape in the eigenvector space.

To summarize, each shape Σ^P is now represented with the following equation:

$$\Sigma^P = \overline{\Sigma^P} + \Pi \mathcal{P}^{*T} (\overline{\Gamma^*} + U^* \alpha_{v_i})$$
(93)

4.5 Experiments

In this section, we evaluate the multiscale shape prior based on WDM and band decomposition for a shape reconstruction task. The basic idea is to learn a prior with a training set, and to project shapes from a test set onto the prior to evaluate how close a projected test shape is to its ground truth.

We have three goals for the evaluation:

- 1. Compare the WDM prior using scale-band decomposition to WDM using only scale decomposition;
- 2. Compare both WDM priors to PDM;
- 3. Test the effect of noise on all priors.

We partition our data with N shapes randomly into T training samples and N-T testing samples, where T = [5, 10, 15, 20] and learn a shape prior from the training set. The prior for PDM consists of the mean shape and the eigenvectors of the landmarks on the shape. The prior for WDM using scale only consists of the mean shape, the mean wavelet coefficient vector, the eigenvectors for coefficients from each shape. The prior for WDM using scale and bands consists of the mean shape, the mean coefficient vector, the eigenvectors for coefficients from each shape.

Once we learn the priors from a training set, we project a each shape in the testing set onto the eigenvectors of the prior and translate the coordinates of the projected test shape to a point lying at a reasonable distance of the training data $(\pm 3 \text{ observed standard deviation})$. We then reconstruct the modified test shape. A mean squared error between the vertices of the ground truth and the reconstructed shape is calculated for all shapes in the testing set.



(a) Hippocampus: Max Squared Recon- (b) Hippocampus: Max Squared Reconstruction Error struction Error with Noise projection



(c) Caudate: Max Squared Reconstruc- (d) Caudate: Max Squared Reconstruction Error tion Error with Noise projection

Figure 54: Max Squared Reconstruction Error (averaged over testing shapes) for various training set sizes

To test the robustness of each prior, we also test the reconstruction in the presence of noise. To add noise to the test shape, we displace each vertex according to a Gaussian probability function with mean 0 and a standard deviation that is 5% of the bounding box of the object, as shown in Figure 55 (columns 2 and 4), producing a shape with noise s_n . Ideally, we would want the prior to not be affected by the noise and the reconstructed shape to be close to the ground truth (the shape without noise). To test this, we project the noisy shape onto the priors, and calculate the mean squared error between the reconstructed shape and the *ground truth* shape.

Figure 54 shows the maximum squared reconstruction error, averaged over all the shape in the testing set, for the various shape priors and various training set sizes of



Figure 55: Reconstruction Task for a test shape using 10 training shapes (first 2 columns) and a test shape using 20 training shapes (last 2 columns)

the hippocampus (top row) and caudate (bottom row) datasets. The left graph show the error using the ground truth as a projection onto the priors, the right graphs show the error using the noisy ground truth as a projection onto the priors (the error is then computed between the reconstructed shape and the original ground truth). As we can see in the graphs, the WDM prior with scale and band decomposition outperform the other techniques, even when the training set size is large (20 shapes). It is also interesting to see that all priors are minimally affected by Gaussian noise. Therefore, although the WDM prior with scale and band decomposition is more specific than PDM (meaning it represents a population more accurately), it is not more sensitive to noise.

As an example, 55 shows the Ground Truth shape, Noisy shape, and reconstruction with PDM and wavelet shape priors with **10** and **20 training samples** for the





Figure 56: Prostate dataset: Ground Truth Test shape, Test shape with noise, and reconstruction with PDM and WDM with scale and bands with **5 training samples**. Color is error from blue (lowest) to red.



(e) WDM with scale and bands recon- (f) WDM with scale and bands reconstruction from Test shape struction from Test shape with **Noise**

Figure 57: Caudate dataset: Ground Truth Test shape, Test shape with noise, and reconstruction with PDM and WDM with scale and bands shape priors with 25 training samples. Color is error from blue (lowest) to red.

hippocampus dataset. The figures show the reconstruction when the Ground Truth shape is projected onto the prior (column 1 and 3), and when the Noisy shape is projected onto the prior (column 2 and 4). We see that details that appear in the WDM (row 3 and 4) are lacking in the PDM (row 2) reconstruction, especially on the posterior side (thinner part of the shape). When comparing WDM with scale only (row 3) and WDM with scale and band decomposition (row 4), we see that the latter has a smaller error and contains finer details.

As another example, Figures 56 and 57 show the Ground Truth Test shape, Noisy shape, and reconstruction with PDM and WDM with scale and band decomposition with **5 training samples** for the prostate dataset and **25 training samples** for the caudate dataset. The figures show the reconstruction when the Test shape both with and without noise is projected onto the priors. We see that again details are lacking in the PDM reconstruction. The WDM technique incorporates both local and global details that PDM does not encode.

4.6 Conclusion

By using the spherical wavelet transform as shape representation, we have been able to take advantage of two types of decomposition: scale decomposition by using the transform structure and space decomposition by using the decorrelation property of the wavelet transform at every scale for the two brain structures we analyzed. We have demonstrated that our spherical wavelet based technique, called wavelet distribution model (WDM), is a better shape prior than ordinary PDM when it is important to represent finer, more localized shape variations for these brain structures. From our results we have shown that WDM with scale and band decomposition outperforms WDM using only scale decomposition for the caudate and hippocampus shapes. We note that if this technique is applied to other anatomical structures where the decorrelation property does not exist at particular scales, our technique will nicely default to including all coefficients from that scale into a single band.

CHAPTER V

3D SEGMENTATION USING THE MULTI-SCALE SPHERICAL WAVELET PRIOR

In this chapter, we derive an efficient active contour segmentation algorithm in the space of the spherical wavelet basis functions in order to directly and naturally include the multiscale prior presented in chapter 4 into a 3D surface evolution framework.

In order to exploit the multiscale prior, we derive a parametric surface evolution equation by *evolving the shapes in the wavelet prior subpspace* directly. This amounts to evolving the shapes in the subspace defined by all the eigenvectors of the bands associated with the wavelet prior. As the surface evolves to fit the image data, we constrain the shape to stay within an allowable part of the subspace spanned by shapes in a training set, or a certain standard deviation away.

The parameters of our model are the shape parameters α , as well as pose parameters that accommodate for shape variability due to a similarity transformation (rotation, scale, translation) which is not explicitly modeled with the shape parameters.

5.1 Related Work

As seen in chapter 2, deformable models offer robustness to both image noise and boundary gaps [71], and have been extensively studied and widely used in medical image segmentation, with good results. There are two types of deformable models, based on which representation is used for the model: parametric deformable models and geometric deformable models. In this work, since we will be segmenting brain structures of a fixed topology, our initial and final contour will remain of the same topology and we will therefore use the parametric model given its computational efficiency. One point of departure with the snakes model is that we will be deriving an evolution equation using the shape parameters directly as opposed to landmarks on the shape, in similar spirit to Tsai *et al.* [64].

Initial formulations of active contours, called "edge-based active contours," combined smoothness constraints with image data forces sampled on the boundary of the model. One issue with edge-based active contours is that they are not robust to noise in the image and the gradient terms can stop the curve evolution at spurious edges. Recently there has been a considerable amount of work on image segmentation using region-based curve evolution techniques, as discussed in Section 2.3.2.5. In those techniques, the force that influences the evolution of the curve depend on region statistics, inspired by the region competition work of Zhu and Yuille [75] and more recently the work of Chan and Vese [8] and Yezzi [73]. Our work uses the region-based active contour formulation in a parametric framework.

5.2 Pose Parameters

Given a surface mesh with N vertices $\Sigma : [1, ..., N] \to \mathbb{R}^4$, expressed in homogeneous coordinates so that a mesh point is denoted by $\Sigma(i) = \mathbf{x}_i = [x_i, y_i, z_i, 1]^T$, a transformed surface $\widetilde{\Sigma}$ is defined by:

$$\widetilde{\Sigma}(i) = T[\mathbf{p}]\Sigma(i).$$
(94)

The transformation matrix $T[\mathbf{p}]$ is the product of a translation matrix with 3 parameters t_x, t_y, t_z , a scaling matrix with 1 parameter s and a rotation matrix with 3 parameters w_x, w_y, w_z , using the exponential map formulation [35].
5.3 Shape Parameters

A surface point $\Sigma(i)$ can be represented in the wavelet basis with homogeneous coordinates by re-arranging Equation (93).

$$\Sigma(i) = \overline{\Sigma(i)} +$$

$$\mathcal{H}\left(\underbrace{\begin{bmatrix} \Phi_{m}(i,:) & 0 & 0 \\ 0 & \Phi_{m}(i,:) & 0 \\ 0 & 0 & \Phi_{m}(i,:) \end{bmatrix}}_{\Pi_{mi}} \mathcal{P}^{*T}(\overline{\Gamma^{*}} + U^{*}\alpha)\right),$$
(95)

where the function $\mathcal{H}: [3N \times 1] \to [4 \times N]$ rearranges a matrix to have the correct homogeneous coordinates and $\Phi_m(i, :)$ is a row vector of all the basis functions evaluated at point \mathbf{x}_i . The weight parameters $\alpha = [\alpha_1, ..., \alpha_L]$ (where L is the total number of eigenvectors of the shape prior) are the shape parameters of our model.

By combining equations (94)-(95), the shape to be evolved is:

$$\widetilde{\Sigma}(i) = T[\mathbf{p}](\overline{\Sigma(i)} + \mathcal{H}(\Pi_{mi}\mathcal{P}^{*T}(\overline{\Gamma^*} + U^*\alpha))).$$
(96)

5.4 Segmentation Energy

We use a region-based energy to drive the evolution of the parametric deformable surface for segmentation. With region-based energies, the force that influences the evolution of a contour depends on more global statistical information [64, 49]. We employ the discrete version of a segmentation energy presented in [49]:

$$E(\alpha, \mathbf{p}) := \sum_{\mathbf{x} \in R} L(\mathbf{x}) \Delta \mathbf{x}, \tag{97}$$

where R is the region inside the evolving surface $\tilde{\Sigma}$ and the force is $L(\mathbf{x}) = -\log(\frac{P_I(I(\mathbf{x}))}{P_O(I(\mathbf{x}))})$ where $I(\mathbf{x})$ is the image intensity at a point \mathbf{x} located inside the region R of the evolving surface, $P_I(I(\mathbf{x}))$ is the probability that a point \mathbf{x} with intensity $I(\mathbf{x})$ belongs to the interior of an object to be segmented in the image, and P_O is the probability that the point belongs to the exterior of the object to be segmented. The segmentation energy is minimized when the surface evolves to include points that have maximum L (points that have a higher P_I than P_O). To estimate the probability density functions P_I and P_O from a training set, we collect sample voxel intensity values inside and outside the segmented shapes in a neighborhood of width 10 pixels around the boundary and use Parzen windows [17].

The surface evolution is defined by a gradient flow of Σ that minimizes the energy in terms of the pose **p** and shape parameters α . We augment the parameters **p** and α with an artificial time parameter t and find gradient descent equations $\frac{d\mathbf{p}}{dt}$ and $\frac{d\alpha}{dt}$ by solving $\frac{dE(\mathbf{p}(t),\alpha(t))}{dt} = 0$.

We now show the derivation of the region flow for the segmentation algorithm. We use the area formula [55], and then the discrete divergence theorem to express the region sum in (109) as a surface sum. Note that this derivation applies to Ndimensions, but here we take the special case N = 3. Let R be an open connected bounded subset of \mathbb{R}^3 (the region inside the surface) with smooth boundary $\Sigma \colon \mathbb{R} \to \mathbb{R}^3$ (the surface) parametrized by a parameter $p_k \in \mathbb{R}$. Let $\mathcal{F}^t : \mathbb{R} \to \mathbb{R}^3$ be a family of embeddings, such that \mathcal{F}^0 is the identity. Here we consider the case where the given surface depends upon parameter p_k that varies with time. Let $L : \mathbb{R}^3 \to \mathbb{R}$ be a C^1 function. Setting $R(t) := \mathcal{F}^t(R)$ and $\Sigma(t) := \mathcal{F}^t(\Sigma)$ to remove the dependence on time, we define the following family of L-weighted volumes:

$$E(p_k(t)) := \sum_{R(t)} L(\mathbf{y}) \Delta \mathbf{y}$$
(98)

$$=\sum_{R} L(\mathcal{F}^{t}(\mathbf{x}))\Delta \mathcal{F}^{t}(\mathbf{x})$$
(99)

We set $f = \frac{d\mathcal{F}^t}{dt}$, then using the area formula [55], and the discrete divergence

theorem:

$$\frac{dE(p_k(t))}{dt}|_{t=0} = \sum_R div(L(\mathbf{x})f(\mathbf{x}))\Delta\mathbf{x}$$
(100)

$$= -\sum_{\Sigma} \left(L(\mathbf{y}) \right) f(\mathbf{y}) \cdot \mathcal{N}(\mathbf{y}) \right) \Delta \mathbf{y}$$
(101)

$$= -\sum_{\Sigma} \left(L(\mathbf{y}) \frac{d\mathbf{y}}{dp_k} \frac{dp_k(t)}{dt} \right) \cdot \mathcal{N}(\mathbf{y})) \Delta \mathbf{y}$$
(102)

$$= -\frac{dp_k(t)}{dt} \sum_{\Sigma} \left(L(\mathbf{y}) \frac{d\mathbf{y}}{dp_k} \right) \cdot \mathcal{N}(\mathbf{y})) \Delta \mathbf{y}$$
(103)

where $\mathcal{N}(\mathbf{y})$ is the inward normal to Σ at point (\mathbf{y}) .

Setting (100) to 0, we obtain:

$$\frac{d(p_k(t))}{dt} = \sum_{\Sigma} \left(L(\mathbf{y}) \frac{d\mathbf{y}}{dp_k} \right) \cdot \mathcal{N}(\mathbf{y}) d\mathbf{y}.$$
 (104)

When p_k is one of the pose (**p**) parameters, then using (94) we have:

$$\frac{d\mathbf{y}}{dp_k} = \frac{d\tilde{\Sigma}(i)}{dp_k} = \frac{dT[\mathbf{p}]}{dp_k}\Sigma(i).$$

When p_k is one of the shape (α) parameters, then using (eq:fullsurf) we have:

$$\frac{d\mathbf{y}}{d\alpha_k} = \frac{d\widetilde{\Sigma}(i)}{d\alpha_k} = T[\mathbf{p}]\mathcal{H}(\Pi_{mi}\mathcal{P}^{*T}U^*(:,k)).$$

5.5 Parameter Optimization via Multiresolution Gradient Descent

We can now use the gradient equation refeq:dp-dt to conduct a parameter optimization via gradient descent for each parameter in \mathbf{p} and α . Explicitly, the update equations are:

$$\mathbf{p}(t+1) = \mathbf{p}(t) + \delta_t^{\mathbf{p}} \frac{d\mathbf{p}}{dt},\tag{105}$$

$$\alpha(t+1) = \alpha(t) + \delta_t^{\alpha} \frac{d\alpha}{dt},$$
(106)

where δ_t^{α} and $\delta_t^{\mathbf{p}}$ are positive step size parameters, $\frac{d\mathbf{p}}{dt}$ is given in (104-105) and $\frac{d\alpha}{dt}$ is given in (104-105) and $\alpha(t+1)$, $\mathbf{p}(t+1)$ denote the values of the parameters α and \mathbf{p} at the $(t+1)^{th}$ iteration of the surface evolution.

We start with an initial shape and iterate between (105) and (106). We update the α parameters in a *multiresolution* fashion. Since each shape parameter α_i corresponds to a band at a wavelet resolution j, we first only update α coefficients corresponding to the coarsest level bands (j = 1). Once α changes less than a threshold value v_{α} , we add the α parameters of the next resolution level to the gradient and update (106). This results in a more stable segmentation since few global parameters are first updated when the shape is far from the solution, and more localized parameters are added as the shape converges to the solution.

We start with (105) until $(\mathbf{p}^{t+1} - \mathbf{p}^t) < v_{\mathbf{p}}$ where $v_{\mathbf{p}}$ is a threshold value. We then run (106) for 1 iteration, and iterate the process. At each α iteration, we ensure that the value of the α parameters stays within ± 3 standard deviation of the observed values in the training set. After each iteration, the updated shape and pose parameters are used to determine the updated surface.

5.6 Results of Segmentation

We have applied our algorithm to the segmentation of the caudate nucleus shapes. We learned a shape prior from a training set of 24 shapes and used the remaining 5 shapes as a test set. We use spherical wavelet basis functions of resolution up to j = 5. In total, we obtained 16 bands in the shape prior. We learned the mean position p_m of the caudate shapes in the MRI scans (in patient RAS coordinates, described in Appendix A). To initialize the segmentation, we use the mean caudate shape learned during the training phase and positioned it at position p_m in the scan to be segmented. We then evolved the surface according to the process described in Section 5.5. The step size parameters were $\delta_t^{\alpha} = 0.5, \delta_t^{\mathbf{p}} = 0.001$ for translation and $\delta_t^{\mathbf{p}} = 0.0001$ for



Figure 58: Surface Evolution using the Ground Truth label-map as the image force for the ASM algorithm. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale).



(c) Mscale, iter=122, levels 1-5 active

Figure 59: Surface Evolution using the Ground Truth label-map as the image force for the Mscale algorithms. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale).



Figure 60: Surface Evolution using the **density estimation** as the image force for ASM. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale)



(c) Mscale, iter=202, levels 1-5 active

Figure 61: Surface Evolution using the density estimation as the image force for Mscale (bottom row) algorithms. The ground truth is shown in red (light-gray if seen in grayscale), the evolving surface in blue (dark gray if seen in grayscale)

scale and rotation and the evolution threshold values were $v_{\mathbf{p}} = v_{\alpha} = 0.02$.

To measure the discrepancy between segmented shape (Σ) and ground truth (G) (obtained from the hand-segmented label-maps), we use the Hausdorff distance $H(G, \Sigma)$ that measures the maximum error between the boundary of two shapes Gand Σ , as well as the partial Hausdorff distance $H_f(G, \Sigma)$ that measures the f% percentile of the Hausdorff distance. We compare our algorithm (called Mscale) to the standard ASM algorithm [11], using the same training, testing shapes and degrees of freedom (we keep 100% of the eigenvectors).

To validate our algorithm, we first use the Ground Truth label-map as the image force in (104) by replacing L with a value of 1 inside the (known) object and -1outside. The end goal is to validate whether the surface evolution converges to the right solution, given perfect image information. Since we are evolving in the space of the shape prior, the discrepancy between the ASM and Mscale algorithm is due to the expressiveness of the shape prior. Figures 58-59 show the result for test shape 5. The final segmentation with the multiscale prior captures more of the shape and finer details than the ASM segmentation. Furthermore, we see that for as the resolution level is increased for the α parameters, the Mscale segmentation is able to capture finer details.

We then validated the full segmentation algorithm, using the proposed image force in (109). The results of the validation for both algorithms is shown in Table 3. For each test shape, the lowest error among the two algorithms is in boldface. The Mscale algorithm consistently outperforms the ASM algorithm. Figures 60-fig:PIPO-mscale qualitatively compares the segmentation of Test shape 3 for both algorithms. The Mscale algorithm is more accurate and captures finer details, especially at the tail of the shape. We note that the segmentation is not fully accurate due to non-perfect image statistics. Our algorithm runs under 5 minutes on a Pentium IV 2GHz using non-optimized MATLAB code.

		N=1	N=2	N=3	N=4	N=5	Mean
$H_{95} (\mathrm{mm})$	Mscale	4.82	2.22	3.03	3.04	3.95	3.16
	ASM	5.51	3.24	3.98	3.18	4.26	3.83
$H (\rm{mm})$	Mscale	5.89	4.06	3.75	5.23	5.57	4.85
	ASM	9.79	5.68	6.33	7.22	6.06	7.07

Table 3: H, H_f distance for 5 test shapes for Mscale and ASM.

5.7 Conclusion

Using the multiscale representation and prior presented in Chapter 4, we presented a computationally efficient segmentation algorithm. Our results show that the proposed segmentation algorithm outperforms standard ASM. One advantage of the technique is the ability to evolve coarse scale parameters first, in order to quickly bring the evolving shape close to the solution and then evolving higher resolution parameters to improve the fit. The technique is general and can be used with any kind of prior based on WDM, regardless of the decompositions used. In our work, we have use the WDM prior with scale-band decomposition given that it outperformed other priors in the reconstruction task.

CHAPTER VI

STATISTICAL SHAPE ANALYSIS OF CAUDATE USING THE SPHERICAL WAVELET SHAPE REPRESENTATION

One important challenge for the medical imaging community is the ability to detect variations between the anatomy of healthy individuals and individuals with a pathology. The neuroimaging community is particularly interested in detecting variations in the shape of brain structures for patients with neurological disorders, such as schizophrenia. In this chapter, we apply the wavelet shape representation of Chapter 4 to a shape analysis task in the study of patients with neuroanatomic abnormalities, such as schizophrenia or schizo-typal personality disorder.

6.1 Related Work

The study of brain morphology has emerged as a new field of computational neuroanatomy and can provide great insights into brain function, development and pathologies. The aim of our work in this chapter is to investigate whether there exists morphological differences of selected brain structures between groups of neuropsychiatric patients with neuroanatomic abnormalities and a group of healthy controls. To reach this aim, we compare structures extracted from MRI images of different subjects using statistical tests.

Both functional and structural neuroimaging studies of subjects with schizophrenia have pointed to abnormalities in the basal ganglia [37]. One common studied brain structure is the caudate nucleus that is part of the striatal nuclei (caudate, putamen, and nucleus accumbens) that serves as the "input nuclei" for the basal ganglia.

Statistical analysis of brain structures is often based on global features, such as volumetric measurements [37]. However, studies have shown that morphometric analysis of brain structures provides new information which is not available by conventional volumetric measurements [24]. To conduct 3D morphometry, various shape representations have been proposed, ranging from dense sampled 3D Point Distribution Models (PDM) [12, 58] to medial shape descriptions [25, 46, 24, 59] and surface parametrization using expansion into a series of Fourier [57] or spherical harmonic basis functions [5]. Combined, these representations provide new complementary measurement tools to answer clinical research questions.

In [59, 60], Styner et. al. use a point distribution model called SPHARM-PDM to test for local shape differences in the caudate nucleus between healthy patients and patients with schizotypal personality disorder (SPD). They first establish correspondences between shapes and align the shapes using spherical harmonics. After the shapes are registered and equally sampled, they test the two groups for differences in the mean spatial location at each point on the surface using a statistical test.

In Hypothesis testing, one picks a test statistic, which assesses a hypothesis of interest. The value of the test statistic is compared to a reference distribution, the distribution of the test statistic assuming the null hypothesis is true. The p-value is the probability that the test statistic would assume a value greater than or equal to the observed value strictly by chance.

In their experiments, they use the Hotelling T^2 statistic to test the null hypothesis that the mean location of a 3D point in two groups of patients does not differ. The Hotelling T^2 statistic is a generalization of Student's t statistic for multivariate hypothesis testing (here the multivariate quantity is the 3D vector). They use nonparametric permutation tests to conduct the hypothesis testing. A non-parametric permutation test is a type of statistical significance test in which a reference distribution is obtained by calculating all possible values of the test statistic under rearrangements on the observed data points. The advantage of such a technique is that there is no need for a parametric reference distribution.

The null hypothesis for their test is that the mean of the location of a point in the two groups is the same. The output of this test is a raw p-value at each point that indicates the probability that the null hypothesis is true. A significance level of 5% is chosen, meaning that any point with a p-value less than 0.05 is considered to have a mean location that differs among the two groups in a statistically significant way. The final step is to correct for multiple comparisons using a technique known as false discovery rate (FDR) that controls for the maximum number of false positives. The final output of the test is a corrected p-value that is pessimistic in order to control for false positives. As a result, they find a significant shape difference between the two populations in the right caudate head, but not in the left caudate head.

In this chapter, we use the framework developed by Styner et. al, but substitute the spherical wavelet coefficients as shape descriptors instead of (x, y, z) coordinates of points on the surface. When representing shape information, spherical wavelet coefficients have a more intuitive interpretation than Fourier or spherical harmonic coefficients due to the localized nature of spherical wavelet basis functions. Each coefficient describes a portion of the surface and the size of that portion depends on the scale of the coefficient. However coefficients are not as localized as points in a PDM representation, potentially capturing shape characteristics that exist at different spatial locations and different spatial extent, i.e a bending of a portion of the shape. We use this shape representation for statistical shape analysis of two brain structures, the caudate nucleus and hippocampus, and compare the results obtained to shape analysis using a SPHARM-PDM representation.

Our motivation is two-fold. First, we would like to test whether the results found

with spherical wavelet features correlate with the results found using vertex coordinates as features. Second, we would like to see if the multiscale nature of the SWC allow for a richer shape description and test whether running the shape analysis at various levels of resolution provide additional information about the difference between groups of healthy patients and patients with SPD not previously found with PDM analysis.

6.2 Methods

Our technique first constructs a SPHARM-PDM representation of surfaces (section 6.2.1) and then represents these surfaces using an expansion in a spherical wavelet series (section 6.2.2). We then conduct statistical tests using both the SPHARM-PDM and SWC representation (Section 6.2.3).

6.2.1 SPHARM-PDM

The input of the proposed shape analysis is a set of binary segmentations of a single brain structure. These segmentations are transformed into a SPHARM-PDM representation using a procedure described in [60]. Here we sketch the major steps of the algorithm. The binary voxel objects provided by expert slice-by-slice segmentations are first preprocessed to fill any interior holes and to smooth boundary voxel noise. The processed binary segmentations are converted to surface meshes, and a spherical parametrization is computed for the surface meshes using an area-preserving, distortion minimizing spherical mapping. The SPHARM description is computed from the mesh and its spherical parametrization. Using the first order ellipsoid from the spherical harmonic coefficients, the spherical parametrizations are aligned to establish correspondence across all surfaces. The SPHARM description is then sampled into a triangulated surface (SPHARM- PDM) via icosahedron subdivision of the spherical parametrization (4 subdivisions). These SPHARM-PDM surfaces are all spatially aligned using rigid Procrustes alignment.

6.2.2 Spherical Wavelet Shape Representation (SWC)

The spherical wavelet description is computed from the SPHARM-PDM surface. Each surface contains N vertices and has a spherical parametrization. The mean shape is substracted from each shape in the dataset, resulting in a deviation signal v for each shape. Each shape is then expressed in the spherical wavelet basis function by representing it as three signals v^x , v^y and v^z on the discrete sphere, corresponding to the x, y and z deviation of all vertices from the mean shape. We then expand each signal into a series of spherical wavelet basis functions using the forward spherical wavelet transform, as described in Chapter 4. For example:

$$v^{x}(n) = \sum_{k} \lambda^{x}_{0,k} \varphi_{0,k}(n) + \sum_{0 \le j} \sum_{m} \gamma^{x}_{j,m} \psi_{j,m}(n).$$
(107)

where $\varphi_{0,k}$ and $\psi_{j,m}$ are basis functions defined over all vertices indexed by $n \in N$, j denotes the scale (spatial extent) and k, m the center of the basis function, and $\lambda_{0,k}^x, \gamma_{j,m}^x$ are the associated wavelet coefficients. As a result, each shape is represented by a series of spherical wavelet coefficients (SWC). The top row of Figure 62 shows the decreasing spatial support of a single basis function at scales j = 1, j = 2 and j = 3. Note that the supports of the basis functions overlap across scales but also slightly within a scale. To locate and visualize the influence of all basis functions at a given scale, each point on the sphere is associated to the basis function with the highest value at that point. These basis membership regions are shown in the middle and bottom row of Figure 62.

6.2.3 Shape Analysis

6.2.3.1 Test Statistic

The difference between a multivariate feature in two groups is computed using a modified Hotelling T^2 two sample metric that is less sensitive to group differences than the standard T^2 metric. Given a group *i* with n_i samples, we calculate the mean



Figure 62: Visualization of spherical wavelet functions and associated membership regions at three levels (columns). Top row : Values of single spherical Wavelet Basis Function shown on the sphere at scales 1 through 3. Middle and Bottom row: Membership regions of spherical wavelet basis functions shown on the sphere and on the original surface, coloring is random.

 μ_i and covariance Σ_i of a 3D feature. The modified T^2 for two groups is given by:

$$T^{2} = (\mu_{1} - \mu_{2})^{T} (\Sigma_{1} \frac{1}{n_{1}} + \Sigma_{2} \frac{1}{n_{2}})^{-1} (\mu_{1} - \mu_{2})$$
(108)

A PDM feature is a point with 3D coordinates. A SWC feature is a basis function $\phi_{j,k}$ with 3D coordinates corresponding to the spherical wavelet coefficients $\gamma_{j,k}^x$, $\gamma_{j,k}^y$ and $\gamma_{j,k}^z$.

6.2.3.2 Non-parametric permutation tests

We want to test the two groups for differences in the means of the T^2 metric at each feature. Permutation tests are a valid and tractable approach for such an application, as they rely on minimal assumptions and can be applied even when the assumptions of the parametric approach are untenable. Our null hypothesis is that the distribution of the value of each feature is the same for every subject regardless of the group. Given n_1 members of the first group $a_k, k = 1, ..., n_1$ and n_2 members of the second group $b_k, k = 1, ..., n_2$, we can create $M \leq {\binom{n_1+n_2}{n_2}}$ permutation samples. A value of M from 20,000 and up should yield results that are negligibly different from using all permutations for a typical experiments of 40 samples in each group [18]. To calculate a P-value for a feature, the real group difference T_0^2 for that feature is compared to the distribution of group differences T_j^2 computed from random permutations of the group labels for that feature. The quantile in the T_j^2 histogram associated with T_0^2 is called the raw P-value. Given a chosen significance value α , the hypothesis that the feature value is the same in both groups is rejected if the P-value for that feature is less than α .

6.2.3.3 Correction for Multiple Comparisons

Since the shape analysis involves testing from a few to many thousands of hypotheses (one per feature), it is important to control for the multiple testing problem. We use a False Discovery Rate (FDR) estimation, a procedure that controls the expected proportion of false positives among those tests for which a local significance has been detected [60]. FDR allows an expected proportion (usually 5%) of the FDR-corrected significance values to be falsely positive.

6.2.4 Significance Map Visualization

For PDM features, we visualize both the raw and FDR corrected P-values as significance color maps on the surface of the mean shape of the structure under study. The color at each point is the P-value. For SWC features, we would also like to build such a significance map. If a feature (basis function) is found significant, we color all points that are in the support of that basis function at that scale with the corresponding P-value. If more than one basis function is found significant and the support of the basis functions overlap, we assign the overlapping region to the function with higher value, and color the region with the P-value of that basis function.

6.3 Results

We applied our shape analysis framework using both the PDM features and SWC features to two studies¹. The first is a schizo-typal personality disorder (SPD) study on the caudate brain structure in female adult patients [43]. 32 SPD subjects and 29 healthy control subjects were analyzed. The second is a schizophrenia study on the hippocampus brain structure in male adult schizophrenia [59]. 56 schizophrenia subjects (age: 30.1y (11.9)) and 26 healthy control (age: 31.2y (10.7)) subjects were analyzed. The subjects in both studies have same handedness and the structures were corrected for difference in head size.

For both structures, we analyzed the right and left hemisphere separately. We present results on left structures for brevity. For SWC features, the shape analysis is conducted at various cumulative scales: all features associated to basis functions up to a given scale are tested for difference among the groups. For all structures we present results up to scale 3 since no new features were discovered at subsequent scales.

6.3.1 Left Caudate Nucleus

The result for PDM is shown in Figure 63. The raw significance map displays an overly optimistic estimate of significance in the superior body and anterior head

¹GE 1.5 Tesla MR system using a 3D IR Prepped SPGR acquisition protocol with a 256x256x124 image matrix at 0.9375x0.9375x1.5mm resolution.

region. The FDR-corrected map is a more pessimistic estimate and does not show any significance. The result for SWC are shown in Figures 64(a)-64(c). For the raw map (top rows), scale 1 displays significance at the anterior inferior head region. At cumulative scales 1-2 and 1-3, the raw map displays additional significance in the anterior superior head region, as well as superior body and posterior tail. Overall, the SWC raw map at scales 1-3 displays similar significant areas than the raw PDM map. However, unlike the PDM FDR map, the SWC FDR map displays significant area in the anterior superior region (raw P-value 5e-5, FDR P-value 0.0085).

6.3.2 Right Caudate Nucleus

The result for PDM is shown in Figure 65. The raw significance map displays regions of significance in the superior body and anterior region that are larger than for the left caudate. The FDR-corrected map preserves most of the areas of significance, except in areas of the tail. The result for SWC are shown in Figures 66(a)-66(c). The raw maps at all scales display similar areas of significance than for the raw PDM results. At all scales, the FDR correction preserves areas of significance in the anterior head. The FDR map at cumulative scales 1-2 is very similar to the PDM FDR map, indicating that most of the differences happen at coarse scales 1 and 2. At scale 3, the FDR correction is more severe due to the increasing number of tests, and only preserves coefficients with very high significance (very small p-values). These regions are again located in the same areas of the anterior head than the PDM FDR maps.

6.3.3 Left Hippocampus

The result for PDM is shown in Figure 67. The raw significance map displays significance both in the superior-anterior and inferior-posterior regions. The FDR-corrected map is a more pessimistic estimate and does not show any significance. The result for SWC are shown in Figures 68(a)-68(c). At scale 1, the raw significance map displays significance in the superior and inferior region, indicating that these differences occur already at a coarse scale. At scales 1-2 and 1-3, additional smaller regions of significance appear in the superior-anterior and inferior-posterior, similar to the PDM regions, indicating those group difference are at a fine scale (small spatial support). At scale 1, the FDR map preserves the significance in most regions, and at scale 1-3 a small region in the medial side is preserved (raw P-value 0.0001, FDR P-value 0.016). The FDR correction is more severe at higher scales due to the increasing number of tests, and only preserves the regions with high significance.

6.3.4 Right Hippocampus

The result for PDM is shown in Figure 69. The raw significance map displays many areas of significance in the superior and inferior regions of the lateral side, and the middle region of the medial side. Again, the results for the SWC shown in Figures 70(a)-70(c) agree with the PDM results and find similar areas of significance.

6.4 Conclusion

We have presented a novel method for statistical analysis of morphological differences of brain structures based on a spherical wavelet (SWC) representation and compared it to a similar analysis with a PDM representation. Overall, the SPHARM-PDM shape analysis shows more pronounced right sided findings compared with left sided findings for both structures. The SWC analysis results are similar to the SPHARM-PDM for the right structures, and it unmasks differences not found by the SPHARM-PDM analysis for the left structures, even with FDR-correction. We notice that for the left structures the coefficients that were found to be different among groups with FDR-correction are located at resolution 1 and 2, which are coarse scales. This might explain why purely local features, such as PDM, might not find differences that SWC features find since the scale of analysis of PDM is too fine. Therefore one key property of the spherical wavelet representation is the ability to conduct statistical tests at various scales, therefore providing features that are targeted to



Figure 63: Left Caudate Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color.

particular types of multiscale group differences. Further studies providing correction for age and medication will be needed to draw clinical conclusions.



Figure 64: Left Caudate Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color.



Figure 65: Right Caudate Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color.



Figure 66: Right Caudate Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color.



Figure 67: Left Hippocampus Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color.



Figure 68: Left Hippocampus Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color.



Figure 69: Right Hippocampus Shape Analysis Results - Significance maps for the PDM features. This figure is best seen in color.



Figure 70: Right Hippocampus Shape Analysis Results - Significance maps for the SWC features. This figure is best seen in color.

CHAPTER VII

CONCLUSION

In this thesis, we proposed and evaluated two novel scale-based decomposable representations of shape for the segmentation of blood vessels and the segmentation and shape analysis of shapes with spherical topology, such as the prostate gland or brain structures. We proposed two representations that are adapted to the class of anatomical structures under study. For blood vessel characterization, we presented a novel scale-space shape filter that measures the deviation from a tubular shape in a local neighborhood of points, given a particular scale of analysis. For characterization of structures with spherical topology, we proposed a multi-scale parametric shape representation using spherical wavelets that can be optimized to compactly encode shape variations in a population at the needed scale and spatial locations. We demonstrated that learning a prior over those representations allow for a richer description and a more fine-grained control in segmentation and shape analysis tasks, when compared to previous techniques. In particular, we showed that the decomposition of these shape representations can be localized both in space and in scale, enabling the construction of more descriptive, non-global, non-uniform shape priors. For each representation, we derived a segmentation algorithm using the parameters of the shape representation to easily include and benefit from the prior in the optimization framework. We also demonstrated that an existing shape analysis algorithm can benefit from using our multiscale shape representation.

There remains challenges to the techniques presented in this thesis, as well as future extensions that could provide exciting research directions. We discuss those in the remainder of this Conclusion.

In Chapter 3, we presented a framework for vessel segmentation using local filters as soft shape priors. Currently the local filters only apply a tubular shape constraint to the evolving contour. A future extension to this work would be to extend the framework to other types shape constraints. One opportunity would be to use local filters to constrain a surface to remain a thin sheet, for example for cortical surface segmentation. A second limitation with the current algorithm is that the scale of the local filters (radius) remains constant throughout the curve evolution. A useful extension to address this limitation would be to add the radius as an extra parameter in the curve evolution and find the optimal radius at each point of the curve. Constraints, such as monotonic radii values along a vessel, should be added in order to model the anatomy of vessels. Another limitation with the current algorithm is that each local filter is modeled as its own independent shape prior term, and the overall shape probability is the product of the probability of each local filter. To address this limitation, one interesting area of research would be to depart from this strictly independent view and to learn the joint probability of nearby filters, a somewhat "localized" data-driven prior. For example, we could learn the joint probability function of neighboring filter responses to learn characteristic parts of the vessel shape such as branching areas, flatter or more curved vessel areas. This shape prior probability could either in segmentation task (i.e to recognize that a part of the vessel surface is a branch as opposed to a leak and not penalize the branch) or even for shape analysis to detect characteristic parts of the vessel.

In Chapter 4 and 5, we presented a framework for segmentation using a multiscale shape representation. Currently, the technique presented is limited to shapes with spherical topology. In the future, one could use wavelets defined on surfaces of genus higher than zero into our framework. Another extension of the framework would be to apply the multiscale representation to groups of shapes with spherical topology. For example, it would be especially interesting to apply our band discovery algorithm to left and right shapes of the same anatomical structure (i.e. left and right caudate nucleus) to test whether bands detect symmetrical variations. One critical step in our framework is the remeshing and registration of the training shapes. If shapes are not correctly aligned, then our representation is sensitive to the misalignment and will not represent the shape population accurately. In the future, we should investigate the sensitivity of our algorithms to misalignment. We note however that this "correspondence" problem exist for most shape representation techniques, whether implicit or parametric. Another limitation of our current framework is that the wavelet coefficient probability estimation assumes a product of multivariate normal probability functions. One could depart from this assumption and attempt to learn a more accurate probability function by using more complex models using for example graphical models, kernel PCA or other non linear PCA estimation techniques.

In Chapter 6, we used the spherical wavelet representation for shape analysis. One extension of the current hypothesis test would be to test for differences in bands of coefficients found during our prior estimation, to test whether bands (that represent areas of covariability in a population on a shape) capture morphological differences among groups of patients. In the future, the multiscale representation could also be used in a classification task for potential diagnosis of a pathology.

In the last two decades, researchers in the medical imaging community have strived to develop algorithms and visualization techniques that leverage the fact that medical scans provide *patient specific* data. In this work, we have also emphasized the need to develop representations and algorithms based on those representations that are as *structure specific* as possible in order to better segment, analyse and visualize patientspecific data. We hope that this contribution will further help physicians diagnose, treat and track the progress of diseases.

APPENDIX A

APPENDIX

A.1 Region-Based Flow derivation

In this section, we show the derivation for the region-based flow. The basic underlying mathematical idea is to write down an energy that minimizes a particular function f inside a region R, enclosed inside the curve $\overrightarrow{\mathcal{C}}$. The goal is then to find a gradient flow of $\overrightarrow{\mathcal{C}}$ that minimizes the following energy:

$$E(\overrightarrow{\mathcal{C}}) = \int_{R} f(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \oint_{\overrightarrow{\mathcal{C}}} \langle \overrightarrow{F}, \overrightarrow{\mathcal{N}} \rangle \, \mathrm{d}s, \qquad (109)$$

where $\overrightarrow{\mathcal{N}}$ denotes the unit normal of $\overrightarrow{\mathcal{C}}$, ds is the Euclidean arclength element, and $\overrightarrow{F}(\mathbf{x})$ is a vector field chosen so that $\nabla \cdot \overrightarrow{F}(\mathbf{x}) = f(\mathbf{x})$. For example, given $\mathbf{x} = (x, y)$

$$\overrightarrow{F}(x,y) = \left(\begin{array}{c} F^x(x,y)\\ F^y(x,y) \end{array}\right),\,$$

where:

$$F^{x}(x,y) = \frac{1}{2} \int_{0}^{x} f(\lambda,y) \, \mathrm{d}\lambda,$$
$$F^{y}(x,y) = \frac{1}{2} \int_{0}^{y} f(x,\lambda) \, \mathrm{d}\lambda,$$

so that:

$$\nabla \cdot \overrightarrow{F}(x,y) = \frac{\partial F^x(x,y)}{\partial x} + \frac{\partial F^y(x,y)}{\partial y} = \frac{1}{2}f(x,y) + \frac{1}{2}f(x,y) = f(x,y).$$
(110)

The equivalence between the region integral based on f and the contour integral based on \overrightarrow{F} in Equation 109 follows from the divergence theorem.

To derive the gradient flow, we start by considering a fixed parameterization $p \in [0, 1]$ of the curve $\overrightarrow{\mathcal{C}}$ which does not vary as the curve evolves in time t so that

(p, t) are indendent variables. By a change of variable we may write E as follows:

$$E(\overrightarrow{\mathcal{C}}) = \int_0^1 \langle \overrightarrow{F}, J \overrightarrow{\mathcal{C}}_p \rangle \, \mathrm{d}p, \qquad (111)$$

where $J = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ denotes a -90° rotation matrix. Differentiating with respect to t yields:

to
$$t$$
 yields:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \int_0^1 \left\langle \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_t, J \overrightarrow{\mathcal{C}}_p \right\rangle + \left\langle \overrightarrow{F}, J \overrightarrow{\mathcal{C}}_{pt} \right\rangle \,\mathrm{d}p,\tag{112}$$

where $\frac{d\vec{F}}{d\mathbf{x}}$ denotes the Jacobian matrix of \vec{F} with respect to \mathbf{x} , that is if $\mathbf{x} = (x, y)$ and

$$\frac{\mathrm{d}\vec{F}}{\mathrm{d}\mathbf{x}} = \begin{bmatrix} \frac{\partial F^x}{\partial dx} \frac{\partial F^x}{\partial dy}\\ \frac{\partial F^y}{\partial dx} \frac{\partial F^y}{\partial dy} \end{bmatrix}.$$
(113)

If we integrate by parts the second term of Equation 112, we obtain

$$\int_{0}^{1} \left\langle \overrightarrow{F}, J \overrightarrow{\mathcal{C}}_{pt} \right\rangle \, \mathrm{d}p = \left\langle \overrightarrow{F}, J \overrightarrow{\mathcal{C}}_{t} \right\rangle |_{1}^{0} - \int_{0}^{1} \left\langle \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_{p}, J \overrightarrow{\mathcal{C}}_{t} \right\rangle \, \mathrm{d}p, \qquad (114)$$

$$\int_{0}^{1} \left\langle \overrightarrow{F}, J \overrightarrow{\mathcal{C}}_{pt} \right\rangle \, \mathrm{d}p = -\int_{0}^{1} \left\langle \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_{p}, J \overrightarrow{\mathcal{C}}_{t} \right\rangle \, \mathrm{d}p, \tag{115}$$

where we are using the fact that for a parametrized closed curve, $\overrightarrow{C}(0) = \overrightarrow{C}(1)$, therefore $\overrightarrow{C}_t(0) = \overrightarrow{C}_t(1)$. Now Equation 112 becomes:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \int_0^1 \left\langle \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_t, J \overrightarrow{\mathcal{C}}_p \right\rangle - \left\langle \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_p, J \overrightarrow{\mathcal{C}}_t \right\rangle \,\mathrm{d}p. \tag{116}$$

Reaaranging terms leads to:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \int_0^1 \left\langle \overrightarrow{\mathcal{C}}_t, \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}}^T J \overrightarrow{\mathcal{C}}_p \right\rangle - \left\langle J^T \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_p, \overrightarrow{\mathcal{C}}_t \right\rangle \,\mathrm{d}p, \tag{117}$$

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \int_0^1 \left\langle \overrightarrow{\mathcal{C}}_t, \left(\frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}}^T J - J^T \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}}\right) \overrightarrow{\mathcal{C}}_p \right\rangle \,\mathrm{d}p.$$
(118)

Since the expression

$$\frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}}^{T}J - J^{T}\frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} = \begin{bmatrix} 0\frac{\partial F^{x}}{\partial dx} + \frac{\partial F^{y}}{\partial dy}\\ -(\frac{\partial F^{x}}{\partial dx} + \frac{\partial F^{y}}{\partial dy})0 \end{bmatrix} = \begin{bmatrix} 0\nabla\cdot\overrightarrow{F}\\ -\nabla\cdot\overrightarrow{F}0 \end{bmatrix} = fJ,$$

Equation 118 becomes:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \int_0^1 \left\langle \overrightarrow{\mathcal{C}}_t, f J \overrightarrow{\mathcal{C}}_p \right\rangle \,\mathrm{d}p = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_t, f \overrightarrow{\mathcal{N}} \right\rangle \,\mathrm{d}s.$$
(119)

We see then that the form of the gradient flow for $\overrightarrow{\mathcal{C}}$ (the negative of the gradient so that the region integral decreases most rapidly) is revealed to be:

$$\frac{\partial \vec{\mathcal{C}}}{\partial t} = -f\vec{\mathcal{N}}.$$
(120)

Thus the flow depends only upon f, not upon our particular choice for F.

A.2 Soft Shape Prior Region-Based Flow derivation

In this section, we consider a more general class of region-based energy functionals where the integrand f depends upon another family of region integrals over R. From Chapter 3, Equation 52, we have the following energy:

$$E(\overrightarrow{\mathcal{C}}) = -\int_{R} \phi \, \mathrm{d}\mathbf{x} + \int_{\overrightarrow{\mathcal{C}}} \, \mathrm{d}s + \alpha \int_{R} \epsilon_{1}^{p}(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$
(121)

We focus on the third term, and therefore want to solve for the minimum of the following energy:

$$E(\overrightarrow{\mathcal{C}}) = \int_{R} f(\epsilon_1(\mathbf{x}, r, t)) \, \mathrm{d}\mathbf{x}, \qquad (122)$$

where $f(x) = x^p$ and

$$\epsilon_1(\mathbf{x}, r, t) = \int_R \mathcal{X}(\mathbf{x}, r, \mathbf{y}) d\mathbf{y} \quad \text{where} \quad \mathcal{X}(\mathbf{x}, r, \mathbf{y}) = \begin{cases} 1 & if \quad \mathbf{y} \in B(\mathbf{x}, r) \\ 0 & if \quad \mathbf{y} \notin B(\mathbf{x}, r) \end{cases}$$
(123)

Note that we make $\epsilon_1^p(\mathbf{x}, t)$ depend on the artificial time parameter t since R (the interior of $\overrightarrow{\mathcal{C}}$) changes as the curve evolves over time.

We start out as in the previous section to rewrite the integral as a contour integral

$$E(\overrightarrow{\mathcal{C}}) = \int_{R} f(\epsilon_{1}(\mathbf{x}, r, t)) \, \mathrm{d}\mathbf{x} = \oint_{\overrightarrow{\mathcal{C}}} \langle \overrightarrow{F}(\mathbf{x}, t), \overrightarrow{\mathcal{N}} \rangle \, \mathrm{d}s, \qquad (124)$$

where $\overrightarrow{\mathcal{N}}$ denotes the unit normal of $\overrightarrow{\mathcal{C}}$, ds is the Euclidean arclength element, and $\overrightarrow{F}(\mathbf{x},t)$ is a vector field chosen so that $\nabla_{\mathbf{x}} \cdot \overrightarrow{F}(\mathbf{x}) = f(\epsilon_1(\mathbf{x},r,t))$ (note that $\nabla_{\mathbf{x}}$ means the divergence operator with respect to \mathbf{x} only).

As in the previous section A.1, to derive the gradient flow, we start by considering a fixed parameterization $p \in [0, 1]$ of the curve \overrightarrow{C} which does not vary as the curve evolves in time t so that (p, t) are indendent variables. By a change of variable we may write E as follows:

$$E(\overrightarrow{\mathcal{C}}) = \int_0^1 \langle \overrightarrow{F}(\mathbf{x}, t), J \overrightarrow{\mathcal{C}}_p \rangle \, \mathrm{d}p, \qquad (125)$$

where $J = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$ denotes a -90° rotation matrix. Differentiating with respect to t yields

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \int_0^1 \left\langle \frac{\mathrm{d}\overrightarrow{F}}{\mathrm{d}\mathbf{x}} \overrightarrow{\mathcal{C}}_t, J\overrightarrow{\mathcal{C}}_p \right\rangle + \left\langle \overrightarrow{F}, J\overrightarrow{\mathcal{C}}_{pt} \right\rangle + \left\langle \overrightarrow{F}_t, J\overrightarrow{\mathcal{C}}_p \right\rangle \,\mathrm{d}p, \tag{126}$$

where $\frac{d\vec{F}}{d\mathbf{x}}$ denotes the Jacobian matrix of \vec{F} with respect to \mathbf{x} . As seen in the previous section A.1, the first two terms of Equation 112 simplify to $[\oint_{\vec{C}} \langle \vec{C}_t f(\epsilon_1) \vec{N} \rangle ds]$ so Equation 126 becomes (with a change of variable for the third term):

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_{t} f(\epsilon_{1}) \overrightarrow{\mathcal{N}} \right\rangle + \left\langle \overrightarrow{F}_{t}, \overrightarrow{\mathcal{N}} \right\rangle \, \mathrm{d}s.$$
(127)

Using Green's theorem,

$$\oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{F}_t, \overrightarrow{\mathcal{N}} \right\rangle \, \mathrm{d}s = \int_R \nabla_{\mathbf{x}} \cdot \overrightarrow{F}_t(\mathbf{x}) \, \mathrm{d}\mathbf{x} = \int_R \frac{\partial f(\epsilon_1(\mathbf{x}, t))}{\partial t} \, \mathrm{d}\mathbf{x} = \int_R f'(\epsilon_1(\mathbf{x}, t))\epsilon_{1t}(\mathbf{x}, t) \, \mathrm{d}\mathbf{x}.$$

So,

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_t f(\epsilon_1) \overrightarrow{\mathcal{N}} \right\rangle \, \mathrm{d}s + \int_R f'(\epsilon_1(\mathbf{x}, t)) \epsilon_{1t}(\mathbf{x}, t) \, \mathrm{d}\mathbf{x}. \tag{128}$$

The second term does not immediately reveal the form of the gradient for \vec{C} (recall that we need to have an inner product with $\langle \vec{C}_t, . \rangle$). We further manipulate this term by noticing that ϵ_1 itself has the form of an energy (region integral whose integrand does not depend on $\overrightarrow{\mathcal{C}}$) and so taking its derivative with respect to t is equivalent to (using the technique of Section A.1):

$$\epsilon_{1t} = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_t, \mathcal{X}(\mathbf{x}, r, \overrightarrow{\mathcal{C}}) \overrightarrow{\mathcal{N}} \right\rangle \, \mathrm{d}s.$$
(129)

Plugging this in to Equation 128, we obtain:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_{t}, f(\epsilon_{1})\overrightarrow{\mathcal{N}} \right\rangle \,\mathrm{d}s + \int_{R} f'(\epsilon_{1}(\mathbf{x},t)) \left[\oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_{t}, \mathcal{X}(\mathbf{x},r,\overrightarrow{\mathcal{C}})\overrightarrow{\mathcal{N}} \right\rangle \,\mathrm{d}s \right] \,\mathrm{d}\mathbf{x},$$
(130)

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_t, f(\epsilon_1) \overrightarrow{\mathcal{N}} \right\rangle \,\mathrm{d}s + \int_R \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_t, f'(\epsilon_1(\mathbf{x}, t)) \mathcal{X}(\mathbf{x}, r, \overrightarrow{\mathcal{C}}) \overrightarrow{\mathcal{N}} \right\rangle \,\mathrm{d}s \,\mathrm{d}\mathbf{x}, \quad (131)$$

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \oint_{\overrightarrow{\mathcal{C}}} \left\langle \overrightarrow{\mathcal{C}}_t, \left[f(\epsilon_1) + \int_R f'(\epsilon_1(\mathbf{x}, t)) \mathcal{X}(\mathbf{x}, r, \overrightarrow{\mathcal{C}}) \,\mathrm{d}\mathbf{x} \right] \overrightarrow{\mathcal{N}} \right\rangle \,\mathrm{d}s.$$
(132)

Therefore the final gradient flow is:

$$\frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = -\left[f(\epsilon_1(\overrightarrow{\mathcal{C}})) + \int_R f'(\epsilon_1(\mathbf{x}, t))\mathcal{X}(\mathbf{x}, r, \overrightarrow{\mathcal{C}}) \, \mathrm{d}\mathbf{x}\right] \overrightarrow{\mathcal{N}},\tag{133}$$

$$\frac{\partial \overrightarrow{\mathcal{C}}}{\partial t} = \epsilon_1^p(\overrightarrow{\mathcal{C}}, r) + p \int_R \epsilon_1^{p-1}(\overrightarrow{\mathcal{C}}) \mathcal{X}(\mathbf{x}, r, \overrightarrow{\mathcal{C}}) \, \mathrm{d}\mathbf{x}.$$
(134)
REFERENCES

- ANGENENT, S., HAKER, S., TANNENBAUM, A., and KIKINIS, R., "Laplacebeltrami operator and brain surface flattening," *IEEE Trans. Medical Imaging*, vol. 18, pp. 700–711, 1999.
- [2] ANGENENT, S., PICHON, E., and TANNENBAUM, A., "Mathematical methods in medical image processing," *Bulletin of the AMS*, 2004. in preparation.
- [3] AUBERT, G., BARLAUD, M., FAUGERAS, O., and JEHAN-BESSON, S., "Image segmentation using active contours: Calculus of variations or shape gradients?," *SIAM Journal on Applied Mathematics*, vol. 63, no. 6, 2003.
- [4] AYACHE, N., "Medical computer vision, virtual reality and robotics," Image and Vision Computing, vol. 13, no. 4, pp. 295–313, 1995.
- [5] BRECHBÜHLER, C., GERIG, G., and KÜBLER, O., "Parametrization of closed surfaces for 3-d shape description," CVIU, vol. 61, pp. 154–170, March 1995.
- [6] CASELLES, V., CATTE, F., COLL, T., and DIBOS, F., "A geometric model for active contours in image processing," *Numerische Mathematik*, vol. 66, pp. 1–31, 1993.
- [7] CASELLES, V., KIMMEL, R., and SAPIRO, G., "Geodesic active contours," International Journal of Computer Vision, vol. 22, no. 11, pp. 61–79, 1997.
- [8] CHAN, T. F., SHEN, J., and VESE, L., "Variational pde models in image processing," *Notice of AMS*, vol. 50, pp. 14–26, January 2003.
- COHEN, L. D., "On active contour models and balloons," CVGIP: Image Understanding, vol. 53, pp. 211–218, March 1991.
- [10] COHEN, L. and COHEN, I., "Finite element methods for active contour models and balloons for 2d and 3d images," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 15, no. 11, pp. 1131–1147, 1993.
- [11] COOTES, T. F. and TAYLOR, C. J., "Combining point distribution models with shape models based on finite element analysis," *Image Vis. Computing*, vol. 13, no. 5, p. 4039, 1995.
- [12] COOTES, T., TAYLOR, C., COOPER, D., and GRAHAM, J., "Active shape models - their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995.
- [13] COURANT, R. and HILBERT, D., Methods of Mathematical Physics. Interscience, 1953.

- [14] DAVATZIKOS, C., TAO, X., and SHEN, D., "Hierarchical active shape models, using the wavelet transform," *IEEE Trans. Medical Imaging*, vol. 22, no. 3, pp. 414–423, 2003.
- [15] DELINGETTE, H. and MONTAGNAT, J., "Topology and shape constraints on parametric active contours." Technical report, INRIA, France, 2000.
- [16] DESCHAMPS, T. and COHEN, L., "Fast extraction of minimal paths in 3d images and application to virtual endoscopy," *Med. Image Anal*, vol. 5, no. 4, 2001.
- [17] DUDA, R., HART, P., and STORK, D., *Pattern Classification*. Wiley-Interscience, 2001.
- [18] EDGINGTON, E., ed., Randomization Tests. Acad Press, 1995.
- [19] FARIN, G., Curves and Surfaces for CAGD. New York: Academic Press.
- [20] FISCHL, B., SERENO, M., and DALE, A., "Cortical surface-based analysis ii: Inflation, flattening, and a surface-based coordinate system," *NeuroImage*, vol. 9, pp. 195–207, 1999.
- [21] FLASQUE, N., DESVIGNES, M., CONSTANS, J., and REVENU, M., "Acquisition, segmentation and tracking of the cerebral vascular tree on 3d magnetic resonance angiography images," *Med Image Anal*, vol. 5, pp. 173–83, Sept 2001.
- [22] FRANGI, A., NIESSEN, W., VINCKEN, K., and VIERGEVER, M., "Multiscale vessel enhancement filtering," in *MICCAI'98*, pp. 130–137, 1998.
- [23] GERIG, G., STYNER, M., JONES, D., WEINBERGER, D., and LIEBERMAN, J., "Shape analysis of brain ventricles using spharm," Workshop on Mathematical Methods in Biomedical Image Analysis (MMBIA), IEEE Computer Society, pp. 171–178, December 2001.
- [24] GERIG, G., STYNER, M., SHENTON, M. E., and LIEBERMAN, J. A., "Shape versus size: Improved understanding of the morphology of brain structures," in *MICCAI*, LNCS 2208, pp. 24–32, 2001.
- [25] GOLLAND, P., GRIMSON, W., and KIKINIS, R., "Statistical shape analysis using fixed topology skeletons: Corpus callosum study," in *IPMI*, LNCS 1613, pp. 382–387, 1999.
- [26] GOLLAND, P., GRIMSON, W. E. L., SHENTON, M. E., and KIKINIS, R., "Detection and analysis of statistical differences in anatomical shape," *Medical Image Analysis*, vol. 9, no. 1, pp. 69–86, 2005.
- [27] GRAYSON, M., "The heat equation shrinks embedded plane curves to round points," J. Differential Geometry, vol. 26, pp. 285–314, 1987.

- [28] GU, X., WANG, Y., CHAN, T., THOMPSON, P., and YAU, S., "Genus zero surface conformal mapping and its application to brain surface mapping," *IEEE Transactions on Medical Imaging*, vol. 23, pp. 949–958, Aug 2004.
- [29] HAN, X., XU, C., and PRINCE, J., "A topology preserving level set method for geometric deformable models," *IEEE Transactions on PAMI*, vol. 25, pp. 755– 768, June 2003.
- [30] HURDAL, M. K. and STEPHENSON, K., "Cortical cartography using the discrete conformal approach of circle packings," *Neuroimage*, vol. 23, pp. 119–128, 2004.
- [31] JU, L., STERN, J., REHM, K., SCHAPER, K., HURDAL, M., and ROTTEN-BERG, D., "Cortical surface flattening using least square conformal mapping with minimal metric distortion," in *Proceedings of the Second IEEE International Symposium on Biomedical Imaging*, pp. 77–80, 2004.
- [32] KASS, M., WITKIN, A., and TERZOPOULOS, D., "Snakes: Active contour models," Int. J. Comput. Vis, vol. 1, no. 4, pp. 321–331, 1987.
- [33] KICHENASSAMY, S., KUMAR, A., OLVER, P., TANNENBAUM, A., and YEZZI, A., "Conformal curvature flows: from phase transitions to active contours," *Archive for Rational Mechanics and Analysis*, vol. 134, pp. 275–301, 1996.
- [34] KRISSIAN, K., MALANDAIN, G., and AYACHE, N., "Model-based detection of tubular structures in 3d images," *Computer Vision and Image Understanding*, vol. 80, no. 2, pp. 130–171, 2000.
- [35] LEPETIT, V. and FUA, P., Foundations and Trends in Computer Graphics and Vision, vol. 1, pp. 1–89. Now Publishing, 2005.
- [36] LEVENTON, M., GRIMSON, E., and FAUGERAS., O., "Statistical shape influence in geodesic active contours," in *CVPR*, June 2000.
- [37] LEVITT, J., RW, R. M., DICKEY, C., VOGLMAIER, M., NIZNIKIEWICZ, M., SEIDMAN, L., HIRAYASU, Y., CISZEWSKI, A., KIKINIS, R., JOLESZ, F., and SHENTON, M., "MRI study of caudate nucleus volume and its cognitive correlates in neuroleptic-naive patients with schizotypal personality disorder," Am J. Psychiatry, vol. 159, no. 7, pp. 1190–97, 2002.
- [38] LORIGO, L. M., FAUGERAS, O., GRIMSON, W. E. L., KERIVEN, R., KIKI-NIS, R., NABAVI, A., and WESTIN, C.-F., "Codimension-two geodesic active contours," in *Comp. Vision and Patt. Recon. (CVPR)*, June 2000.
- [39] MALLAT, S., Wavelet Tour of Signal Processing. Academic Press, 1999.
- [40] MCINERNEY, T. and TERZOPOULOS, D., "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, no. 1, pp. 69–83, 1995.

- [41] MCINERNEY, T. and TERZOPOULOS, D., "Deformable models in medical image analysis: a survey," *Medical Image Analysis*, vol. 1, no. 2, pp. 91–108, 1996.
- [42] MCINERNEY, T. and TERZOPOULOS, D., "T-snakes:topology adaptive snakes," Medical Image Analysis, vol. 4, no. 2, pp. 73–91, 2000.
- [43] MS, K., JJ, L., RW, M., LJ, S., CC, D., MA, N., MM, V., ANDLONG KL, Z. P., SS, K., and ME., S., "Reduction of caudate volume in neuroleptic-naive female subjects with schizotypal personality disorder," *Biol Psychiatry 2006*, vol. 1, no. 60, pp. 40–48.
- [44] OSHER, S. and FEDKIW, R., Level Set Methods and Dynamic Implicit Surfaces. Springer Verlag, 2002.
- [45] PICHON, E., Novel Methods for Multidimensional Image Segmentation. PhD thesis, Georgia Institute of Technology, 2005.
- [46] PIZER, S., FRITSCH, D., YUSHKEVICH, P., JOHNSON, V., and CHANEY, E., "Segmentation, registration, and measurement of shape variation via image object shape," *IEEE Trans. Med. Imaging*, vol. 18, pp. 851–865, October 1999.
- [47] PRESS, W., TEUKOLSKY, S., VETTERLING, W., and FLANNERY, B., Numerical Recipes in C++. Cambridge University Press.
- [48] R. MALLADI, J. S. and VEMURI, B., "Shape modeling with front propagation: a level set approach," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 17, pp. 158–175, 1995.
- [49] ROUSSON, M. and CREMERS, D., "Efficient kernel density estimation of shape and intensity priors for level set segmentation," in *MICCAI*, vol. 3750, pp. 757 - 764, 2005.
- [50] SATO, Y., S.NAKAJIMA, SHIRAGA, N., ATSUMI, H., YOSHIDA, S., KOLLER, T., GERIG, G., and KIKINIS., R., "Three dimensional multi-scale line filter for segmentation and visualization of curvilinear structures in medical images," *Med.Imag Anal*, vol. 2, pp. 143–168, 1998.
- [51] SCHRÖDER, P. and SWELDENS, W., "Spherical wavelets: Texture processing," in *Rendering Techniques '95*, Springer Verlag, 1995.
- [52] SCHRÖDER, P. and SWELDENS, W., "Spherical wavelets: Efficiently representing functions on the sphere," *Computer Graphics Proceedings (SIGGRAPH 95)*, pp. 161–172, 1995.
- [53] SETHIAN, J., Level Set Methods and Fast Marching Methods. Cambridge University Press, 1999.
- [54] SHI, J. and MALIK, J., "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

- [55] SIMON, L., "Lectures on geometric measure theory," in *Proceedings of the Centre* for Mathematical Analysis, Australian National University, Canberra, 1983.
- [56] STAIB, L. and DUNCAN, J., "Boundary finding with parametrically deformable models," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 14, no. 11, pp. 1061–1075, 1992.
- [57] STAIB, L. and DUNCAN, J., "Deformable fourier models for surface finding in 3d images," in *Proc. 2nd. Conf. on Vis. in Biomed. Computing*, vol. 1808, pp. 90–104, 1992.
- [58] STYNER, M., LIEBERMAN, J., MCCLURE, R., WEINGBERGER, D., JONES, D., and GERIG, G., "Morphometric analysis of lateral ventricles in schizophrenia and healthy controls regarding genetic and disease- specific factors," *Proceedings* of the National Academy of Science, vol. 102, pp. 4872–4877, March 2005.
- [59] STYNER, M., LIEBERMAN, J., PANTAZIS, D., and GERIG, G., "Boundary and medial shape analysis of the hippocampus in schizophrenia," *Med Image Anal.*, vol. 8, no. 3, pp. 197–203, 2004.
- [60] STYNER, M., OGUZ, I., XU, S., BRECHBUHLER, C., PANTAZIS, D., LEVITT, J., SHENTON, M., and GERIG, G., "Framework for the statistical shape analysis of brain structures using spharm-pdm," Open Science Workshop at MICCAI 2006.
- [61] S.VETSA, STYNER, M., PIZER, S., LIEBERMAN, J., and GERIG, G., "Caudate shape discrimination in schizophrenia using template-free non-parametric tests," in *MICCAI*, 2003.
- [62] T. DESCHAMPS, L. C., "Fast extraction of tubular and tree 3d surfaces with front propagation methods," *International Conference on Pattern Recognition*, 2002.
- [63] TER HAAR ROMENY, B. M., ed., Computational Imaging and Vision. Kluwer, 1994.
- [64] TSAI, A., YEZZI, A. J., III, W. M. W., TEMPANY, C., TUCKER, D., FAN, A., GRIMSON, W. E. L., and WILLSKY, A. S., "A shape-based approach to the segmentation of medical imagery using level sets," *IEEE Trans. Med. Imaging*, vol. 22, no. 2, pp. 137–154, 2003.
- [65] VASILEVSKIY, A. and SIDDIQI, K., "Flux-maximizing geometric flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1565–1578, 2002.
- [66] WANG, K., DUTTON, R., and TAYLOR, C., "Improving geometric model construction for blood flow modeling," *IEEE Engineering in Medicine and Biology Magazine*, vol. 18, no. 6, pp. 58–69, 1999.

- [67] WEBSITE, N., "National alliance for medical image computing." http://www.namic.org.
- [68] WIKIPEDIA, "Caudate nucleus." http://en.wikipedia.org/wiki/Caudate. Last Accessed on May 15, 2006.
- [69] WIKIPEDIA, "Hippocampus." http://en.wikipedia.org/wiki/Hippocampus. Last Accessed on May 15, 2006.
- [70] WIKIPEDIA, "Wavelet." http://en.wikipedia.org/wiki/Wavelet. Last Accessed on May 15, 2006.
- [71] XU, C., PHAM, D. L., and PRINCE, J. L., Handbook of Medical Imaging, Volume 2, ch. Medical Image Segmentation Using Deformable Models, pp. 129– 174. SPIE Press, May 2000.
- [72] YANG, Y., TANNENBAUM, A., and GIDDENS, D., "Knowledge-based 3d segmentation and reconstruction of coronary arteries using ct images," in *IEEE EMBS*, pp. 1664–1666, 2004.
- [73] YEZZI, A. J., TSAI, A., and WILLSKY, A., "A fully global approach to image segmentation via coupled curve evolution equations," J. Vis. Comm. Im. Rep., vol. 13, pp. 195–216, 2002.
- [74] YU, P., SEGONNE, F., HAN, X., and FISCHL, B., "Shape analysis of neuroanatomical structures based on spherical wavelets," in *Human Brain Mapping* (*HBM*), 2005.
- [75] ZHU, S. C. and YUILLE, A., "Region competition: Unifying snakes, region growing, and bayes/mdl for multiband image segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, pp. 884–900, September 1996.

VITA

Delphine Nain was born in Chatenay Malabry in France on June 11th, 1979. She received both a Bachelor of Science in Electrical Engineering and Computer Science in 2001 and a Master of Engineering in Electrical and Computer Engineering in 2002 from the Massachusetts Institute of Technology. Her doctorate research focuses on computer vision and its applications to the analysis of medical imaging datasets. During her Ph.D, she also explored technology related policy issues as a fellow for the Sam Nunn/MacArthur Security Program at Georgia Tech.