Learning Task-Optimal Image Registration with Applications in Localizing Structure and Function in the Cerebral Cortex

by

B.T. Thomas Yeo

B.S., Electrical Engineering, Stanford University (2002)
M.S., Electrical Engineering, Stanford University (2002)

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

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2010

Author . . . .
Department of Electrical Engineering and Computer Science
September 15, 2009

Certified by . . . .
Polina Golland
Associate Professor of Electrical Engineering and Computer Science
Distinguished Alumnus (1964) Career Development Chair
Thesis Supervisor

Certified by . . . .
Bruce Fischl
Associate Professor of Radiology at Harvard Medical School
Director, Computational Core, Martinos Center, Massachusetts General Hospital
Thesis Supervisor

Accepted by . . . .
Terry P. Orlando
Chairman, Department Committee on Graduate Students

Learning Task-Optimal Image Registration with Applications in Localizing Structure and Function in the Cerebral Cortex

by

B.T. Thomas Yeo

Submitted to the Department of Electrical Engineering and Computer Science
on September 15, 2009, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

In medical image analysis, registration is necessary to establish spatial correspondences across two or more images. Registration is rarely the end-goal, but instead, the results of image registration are used in other tasks, such as voxel-based morphometry, functional group analysis, image segmentation and tracking.

In this thesis, we argue that the quality of image registration should be evaluated in the context of the application. Consequently, we develop a framework for learning registration cost functions optimized for specific tasks. We demonstrate that by taking into account the application, we not only achieve better registration, but also potentially resolve certain ambiguities and ill-posed nature of image registration.

We first develop a generative model for joint registration and segmentation of images. By jointly modeling registration and the application of image segmentation, we demonstrate improvements in parcellation of the cerebral cortex into different structural units.

In this thesis, we work with spherical representations of the human cerebral cortex. Consequently, we develop a fast algorithm for registering spherical images. Application to the cortex shows that our algorithm achieves state-of-the-art accuracy, while being an order of magnitude faster than competing diffeomorphic, landmark-free algorithms.

Finally, we consider the problem of automatically determining the “free” parameters of registration cost functions. Registration is usually formulated as an optimization problem with multiple tunable parameters that are manually set. By introducing a second layer of optimization over and above the usual registration, this thesis provides the first effective approach to optimizing *thousands* of registration parameters to improve alignment of a new image as measured by an application-specific performance measure.

Much previous work has been devoted to developing generic registration algorithms, which are then specialized to particular imaging modalities (e.g., MR), particular imaging targets (e.g., cardiac) and particular post-registration analyses (e.g., segmentation). Our framework provides a principled method for adapting generic algorithms to specific applications. For example, we estimate the optimal weights or cortical folding template of the generic weighted Sum of Squared Differences dissimilarity measure for localizing underlying cytoarchitecture and functional regions of the cerebral cortex. The generality of the framework suggests potential applications to other problems in science and engineering formulated as optimization problems.

Thesis Supervisor: Polina Golland
Title: Associate Professor of Electrical Engineering and Computer Science
Distinguished Alumnus (1964) Career Development Chair

Thesis Supervisor: Bruce Fischl
Title: Associate Professor of Radiology at Harvard Medical School
Director, Computational Core, Martinos Center, Massachusetts General Hospital

Thesis Reader: Nicholas Ayache
Title: Research Director, Asclepios Project-Team, INRIA
Professor, École Centrale, Paris

Thesis Reader: William T. Freeman
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I like to thank my advisors, Polina Golland and Bruce Fischl for their support and guidance over the years. When I first came to MIT, I simply have no idea the effort and dedication good research entails. It is because of them that I am what I am today — in other words, I blame them for any bad paper I have written. Jokes aside, I like to thank Polina for our brainstorming sessions, her insights into my research problems, her unbounded optimism and just as importantly, her meticulous efforts in improving the clarity and focus of my papers and presentations. I have never met anyone who has such a focused attitude towards improving a paper, even when she is not a major co-author. I like to thank Bruce for being always there to discuss the tiniest detail about our algorithms or to listen to explanations of my code during the debugging process. I swear that I occasionally get a reply from Bruce before I send out my email.

I would also like to thank my committee members, Nicholas Ayache and Bill Freeman, for their useful suggestions for the thesis. Nicholas is especially kind for flying all the way from France to Boston to attend my defense. Among my mentors, I will not forget Balaji Prabhakar and Dwight Nishimura, both at Stanford University, for encouraging me to tackle challenging problems. I have been fortunate to enjoy A*STAR's financial support for the past five years. I am also deeply appreciative of the special understanding and kindness provided (and continue to be provided) by Professor Michael Chee, Chairman Lim Chuan Poh, Professor Miranda Yap and A*GA HR staff during a trying period this past year.

My research is made possible because of the input of so many brilliant collaborators: Nicholas Ayache, Olivier Clatz, Rahul Desikan, Pierre Fillard, Ellen Grant, Wanmei Ou, Xavier Pennec, Jean-Marc Peyrat, Mert Sabuncu, Tom Vercauteren, Peng Yu. Tom and Mert have been especially critical to my research. Tom has taught (and is still teaching) me everything I know about diffeomorphisms. He is always resourceful and full of good ideas. Mert is instrumental to almost every paper I have written. He always has valuable insights to whatever problem I have. I have lost count of the number of times Mert walks into my office (while I am reading the New York Times) to discuss with me a brilliant new idea. I also have to thank Katrin Amunts, Randy Buckner, Rahul Desikan, Ellen Grant, Daphne Holt, Hartmut Mohlberg and Karl Zilles for their valuable data, without which our algorithms would be meaningless.

Biswajit Bose, Kilian Pohl, Florent Segonne, Michael Siracusa, Marshall Tappen and Lilla Zollei have given (and are still giving) me good advice since my first day at MIT. I will always remember Kilian's (hard-to-follow) advice of not being too exhilarated when things are going well, and not being too worried when things are going poorly. Special thanks must be offered to Serdar Balci, Wanmei Ou and Jenny Yuen for putting up with such a noisy officemate as me. Discussions with Stanley Durrleman, Danial Lashkari, Koen Van

Leemput, Wanmei Ou and Archana Venkataraman have always been stimulating. In short, I owe a tremendous amount of gratitude to my labmates at MIT, MGH and INRIA, too many to be listed here.

My stay at MIT would not be complete without the friends I have met here. I like to thank Biswajit Bose, Wee Peng Tay and Lav Varshney for being such wonderful apartment mates. It is always fun to hang out with Lynette Cheah, Jiamin Chin, Chiao-Lun Cheng, Yidong Chong, Lynn Chua, Shireen Goh, Angela Ho, Kongjie Kah, Henry Koh, Trina Koh, Nai Jia Lee, Kenneth Lim, Trevor Ng, Wui Siew Tan and many others at the Singaporean potlucks, enjoying the super-elaborate dishes that are made from scratch (think Bak Kua and Singaporean curry puffs). I cherish the many times I went with Andy Eow to Royal East, enjoying fried squids and ducklings, completely disregarding our cholesterol problems. Sybor Wang has been wonderful in showing me the best eats in town, despite our graduate student budget. Our road trip to Arcadia National Park was memorable for the lack of hiking and the amount of food we ate. I have also enjoyed hanging out with Kevin Lee, Hsu-Yi Lee and other Taiwanese students. I remember the awesome Christmas that Nancy and I spent in LA with Kevin and Hsu-Yi, watching the entire first season of Prison Break, breaking only for food.

I must also thank my Stanford friends, Kenny Cheung, Matt Corpos (and his family!), Jose Paterno, Chris Sprague, Erik Vandekieft and others for showing me a wonderful time every time I visit the Bay area to take a break from graduate school. Chris is particularly instrumental in ensuring everyone having a good time with his very persuasive mantra "Dude! just one more. Do it for Yeo".

Last but certainly not least, I like to thank my family for their support over the years. I like to especially thank my wife Nancy Chen, whose unrelenting support is critical to my graduation. I hope I have been just as supportive of her roller-coastal ride through graduate school.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Advances in neuroimaging have provided neuroscientists with powerful tools to study different aspects of the human brain (Fig. 1-1). For example, structural Magnetic Resonance Imaging (MRI) has allowed us to build detailed models of the cortical surface and its folding pattern. Diffusion Tensor Imaging (DTI) non-invasively measures the diffusion of water in *in-vivo* biological tissues. The diffusion is anisotropic in tissues such as cerebral white matter and therefore DTI is a powerful imaging modality for studying white matter structures in the brain. Histology and high-resolution MRI of *ex-vivo* brains offers us a glimpse into the cellular architecture of the cortex. Finally, functional Magnetic Resonance Imaging (fMRI) and Electroencephalography/Magnetoencephalography (EEG/MEG) provide insights into *in-vivo* brain activity. With so many imaging modalities, it is necessary to build models for individual modalities and to fuse data from different modalities.

Building these models usually requires us to establish spatial correspondences across images. These images can be snapshots of a single or multiple subjects using a single or multiple imaging modalities at single or multiple time instances. This process of bringing images into the same coordinate system is known as image registration and is perhaps one of the most fundamental problems in medical image analysis [31, 103, 122, 194]. Indeed, the concept of a common coordinate system is implicitly or explicitly assumed in almost every medical imaging application involving more than one image.

Traditionally, registration is considered a pre-processing step. Images are registered and are then used for other image analysis applications, such as voxel-based morphometry (VBM), functional group analysis, image segmentation, shape analysis and tracking anatomical changes or tumor growth. These applications aim at better understanding of brain function and neurological diseases, as well as at aiding surgical planning and early diagnosis of neuropathology.

This thesis is motivated by our belief that the notion of "correct registration" makes sense only within the context of the application. By properly considering the application when performing registration, we significantly improve the application performance and

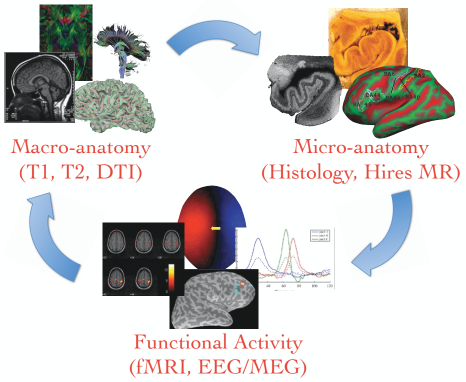


Figure 1-1: Different imaging modalities allow neuroscientists to study different aspects of the human brain.

potentially resolve certain ambiguities of image registration.

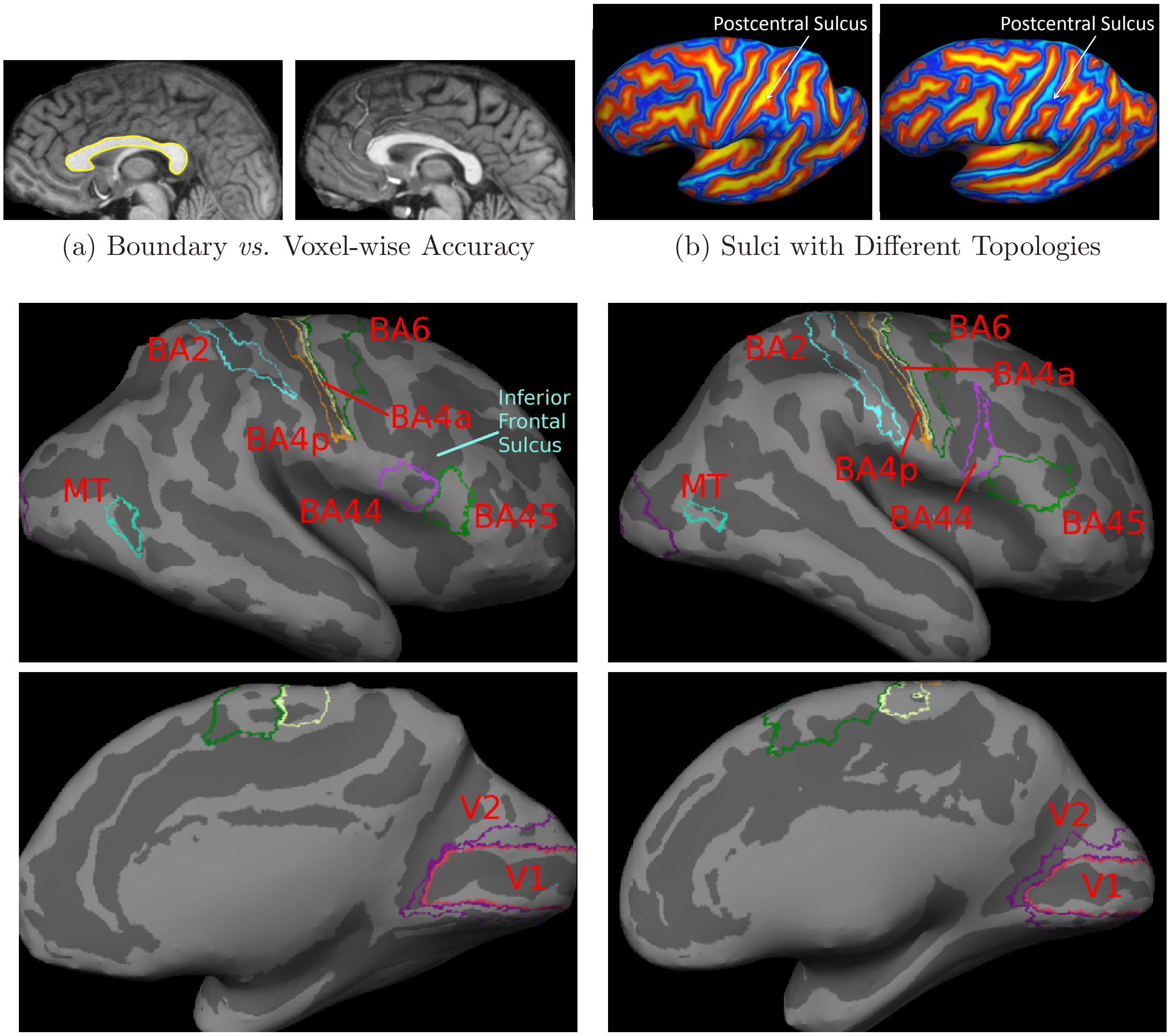
We begin this chapter by presenting examples demonstrating the ill-posed nature of image registration. We then discuss our primary motivation in this thesis: the localization of structures and functions in the human cerebral cortex. We conclude this chapter with a summary of our contributions and a preview of the results.

1.1 Motivating Examples

We will now discuss three different examples that illustrate ambiguities and difficulties that arise in medical image registration when we ignore the end-goal application.

1. Fig. 1-2(a) shows the sagittal slices of the brains of two subjects. Depending on the application when registering the brain images, one might be more concerned with accuracy in registering boundaries of structures, rather than voxel-wise accuracy. For example, suppose the goal is to segment the corpus callosum in the second image. A common approach is to register the first image to the second image, and to transfer the manual labels from the first image to second image. Therefore, for accurate segmentation, it is sufficient to accurately align the boundaries of corpus callosum in the two images. On the other hand, deformation-based morphometry [13, 36] extracts local morphometric information, such as local volumetric changes, from dense deformation fields computed between images. Because deformation-based morphometry is typically performed on entire images, rather than regions of interests, voxel-wise accuracy is desirable. This suggests that for the same pair of images, depending on the application, the requirements for registration can be quite different.

2. The variability of the folding pattern in the human cerebral cortex is well-documented



(a) Boundary *vs.* Voxel-wise Accuracy

(b) Sulci with Different Topologies

(c) Brodmann Areas Overlaid on Cortex

Figure 1-2: Examples of ambiguities and ill-posedness in image registration, which can potentially be resolved by taking the application at hand into account.

(see e.g. [116]). Fig. 1-2(b) shows postcentral sulci of two different subjects. Note the differences in topology between the two sulci. When matching cortical folds, even neuroanatomical experts do not agree on whether to join the ends of the broken sulcus or to break up the uninterrupted sulcus.

3. In population studies of human brain mapping, it is common to align subjects into a single coordinate system by aligning macroanatomy or cortical folding patterns. The pooling of functional data in this common coordinate system boosts the statistical power of group analysis and allows functional findings to be compared across different studies. However, substantial cytoarchitectonic [4, 5, 56] and functional [108, 149, 150, 151, 172, 176] variability is widely reported. One reason for this variability is

certainly mis-registration of the highly variable macroanatomy. However, even if we perfectly align the macroanatomy, the underlying function and cellular architecture of the cortex will not be aligned because the cortical folds do not completely predict the underlying brain function. To illustrate this, Fig. 1-2(c) shows nine Brodmann Areas (BAs) projected on the cortical surfaces of two different subjects, obtained from histology. BAs define cytoarchitectonic parcellation of the cortex closely related to brain function [30]. Here, we see that perfectly aligning the inferior frontal sulcus (Fig. 1-2(c)) will misalign the superior end of BA44 (Broca's language area). If our goal is to segment sulci and gyri, perfect alignment of the cortical folding pattern is ideal. However, it is unclear that perfectly aligning cortical folds is optimal for function localization.

1.2 Thesis Overview

The examples above motivate our argument that the quality of image registration should be evaluated in the context of the application. We will show in this thesis that by taking into account the application, we achieve better registration as measured by the application performance.

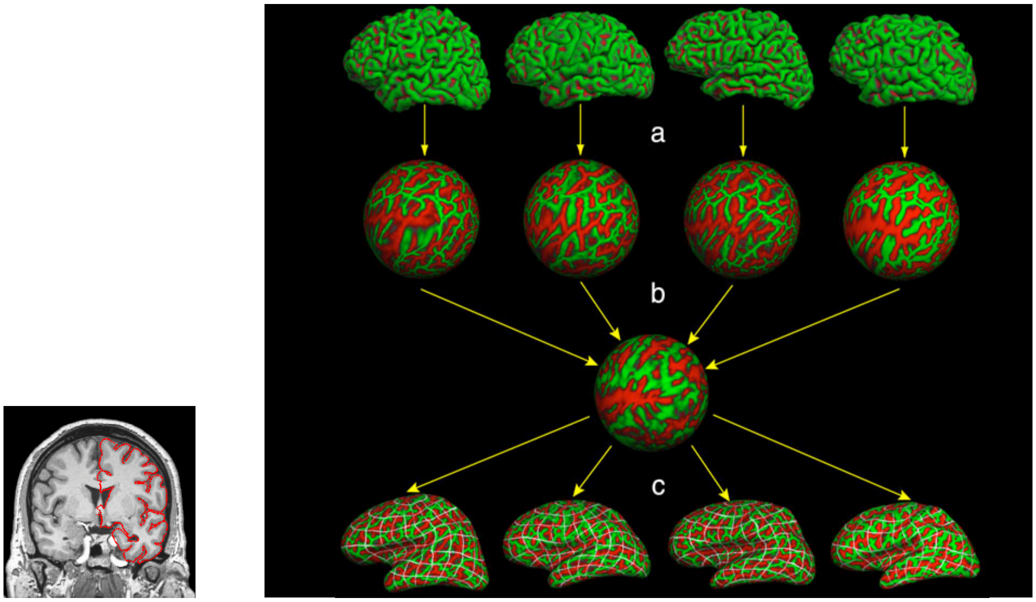
1.2.1 Cerebral Cortex

Although the formulations we develop in this thesis are general, we will focus mostly on the human cerebral cortex. The cortex is a thin, convoluted sheet about 2-4mm thick, forming the outer layer of the brain. The cortex plays an important role in cognitive functions, including memory, awareness and language. The cortical folding pattern is correlated with both cyto-architectonics and function.

With the use of modern 3D MRI, each half of the cortex can be accurately segmented and modeled as a closed 2D mesh[1] in 3D (Fig. 1-3(i)). Because the resulting hemispherical mesh has the same topology as a sphere, we can impose a spherical coordinate system on the mesh (Fig. 1-3(iia)). Here, we choose a spherical coordinate system that minimizes metric distortion [42, 58]. The result is a spherical image representation of the cortex, where the image intensity at each vertex encodes the local geometry of the cortex. While there are many possible choices of geometric features [28, 89, 177, 178, 187, 190], in this thesis, we choose to work with average convexity and mean curvatures of the partially inflated (bottom row of Fig. 1-3(ii)) and original cortical surface [58, 59].

Since cortical areas – both structure and function – are arranged in a mosaic across the cortical surface, a major advantage of the spherical representation is that the cortical

[1] Technically, the cortex does not have the same topology as a sphere because of the corpus callosum connecting the hemispheres and the brain stem carrying signals to and from the spinal cord. The corpus callosum and brain stem are typically removed as a preprocessing step [42].



(i) Coronal MR Slice (ii) Spherical Representation and Registration

Figure 1-3: Extraction, representation and registration of cortical surfaces. In (a), the segmented cortical surface is overlaid onto the coronal slice of a 3D MRI volume. In (b), the green regions indicate gyri (ridges) and the red regions indicate sulci (valleys). The bottom row in (b) displays partially inflated cortical surfaces.

regions that are far apart as measured along the cortical surface remains far apart on the sphere, which is not the case in the original 3D coordinates. Registration of the cortical surfaces then involves warping the spherical coordinate systems (Fig. 1-3(iib)). We note that in this case, registration establishes correspondences between the surfaces, but does not actually change the 3D vertex locations of the original surfaces (Fig. 1-3(iic)).

1.2.2 Applications

In this thesis, we will consider three applications of localizing structure and function in the cerebral cortex.

1. We consider automatic parcellation of the cortex into different sulci (grooves) and gyri (ridges). Automatic labeling of surface models of the cerebral cortex is important for identifying regions of interest for clinical, functional and structural studies [43, 130]. Recent efforts range from identification of sulcal or gyral ridge lines [148, 159] to segmentation of sulcal or gyral basins [43, 60, 88, 97, 105, 128, 130]. Here, we consider the left and right cortical surfaces of 39 subjects. The data set exhibits significant anatomical variability since it contains young, middle-aged, elderly subjects and Alzheimer's patients. The surfaces are topologically corrected [55, 145] and a spherical coordinate system is established by minimizing metric distortion [58].The surfaces are manually parcellated by a neuroanatomical expert into 35 labels [43] so that they can be used

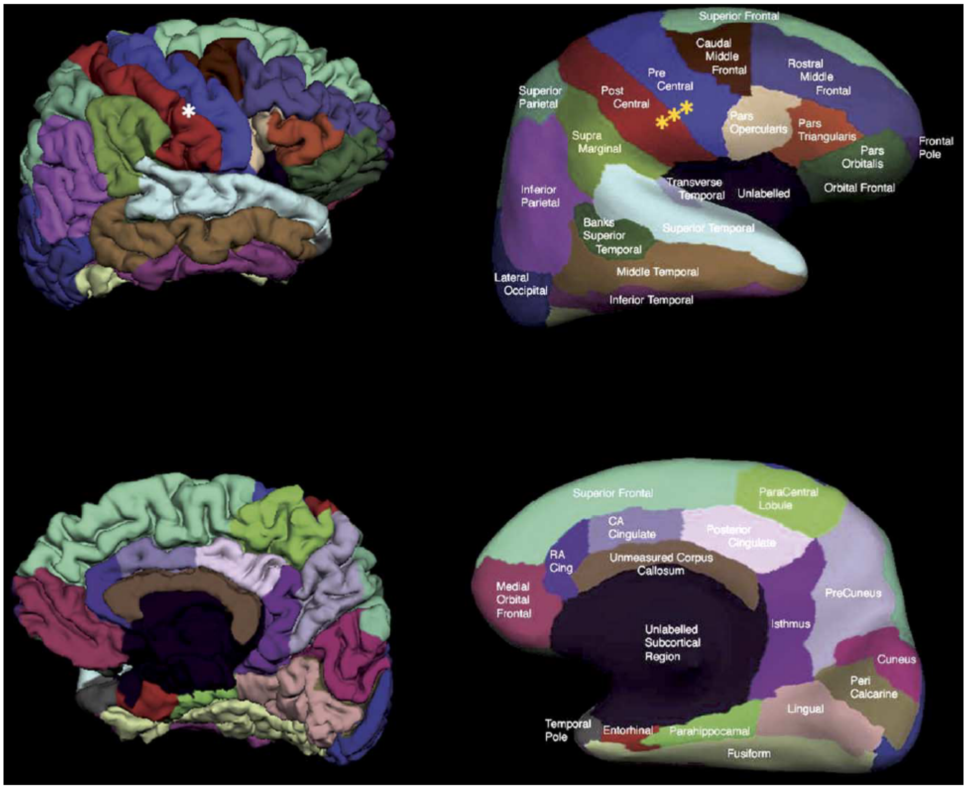


Figure 1-4: Example of manual parcellation shown on a partially inflated cortical surface. In our data set, the neuroanatomist preferred gyral labels to sulcal labels. There are also regions where sulci and gyri are grouped together as one label, such as the superior and inferior parietal complexes.

both for training and evaluating the accuracy of the segmentation algorithm. Figure 1-4 illustrates the manual parcellation for one subject. A complete list of the parcellation units is included in Table 1.1.

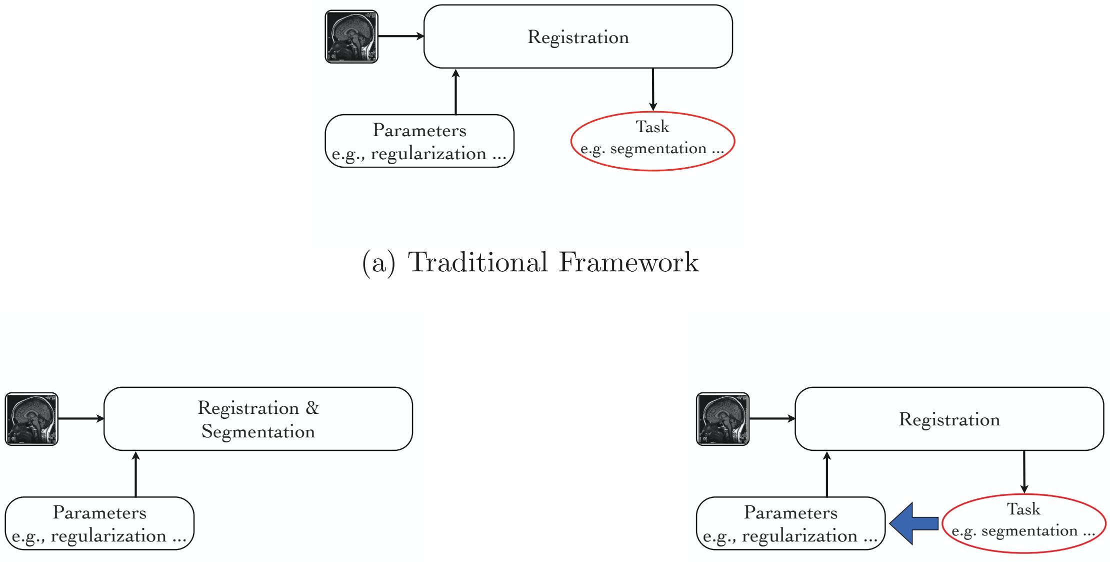
2. We consider the problem of localizing Brodmann Areas (BAs) in the cortex using only cortical folding patterns. As discused in Section 1.1, BAs are cyto-architectonically defined parcellations of the cerebral cortex [30], more closely related to brain function than macro-anatomy. They can be observed through histology and more recently, through *ex-vivo* high resolution MRI [14]. Unfortunately, much of the cytoarchitectonics cannot be observed with current *in-vivo* imaging. Nevertheless, most studies today report their functional findings with respect to Brodmann areas, usually estimated by visual comparison of cortical folds with Brodmann's original drawings without quantitative analysis of local accuracy. By combining histology and MRI, recent methods for creating probabilistic Brodmann area maps in the Talairach, Colin27 and cortical folding normalized space promise a more principled approach [4, 49, 56, 143, 144, 193].

Table 1.1: List of Parcellation Structures

1.	Sylvian Fissure / Unknown	2.	Bank of the Superior Temporal Sulcus
3.	Caudal Anterior Cingulate	4.	Caudal Middle Frontal Gyrus
5.	Corpus Callosum	6.	Cuneus
7.	Entorhinal	8.	Fusiform Gyrus
9.	Inferior Parietal Complex	10.	Inferior Temporal Gyrus
11.	Isthmus Cingulate	12.	Lateral Occipital
13.	Lateral Orbito Frontal	14.	Lingual
15.	Medial Orbito Frontal	16.	Middle Temporal Gyrus
17.	Parahippocampal	18.	Paracentral
19.	Parsopercularis	20.	Parsorbitalis
21.	Parstriangularis	22.	Peri-calcarine
23.	Post-central Gyrus	24.	Posterior Cingulate
25.	Pre-central Gyrus	26.	Pre-cuneus
27.	Rostral Anterior Cingulate	28.	Rostral Middle Frontal
29.	Superior Frontal Gyrus	30.	Superior Parietal Complex
31.	Superior Temporal Gyrus	32.	Supramarginal
33.	Frontal Pole	34.	Temporal Pole
35.	Transverse Temporal		

Here, we work with postmortem histological images of ten brains created using the techniques described in [5, 143, 193]. The histological sections were aligned to postmortem MR with nonlinear warps to build a 3D histological volume. These volumes were segmented to separate white matter from other tissue classes, and the segmentation was used to generate topologically correct and geometrically accurate surface representations of the cerebral cortex using FreeSurfer [42]. We consider nine semi-automatically labeled Brodmann area maps (areas 2, 4a, 4p, 6, 44, 45, V1, V2 and MT) mapped onto the surface representations of each hemisphere. Errors in this mapping were manually corrected (e.g., when a label was erroneously assigned to both banks of a sulcus). A morphological close was then performed on each label to remove small holes. We note that Brodmann areas 4a, 4p and 6 were mapped in only eight of the ten subjects. Fig. 1-2(c) shows these nine Brodmann areas on the resulting cortical representations for two subjects.

3. We consider the application of localizing fMRI-defined functional areas in the cortex using only cortical folding patterns. FMRI measures the haemodynamic response related to neural activity in the brain [92] and is perhaps the most popular measure of brain activity locations. Here, we consider the so-called MT+ area localized in 42 *in-vivo* subjects using fMRI. The MT+ area defined functionally is thought to include primarily the cytoarchitectonically-defined MT and a small part of the medial supe-



(a) Traditional Framework

(b) Joint Registration-Segmentation

(c) Task-Optimal

Figure 1-5: Constrast between our approach and the traditional one.

rior temporal (MST) area (hence the name MT+). The imaging paradigm involved subjects viewing an alternating 16 second blocks of moving and stationary concentric circles. The structural scans were processed using the FreeSurfer pipeline [62], resulting in spherically parameterized cortical surfaces [42, 59]. The functional data were analyzed using the general linear model [63]. The resulting activation maps were thresholded by drawing the activation boundary centered around the vertex with maximal activation. The threshold was varied across subjects in order to maintain a relatively fixed ROI area of about 120mm^2($\pm$5%) as suggested in [156].

1.2.3 Contributions

The contributions of this thesis are as follows:

1. We propose a generative model for joint registration and segmentation (Fig. 1-5(b), Fig. 1-6(a)) [179, 180]. By jointly modeling both registration and the application of segmentation, we achieve improved parcellation of the cerebral cortex into different gyral units in *in-vivo* brain images.

2. We develop a fast algorithm for registering spherical images [183, 184]. By leveraging the tools of differential geometry (Fig. 1-6(b)) and spherical spline vector interpolation (Fig. 1-6(c)), the resulting Spherical Demons algorithm achieves state-of-the-art accu-

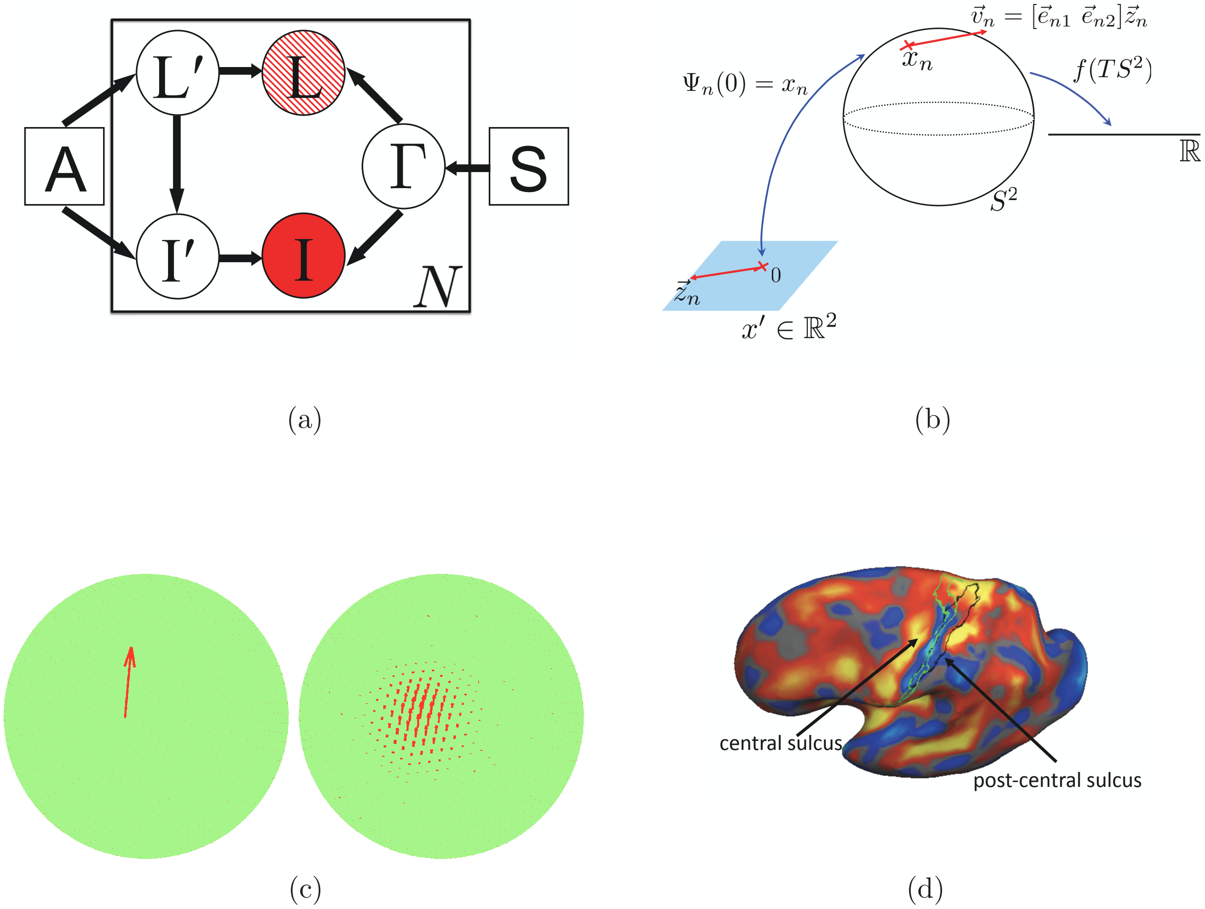


Figure 1-6: Preview. (a) Generative model for joint registration and segmentation. (b) Coordinate chart parameterization of the sphere S^2. (c) Spherical vector spline interpolation. Left image: input vector field; Right image: output smoothed vector field. (d) Task-optimal template for localizing Brodmann Area 2.

racy, while being an order magnitude faster than competing diffeomorphic, landmark-free algorithms. We demonstrate our algorithms in two different applications that use registration to transfer segmentation labels onto a new image: (1) parcellation of *in-vivo* cortical surfaces and (2) Brodmann area localization in *ex-vivo* cortical surfaces.

3. We develop the task-optimal registration framework [181] that optimizes the parameters of any differentiable family of registration cost functions on a training set, with the aim of improving the performance of a particular task for a new image (Fig. 1-5(c)). This thesis provides the first effective approach of optimizing *thousands* of registration parameters to improve alignment of a new image as measured by an application-specific performance measure. Our framework leads to the estimation of optimal

cortical folding template and optimal weighting of different cortical regions for localizing cytoarchitectural and functional regions in the cerebral cortex (Fig. 1-6(d)). The generality of the framework suggests potential applications to other problems in science and engineering formulated as optimization problems.

1.3 Thesis Outline

In the next chapter, we present a review of related work in registration and segmentation. Chapter 3 discusses the generative model for joint registration and segmentation. It is followed by the presentation of the fast Spherical Demons registration algorithm in Chapter 4. Finally, we develop the task-optimal registration framework for estimating the parameters of registration cost functions in Chapter 5. We conclude with a discussion of the implications of this dissertation and future work in Chapter 6.

Chapter 2

Registration, Segmentation and Atlas Construction

In this chapter, we present a survey of related work in medical image registration, segmentation and atlas construction. More specifically, we focus on intensity-based (as opposed to landmark-based) registration and atlas-based medical image segmentation. For a more complete review, we refer the readers to the multiple surveys of image registration [31, 75, 103, 122, 194] and segmentation [24, 41, 109, 117, 121].

In the next section, we provide a brief overview of modern pairwise and groupwise intersubject registration methods. In Section 2.2, we describe existing approaches to atlas-based segmentation. This is followed by an in-depth discussion of the contributions of this thesis in specific areas of registration and segmentation in Section 2.3.

2.1 Registration

Registration is the process of establishing spatial correspondences across images. These images can be snapshots of a single or multiple subjects using a single or multiple imaging modalities at single or multiple time instances.

Intra-subject intra-modality registration involves normalizing images of a single subject of the same modality, such as when tracking the disease progression of patients [61]. Examples of intra-subject inter-modality registration include the alignment of fMRI and/or anatomical images into the 3D space that the patient and surgeon reside in during a surgical procedure [171] and the registration of functional scans to the anatomical image of the same subject [6].

In population group analysis, inter-subject registration is frequently needed. For example, inter-subject intra-modality registration is required when analyzing structural MR [10, 11, 16, 86, 196], diffusion MR [1, 118, 185, 186, 192, 191] or fMRI [7, 59, 64, 110, 127, 163].

This thesis focuses on inter-subject registration, which is especially challenging due to high anatomical and functional variability in the human population. One example is the folding patterns of the cerebral cortex (Fig. 1-2(b)). Because cortical folding patterns correlate with both cytoarchitectural [56] and functional regions [59], in group studies of structure or function, it is desirable to normalize the sulci (valleys) and gyri (ridges) of different subjects.

While the applications of interest utilize multiple modalities for inference on the functional organization of the brain, this thesis does not deal with the intra-subject inter-modality registration required (e.g., histology with MRI and fMRI with MRI) in the creation of the ground truth data. In other words, we assume that the distribution of inter-modality misregistration within any subject is such that the comparison of different algorithms is not biased, i.e., algorithm A is better than algorithm B regardless of the inaccuracies in the data. Obviously, an improvement in the intra-subject inter-modality registration can improve the accuracy of the data and thus the applications of interest.

2.1.1 Pairwise Registration

We first discuss pairwise registration since registration of a group of images can typically be reduced to registration of pairs of images. Let $F : \Omega_F \mapsto \mathbb{R}$ and $M : \Omega_M \mapsto \mathbb{R}$ be two images defined on the discrete sets of voxels or vertices, $\Omega_F, \Omega_M \subset D$. Classically, the domain D is either $\mathbb{R}^2$ or $\mathbb{R}^3$. In this thesis, we deal with spherically parameterized meshes, so that D corresponds to the sphere S^2. Let $\Gamma : D \mapsto D$ denote the spatial transformation that maps voxel (vertex) coordinates x in image F to the coordinates $\Gamma(x)$ in image M. Registration of images F and M is traditionally formulated as an optimization problem over the deformation Γ:

$$\Gamma^* = \underset{\Gamma}{\operatorname{argmin}} \, \lambda \, \text{Dissim}(F, M \circ \Gamma) + \text{Reg}(\Gamma), \tag{2.1}$$

where the first term is the dissimilarity between the image F and the deformed image $M \circ \Gamma$ and the second term is the regularization on the deformation Γ. Consequently, F is known as the fixed image and M is known as the moving image, since interpolation is performed on M. λ is the tradeoff between the dissimilarity term and regularization term. For appropriate similarity measures and regularization, Eq. (2.1) can be probabilistically interpreted as

$$\begin{aligned} \Gamma^* &= \underset{\Gamma}{\operatorname{argmax}} \log p(M|\Gamma; F) + \log p(\Gamma) && (2.2) \\ &= \underset{\Gamma}{\operatorname{argmax}} \, p(\Gamma|M; F), && (2.3) \end{aligned}$$

where $\text{Dissim}(F, M \circ \Gamma) = -\log p(M|\Gamma; F)$ and $\text{Reg}(\Gamma) = -\log p(\Gamma)$. In other words, registration can be interpreted as Maximum-A-Posteriori estimation of the unknown random deformation Γ.

To instantiate the optimization problem in Eq. (2.1), we need to define the dissimilarity measure, the regularization, the feasible set of allowable deformations and the choice of interpolation. We discuss the possible choices in the context of our thesis contributions in Section 2.3.

2.1.2 Groupwise Registration and Templates

To register a group of images $\{I_n\}$, one generalizes the pairwise formulation as follows:

$$\{\Gamma_n^*\} = \operatorname*{argmin}_{\{\Gamma_n\}} \lambda \, \text{Dissim}(\{I_n \circ \Gamma_n\}) + \text{Reg}(\{\Gamma_n\}). \tag{2.4}$$

Since pairwise registration is a well-studied problem, most instantiations of Eq. (2.4) involve defining a template image T, so that the groupwise registration of N images can be formulated as

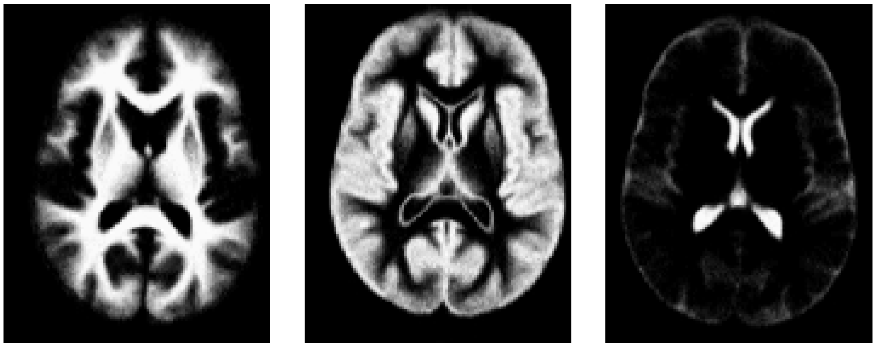
$$\Gamma_n^* = \operatorname*{argmin}_{\Gamma_n} \lambda \text{Dissim}(T, I_n \circ \Gamma_n) + \text{Reg}(\Gamma_n) \quad n = 1, \cdots, N. \tag{2.5}$$

The resulting groupwise registration problem is therefore reduced to N independent pairwise registration problems. The template T can simply be one of the images $\{I_n\}$. To avoid bias induced by choosing a particular subject, the template T can also be estimated together with the deformations [2, 16, 59, 86]:

$$(\{\Gamma_n^*\}, T^*) = \operatorname*{argmin}_{\{\Gamma_n\}, T} \lambda \sum_n \text{Dissim}(T, I_n \circ \Gamma_n) + \text{Reg}(\Gamma_n). \tag{2.6}$$

Optimization typically proceeds via coordinate descent, by iterating between (1) fixing the template T and registering each image I_n to the template and (2) fixing the deformations and estimating the template T using Eq. (2.6). The template is also often referred to as an atlas. We use the two words interchangeably in this thesis.

We note that Eq. (2.6) is quite general. Even the "template-free" congealing groupwise registration [196] can be reformulated as Eq. (2.6), where T is a template histogram rather than a template image. We also note that recent advances have generalized Eq. (2.6) to estimate multiple templates within a given population of subjects [2, 27, 136].



(a) WM (b) GM (c) CSF

Figure 2-1: Tissue probability maps of white matter (WM), gray matter (GM) and cerebral spinal fluid (CFS). Images from [19].

2.2 Template-based Segmentation

Segmentation is the partitioning of an image into different regions or the delineating of a region of interest within an image. Here, we focus on the classification of individual image voxels (or mesh vertices) into different segmentation labels. The methods proposed for voxel-based segmentation of MR images can be grouped into two broad categories. The first approach involves an unsupervised classification of each image voxel into different segmentation labels (e.g., tissue types) [170].

The second approach performs registration of the image to an atlas template [38, 101]. The atlas encodes prior segmentation knowledge, leading to better segmentation. Probabilistic atlases provide a principled approach for encoding prior segmentation knowledge [12, 37, 43, 57, 60, 123]. The simplest probabilistic segmentation atlases provide only the prior probability of labels at a spatial position and no information about the expected appearance of the image (Fig. 2-1). The label probabilities are combined with the intensity of the new image to produce the final segmentation [12]. This approach estimates the relationship between labels and image intensities in the new image, rather than assuming the intensity distribution of a new image is the same as that of training images. This is usually achieved by essentially clustering the voxel intensities [93]. Consequently, this approach tends to have difficulties distinguishing fine structures that have similar image intensities (Fig. 2-2). Recent advances have made significant progress in resolving structures with similar image intensities, such as by the use of topological priors of segmented structures [20].

More complex probabilistic atlases provide statistics on the relationship between the segmentation labels and the image intensities [43, 51, 57, 60, 123] (Fig. 2-2). The intensity model in the atlas is used to bring the atlas and the new image into the same space. The label probabilities and intensity model are then used to segment the new image. Such a registration process can be done sequentially [43, 57, 60] or jointly with segmentation [123]. Since this complex atlas contains more prior information than the simpler atlases that

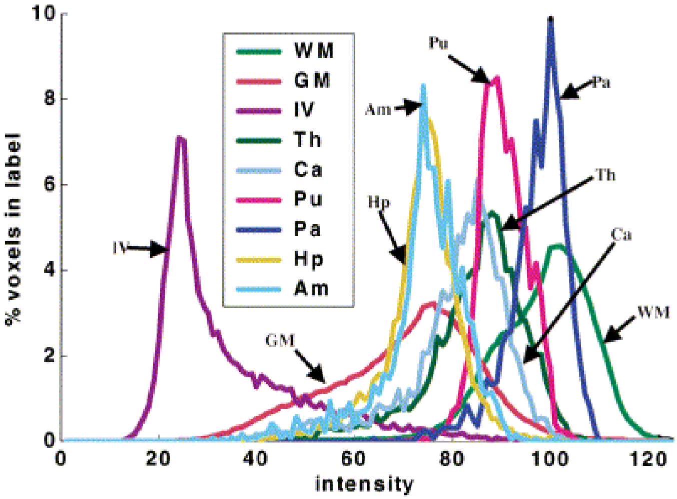


Figure 2-2: Intensity histograms of various cortical and subcortical structures. Images from [57].

capture label statistics only, it can be a more powerful tool for segmentation. However, extra care is needed if the imaging modalities of the new image and of the training images are different.

2.2.1 Atlas Construction

An initial step in probabilistic atlas computation is the spatial normalization of the training images. The features used for co-registering images are usually derived from the images themselves [2, 25, 44, 58, 69, 86, 107, 120, 160] or from the segmentation labels [40, 99, 165, 169, 182]. After normalization to a common space, the statistics of the intensity values (and possibly of the labels) in the atlas space are computed. This method is based on the assumption that a new image registered into the atlas space exhibits the same statistics as those computed from the registered training images.

Spatial normalization of the training subjects can be achieved with different registration algorithms that vary in the flexibility of warps they allow. Both low-dimensional warps, e.g., affine [123], and richer, more flexible warps can be employed, e.g., splines [25, 160, 169], dense displacement fields [40, 69, 58, 60] and velocity fields [86, 99]. Yet another route to vary the warp flexibility is to tune the tradeoff λ between the regularization and the dissimilarity measure. More restricted warps yield blurrier atlases that capture inter-subject variability of structures, enlarging the basin of attraction for the registration of a new subject. However, label prediction accuracy is limited by the sharpness of the atlas. The choice of the deformation flexibility is a major problem in registration, which we will address in Section 2.3.2.

2.3 Thesis Contributions

We now discuss the contributions of this thesis towards different aspects of registration and atlas-based segmentation. In particular, we will discuss our contributions to (1) spherical image registration and (2) general registration and template construction theory.

2.3.1 Spherical Image Registration

As mentioned in the introduction, the cortex is typically modeled as a 2D mesh embedded in 3D. There has been much effort focused on registering cortical surfaces directly in 3D [46, 47, 66, 81, 82, 153]. Since cortical areas — both structure and function — are arranged in a mosaic across the cortical surface, an alternative approach is to model the surface as a 2D closed manifold in 3D and to warp the underlying spherical coordinate system [50, 59, 126, 146, 157, 154, 163]. Modeling the cortex as a surface has led to multiple discoveries in neuroscience, such as the effects of aging and diseases [54, 91, 133, 141]. More recent efforts have sought to combine both approaches [84, 85, 124].

Nonlinear registration of spherical images presents distinct challenges due to differences between the geometry of the sphere and the standard Euclidean spaces. In Chapter 4, we develop the Spherical Demons algorithm to greatly speed up the registration of spherical images. We note that while fast algorithms are useful for deploying the developed tool on large datasets, they can further allow for complex applications that were previously computationally intractable. For example, incorporating the ideas behind Spherical Demons renders the task-optimal registration framework in Chapter 5 tractable. Spherical Demons speed up spherical image registration by focusing on the challenges of warp parameterization and regularization.

Warp Parameterization

In registration, there is frequently a need for invertible deformations. The assumption is that the topology of structural or functional regions are preserved across subjects. Unfortunately, this causes many spherical warping algorithms to be computationally expensive. Previously demonstrated methods for cortical registration [59, 157, 180] rely on soft regularization constraints to encourage invertibility. These require unfolding the mesh triangles, or limiting the size of optimization steps to achieve invertibility [59, 180]. Elegant regularization penalties that guarantee invertibility exist [10, 114] but they explicitly rely on special properties of the Euclidean image space that do not hold for the sphere.

An alternative approach to achieving invertibility is to work in the group of diffeomorphisms [9, 16, 21, 46, 67, 106, 167, 168]. In this case, the underlying theory of flows of vector fields can be extended to manifolds [26, 111, 115]. The Large Deformation Diffeomorphic Metric Mapping (LDDMM) [16, 21, 46, 67, 106] is a popular framework that seeks a time-varying velocity field representation of a diffeomorphism. Because LDDMM optimizes over the entire path of the diffeomorphism, the resulting method is slow and memory intensive. In contrast, Ashburner [9] and Hernandez *et al.* [73] consider diffeomorphic transformations parameterized by a single stationary velocity field. While restricting the space of solutions reduces the memory needs relative to LDDMM, these algorithms still have to consider the entire trajectory of the deformation induced by the velocity field when computing the gradients of the objective function, leading to a long run time. We note that recent algorithmic advances [74, 106] promise to improve the speed and relieve the memory requirements of both LDDMM and the stationary velocity approach.

We choose to adopt the approach of the Diffeomorphic Demons algorithm [168], originally demonstrated in the Euclidean image space, which constructs the deformation space that contains compositions of diffeomorphisms, each of which is parameterized by a stationary velocity field. Unlike the Euclidean Diffeomorphic Demons, the Spherical Demons algorithm utilizes velocity vectors tangent to the sphere and not arbitrary 3D vectors. This constraint does not require additional explicit steps in our algorithm since we directly work with the tangent spaces. In each iteration, the method greedily seeks the locally optimal diffeomorphism to be composed with the current transformation. As a result, the approach is much faster than LDDMM [16, 21, 46, 67] and its simplifications [9, 73]. A drawback is that the path of deformation is no longer a geodesic in the group of diffeomorphisms, which is problematic if the deformation path is used to measure similarities between images.

Warp Regularization

Since most types of regularization favor smooth deformations, the gradient computation of the cost function in Eq. (2.1) is complicated by the need to take into account the deformation in neighboring regions. This coupling among the voxels or vertices of an image slows down the registration algorithm. For Euclidean images, the popular Demons algorithm [152] can be interpreted as optimizing an objective function with two regularization terms [32, 168] instead of one in Eq. (2.1). The special form of the objective function facilitates a fast two-step optimization where the second step handles the warp regularization via a single convolution with a smoothing filter.

Using spherical vector spline interpolation theory [67] and other differential geometric tools, we show that the two-stage optimization procedure of Demons can be efficiently approximated on the sphere. We note that the problem is not trivial since tangent vectors at different points on the sphere are not directly comparable. We also emphasize that this decoupling of the image similarity and the warp regularization could also be accomplished with a different space of admissible warps, e.g., spherical thin plate splines [197].

One problem we will not consider is the difficulty of performing interpolation on a spherical grid, unlike a regular Euclidean grid. In this work, we use existing methods for interpolation, requiring about one second to interpolate data from a spherical mesh of 160k vertices onto another spherical mesh of 160k vertices. Recent work on using different coordinate charts of the sphere [161] promises to further speed up our implementation of the Spherical Demons algorithm.

2.3.2 Registration and Template Construction

We now discuss our contributions towards general theories of registration and atlas construction.

Joint Registration-Segmentation

Joint registration-segmentation algorithms are generally more effective than sequential procedures that first register the new image to the atlas and then segment the new image based on the priors and model parameters provided by the atlas since registration and segmentation benefit from the additional knowledge that each provides the other [12, 123, 173, 174, 175, 188]. In joint methods, the registration and segmentation of images are estimated simultaneously under a single framework. Previous work on joint registration and segmentation typically assumes a precomputed registration-segmentation atlas.

In Chapter 3, we develop a generative model for the joint registration and segmentation of images, such that the atlas construction arises naturally as estimation of the model parameters. The estimation process proceeds via coordinate descent, by iterating between (1) fixing the atlas parameters and registering the training subjects using both image intensities and manual segmentations and (2) fixing the deformations and estimating the atlas parameters. Given a new image, the requirement to jointly perform registration to the atlas and atlas-guided segmentation is a natural consequence of the generative model.

Regularization Tradeoff

As discussed in Section 2.2.1, the choice of the deformation flexibility can influence the quality of the atlas constructed. Indeed, recent work has shown that taking into account the tradeoff between the regularization and dissimilarity measure in registration can significantly improve population analysis [104] and segmentation quality [39, 180].

As a classical example, the traditional choice of image dissimilarity for unimodal images is the weighted Sum of Squared Differences (wSSD) similarity measure:

$$\text{Dissim}(T, I \circ \Gamma) = \sum_i \lambda_i^2 \left(T(x_i) - I(\Gamma(x_i))\right)^2, \quad x_i \in \Omega_T. \tag{2.7}$$

The wSSD similarity measure implicitly assumes an independent Gaussian distribution on the image intensities, where the weights λ_i^2 correspond to the precision (reciprocal of the variance) and T is the mean of the Gaussian distribution. The parameterization of the weights as λ_i^2 ensures non-negative weights. The weights are typically set to a constant value [86, 168]. Alternatively, a spatially-varying variance can be estimated from the intensities of registered images [59]. However, depending on the tradeoff between the wSSD criterion and the regularization term, there is an arbitrary choice of the scale of the variance. With weaker regularization, the training images will be better aligned, resulting in lower variance estimates.

Recent work in probabilistic template construction resolves this problem by either marginalizing the tradeoff under a Bayesian framework [2] or estimating the tradeoff with the Minimum Description Length principle [160]. While these methods are optimal for the assumed generative models, it is unclear whether the estimated parameters are optimal for application-specific tasks. After all, the parameters for optimal image segmentation might be different from those for optimal group analysis. In contrast, Van Leemput [165] proposes a generative model for image segmentation. The estimated parameters are therefore Bayes-optimal for segmentation.

In the case of our joint registration-segmentation generative model in Chapter 3, there is a single smoothness parameter. We show that the direct approach of cross-validating the segmentation accuracy through an exhaustive search over the values of the tradeoff parameter offers an effective approach to selecting the value of the regularization parameter.

Estimation of Multiple Parameters in the Registration Cost Function

When there are multiple "free" parameters in the registration cost function, the exhaustive cross-validation approach becomes infeasible. Consequently, in Chapter 5, we propose a task-optimal registration framework that introduces a second layer of optimization over and above the usual registration step. This second layer of optimization traverses the space of parameters of any differentiable family of registration cost functions, selecting the registration parameter setting that results in better registration as measured by the performance of the specific application in a training data set. We assume that given a set of deformation fields or warped images obtained from registration, there exists a differentiable cost function that accurately measures the task performance; this task performance is used as a proxy of registration accuracy. The training data provides additional information not present in a test image, allowing the task-specific cost function to be evaluated during training. For example, if the task is segmentation, we assume the existence of a training data set with ground truth segmentation and a smooth cost function that evaluates segmentation accuracy. We demonstrate the optimization of thousands of parameters by gradient descent on the task performance of the training data.

Our formulation is related to the use of continuation methods [3] in computing the entire path of solutions of learning problems (e.g., SVM or Lasso) as a function of a single regularization parameter [48, 71, 119]. For example, the cost function in Lasso [155] consists of the tradeoff between the least-squares term and the L_1 regularization term. Least-angles Regression (LARS) allows one to compute the entire set of solutions of Lasso as a function of the tradeoff parameter [48]. Because we deal with multiple parameters, it is impossible for us to compute a solution manifold. Instead, we trace a path within the solution manifold that improves the task-specific cost function. Furthermore, registration is not convex (unlike SVM and Lasso), resulting in several theoretical and practical conundrums that we have to overcome, some of which we leave for future work.

Traditionally, much of the research efforts have been devoted to developing generic registration algorithms. These algorithms are then specialized to particular imaging modalities (e.g., MR, CT, histology, etc.), particular imaging targets (e.g., cardiac, brain, etc.) and particular post-registration analysis (e.g., segmentation, voxel-based morphometry, functional group analysis, etc.). The task-optimal framework provides a principled method for adapting these generic algorithms to specific applications.

In this thesis, we focus mostly on variations of the generic wSSD criterion for measuring image dissimilarity. In particular, we seek to learn the weights of the weighted Sum of Squared Differences (wSSD) family of registration cost functions and to estimate an optimal macroanatomical template for localizing the cytoarchitectural and functional regions based on the cortical folding pattern. Our work can be in principle generalized and applied to the estimation of parameters in similarity measures such as cross-correlation [15, 31, 32] or mutual information [102, 171], and a variety of regularization terms.

We emphasize that the task-optimal template we estimate based on the wSSD dissimilarity measure is different from the mean image that is traditionally estimated when substituting the wSSD criterion (Eq. (2.7)) into the groupwise registration cost function in Eq. (2.6). Optimizing Eq. (2.6) leads to a template image that is optimal at "predicting" the intensities of new image, but may not be optimal for application-specific performance.

Lack of Ground Truth Deformations

By learning the weights of the wSSD, we implicitly learn the tradeoff betweeen the dissimilarity measure and regularization. Furthermore, the tradeoff we learn is spatially varying. Previous work on learning a spatially varying regularization prior suffers from the lack of ground truth (nonlinear) deformations. For example, [39, 68] assume that the deformations obtained from registering a set of training images can be used to estimate a registration regularization to register new images. However, a change in the parameters of the registration cost function used by these methods to register the training images would lead to a different set of training deformations and thus a different prior for registering new images. Furthermore, the methods are inconsistent in the sense that the learned prior applied on the training images will not result in the same training deformations obtained previously.

While there has been efforts in obtaining ground truth human-annotated deformation fields [96], the images considered typically have well-defined correspondences, rather than for example, the brain images of two different subjects. As suggested in the previously presented examples (Fig. 1-2), the concept of "ground truth deformations" may not always be well-defined, since the optimal registration may be a function of the application at hand. In contrast, our approach avoids the need for ground truth deformations by focusing on the application performance, where ground truth (e.g., via segmentation labels) is better defined.

Function Localization

We note that an alternative approach to overcome the imperfect correlation between anatomy and function is to directly use the functional data for establishing across-subject *functional* correspondence [137, 142]. However, these approaches require extra data acquisition (such as fMRI scans) of all future test subjects. In contrast, our method aims to take a training data set containing information about both anatomy and function (or cytoarchitectonics) and generalize the prediction patterns in neuroanatomy for functional localization in future subjects.

2.4 Summary

In this chapter, we provide a review of previous work on registration, atlas-based segmentation and atlas construction. We also discuss our contributions with respect to current literature. In the next chapter, we present our first contribution to the atlas construction literature: a generative model for joint registration and segmentation.

Chapter 3

Generative Model for Joint Registration-Segmentation

In non-rigid registration, the tradeoff between warp regularization and image fidelity is typically determined empirically. In atlas-based segmentation, this leads to a probabilistic atlas of arbitrary sharpness: weak regularization results in well-aligned training images and a sharp atlas; strong regularization yields a "blurry" atlas.

In this chapter, we formulate a generative model for the joint registration and segmentation of images. The atlas construction process arises naturally as estimation of the model parameters. This model allows for computation of unbiased atlases from manually labeled data at various degrees of "sharpness", as well as a procedure for the simultaneous registration and segmentation of a novel brain consistent with the atlas construction.

We study the effects of the tradeoff of atlas sharpness and warp smoothness in the context of cortical surface parcellation. By cross-validating the segmentation performance, we find that the optimal segmentation corresponds to a unique balance of atlas sharpness and warp regularization, yielding statistically significant improvements over the FreeSurfer[1] parcellation algorithm [43, 60].

Furthermore, we conclude that one can simply use a single atlas computed at an optimal sharpness for the registration-segmentation of a new subject with a pre-determined optimal warp constraint. While this might be obvious from our explicit generative model, there is an increasing availability of atlases in public databases, such as MNI305 [51], and registration algorithms are commonly developed separately from the atlas construction process. Our experiments suggest that segmentation accuracy is *tolerant* up to a small mismatch between atlas sharpness and warp smoothness.

A preliminary version of this work was published at the International Conference on

[1] FreeSurfer is a freely available software package [62] that is widely used in the neuroscientific community. It is considered to contain some of most accurate algorithms that have been repeatedly validated. It is also commonly used as a baseline for evaluation of other algorithms.

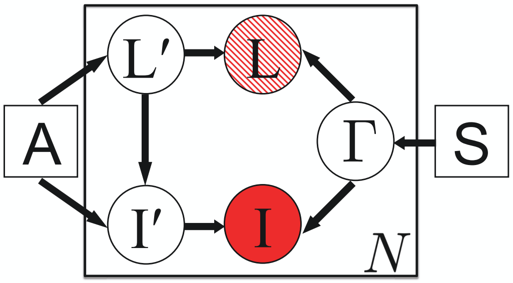


Figure 3-1: Generative model for joint registration and segmentation. A is an atlas used to generate the label map L' in some universal atlas space. The atlas A and label map L' generate image I'. S is the smoothness parameter that generates random warp field Γ. This warp is then applied to the label map L' and image I' to create the label map L and the image I. We assume the label map L is available for the training images, but not for the test image. The image I is observed in both training and test cases.

Medical Image Computing and Computer Assisted Intervention [179]. An expanded version was published in the Journal of Medical Image Analysis [180] and forms the core of this chapter.

In the next section, we introduce the generative model, describe the resulting atlas estimation process, and the registration and segmentation of a novel image. Section 3.2 introduces the cortical surface parcellation problem and describes further modeling assumptions made in the context of this problem. We present experimental results in Section 3.3.

3.1 Theory

Given a training set of N images $I_{1:N} = \{I_1, \cdots, I_N\}$ with label maps $L_{1:N} = \{L_1, \cdots, L_N\}$, joint registration-segmentation aims to infer the registration parameters Γ and segmentation labels L of a new image I. To achieve this goal, we first learn the parameters of the generative model (Figure 3-1) from the training images. These parameters correspond to the atlas A and warp smoothness parameter S. Estimating the parameters S and A involves co-registration of the training images and labels into a common space.

3.1.1 Generative Model for Registration and Segmentation

We consider the generative model of Figure 3-1. L' is a label map in the atlas space generated probabilistically by the atlas A. For example, L' could be the tissue type at each MRI pixel, generated by a tissue probability map. Given the label map, the atlas then generates image

I'. For example, at each pixel, we can generate an MR intensity conditioned on the tissue type and spatial location. Finally, we generate a random warp field Γ controlled by the smoothness parameter S. For example, we can generate a random displacement field, with neighboring displacements encouraged to be "close". For example, S can be the spacing of spline control points or the penalty in a cost function that discourages large or discontinuous deformation fields. The random warp Γ is then applied to the label map L' and image I' to create the observed label map L and observed image I, i.e., $I(\Gamma(x)) = I'(x)$ and $L(\Gamma(x)) = L'(x)$. Thus a location x in the atlas space is mapped to a location $\Gamma(x)$ in the native (or raw) image space. We defer a detailed instantiation of the model to Section 3.2.

During co-registration, a small value of smoothness parameter S leads to less constrained warps, resulting in better alignment of the training images[2]. This results in a sharper atlas. On the other hand, a larger smoothness parameter yields more regularized warps and a blurrier atlas.

3.1.2 Atlas Building: Estimating Parameters of Generative Model

To estimate the parameters of the generative model, we maximize the likelihood of the observed images $I_{1:N}$ and $L_{1:N}$ over the values of the non-random smoothness parameter S and atlas A.

$$\begin{aligned}(S^*, A^*) &= \operatorname*{argmax}_{S,A} \log p(I_{1:N}, L_{1:N}; S, A) &(3.1)\\ &= \operatorname*{argmax}_{S,A} \log \int p(I_{1:N}, L_{1:N}, \Gamma_{1:N}; S, A) d\Gamma_{1:N}. &(3.2)\end{aligned}$$

Here $p(a; b)$ indicates the probability of random variable a parameterized by a non-random parameter b while $p(a|b)$ indicates the probability of random variable a conditioned on the observation of a random variable b.

In this case, we need to marginalize over the registration warps $\Gamma_{1:N}$, which is difficult. Previously demonstrated methods for atlas construction use various approximations to evaluate the integral. In this paper, we adopt the standard mode approximation that replaces the integral with the value of the integrand estimated at the Maximum-A-Posteriori (MAP) estimate of deformation. For large number of training images, such a mode approximation is justified as the posterior distribution is presumably peaky. In contrast, [129] uses a sampling method while [164] uses the Laplace approximation, which essentially models the distribution to be a Gaussian centered at the MAP estimate. It is unclear that these more computationally complex methods, while theoretically more sound, lead to (practi-

[2] By a "better alignment of images", we mean that the warped images look more similar, i.e., the similarity measure is improved. However, an improved similarity measure does not necessarily imply deformations with better label alignment. In fact, we show in the paper that the best segmentation occurs when warps are not overly flexible.

cally) better approximation. A possible avenue of future research is the investigation of variational methods to approximate the integral, since variational methods are known to be computationally efficient, although not as accurate as sampling methods. Based on the mode approximation, we seek

$$\begin{aligned}(S^*, A^*, \Gamma^*_{1:N}) &= \operatorname*{argmax}_{S,A,\Gamma_{1:N}} \log p(I_{1:N}, L_{1:N}, \Gamma_{1:N}; S, A) &(3.3)\\ &= \operatorname*{argmax}_{S,A,\Gamma_{1:N}} \sum_{n=1}^{N} \log p(I_n, L_n, \Gamma_n; S, A) &(3.4)\\ &= \operatorname*{argmax}_{S,A,\Gamma_{1:N}} \sum_{n=1}^{N} \left[\log p(\Gamma_n; S) + \log p(I_n, L_n|\Gamma_n; A)\right]. &(3.5)\end{aligned}$$

The second equality comes from the fact that the training images are independent of each other given the atlas A and smoothness parameter S. The last equality is implied by the independence relations specified by the graphical model in Figure 3-1.

Optimizing the above expression yields the atlas A^* and smoothness parameter S^*. As mentioned before, a smaller value of the smoothness parameter S results in a sharper atlas. Since we are interested in how atlas sharpness affects segmentation accuracy, instead of estimating one single optimal S^*, we construct a series of atlases corresponding to different values of the regularization parameter S. In particular, we discretize S into a finite set $\{S_k\} = \{S_1 > S_2 > \cdots > S_K\}$. For each value S_k, we seek the optimal atlas and set of deformations:

$$(A^*, \Gamma^*_{1:N}) = \operatorname*{argmax}_{A,\Gamma_{1:N}} \sum_{n=1}^{N} \left[\log p(\Gamma_n; S_k) + \log p(I_n, L_n|\Gamma_n; A)\right]. \qquad (3.6)$$

We shall refer to the atlas computed using a particular S_k as $A_{\alpha=S_k}$. We use alternating optimization to maximize Eq. (3.6). In each step, we fix the set of registration warps $\Gamma_{1:N}$ and estimate the atlas $A_{\alpha=S_k}$:

$$A^*_{S_k} = \operatorname*{argmax}_{A_{S_k}} \sum_{n=1}^{N} \log p(I_n, L_n|\Gamma_n; A_{S_k}). \qquad (3.7)$$

We then fix the atlas $A_{\alpha=S_k}$, and optimize the registration warps $\Gamma_{1:N}$ by optimizing each warp independently of others:

$$\Gamma^*_n = \operatorname*{argmax}_{\Gamma_n} \log p(\Gamma_n; S_k) + \log p(I_n, L_n|\Gamma_n; A_{S_k}), \quad \forall n = 1, \ldots, N. \qquad (3.8)$$

This process is repeated until convergence. Convergence is guaranteed since this is essentially a coordinate-ascent procedure operating on a bounded function.

We can think of Eq. (3.8) as the atlas co-registration objective function by treating $\log p(I_n, L_n | \Gamma_n; A_{S_k})$ as the data fidelity function and $\log p(\Gamma_n; S_k)$ as the regularization term. We effectively register each image independently to the atlas $A_{\alpha=S_k}$. This iterated process of updating the atlas is similar to [16, 86], except we include training labels in the co-registration. We will show a concrete instantiation of this formulation in Section 3.2.

In practice, we first create atlas A_∞ based on a simple rigid-body registration on the sphere and use it to initialize the atlas A_{S_1}, where S_1 is large enough such that the resultant warp is almost rigid. We then use atlas A_{S_1} to initialize the atlas A_{S_2} where $S_1 > S_2$, and so on. The result is a set of atlases $\{A_\alpha\} = \{A_{S_1} \cdots A_{S_K}\}$. With enough samples, the finite set $\{S_k, A_{\alpha=S_k}\}$ accurately represents the underlying continuous space of atlases at different levels of sharpness.

3.1.3 Registration and Segmentation of a New Image

Given an atlas A and smoothness parameter S, the registration and segmentation of a new image I can be computed using a MAP estimate. This involves finding the mode of the posterior distribution of the new image labels L and registration Γ given the observed image I and atlas A and smoothness parameter S:

$$(L^*, \Gamma^*) = \underset{L,\Gamma}{\operatorname{argmax}} \log p(L, \Gamma | I; S, A) \tag{3.9}$$

$$= \underset{L,\Gamma}{\operatorname{argmax}} \log p(I, L, \Gamma; S, A) \tag{3.10}$$

$$= \underset{L,\Gamma}{\operatorname{argmax}} \log p(\Gamma; S) + \log p(I, L | \Gamma; A). \tag{3.11}$$

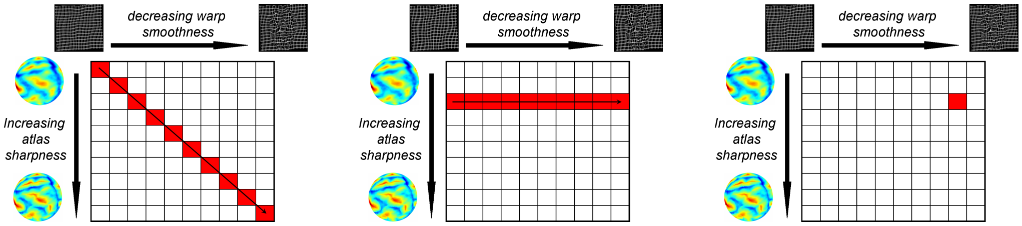
The second equality follows from the definition of the conditional probability. The last equality follows from the independence relations specified by the graphical model in Figure 3-1. In prior work, this MAP approach is favored by some [173, 174, 175], while others suggest estimating the MAP solution for the registration warp alone [12, 123]:

$$\Gamma^* = \underset{\Gamma}{\operatorname{argmax}}\, p(I, \Gamma; S, A) \tag{3.12}$$

$$= \underset{\Gamma}{\operatorname{argmax}} \log \sum_L p(L, I, \Gamma; S, A), \tag{3.13}$$

and recovering the segmentation labels as a final step after recovering the optimal registration Γ^*. Prior work in joint registration and segmentation did not consider atlas construction in the same framework [12, 123, 173, 174, 175]. Furthermore, S is usually set by an expert rather than estimated from the data.

We previously reported results based on the latter two-step approach [179]. In this version, we use the former MAP framework since it requires fewer assumptions for practical


decreasing warp smoothness
Increasing atlas sharpness
decreasing warp smoothness
Increasing atlas sharpness
decreasing warp smoothness
Increasing atlas sharpness

(a) Multiple Atlases, Multiple Smoothness (MAMS)

(b) Single Atlas, Multiple Smoothness (SAMS)

(c) Single Atlas, Single Smoothness (SASS)

Figure 3-2: Strategies for exploring space of atlas sharpness and warp smoothness of a new image.

optimization. As we show in Section 3.2, optimizing Eq. (3.11) using a soft-MAP coordinate ascent approach using the Mean Field approximation [79, 87] results in the same update rule used by our previously demonstrated method [179].

To optimize Eq. (3.11), we use a coordinate ascent scheme. In step t, we fix the registration parameters $\Gamma^{(t)}$ and estimate the labels $L^{(t+1)}$:

$$L^{(t+1)} = \underset{L}{\operatorname{argmax}} \log p(\Gamma^{(t)}; S) + \log p(I, L|\Gamma^{(t)}; A) \tag{3.14}$$

$$= \underset{L}{\operatorname{argmax}} \log p(I, L|\Gamma^{(t)}; A) \tag{3.15}$$

$$= \underset{L}{\operatorname{argmax}} \log p(L|I, \Gamma^{(t)}; A). \tag{3.16}$$

Eq. (3.16) optimizes the log posterior probability of the label map L given the image I, atlas A and current estimate of the registration parameters $\Gamma^{(t)}$. Next, we fix the label estimate $L^{(t+1)}$ and re-estimate the registration parameters $\Gamma^{(t+1)}$:

$$\Gamma^{(t+1)} = \underset{\Gamma}{\operatorname{argmax}} \log p(\Gamma; S) + \log p(I, L^{(t+1)}|\Gamma; A). \tag{3.17}$$

We can think of Eq. (3.17) as the new image registration objective function by treating $\log p(I, L^{(t+1)}|\Gamma; A)$ as the data fidelity term and $\log p(\Gamma; S)$ as the regularization term.

To maintain generality, we allow the use of a smoothness parameter S_k for the registration and segmentation of a new subject with an atlas A_α where S_k may not be equal to α. In other words, we can, for instance, compute an atlas using affine transformations, while using a flexible deformation model for the registration of a new subject. Strictly speaking, S_k should theoretically be equal to α in Eq. (3.11) from the perspective of probabilistic inference. Here, we examine whether using different warp smoothness and atlas sharpness is necessarily detrimental to segmentation in practice.

More specifically, we investigate three strategies for exploring the space of atlas sharpness and warp smoothness to register and segment a new image as illustrated in Figure 3-2.

1. Multiple Atlas, Multiple Smoothness (MAMS): A multiscale approach where we optimize Eq. (3.16) and Eq. (3.17) w.r.t. Γ and L with the blurry atlas A_{S_1} and warp regularization S_1. The resulting alignment is used to initialize the registration with a sharper atlas A_{S_2} and a correspondingly flexible warp regularization S_2, and so on (Figure 3-2a).

2. Single Atlas, Multiple Smoothness (SAMS): We use a fixed atlas of a fixed sharpness scale A_{S_k} to compute Γ and L according to Eq. (3.16) and Eq. (3.17) using a progressively decreasing warp smoothness S (Figure 3-2b).

3. Single Atlas, Single Smoothness (SASS): We optimize Eq. (3.16) and Eq. (3.17) w.r.t. Γ and L with a fixed atlas A_{S_k} and warp regularization S_m, where S_k might not be equal to S_m (Figure 3-2c).

So far, the derivations have been general without any assumption about the atlases A_α, the prior $p(\Gamma; S_k)$ or the image-segmentation fidelity $p(I, L|\Gamma; A_\alpha)$. In the next section, we instantiate this approach for a concrete example of cortical surface registration and parcellation.

3.2 Cortical Surface Parcellation

We now demonstrate the framework developed in the previous section for the joint registration and parcellation of surface models of the cerebral cortex. As discussed in Chapter 1.2.1, each half of the cortex is modeled as triangular meshes with a spherical coordinate system. The aim is to register an unlabeled cortical surface to a set of manually labeled surfaces and to classify each vertex of the triangular mesh into sulcal and gyral units.

To instantiate the generative model for this problem, we need to define the prior on the registration parameters $p(\Gamma; S)$ and the model for label and image generation $p(I, L|\Gamma; A_\alpha)$. In general, our model decisions were inspired by previous work on cortical surface parcellation [43, 58, 55, 60].

3.2.1 Generative Model for Registration and Segmentation

We model the warp regularization with an MRF parameterized by S:

$$p(\Gamma; S) \triangleq \frac{F(\Gamma)}{Z_1(S)} \exp\Big\{ -S\Big[\sum_i \sum_{j \in \mathcal{N}_i} \Big(\frac{d_{ij}^\Gamma - d_{ij}^0}{d_{ij}^0}\Big)^2\Big]\Big\} \tag{3.18}$$

where d_{ij}^Γ is the distance between vertices i and j under registration Γ, d_{ij}^0 is the original distance, $\mathcal{N}_i$ is a neighborhood of vertex i and $Z_1(S)$ is the partition function. Our regularization penalizes percentage metric distortion weighted by a scalar S that reflects the amount of smoothness (rigidity) of the final warp. We choose a percentage metric distortion instead of an absolute metric distortion [58] to ensure that the tradeoff between the regularization and the data fidelity term is invariant to scale in our multi-resolution framework. This is especially important in our work since we explore a large range of warp smoothness parameter S. $F(\cdot)$ ensures invertibility. It is zero if any triangle is folded by warp Γ and one otherwise. We represent the warp Γ as a displacement field on the sphere. Therefore, a term like $F(\Gamma)$ is necessary. One could replace $F(\Gamma)$ with a more sophisticated invertibility prior (e.g., [10, 114]) or restrict the space of feasible warps to be the space of diffeomorphisms [86, 67].

We first decompose the label and image prior

$$p(I, L|\Gamma; A_\alpha) = p(L|\Gamma; A_\alpha)p(I|L, \Gamma; A_\alpha). \tag{3.19}$$

and impose an MRF prior on the parcellation labels

$$p(L|\Gamma; A_\alpha) \triangleq \frac{1}{Z_2(A_\alpha)} \exp\Big\{ \sum_i U_i(A_\alpha)L(\Gamma(x_i)) + \sum_i \sum_{j\in\mathcal{N}_i} L^T(\Gamma(x_i))V_{ij}(A_\alpha)L(\Gamma(x_j)) \Big\}. \tag{3.20}$$

Here, we use the vectorized MRF notation of [87]. Assuming the total number of labels is M, $L(\Gamma(x_i))$ is a column vector of size $M \times 1$ that sums to 1. Each component of $L(\Gamma(x_i))$ is an indicator variable. In particular, suppose the image has label m at location $\Gamma(x_i)$, then $L(\Gamma(x_i))$ is zero for all entries, except for the m-th entry, which is equal to 1. It is a common practice to relax the constraint so that L still sums to 1 but the entries might take on fractional values to indicate uncertainty in the segmentation results [87]. $U_i(A_\alpha)$ is a $1 \times M$ local potential vector that captures the frequency of label $L(\Gamma(x_i))$ at vertex i of the atlas mesh. The $M \times M$ compatibility matrix $V_{ij}(A_\alpha)$ reflects the likelihood of two labels at vertices i and j being neighbors. $Z_2(A_\alpha)$ is the partition function dependent on the atlas A_α, ensuring the given MRF is a valid probability distribution.

We further assume that the noise added to the mean image intensity at each vertex location is independent, given the label at that location.

$$p(I|L, \Gamma; A_\alpha) \triangleq \prod_i W_i(I(\Gamma(x_i)); A_\alpha)L(\Gamma(x_i)), \tag{3.21}$$

where $W_i(I(\Gamma(x_i)); A_\alpha)$ is a $1 \times M$ observation potential vector defined at each atlas vertex i. The m-th entry corresponds to the likelihood of observing a particular image intensity or vectors of intensity (in this case, the local surface geometries) at location $\Gamma(x_i)$ given a particular label m. We assume W_i follows a Gaussian distribution, e.g., given that the post-central gyrus is at location $\Gamma(x_i)$ of the image, we expect the local curvature and/or sulcal depth $I(\Gamma(x_i))$ to follow a Gaussian distribution whose parameters are estimated from the training images.

The collection of model components $\{U_i, V_{ij}, W_i\}$ define the atlas A_α. Eq. (3.20) defines an isotropic prior on the labels, which is a simpler model than that used by modern approaches. The FreeSurfer parcellation algorithm uses a spatially varying and anisotropic MRF model [60], whose parameters change dynamically with the geometries of the subject being segmented. An anisotropic MRF improves the parcellation accuracy because the boundaries of certain gyral regions are predicted robustly by the variation in curvature. For example, the boundary between the pre-central and post-central gyrus is the central sulcus. Along the boundary, there is high curvature, while across the boundary, the curvature drops off sharply.

We made the explicit choice of warping an image (interpolating an image) to the atlas space. The alternative of warping the atlas (interpolating the atlas) to image space would require us to interpolate the MRF, which is non-trivial. Interpolating U would result in the partition function being a function of the warp, which is exponentially hard to compute. In addition, if we were to use the dynamic model of FreeSurfer, since we have made the choice of warping the subject, this would mean that the model parameters and hence the partition function of the MRF would change during the registration step. FreeSurfer does not have this problem because it does not perform joint registration and segmentation. Therefore, we assume V to be spatially stationary and isotropic and drop the subscripts i, j. However, we note that recent work has suggested that warping an atlas is fundamentally more correct compared with warping the image [2].

3.2.2 Atlas Building: Estimating Parameters of Generative Model

Substituting Eq. (3.18), Eq. (3.20) and Eq. (3.21) into the atlas co-registration objective function in Eq. (3.8), we obtain:

$$\begin{aligned}
\Gamma_n^* &= \underset{\Gamma_n}{\operatorname{argmax}} \log p(\Gamma_n; S_k) + \log p(I_n, L_n | \Gamma_n; A_{S_k}) \\
&= \underset{\Gamma_n}{\operatorname{argmax}} \log F(\Gamma_n) - S_k \sum_i \sum_{j \in \mathcal{N}_i} \left(\frac{d_{ij}^{\Gamma_n} - d_{ij}^0}{d_{ij}^0} \right)^2 + \\
&\quad + \sum_i U_i(A_\alpha) L(\Gamma_n(x_i)) + \sum_i \sum_{j \in \mathcal{N}_i} L^T(\Gamma_n(x_i)) V(A_\alpha) L(\Gamma_n(x_j)) + \\
&\quad + \sum_i W_i(I(\Gamma_n(x_i)); A_\alpha) L(\Gamma_n(x_i)) + \text{const.} \qquad (3.22)
\end{aligned}$$

where the first term prevents folding triangles, the second term penalizes metric distortion, the third and fourth terms are the Markov prior on the labels and the last term is the likelihood of the surface geometries given the segmentation labels of the n-th training image.

We can then fix Γ_n^* and estimate the atlas parameters $A_\alpha = \{U_i, V, W_i\}$ using Eq. (3.7). In practice, we use the naive approach of frequency counts [60] to estimate the clique potentials U, V and the maximum likelihood estimation of the Gaussian likelihood terms W. Appendix A.2 provides the implementation details.

3.2.3 Registration and Segmentation of a New Image

Similarly, we substitute Eq. (3.18), Eq. (3.20) and Eq. (3.21) into the update rules for the new subject segmentation in Eq. (3.16) and registration in Eq. (3.17).

Warping the subject to the atlas, optimization in Eq. (3.16) with a fixed $\Gamma^{(t)}$ involves estimating the segmentation labels at positions $\Gamma^{(t)}(\{x_i\})$ of the subject, where $\{x_i\}$ are vertices of the atlas mesh. We will denote this segmentation estimate by $\widehat{L}^{(t+1)}$. Eq. (3.16) becomes

$$\widehat{L}^{(t+1)} = \underset{\widehat{L}}{\operatorname{argmax}} \log p(\widehat{L} | I(\Gamma^{(t)}(\{x_i\})); A_\alpha). \tag{3.23}$$

Even with fixed $\Gamma^{(t)}$, solving the MAP segmentation Eq. (3.23) is NP-hard. We adopt the Mean Field approximation [79, 87]. We then use the complete approximate distribution provided by the Mean Field solver in optimizing Eq. (3.17). This approximation effectively creates a soft segmentation estimate $\widehat{L}_i^{(t+1)}$ at each location $\Gamma^{(t)}(x_i)$ of the new subject. $\widehat{L}_i^{(t+1)}$ is a column vector of size M x 1. The m-th component of $\widehat{L}_i^{(t+1)}$ is the probability of finding label m at location $\Gamma^{(t)}(x_i)$ of the new subject. To estimate the label $\widehat{L}(x)$ at an arbitrary location x in the subject space, we interpolate from $\widehat{L}^{(t+1)}$ defined at $\Gamma^{(t)}(\{x_i\})$ onto location x.

This optimization procedure leads to fewer local minima since it avoids commitment to the initial estimates obtained through hard thresholding that might be very far from a good solution if the new image is originally poorly aligned to the atlas. Appendix A.1 provides an outline for computing $\widehat{L}^{(t+1)}$ via the Mean Field approximation. Warping the subject to the atlas, Eq. (3.17) becomes:

$$\begin{aligned}\Gamma^{(t+1)} = \underset{\Gamma}{\operatorname{argmax}} \log F(\Gamma) - S_k \sum_i \sum_{j \in \mathcal{N}_i} \left(\frac{d_{ij}^{\Gamma} - d_{ij}^0}{d_{ij}^0} \right)^2 + \\ + \sum_i U_i(A_\alpha) \widehat{L}(\Gamma(x_i)) + \sum_i \sum_{j \in \mathcal{N}_i} \widehat{L}^T(\Gamma(x_i)) V(A_\alpha) \widehat{L}(\Gamma(x_j)) + \\ + \sum_i W_i(I(\Gamma(x_i)); A_\alpha) \widehat{L}(\Gamma(x_i)) + \text{const.}\end{aligned} \tag{3.24}$$

Further implementation details can be found in Appendix A.2.

3.3 Experiments and Discussion

In this experiment, we consider the automatic parcellation of the 39 pairs of hemispheres described in Chapter 1.2.2. For each hemisphere, the 39 cortical surfaces are first rigidly aligned on the sphere, which corresponds to rotation only. For an illustration of the manual parcellation of the cortex and a list of parcellation units, we refer the readers to Fig. 1-4 and Table 1.1 respectively.

We perform cross-validation by leaving out subjects 1 through 10 in the atlas construction, followed by the joint registration-segmentation of left-out subjects. We repeat with subjects 11 through 20, 21 through 30 and finally 31 through 39. We select S to be the set $\{100, 50, 25, 12.5, 8, 5, 2.5, 1, 0.5, 0.25, 0.1, 0.05, 0.01\}$. We find that in practice, $S = 100$ corresponds to allowing minimal metric distortion and $S = 0.01$ corresponds to allowing almost any distortion. The intervals in the set S are chosen so that each decrease in the value of S roughly corresponds to an average of $1mm$ increased displacement in registration.

Since we treat the subject mesh as the moving image, both registration and parcellation are performed on the fixed atlas mesh. The segmentation is interpolated from the atlas mesh onto the subject mesh to obtain the final segmentation. We compute segmentation quality by comparing this final segmentation with the "ground truth" manual parcellation.

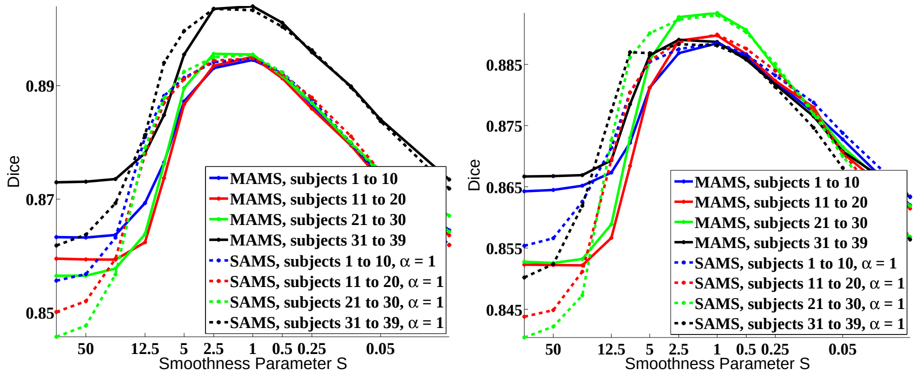
To speed up the algorithm, we construct the atlas on a sub-divided icosahedron mesh with about 40k vertices. Typically, each subject mesh has more than 100k vertices. The segmentation labels inferred on the low resolution atlas mesh are therefore computed on a coarser grid than that of the manual parcellation. Yet, as we discuss in the remainder of this section, the proposed implementation on average outperforms the FreeSurfer parcellation algorithm [60].

Despite working with a lower resolution icosahedron mesh, registration at each smoothness scale still takes between 20 minutes to 1.5 hours per subject per atlas. Registration with weakly constrained warps ($S \leq 0.1$) requires more time because of the need to preserve the invertibility of the warps. The entire set of experiments took approximately 3 weeks to complete on a computing cluster, using up to 80 nodes in parallel.

3.3.1 Exploration of Smoothness S and Atlas A_α

In this section, we discuss experimental results for the exploration of the smoothness parameter S and atlases A_α. We measure the segmentation accuracy using the Dice coefficient, defined as the ratio of cortical surface area with correct labels to the total surface area, averaged over the test set.

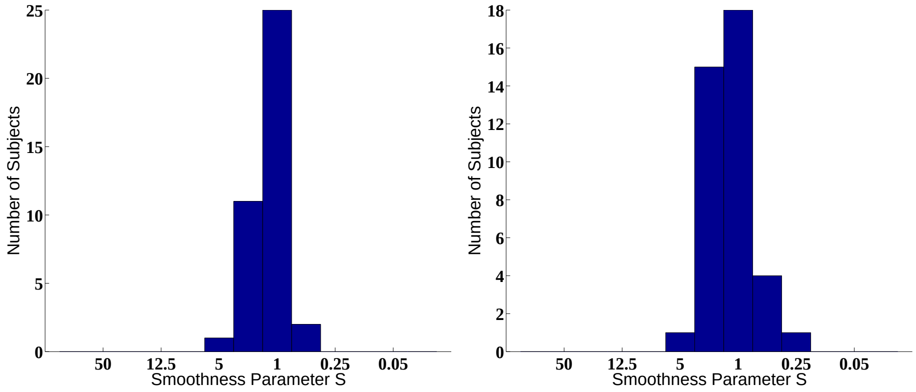
Figure 3-3 shows the segmentation accuracy for SAMS ($A_\alpha = A_1$) and MAMS as we vary S. The average Dice peaks at approximately $S = 1$ for all cross-validation trials,



(a) Left Hemisphere

(b) Right Hemisphere

Figure 3-3: Parcellation accuracy as a function of warp smoothness. S is plotted on a log scale.



(a) Left Hemisphere

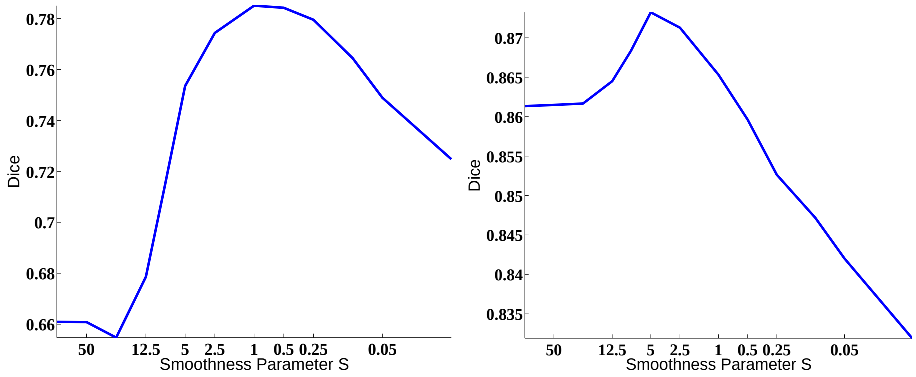
(b) Right Hemisphere

Figure 3-4: Histogram of optimal warp smoothness S across subjects (MAMS).

although individual variation exists, as shown in Figure 3-4[3]. For a particular value of S, outliers warp more because the tradeoff between the data-fidelity and regularization is skewed towards the former. However, it is surprising to find that the optimal S is mostly constant across subjects. It also appears that peaks of the segmentation accuracy plots are relatively broad, implying that good parcellation results can be obtained for a range of S between 1 and 2.5.

For large S (highly constrained warps), MAMS consistently outperforms SAMS. Because the surfaces are initially misaligned, using a sharp atlas (in the case of SAMS, atlas A_1) results in poor segmentation accuracy due to a mismatch between the image features and the atlas statistics. As we decrease the smoothness S, SAMS allows for more flexible warps towards the population average than MAMS since it uses a sharper atlas. The similarity measure is therefore given higher weight to overcome the regularization. This results in

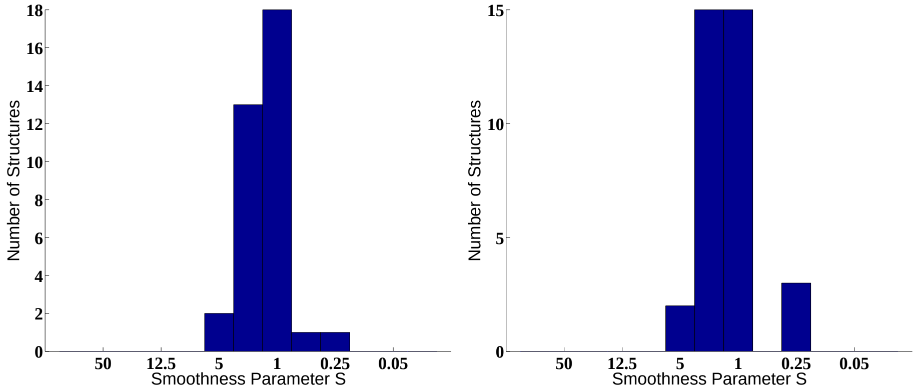
[3] The optimal value of $S = 1$ is coincidental in the sense that it depends on the unit chosen for metric distortion.



(a) Right Inferior Temporal Gyrus

(b) Right Temporal Pole

Figure 3-5: (a) Typical plot of Dice against smoothness S. (b) A noisy plot of Dice against smoothness S.



(a) Left Hemisphere

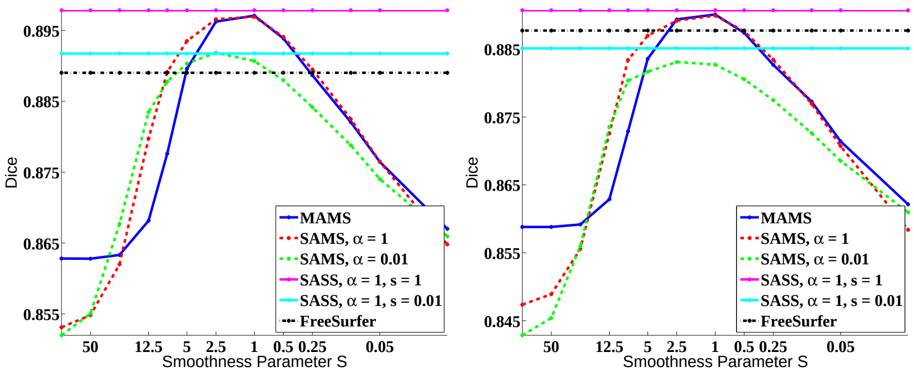
(b) Right Hemisphere

Figure 3-6: Histogram of optimal smoothness S across structures (MAMS).

better segmentation accuracy than MAMS. Eventually, SAMS and MAMS reach similar maximal values at the same optimal smoothness S. Beyond the optimal S, however, both MAMS and SAMS exhibit degraded performance. This is probably due to overfitting and local optima created by more flexible warps.

We also examine the Dice measure of each parcellation structure as a function of the warp smoothness S. In general, the curves are noisier but follow those of Figure 3-3. Figure 3-5(a) shows a typical curve that peaks at $S = 1$, while Figure 3-5(b) shows a curve that peaks at $S = 5$. However, in general, for most structures, the optimal smoothness S occurs at approximately $S = 1$ (Figure 3-6), which is not surprising since for most subjects, the optimal overall Dice (computed over the entire surface) also occurs at $S = 1$ (Figure 3-4).

We now explore the effects of both warp smoothness S and atlas sharpness α on parcellation accuracy. Figure 3-7 shows a plot of Dice averaged over all 39 subjects. The performances of MAMS, SAMS and SASS at $(S = 1, \alpha = 1)$ are indistinguishable. As an example, we see that for both hemispheres, SAMS with $\alpha = 0.01$ (green line) starts off well,



(a) Left Hemisphere

(b) Right Hemisphere

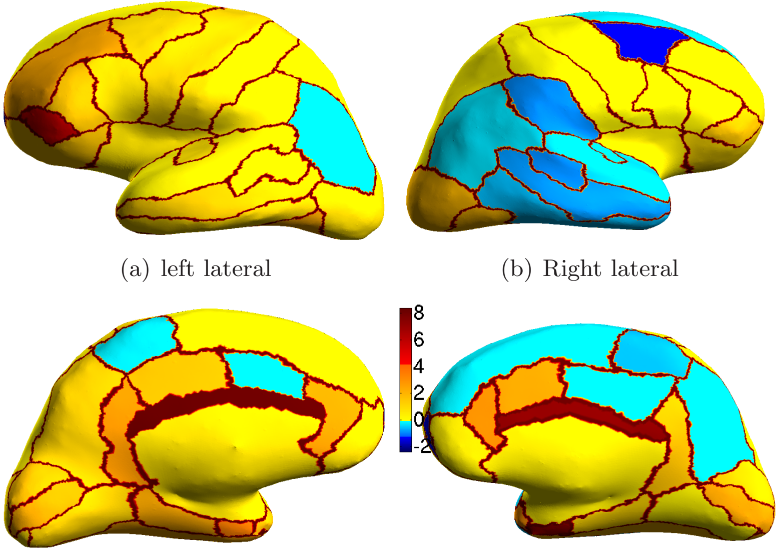
Figure 3-7: Overall Dice versus smoothness. S is plotted on a log scale.

but eventually overfits with a worse peak at $S = 1$ ($p < 10^{-5}$ for one-sided paired-sampled t-test, statistically significant even when corrected for multiple comparisons). Similarly for SASS, the best values of α and S are both equal to 1. We also show SASS with $\alpha = 0.01$ and $S = 1$ in Figure 3-7. In general, there is no statistical difference between MAMS, SAMS or SASS at their optima: $\alpha = 1$, $S = 1$.

Originally, MAMS and SAMS were introduced to reduce local optima in methods, such as SASS. It is therefore surprising that the performance of all three methods is comparable. While using the correct smoothness and atlas sharpness is important, our "annealing" process of gradual reduction of smoothness (MAMS and SAMS) does not seem to increase the extent of the basins of attraction in the context of cortical parcellation. One possible reason is that on the cortical surfaces, two adjacent sulci might appear quite similar locally. Smoothing these features might not necessarily remove the local minima induced by such similarity. Incorporating multiscale features [80, 112, 190, 189] with multiple smoothness offers a promising approach for avoiding such local optimal issues on the cortex.

The fact that the optimal smoothness parameter S^* corresponds to the optimal atlas sharpness parameter α^* is not surprising. According to the graphical model in Figure 3-1 and as mentioned in the derivations in Section 3.1.2 and Section 3.1.3, theoretically, we do expect $S^* = \alpha^*$. Learning this optimal S^* in the atlas construction process is a future avenue of investigation.

Alternatively, we can also determine the best S and A_α for a new image registration by optimizing the objective function in Eq. (3.11). Unfortunately, there are technical difficulties in doing this. First, we notice that the objective function in Eq. (3.11) increases with decreasing S. This model contains no Occam's razor regularization terms that penalize overfitting due to flexible warps. This is mainly because Eq. (3.11) omits certain difficult-to-compute normalization terms, such as the partition function that depends on U, V and W and thus dependent on S and α. These terms are ignored for fixed values of S and α.



(a) left lateral (b) Right lateral

(c) Left medial (d) Right medial

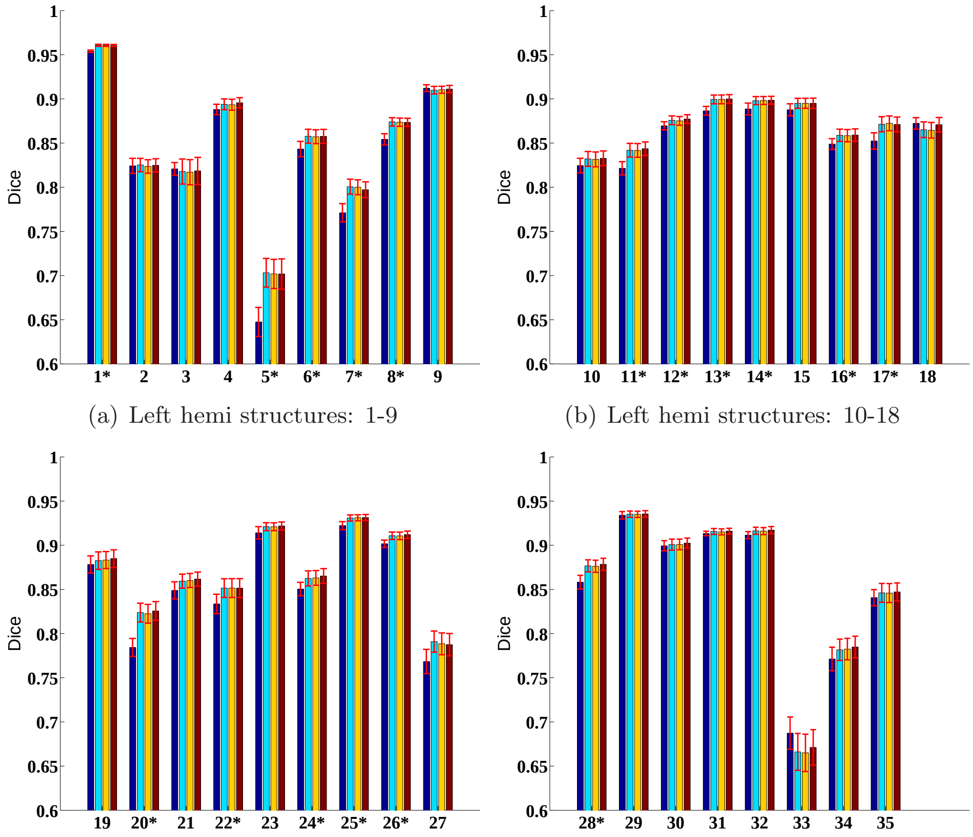
Figure 3-8: Percentage improvement of SASS over FreeSurfer. The boundaries between parcellation regions are set to reddish-brown so that the different regions are more visible.

We can use various approximation strategies to compute the normalization terms. But it is not clear whether these approximations yield sufficient accuracy to determine the optimal values for S and α. On the other hand, empirically we find that $S = \alpha = 1$ consistently yields the optimal (or very close to the optimal) segmentation performance. This suggests that one can probe the training data using a MAMS-type strategy to determine the optimal values of warp smoothness and atlas sharpness, and then use the SASS strategy for future registration and segmentation of new subjects. Furthermore, our experiments suggest that segmentation accuracy is tolerant up to a small mismatch between atlas sharpness and warp smoothness.

3.3.2 Comparison with FreeSurfer

We now compare the performance of our algorithm with the FreeSurfer parcellation algorithm, described in [60] and extensively validated in [43].

It is unclear which algorithm has a theoretical advantage. The FreeSurfer algorithm is essentially a "Single Atlas, Single Smoothness" (SASS) method that uses a sequential registration-segmentation approach and a more complicated anisotropic MRF model that has been specifically designed and fine-tuned for the surface parcellation application. Our model lacks the anisotropic MRF and introducing it would further improve its performance. On the other hand, FreeSurfer uses Iterated Conditional Modes [23] to solve the MRF, while we use the more powerful Mean Field approximation [79, 87]. FreeSurfer also treats the subject mesh as a fixed image and the parcellation is performed directly on the subject



(a) Left hemi structures: 1-9

(b) Left hemi structures: 10-18

(c) Left hemi structures: 19-27

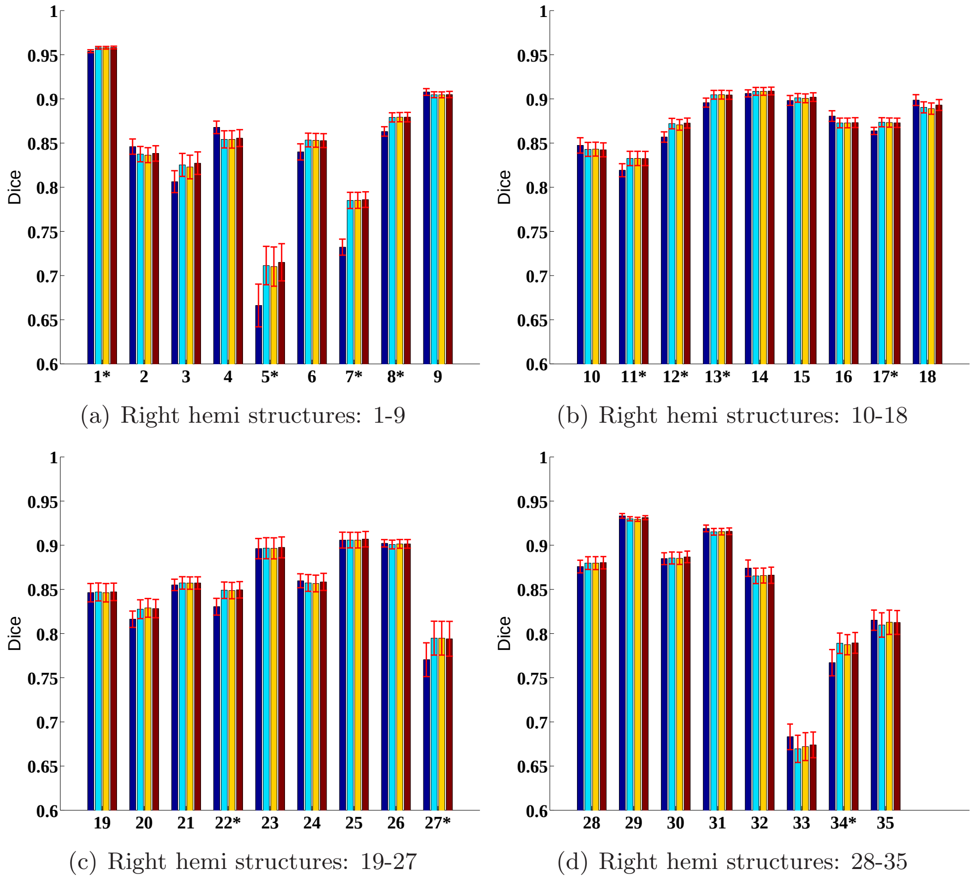
(d) Left hemi structures: 28-35

Figure 3-9: Structure-specific parcellation accuracy for the left hemisphere. First column (dark blue) corresponds to FreeSurfer. Second (light blue), third (yellow) and fourth (brown) columns correspond to MAMS, SAMS and SASS respectively. $(S = 1, \alpha = 1)$. * indicates structures where SASS shows statistically significant improvement over FreeSurfer. There is no structure that becomes worse.

mesh. Therefore, unlike our approach, no interpolation is necessary to obtain the final segmentation.

As shown in Figure 3-7, the optimal performances of MAMS, SAMS and SASS are statistically significantly better (even when corrected for multiple comparisons) than the FreeSurfer, with p-value 1 x 10^{-8} (SASS) for the left hemisphere and 8×10^{-4} (SASS) for the right hemisphere.

Because Dice computed over the entire surface can be deceiving by suppressing small structures, we show in Figure 3-8 the percentage improvement of SASS over FreeSurfer on inflated cortical surfaces. Qualitatively, we see that SASS performs better than FreeSurfer since there appears more orange-red regions than blue regions. The fact that the colorbar has significantly higher positive values than negative values indicates that there are parcellation regions with significantly greater improvements compared with regions that suffer significant penalties.



(a) Right hemi structures: 1-9

(b) Right hemi structures: 10-18

(c) Right hemi structures: 19-27

(d) Right hemi structures: 28-35

Figure 3-10: Structure-specific parcellation accuracy for the right hemisphere. First column (dark blue) corresponds to FreeSurfer. Second (light blue), third (yellow) and fourth (brown) columns correspond to MAMS, SAMS and SASS respectively. $(S = 1, \alpha = 1)$. * indicates structures where SASS shows statistically significant improvement over FreeSurfer. There is no structure that becomes worse.

More quantitatively, Figure 3-9 and 3-10 display the average Dice per structure for FreeSurfer, MAMS, SAMS and SASS at $(S = 1, \alpha = 1)$ for the left and right hemispheres respectively. Standard errors of the mean are displayed as red bars. The numbering of the structures correspond to Table 1.1. The structures with the worst Dice are the frontal pole, corpus callosum and entorhinal cortex. These structures are small and relatively poorly defined by the underlying cortical geometry. For example, the entorhinal cortex is partially defined by the rhinal sulcus, a tiny sulcus that is only visible on the pial surface. On the other hand, the corpus callosum is mapped from the white matter volume onto the cortical surface. Its boundary is thus defined by the surrounding structures, rather than by the cortical geometry.

For each structure, we perform a one-sided paired-sampled t-test between SASS and FreeSurfer, where each subject is considered a sample. We use the False Discovery Rate (FDR) [22] to correct for multiple comparisons. In the left hemisphere, SASS achieves statistically significant improvement over FreeSurfer for 17 structures (FDR < 0.05), while the remaining structures yield no statistical difference. In the right hemisphere, SASS achieves improvement for 11 structures (FDR < 0.05), while the remaining structures yield no statistical difference. The p-values for the left and right hemispheres are pooled together for the False Discovery Rate analysis.

3.4 Summary

In this chapter, we proposed a generative model for the joint registration and segmentation of images. The atlas construction process is formulated as estimation of the graphical model parameters. The framework incorporates consistent atlas construction, multiple atlases of varying sharpness and MRF priors on both registration warps and segmentation labels. We show that atlas sharpness and warp regularization are important factors in segmentation and that the optimal smoothness parameters are stable across subjects in the context of cortical parcellation. The experimental results imply that the optimal atlas sharpness and warp smoothness can be determined by cross-validation. Furthermore, segmentation accuracy is tolerant up to a small mismatch between atlas sharpness and warp smoothness. With the proper choice of atlas sharpness and warp regularization, even with a less complex MRF model, the joint registration-segmentation framework achieves better segmentation accuracy than the state-of-the-art FreeSurfer algorithm [43, 60].

A serious drawback of the algorithm presented in this chapter is the registration-segmentation runtime. There are two reasons for this: (1) coupling of neighboring voxels or vertices introduced by the registration regularization and (2) the desire to maintain deformation invertibility. In the next chapter, we introduce the Spherical Demons algorithm that significantly reduces the time needed to register closed 2D surfaces. Furthermore, while we have shown that cross-validation is an effective way for estimating an optimal tradeoff, it is simply not feasible with multiple parameters. We will address this problem in Chapter 5.

Chapter 4

Spherical Demons: Fast Surface Registration

In this chapter, we present the Spherical Demons algorithm for registering two spherical images. By exploiting spherical vector spline interpolation theory, we show that a large class of regularizors for the modified Demons objective function can be efficiently approximated on the sphere using iterative smoothing. Based on a one-parameter subgroup of diffeomorphisms, the resulting registration method is diffeomorphic and fast. Furthermore, the accuracy of our method compares favorably to the state-of-the-art registration algorithms. We validate the technique in two different applications that use registration to transfer segmentation labels onto a new image: (1) parcellation of *in-vivo* cortical surfaces and (2) Brodmann area localization in *ex-vivo* cortical surfaces.

A preliminary version of this work was published at the International Conference on Medical Image Computing and Computer Assisted Intervention [183]. An expanded version has been accepted to the IEEE Transactions on Medical Imaging and forms the core of this chapter [184]. We also note that our Spherical Demons code is freely available[1].

While most discussion in this chapter concentrates on pairwise registration of spherical images, the proposed Spherical Demons algorithm can be modified to incorporate a probabilistic atlas. We derive and implement two variants of the algorithm for registration to an atlas corresponding to whether we warp the atlas or the subject. On a Xeon 3.2GHz single processor machine, registration of a cortical surface mesh to an atlas mesh, both with more than 160k nodes, requires less than 5 minutes when warping the atlas and less than 3 minutes when warping the subject. The runtime is therefore comparable to other non-linear landmark-free cortical surface registration algorithms whose runtime ranges from

[1] There are two versions of the code (matlab and ITK) available at http://sites.google.com/site/yeoyeo02/software/sphericaldemonsrelease. The matlab code is used in the experiments of this chapter. The preliminary ITK code [76, 77, 78] can also be found at http://hdl.handle.net/10380/3117.

minutes [47, 157] to more than an hour [59, 180]. However, the other fast algorithms suffer from folding spherical triangles [157] and intersecting triangles in 3D [47] since only soft constraints are used. No runtime comparison can be made with spherical registration algorithm of the LDDMM type because to the best of our knowledge, no landmark-free LDDMM spherical registration algorithm that handles cortical surfaces has been developed yet.

Unlike landmark-based methods for surface registration [18, 46, 53, 67, 126, 153, 163], we do not assume the existence of landmarks that can be easily matched across images. Landmark-free methods have the advantage of allowing for a fully automatic processing and analysis of medical images. Unfortunately, landmark-free registration is also more challenging, because no information about correspondences is provided. The difficulty is exacerbated for the cerebral cortex since different sulci and gyri appear locally similar. Nevertheless, we demonstrate that our algorithm is accurate in both cortical parcellation and cyto-architectonic localization applications.

The Spherical Demons algorithm for registering cortical surfaces presented here does not take into account the metric properties of the original cortical surface. FreeSurfer [59] uses a regularization that penalizes deformation of the spherical coordinate system based on the distortion computed on the original cortical surface. Thompson *et al.* [154] suggest the use of Christoffel symbols [90] to correct for the metric distortion of the initial spherical coordinate system during the registration process. However, it is unclear whether correcting for the metric properties of the cortex is important in practice, since we demonstrate that the accuracy of the Spherical Demons algorithm compares favorably to that of FreeSurfer. A possible reason is that we initialize the registration with a spherical parametrization that minimizes metric distortion between the spherical representation and the original cortical surface [59], so that inter-subject variability is greater than the initial metric distortion.

This chapter is organized as follows. In the next section, we discuss the classical Demons algorithm [152] and its diffeomorphic variant [168]. In Section 4.2, we present the extension of the Diffeomorphic Demons algorithm to the sphere. We conclude with experiments in Section 4.3 and further discussion in Section 4.4. Appendix B at the end of the thesis provides technical and implementation details of the Spherical Demons algorithm and the extension to probabilistic atlases.

4.1 Background - Demons Algorithm

We choose to work with the modified Demons objective function [32, 168]. Let F be the fixed image, M be the moving image and c be the desired transformation that deforms the moving image M to match the fixed image F. Throughout this chapter, we assume that F and M are scalar images, even though it is easy to extend the results to vector

```
Algorithm 1: Demons Algorithm
  Data: A fixed image $F$ and moving image $M$.
  Result: Transformation $\Gamma$ so that $M \circ \Gamma$ is "close" to $F$.
  Set $\Upsilon^0$ = identity transformation (or some a-priori transformation, e.g., from a
  previous registration)
  repeat
      Step 1. Given $\Upsilon^{(t)}$,
          Minimize the first two terms of Eq. (4.1)
          $u^{(t)} = \underset{u}{\operatorname{argmin}} \left\| \Sigma^{-1} \left( F - M \circ \{\Upsilon^{(t)} \circ u\} \right) \right\|^2 + \frac{1}{\sigma_x^2} \operatorname{dist}\left( \Upsilon^{(t)}, \{\Upsilon^{(t)} \circ u\} \right), \quad (4.2)$
          where $u$ is any admissible transformation. Compute $\Gamma^{(t)} = \Upsilon^{(t)} \circ u^{(t)}$.
      Step 2. Given $\Gamma^{(t)}$,
          Minimize the last two terms of Eq. (4.1):
          $\Upsilon^{(t+1)} = \underset{\Upsilon}{\operatorname{argmin}} \frac{1}{\sigma_x^2} \operatorname{dist}(\Upsilon, \Gamma^{(t)}) + \frac{1}{\sigma_T^2} \operatorname{Reg}(\Upsilon). \quad (4.3)$
  until convergence ;
```

images [185, 186]. We introduce a hidden transformation Υ and seek

$$(\Upsilon^*, \Gamma^*) = \underset{\Upsilon,\Gamma}{\operatorname{argmin}} \|\Sigma^{-1} (F - M \circ \Gamma) \|^2 + \frac{1}{\sigma_x^2} \operatorname{dist}(\Upsilon, \Gamma) + \frac{1}{\sigma_T^2} \operatorname{Reg}(\Upsilon). \tag{4.1}$$

In this case, the fixed image F and warped moving image $M \circ \Gamma$ are treated as $N \times 1$ vectors. Typically, $\operatorname{dist}(\Upsilon, \Gamma) = \|\Upsilon - \Gamma\|^2$, encouraging the resulting transformation Γ to be close to the hidden transformation Υ and $\operatorname{Reg}(\Upsilon) = \|\nabla(\Upsilon - \operatorname{Id})\|^2$, i.e., the regularization penalizes the gradient magnitude of the displacement field $\Upsilon - \operatorname{Id}$ of the hidden transformation Υ. σ_x and σ_T provide a tradeoff among the different terms of the objective function. Σ is typically a diagonal matrix that models the variability of a feature at a particular voxel. It can be set manually or estimated during the construction of an atlas.

This formulation facilitates a two-step optimization procedure that alternately optimizes the first two (first and second) and last two (second and third) terms of Eq. (4.1). Starting from an initial displacement field Υ^0, the Demons algorithm iteratively seeks an update transformation to be composed with the current estimate, as summarized in Algorithm 1.

In the original Demons algorithm [152], the space of admissible warps includes all 3D displacement fields, and the composition operator $\circ$ corresponds to the addition of displacement fields. The resulting transformation might therefore be not invertible. In the Diffeormorphic Demons algorithm [168], the update u is a diffeormorphism from $\mathbb{R}^3$ to $\mathbb{R}^3$ parameterized by a stationary velocity field $\vec{v}$. Note that $\vec{v}$ is a function that associates a tangent vector with each point in $\mathbb{R}^3$. Under certain mild smoothness conditions, a stationary velocity field $\vec{v}$ is related to a diffeomorphism through the exponential mapping $u = \exp(\vec{v})$. In this case, the stationary ODE $\partial x(t)/\partial t = \vec{v}(x(t))$ with the initial condition $x(0) \in \mathbb{R}^3$ yields $\exp(\vec{v})$ as a solution at time 1, i.e., $x(1) = \exp(\vec{v})(x(0)) \in \mathbb{R}^3$. In this case, $\exp(\vec{v})(x(0))$ maps point $x(0)$ to point $x(1)$.

The Demons algorithm and its variants are fast because for certain forms of $\text{dist}(\Upsilon, \Gamma)$ and $\text{Reg}(\Upsilon)$, Step 1 reduces to a non-linear least-squares problem that can be efficiently minimized via Gauss-Newton optimization and Step 2 can be solved by a single convolution of the displacement field Γ with a smoothing kernel. The proof of the reduction of Step 2 to a smoothing operation is illuminating and holds for $\text{dist}(\Upsilon, \Gamma) = \|\Upsilon - \Gamma\|^2$ and any Sobolev norm $\text{Reg}(\Upsilon) = \sum_i \sigma_i \|\nabla^i(\Upsilon - \text{Id})\|^2$ [32, 113]. In practice, a Gaussian filter is used without consideration of the actual induced norm [32, 168]. The proof uses Fourier transforms and is therefore specific to the Euclidean domain. Due to differences between the geometry of the sphere and Euclidean spaces, we will see in Section 4.2.4 that the reduction of Step 2 to a smoothing operation is only an approximation on the sphere.

4.2 Spherical Demons

In this section, we demonstrate suitable choices of $\text{dist}(\Upsilon, \Gamma)$ and $\text{Reg}(\Upsilon)$ that lead to efficient optimization of the modified Demons objective function in Eq. (4.1) on the unit sphere S^2. We construct updates u as diffeomorphisms from S^2 to S^2 parameterized by a stationary velocity field $\vec{v}$. We emphasize that unlike Diffeomorphic Demons [168], $\vec{v}$ is a tangent vector field on the sphere and not an arbitrary 3D vector field. A glossary of common terms used throughout the chapter is found in Table 4.1.

4.2.1 Choice of dist(Υ, Γ)

Suppose the transformations Γ and Υ map a point $x_n \in S^2$ to two different points $\Gamma(x_n) \in S^2$ and $\Upsilon(x_n) \in S^2$ respectively. An intuitive notion of distance between $\Gamma(x_n)$ and $\Upsilon(x_n)$ would be the geodesic distance between $\Gamma(x_n)$ and $\Upsilon(x_n)$. Therefore, we could define $\text{dist}(\Upsilon, \Gamma) = \sum_{n=1}^N \text{geodesic}(\Upsilon(x_n), \Gamma(x_n))$. For reasons that will become clear in Section 4.2.4, we prefer to define $\text{dist}(\Upsilon, \Gamma)$ in terms of a tangent vector representation of the transformations Γ and Υ, illustrated in Fig. 4-1, where the length of the tangent vector encodes the amount of deformation.

Let $T_{x_n}S^2$ be the tangent space at x_n. We define $\vec{\Gamma}_n \in T_{x_n}S^2$ to be the tangent vector at x_n pointing along the great circle connecting x_n to $\Gamma(x_n)$. In this work, we set the length of $\vec{\Gamma}_n$ to be equal to the sine of the angle between x_n and $\Gamma(x_n)$. With this particular choice of length, there is a one-to-one correspondence between $\Gamma(x_n)$ and $\vec{\Gamma}_n$, assuming the angle between x_n and $\Gamma(x_n)$ is less than $\pi/2$, which is a reasonable assumption even for relatively

Table 4.1: Glossary of terms used throughout the chapter.

F, M	Fixed image F, moving image M.
Σ	Typically a diagonal matrix that models variability of feature values at a particular vertex.
σ_x, σ_T	Parameters of Demons cost function in Eq. (4.1).
Γ, Υ	Transformations from S^2 to S^2. Γ is the transformation we are seeking. Υ is the smooth hidden transformation close to Γ.
$\vec{\Gamma} \triangleq \{\vec{\Gamma}_n\}$, $\vec{\Upsilon} \triangleq \{\vec{\Upsilon}_n\}$	Discrete tangent vector representation of the deformations (see Fig. 4-1 and Eq. (4.5)). For example, given the tangent vector $\vec{\Gamma}_n$ at $x_n \in S^2$, one can compute $c(x_n)$.
$\vec{v} \triangleq \{\vec{v}_n\}$	We parameterize diffeomorphic transformations from S^2 to S^2 by a composition of diffeomorphisms, each parameterized by a stationary velocity field $\vec{v}$. $\vec{v}_n$ is the velocity vector at x_n.
$u(\cdot) \triangleq \exp(\vec{v})(\cdot)$	The diffeomorphism parameterized by the stationary velocity field $\vec{v}$ is the solution of a stationary ODE at time 1.
$E_n \triangleq [\vec{e}^{n1} \ \vec{e}^{n2}]$	$\vec{e}^{n1}$ and $\vec{e}^{n2}$ are orthonormal vectors tangent to the sphere at x_n
Ψ_n	Coordinate chart defined in Eq. (4.10): $\Psi_n(x') = \frac{x_n + E_n x'}{\|x_n + E_n x'\|}$. Ψ_n is a diffeomorphism between $\mathbb{R}^2$ and a hemisphere centered at $x_n \in S^2$.
$\vec{z}_n$	$\vec{z}_n$ is a tangent vector at the origin of $\mathbb{R}^2$. At x_n, the velocity vector $\vec{v}_n = E_n \vec{z}_n$ via the coordinate chart Ψ_n (see Eq. (4.14)).

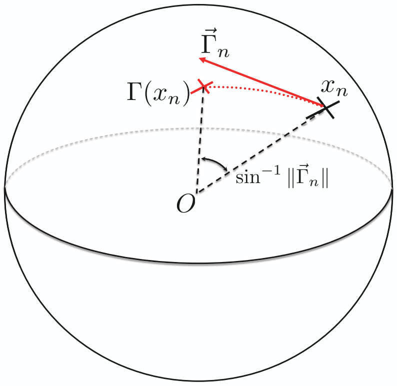


Figure 4-1: Tangent vector representation of transformation Γ. See text for more details.

large deformations. The choice of this length leads to a compact representation of $\vec{\Gamma}_n$ via vector products. We define G_n to be the 3×3 skew-symmetric matrix representing the cross-product of x_n with any vector:

$$G_n = \begin{pmatrix} 0 & -x_n(3) & x_n(2) \\ x_n(3) & 0 & -x_n(1) \\ -x_n(2) & x_n(1) & 0 \end{pmatrix}, \tag{4.4}$$

where $x_n(i)$ is the i-th coordinate of x_n. Thus, $x_n \times \Gamma(x_n) = G_n \Gamma(x_n)$. Then on a unit sphere, we obtain

$$\vec{\Gamma}_n = -x_n \times (x_n \times \Gamma(x_n)) = -G_n^2 \Gamma(x_n). \tag{4.5}$$

A more intuitive choice for the length of $\vec{c}_n$ might be the geodesic distance between x_n and $\Gamma(x_n)$. If we restrict $\vec{c}_n$ to be at most length π, there is a one-to-one mapping between this choice of the tangent vector $\vec{\Gamma}_n$ and the resulting transformation $\Gamma(x_n)$. Indeed, such a choice of a tangent vector corresponds to an exponential map of S^2 [90]. The resulting expression for $\vec{\Gamma}_n = \frac{-2G_n^2\Gamma(x_n)}{\|G_n^2\Gamma(x_n)\|} \sin^{-1}\left(\frac{\Gamma(x_n)-x_n}{2}\right)$ is feasible, but more complicated than Eq. (4.5). In this work, for simplicity, we follow the definition in Eq. (4.5).

Given N vertices $\{x_n\}_{n=1}^N$, the set of transformed points $\{\Gamma(x_n)\}_{n=1}^N$ – or equivalently the tangent vectors $\{\vec{\Gamma}_n\}_{n=1}^N$ – together with a choice of an interpolation function define the transformation Γ everywhere on S^2. Similarly, we can define the transformation Υ or the equivalent tangent vector field $\vec{\Upsilon}$ through a set of N tangent vectors $\{\vec{\Upsilon}_n\}_{n=1}^N$. We emphasize that these tangent vector fields are simply a convenient representation of the transformations Υ and Γ and should not be confused with the stationary velocity field $\vec{v}$ that will be used later on. We now set

$$\text{dist}(\Upsilon, \Gamma) = \sum_{n=1}^{N} \|\vec{\Upsilon}_n - \vec{\Gamma}_n\|^2, \tag{4.6}$$

which is well-defined since both $\vec{\Gamma}_n$ and $\vec{\Upsilon}_n$ belong to $T_{x_n}S^2$ for each $n = 1, \cdots, N$.

4.2.2 Spherical Demons Step 1

In this section, we show that the update for Step 1 of the Spherical Demons algorithm can be computed independently for each vertex. With our choice of dist(Υ, Γ), step 1 of the algorithm becomes a minimization with respect to the velocity field $\vec{v} \triangleq \{\vec{v}_n \in T_{x_n}S^2\}_{n=1}^N$.

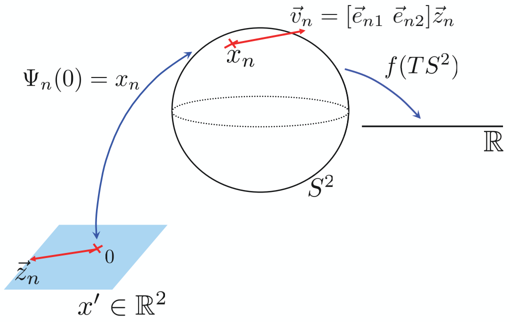


Figure 4-2: Coordinate chart of the sphere S^2. The chart allows a reparameterization of the constrained optimization problem f in step 1 of the Spherical Demons algorithm into an unconstrained one.

By substituting $u = \exp(\vec{v})$ and $\text{dist}(\Upsilon, \Gamma) = \sum_{n=1}^{N} \|\vec{\Upsilon}_n - \vec{\Gamma}_n\|^2$ into Eq. (4.2), we obtain

$$\vec{v}^{(t)} = \underset{\vec{v}}{\text{argmin}}\, f(\vec{v}) \tag{4.7}$$

$$= \underset{\vec{v}}{\text{argmin}} \left\| \Sigma^{-1} \left(F - M \circ \{\Upsilon^{(t)} \circ \exp(\vec{v})\} \right) \right\|^2 + \frac{1}{\sigma_x^2} \text{dist} \left(\Upsilon^{(t)}, \{\Upsilon^{(t)} \circ \exp(\vec{v})\} \right) \tag{4.8}$$

$$= \underset{\vec{v}}{\text{argmin}} \sum_{n=1}^{N} \frac{1}{\sigma_n^2} \left(F(x_n) - M \circ \{\Upsilon^{(t)} \circ \exp(\vec{v})\}(x_n) \right)^2 + \frac{1}{\sigma_x^2} \sum_{n=1}^{N} \left\| \vec{\Upsilon}_n^{(t)} + G_n^2\{\Upsilon^{(t)} \circ \exp(\vec{v})\}(x_n) \right\|^2 , \tag{4.9}$$

where σ_n^2 is the n-th diagonal entry of Σ and $\circ$ denotes warp composition.

Defining Coordinate Charts on the Sphere. The cost function in Eq. (4.9) is a mapping from the tangent bundle TS^2 to the real numbers $\mathbb{R}$. We can think of each tangent vector $\vec{v}_n$ as a 3×1 vector in $\mathbb{R}^3$ tangent to the sphere at x_n. Therefore $\vec{v}_n$ has 2 degrees of freedom and Eq. (4.9) represents a constrained optimization problem. Instead of dealing with the constraints, we employ coordinate charts that are diffeomorphisms (smooth and invertible mappings) between open sets in R^2 and open sets on S^2. The differential of the coordinate chart establishes correspondences between the tangent bundles $T\mathbb{R}^2$ and TS^2 [90, 111], so we can reparameterize the constrained optimization problem into an unconstrained one in terms of $T\mathbb{R}^2$ (see Fig. 4-2).

It is a well-known fact in differential geometry that covering S^2 requires at least two coordinate charts. Since the tools of differential geometry are coordinate-free [90, 111], our results are independent of the choice of the coordinate charts. Let $\vec{e}^{n1}$, $\vec{e}^{n2}$ be any two orthonormal 3 × 1 vectors tangent to the sphere at x_n, where orthonormality is defined via the usual Euclidean inner product in $3D$. In this work, for each mesh vertex x_n, we define a local coordinate chart $\Psi_n : \mathbb{R}^2 \mapsto S^2$,

$$\Psi_n(x') = \frac{x_n + E_n x'}{\|x_n + E_n x'\|}, \quad \text{where} \;\; E_n = [\vec{e}^{n1} \;\; \vec{e}^{n2}]. \tag{4.10}$$

As illustrated in Fig. 4-2, $\Psi_n(0) = x_n$. Let $\vec{z}_n$ be a 2 × 1 tangent vector at the origin of $\mathbb{R}^2$. With the choice of the coordinate chart above, the corresponding tangent vector at x_n is given by the differential of the mapping $D\Psi_n(\cdot)$ evaluated at $x' = 0$:

$$\begin{aligned}
\vec{v}_n &= D\Psi_n(0)\vec{z}_n \qquad &(4.11)\\
&= \frac{I_{3\times 3} - \Psi_n(0)\Psi_n^T(0)}{\|\Psi_n(0)\|} E_n \vec{z}_n \qquad &(4.12)\\
&= \frac{I_{3\times 3} - x_n x_n^T}{\|x_n\|} E_n \vec{z}_n \qquad &(4.13)\\
&= E_n \vec{z}_n = [\vec{e}^{n1} \;\; \vec{e}^{n2}]\vec{z}_n. \qquad &(4.14)
\end{aligned}$$

The above equation defines the mapping of a tangent vector $\vec{z}_n$ at the origin of $\mathbb{R}^2$ to the tangent vector $\vec{v}_n$ at x_n via the differential of the coordinate chart $D\Psi_n$ at $x' = 0$. We note that for a tangent vector at an arbitrary point in $\mathbb{R}^2$, the expression for the corresponding tangent vector on the sphere is more complicated. This motivates our definition of a separate chart for each mesh vertex, to simplify the derivations.

Gauss-Newton Step of Spherical Demons. From Eq. (4.14), we obtain $\exp(\vec{v}) = \exp(\{\vec{v}_n\}) = \exp(\{E_n\vec{z}_n\})$ and rewrite Eq. (4.9) as an unconstrained optimization problem:

$$\begin{aligned}
\{\vec{z}_n^{(t)}\} = \operatorname*{argmin}_{\{\vec{z}_n\}} \sum_{n=1}^{N} \frac{1}{\sigma_n^2} \Big(F(x_n) - M \circ \{\Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})\}(x_n) \Big)^2 \\
+ \frac{1}{\sigma_x^2} \sum_{n=1}^{N} \Big\| \vec{\Upsilon}_n^{(t)} + G_n^2 \{\Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})\}(x_n) \Big\|^2,
\end{aligned} \tag{4.15}$$

$$\triangleq \operatorname*{argmin}_{\{\vec{z}_n\}} \sum_{n=1}^{N} \frac{1}{\sigma_n^2} f_n^2(\vec{z}) + \frac{1}{\sigma_x^2} \sum_{n=1}^{N} \|g_n\|^2(\vec{z}) \tag{4.16}$$

This non-linear least-squares form can be optimized efficiently with the Gauss-Newton method, which requires finding the gradient of both terms with respect to $\{\vec{z}_n\}$ at $\{\vec{z}_n = 0\}$ and solving a linearized least-squares problem.

We let $\vec{m}_n^T$ be the 1 × 3 spatial gradient of the warped moving image $M \circ \Upsilon^{(t)}(\cdot)$ at x_n and note that $\vec{m}_n^T$ is tangent to the sphere at x_n. The computation of $\vec{m}_n^T$ is discussed in

Appendix B.1.1. Defining $u_n \triangleq \exp(\{E_n\vec{z}_n\})(x_n)$, we differentiate the first term of the cost function $f_n(\vec{z})$ in Eq. (4.15) using the chain rule, resulting in the 1×2 vector:

$$\begin{aligned}
&\frac{\partial}{\partial \vec{z}_k}\left[F(x_n) - M \circ \{\Upsilon^{(t)} \circ \exp(\{E_n\vec{z}_n\})\}(x_n)\right]_{\vec{z}=0} \\
&= -\frac{\partial}{\partial \vec{z}_k} M \circ \{\Upsilon^{(t)} \circ \exp(\{E_n\vec{z}_n\})\}(x_n)\Big|_{\vec{z}=0} && (4.17)\\
&= -\frac{\partial M \circ \{\Upsilon^{(t)} \circ \exp(\{E_n\vec{z}_n\})\}(x_n)}{\partial \exp(\{E_n\vec{z}_n\})(x_n)}\left[\frac{\partial \exp(\{E_n\vec{z}_n\})(x_n)}{\partial \vec{z}_k}\right]\Bigg|_{\vec{z}=0} && (4.18)\\
&= -\frac{\partial M \circ \Upsilon^{(t)}(u_n)}{\partial u_n}\bigg|_{u_n=x_n}\left[\frac{\partial \exp(\{E_n\vec{z}_n\})(x_n)}{\partial E_k\vec{z}_k}\frac{\partial E_k\vec{z}_k}{\partial \vec{z}_k}\right]\Bigg|_{\vec{z}=0} && (4.19)\\
&= -\vec{m}_n^T E_n \delta(k,n), && (4.20)
\end{aligned}$$

where $\delta(k,n) = 1$ if $k = n$ and 0 otherwise. Eq. (4.20) uses the fact that the differential of $\exp(\vec{v})$ at $\vec{v} = 0$ is the identity [115], i.e, $[D\exp(0)]\,\vec{v} = \vec{v}$. In other words, a change in velocity $\vec{v}_k$ at vertex x_k does not affect $\exp(\vec{v})(x_n)$ for $n \neq k$ up to the first order derivatives.

Similarly, we define S_n^T to be the 3×3 Jacobian of $\Upsilon^{(t)}(\cdot)$ at x_n. The computation of S_n^T is discussed in Appendix B.1.2. Differentiating the second term of the cost function $g_n(\vec{z})$ in Eq. (4.15) using the chain rule, we get the 3×2 matrix:

$$\frac{\partial}{\partial \vec{z}_k}\left[\vec{\Upsilon}_n^{(t)} + G_n^2\{\Upsilon^{(t)} \circ \exp(\{E_n\vec{z}_n\})\}(x_n)\right]_{\vec{z}=0} = G_n^2 S_n^T E_n \delta(k,n), \qquad (4.21)$$

where G_n is the skew-symmetric matrix defined in Eq. (4.4).

Once the derivatives are known, we can compute the corresponding gradients based on our choice of the metric of vector fields on S^2. In this work, we assume an l_2 inner product, so that the inner product of vector fields is equal to the sum of the inner product of the individual vectors. The inner product of individual vectors is in turn specified by the choice of the Riemannian metric on S^2. Assuming the canonical metric, so that the inner product of two tangent vectors is the usual inner product in the Euclidean space [90], the gradients are equal to the transpose of the derivatives Eqs. (4.20), (4.21) (see Appendix B.1.3). We can then rewrite Eq. (4.15) as a linearized least-squares objective function:

$$\begin{aligned}
\{\vec{z}_n^{(t)}\} &\approx \operatorname*{argmin}_{\{\vec{z}_n\}} \sum_{n=1}^{N} \frac{1}{\sigma_n^2}\Big(f_n(\vec{z}=0) + \nabla_{l_2} f_n \vec{z}\Big)^2 + \frac{1}{\sigma_x^2}\sum_{n=1}^{N}\Big\| g_n(\vec{z}=0) + \nabla_{l_2} g_n \vec{z}\Big\|^2 && (4.22)\\
&= \operatorname*{argmin}_{\{\vec{z}_n\}} \sum_{n=1}^{N} \frac{1}{\sigma_n^2}\Big(\Big(F(x_n) - M \circ \Upsilon^{(t)}(x_n)\Big) - \vec{m}_n^T E_n \vec{z}_n\Big)^2 + \frac{1}{\sigma_x^2}\sum_{n=1}^{N}\big\|G_n^2 S_n^T E_n \vec{z}_n\big\|^2 && (4.23)\\
&= \operatorname*{argmin}_{\{\vec{z}_n\}} \sum_{n=1}^{N} \left\| \begin{pmatrix} \frac{1}{\sigma_n}\left(F(x_n) - M \circ \Upsilon^{(t)}(x_n)\right) \\ 0 \end{pmatrix} + \begin{pmatrix} -\frac{1}{\sigma_n}\vec{m}_n^T \\ \frac{1}{\sigma_x} G_n^2 S_n^T \end{pmatrix} E_n \vec{z}_n \right\|^2. && (4.24)
\end{aligned}$$

Because of the delta function $\delta(k,n)$ in the derivatives in Eqs. (4.20), (4.21), $\vec{z}_n$ only appears in the n-th term of the cost function Eq. (4.24). The solution of Eq. (4.24) can therefore be computed independently for each $\vec{z}_n$. Solving this linear least-squares equation yields an update rule for $\vec{z}_n$:

$$\vec{z}_n^{(t)} = \frac{F(x_n) - M \circ \Upsilon^{(t)}(x_n)}{\sigma_n^2} \left(E_n^T \left[\frac{1}{\sigma_n^2} \vec{m}_n \vec{m}_n^T + \frac{1}{\sigma_x^2} S_n (G_n^2)^T G_n^2 S_n^T \right] E_n \right)^{-1} E_n^T \vec{m}_n \ . \tag{4.25}$$

For each vertex, we only need to perform matrix-vector multiplication of up to 3×3 matrices and matrix inversion of 2×2 matrices. This implies the update rule for $\vec{v}_n$:

$$\vec{v}_n^{(t)} = E_n \vec{z}_n^{(t)} \tag{4.26}$$

$$= \frac{F(x_n) - M \circ \Upsilon^{(t)}(x_n)}{\sigma_n^2} E_n \left(E_n^T \left[\frac{1}{\sigma_n^2} \vec{m}_n \vec{m}_n^T + \frac{1}{\sigma_x^2} S_n (G_n^2)^T G_n^2 S_n^T \right] E_n \right)^{-1} E_n^T \vec{m}_n \ . \tag{4.27}$$

In practice, we use the Levenberg-Marquardt modification of Gauss-Newton optimization [125] to ensure matrix invertibility:

$$\vec{v}_n^{(t)} = \frac{F(x_n) - M \circ \Upsilon^{(t)}(x_n)}{\sigma_n^2} E_n \left(E_n^T \left[\frac{1}{\sigma_n^2} \vec{m}_n \vec{m}_n^T + \frac{1}{\sigma_x^2} S_n (G_n^2)^T G_n^2 S_n^T \right] E_n + \epsilon I_{2\times 2} \right)^{-1} E_n^T \vec{m}_n \ . \tag{4.28}$$

where ϵ is a regularization constant. We note that in the classical Euclidean Demons [152, 32], the term $E_n^T S_n (G_n^2)^T G_n^2 S_n^T E_n$ turns out to be the identity, so it can also be seen as utilizing Levenberg-Marquardt optimization. Once again, we emphasize that a different choice of the coordinate charts will lead to the same update.

Given $\{\vec{v}_n^{(t)}\}_{n=1}^N$, we use "scaling and squaring" to compute $\exp(\vec{v}^{(t)})$ [8], which is then composed with the current transformation estimate $\Upsilon^{(t)}$ to form $\Gamma^{(t)} = \Upsilon^{(t)} \circ \exp(\vec{v}^{(t)})$. Appendix B.4 discusses implementation details of extending the "scaling and squaring" procedure in Euclidean spaces to S^2.

4.2.3 Choice of Reg(Υ)

We now define the Reg(Υ) term using the corresponding tangent vector field representation $\vec{\Upsilon}$. Following the work of [67, 158], we let H be the Hilbert space of square integrable vector fields on the sphere defined by the inner product:

$$\langle \vec{u}_1, \vec{u}_2 \rangle_H = \int_{S^2} \langle \vec{u}_1(x), \vec{u}_2(x) \rangle_R \, dS^2 \ , \tag{4.29}$$

where $\vec{u}_1, \vec{u}_2 \in H$ and $\langle \cdot, \cdot \rangle_R$ refers to the canonical metric. Because vector fields from H are not necessarily smooth, we restrict the deformation $\vec{\Upsilon}$ to belong to the Hilbert space

$V \subset H$ of vector fields obtained by the closure of the space of smooth vector fields on S^2 via a choice of the so-called energetic inner product denoted by

$$\langle \vec{u}, \vec{v} \rangle_V = \langle L\vec{u}, \vec{v} \rangle_H \ , \tag{4.30}$$

where L could for example be the Laplacian operator on smooth vector fields on S^2 [67, 158].

We define $\text{Reg}(\Upsilon) \triangleq \|\vec{\Upsilon}\|_V$. With a proper choice of the energetic inner product (e.g., Laplacian), a smaller value of $\|\vec{\Upsilon}\|_V$ corresponds to a smoother vector field and thus smoother transformation Υ. As we will see later in this section, the exact choice of the inner product is unimportant in our implementation.

4.2.4 Optimizing Step 2 of Spherical Demons

With our choice of $\text{dist}(\Upsilon, \Gamma)$ in Section 4.2.1 and $\text{Reg}(\Upsilon)$ in Section 4.2.3, the optimization in Step 2 of the Spherical Demons algorithm

$$\vec{\Upsilon}^{(t+1)} = \underset{\vec{\Upsilon}}{\text{argmin}} \frac{1}{\sigma_x^2} \sum_{n=1}^{N} \|\vec{\Upsilon}_n - \vec{\Gamma}_n^{(t)}\|^2 + \frac{1}{\sigma_T^2} \|\vec{\Upsilon}\|_V \tag{4.31}$$

seeks a smooth vector field $\vec{\Upsilon} \in V$ that approximates the tangent vectors $\{\vec{\Gamma}_n^{(t)}\}_{n=1}^N$. This problem corresponds to the inexact vector spline interpolation problem solved in [67], motivating our use of tangent vectors in the definition of $\text{dist}(\Upsilon, \Gamma)$ in Section 4.2.1, instead of the more intuitive choice of geodesic distance.

We can express $\vec{\Gamma}_n$ and $\vec{\Upsilon}_n$ as $E_n\Gamma_n$ and $E_n\Upsilon_n$ respectively. Let $\hat{\Gamma}$ and $\hat{\Upsilon}$ be $2N \times 1$ vectors corresponding to stacking Γ_n and Υ_n respectively. The particular optimization formulated in Eq. (4.31) has a unique optimum [67], given by

$$\hat{\Upsilon} = K \left(\frac{\sigma_x^T}{\sigma_T^2} I_{2N \times 2N} + K \right)^{-1} \hat{\Gamma}, \tag{4.32}$$

where K is a $2N \times 2N$ matrix consisting of $N \times N$ blocks of 2×2 matrices: the (i, j) block corresponds to $k(x_i, x_j)T_{x_i,x_j}$. The 2×2 linear transformation $T_{x_i,x_j}(\cdot)$ parallel transports a tangent vector along the great circle from $T_{x_i}S^2$ to $T_{x_j}S^2$. $k(x_i, x_j)$ is a non-negative scalar function uniquely determined by the choice of the energetic norm. Typically, $k(x_i, x_j)$ monotonically decreases as a function of the distance between x_i and x_j. The proof of the uniqueness of the global optimum and the form of solution in Eq. (4.32) follow from the fact that the Hilbert space V is a reproducing kernel hilbert space (RKHS), allowing the exploitation of the Riesz representation theorem [67]. This offers a wide range of choices of regularization depending on the choice of the energetic norm and the corresponding RKHS.

In [67], the spherical vector spline interpolation problem was applied to landmark matching on S^2, resulting in a reasonable sized linear system of equations. Solving the matrix inversion shown in Eq. (4.32) is unfortunately prohibitive for cortical surfaces with more than $100,000$ vertices. If one chooses a relatively wide kernel $k(x_i, x_j)$, the system is not even sparse.

Inspired by the convolution method of optimizing Step 2 in the Demons algorithm [32, 152, 168] and the convolution-based fast fluid registration in the Euclidean space [29], we propose an iterative smoothing approximation to the solution of the spherical vector spline interpolation problem.

In each smoothing iteration, for each vertex x_i, tangent vectors of neighboring vertices x_j are parallel transported to x_j and linearly combined with the tangent vector at x_i. The weights for the linear combination are set to $\lambda(x_i, x_i) = \frac{1}{1+|N_i|\exp(-\frac{1}{2\gamma})}$ and $\lambda(x_i, x_j) = \frac{\exp(-\frac{1}{2\gamma})}{1+|N_i|\exp(-\frac{1}{2\gamma})}$ for $i \neq j$, where $|N_i|$ is the number of neighboring vertices of x_i. Therefore, larger number of iterations m and values of γ results in greater amount of smoothing.

We note that the iterative smoothing approximation to spline interpolation is not exact because parallel transport is not transitive on S^2 due to the non-flat curvature of S^2 (unlike in Euclidean space), i.e., parallel transporting a tangent vector from point a to b to c results in a vector different from the result of parallel transporting a tangent vector from a to c. Furthermore, the approximation accuracy degrades as the distribution of points becomes less uniform. In Appendix B.2, we demonstrate empirically that iterative smoothing provides a good approximation of spherical vector spline interpolation for a relatively uniform distribution of points corresponding to those of the subdivided icosahedron meshes used in this work.

4.2.5 Remarks

The Spherical Demons algorithm is summarized in Algorithm 2.

We run the Spherical Demons algorithm in a multi-scale fashion on a subdivided icosahedral mesh. We begin from a subdivided icosahedral mesh (ic4) that contains 2,562 vertices and work up to a subdivided icosahedral mesh (ic7) that contains 163,842 vertices, which is roughly equal to the number of vertices in the cortical meshes we work with. We perform 15 iterations of Step 1 and Step 2 at each level. Because of the fast convergence rate of the Gauss-Newton method, we find that 15 iterations are more than sufficient for our purposes. We also perform a rotational registration at the beginning of each multi-scale level via a sectioned search of the three Euler angles.

Empirically, we find the computation time of the Spherical Demons algorithm is roughly divided equally among the four components: registration by rotation, computing the Gauss-Newton update, performing "scaling and squaring" and smoothing the vector field.

In practice, we work with spheres that are normalized to be of radius 100, because we

Algorithm 2: Spherical Demons Algorithm

```
Data: A fixed spherical image $F$ and moving spherical image $M$.
Result: Diffeomorphism $c$ so that $M \circ c$ is “close” to $F$.
Set $s^0$ = identity transformation (or some a-priori transformation, e.g., from a
previous registration)
repeat
    Step 1. Given $s^{(t)}$,
        foreach vertex $n$ do
            Compute $\vec{v}_n^{(t)}$ using Eq. (4.28).
        end
        Compute $c^{(t)} = \exp(\vec{v})$ using “scaling and squaring”.

    Step 2. Given $c^{(t)}$,
        foreach vertex $n$ do
            Compute $\vec{s}_n^{(t)}$ using Eq. (B.16) implemented via iterative smoothing.
        end

until convergence ;
```

find that at ic7, the average edge length of 1mm corresponds to that of the original cortical surface meshes. This allows for meaningful interpretation of distances on the sphere. This requires slight modification of the equations presented previously to keep track of the radius of the sphere.

The Spherical Demons algorithm presented here registers pairs of spherical images. To incorporate a probabilistic atlas defined by a mean image and a standard deviation image, we modify the Demons objective function in Eq. (4.1), as explained in Appendix B.3. This requires a choice of warping the subject or warping the atlas. We find that interpolating the atlas gives slightly better results, compared with interpolating the subject. However, interpolating the subject results in a runtime of under 3 minutes, while the runtime for interpolating the atlas is less than 5 minutes. In the next section, we report results for interpolating the atlas.

4.3 Experiments

We use two sets of experiments to evaluate the performance of the Spherical Demons algorithm by comparing it to the widely used and freely available FreeSurfer [59] software. The FreeSurfer registration algorithm uses the same similarity measure as Demons, but explicitly penalizes for metric and areal distortion. As we will show, even though the Spherical Demons algorithm does not specifically take into account the original metric properties of the cortical surface, we still achieve comparable if not better registration accuracy than

FreeSurfer. Furthermore, FreeSurfer runtime is more than an hour while Spherical Demons runtime is less than 5 minutes on a Xeon 3.2GHz single processor machine.

There are four parameters in the algorithm. $1/\sigma_x^2$ and ϵ appear in Eq. (4.28). Larger values of $1/\sigma_x^2$ and ϵ decrease the size of the update taken in Step 1 of the Spherical Demons algorithm. In the experiments that follow, we set $1/\sigma_x^2 = \epsilon$ and set their values such that the largest vector of the update velocity field is roughly two times the edge lengths of the mesh. The number of iterations m and the weight $\exp(-\frac{1}{2\gamma})$ determine the degree of smoothing. We set $\gamma = 1$ and explore a range of smoothing iterations m in the following experiments.

4.3.1 Parcellation of *In-vivo* Cortical Surfaces

We validate Spherical Demons in the context of automatic cortical parcellation described in Chapter 1.2.2. We consider the same set of 39 left and right cortical surface models as the previous chapter. For an illustration of the manual parcellation of the cortex and a list of parcellation units, we refer the readers to Fig. 1-4 and Table 1.1 respectively.

We co-register all 39 spherical images of cortical geometry with Spherical Demons by iteratively building an atlas and registering the surfaces to the atlas. The atlas consists of the mean and variance of cortical geometry represented by the surface features described above. We then perform 4-fold cross-validation of the parcellation of the co-registered cortical surfaces. In each iteration of cross-validation, we leave out ten subjects and use the remainder of the subjects to train a classifier [43, 60] that predicts the labels based on location and geometric features. We then apply the classifier to the hold-out set of ten subjects. We perform each iteration with a different hold-out set, i.e., subjects 1-10, 11-20, 21-30 and 31-39.

As mentioned previously, increasing the number of iterations of smoothing results in smoother warps. As discussed in [180], the choice of the tradeoff between the similarity measure and regularization is important for segmentation accuracy. We repeat the above experiments using $\{6, 8, 10, 12, 14\}$ iterations of smoothing. For brevity, we will focus the discussion on using 10 iterations of smoothing and comment on results obtained with the other levels of smoothing.

We repeat the above procedure of performing co-registration and cross-validation with the FreeSurfer registration algorithm [59] using the default FreeSurfer settings. Once again, we use the same features and parcellation algorithm [43, 60]. As before, the atlas consists of the mean and variance of cortical geometry.

To compare the cortical parcellation results, we compute the average Dice measure, defined as the ratio of cortical surface area with correct labels to the total surface area averaged over the test set. Because the average Dice can be misleading by suppressing small structures, we also compute the Dice measure for each structure.

On the left hemisphere, FreeSurfer achieves an average Dice of 88.9, while Spherical

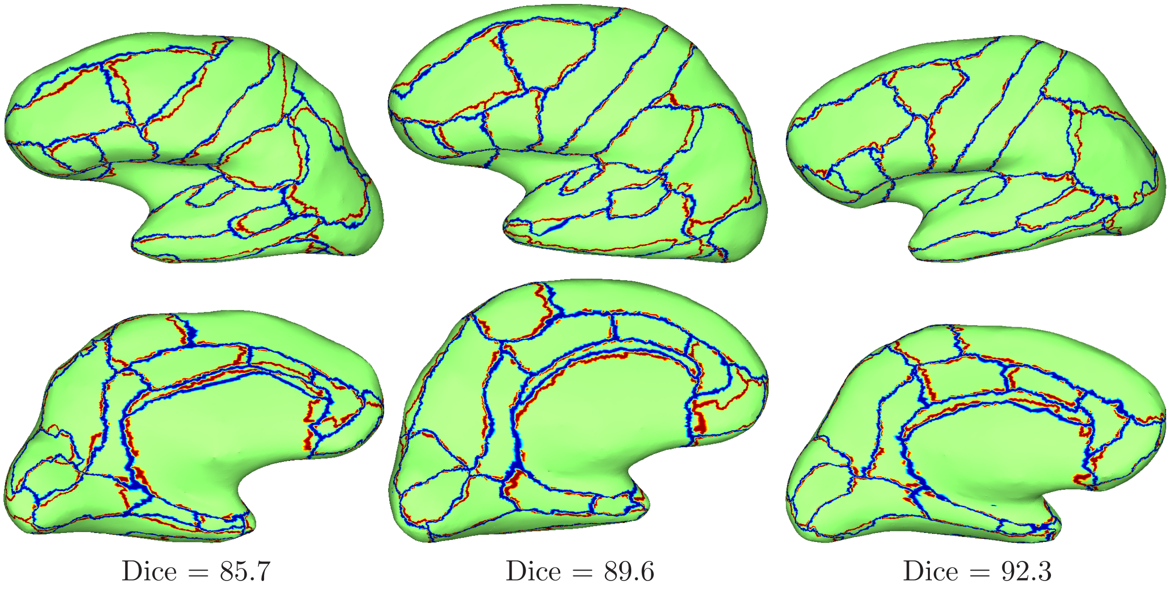


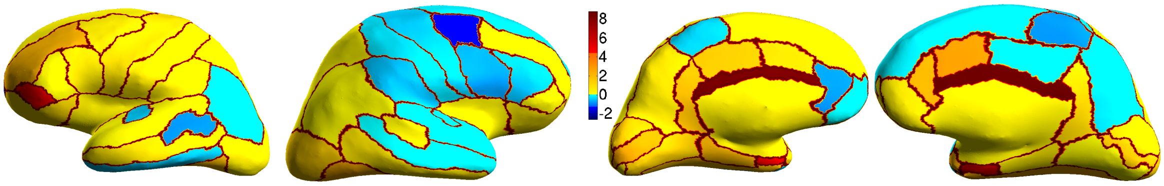
Figure 4-3: Example parcellations of three different subjects. Blue corresponds to ground truth segmentation. Red corresponds to automatic segmentation using Spherical Demons for registration.

Demons achieves an average Dice of 89.6 with 10 iterations of smoothing. While the improvement is not big, the difference is statistically significant for a one-sided t-test with the Dice measure of each subject treated as an independent sample ($p = 2 \times 10^{-6}$). Furthermore, the overall Dice is statistically significantly better than FreeSurfer for all levels of smoothing we considered, with the best overal dice achieved with 12 iterations of smoothing.

On the right hemisphere, FreeSurfer obtains a Dice of 88.8 and Spherical Demons achieves 89.1 with 10 iterations of smoothing. Here, the improvement is smaller, but still statistically significant ($p = 0.01$). Furthermore, the overall dice is statistically significantly better than FreeSurfer for all levels of smoothing we considered, except when 6 iterations of smoothing is used ($p = 0.06$). All results we report in the remainder of this section use 10 iterations of smoothing.

As an illustration, Fig. 4-3 shows the example parcellation of three subjects, where blue corresponds to ground truth segmentation and red corresponds to the automatic segmentation results from using Spherical Demons with 10 iterations of smoothing. The left most subject corresponds to the worst subject, while the right most subject corresponds to the best. The middle subject achieves median segmentation accuracy among the left hemispheres of 39 subjects.

We analyze the segmentation accuracy separately for each structure. To compare Spherical Demons with FreeSurfer, we perform a one-sided paired-sampled t-test treating each subject as an independent sample and correct for multiple comparisons using a False Discovery Rate (FDR) of 0.05 [22]. On the left (right) hemisphere, the segmentations of 16



(a) Lateral View (b) Medial View

Figure 4-4: Percentage Improvement over FreeSurfer. Yellow regions indicate structures scoring better than FreeSurfer. Blue regions correspond to decrease in accuracy. Note that none of these blue regions are statistically significant. The boundaries between parcellation regions are set to reddish-brown to improve visualization of the regions.

(8) structures are statistically significantly improved by Spherical Demons with respect to FreeSurfer, while no structure is significantly worse.

Fig. 4-4 shows the percentage improvement of individual structures over FreeSurfer. Fig. 4-5 displays the average Dice per structure for FreeSurfer and Spherical Demons (10 iterations of smoothing) for the left and right hemispheres. Standard errors of the mean are displayed as red bars. The numbering of the structures correspond to Table 1.1. Parcellation improvements suggest that our registration is at least as accurate as FreeSurfer.

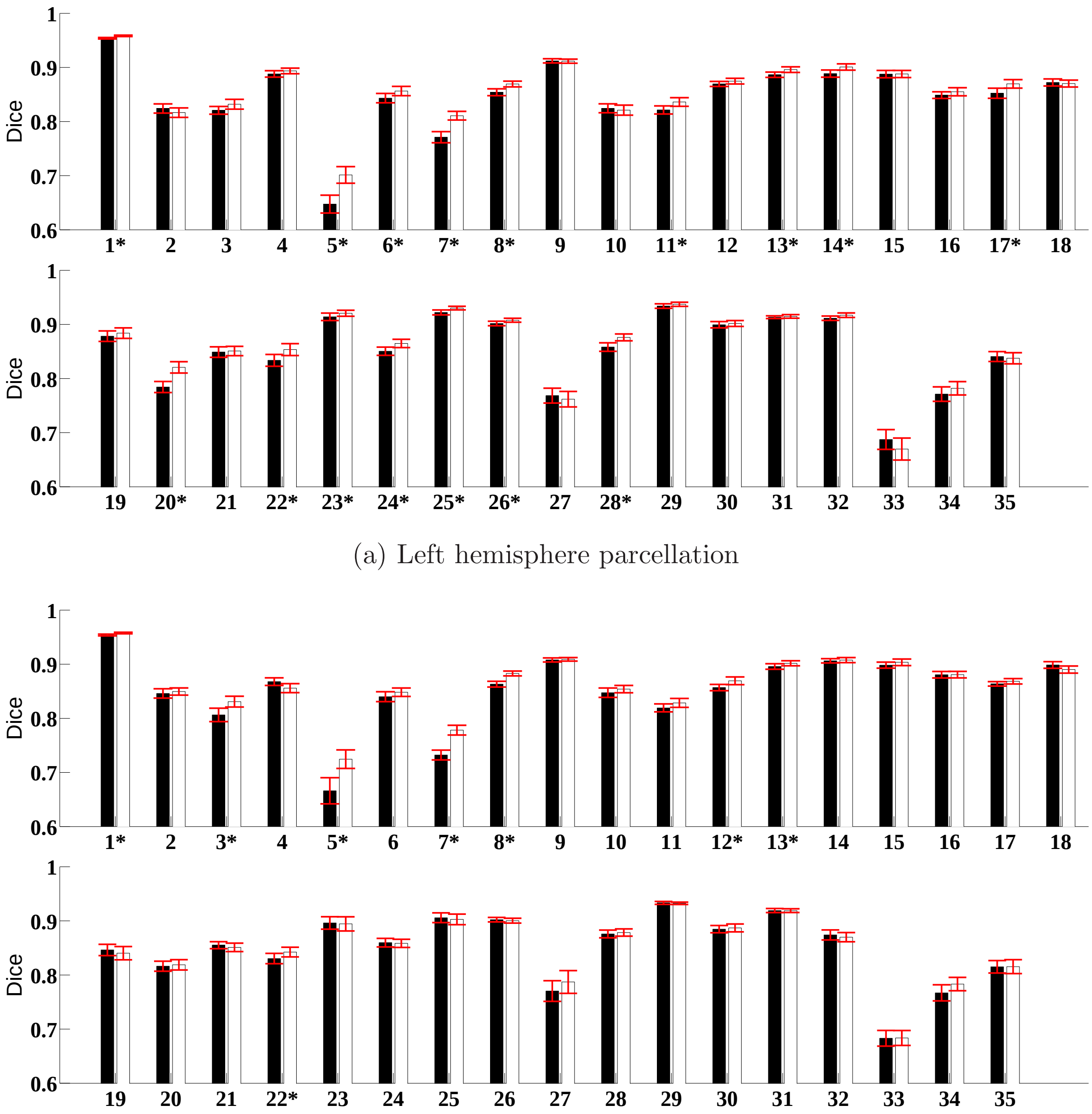
The structures with the worst Dice are the frontal pole and entorhinal cortex. These structures are small and relatively poorly defined by the underlying cortical geometry. For example, the entorhinal cortex is partially defined by the rhinal sulcus, a tiny sulcus that is only visible on the pial surface. The frontal pole is defined by the surrounding structures, rather than by the underlying cortical geometry.

4.3.2 Brodmann Area Localization on *ex-vivo* Cortical Surfaces

We now consider the localization of Brodmann Areas (BAs) in the cortex of the 10 *ex-vivo* subjects described in Chapter 1.2.2. For an illustration of the BAs on the cortex, we refer the readers to Fig. 1-2(c). Here, we do not consider MT because it was not yet available at the time of this experiment.

It has been shown that nonlinear surface registration of cortical folds can significantly improve Brodmann area overlap across different subjects [56, 181, 182] compared with volumetric registration. Registering the *ex-vivo* surfaces is more difficult than *in-vivo* surfaces because the reconstructed volumes are extremely noisy due to the distortions introduced by the histology, resulting in noisy geometric features, as shown in Fig. 4-6.

We consider two strategies for aligning Brodmann areas. For both strategies, we will use 10 iterations of smoothing for Spherical Demons as it proved reasonable in the previous set of experiments. The first strategy involves co-registering the 10 *ex-vivo* surfaces using cortical geometry by repeatedly building an atlas and registering the surfaces to the atlas, similar to the previous experiment on cortical parcellation. We use either Spherical Demons



(a) Left hemisphere parcellation

(a) Right hemisphere parcellation

Figure 4-5: (a) Dice measure for each structure in the left hemisphere. (b) Dice measure for each structure in the right hemisphere. Black columns correspond to FreeSurfer. White columns correspond to Spherical Demons. * indicates structures where Spherical Demons shows statistically significant improvement over FreeSurfer (FDR = 0.05). No structure exhibit significant decrease in accuracy.

or FreeSurfer for registration. We refer to the co-registration using Spherical Demons and FreeSurfer as SD10 and FS10 respectively (10 refers to the number of subjects in the study, not the number of smoothing iterations).

The second strategy involves registering the 10 *ex-vivo* surfaces to the *in-vivo* “Buck-

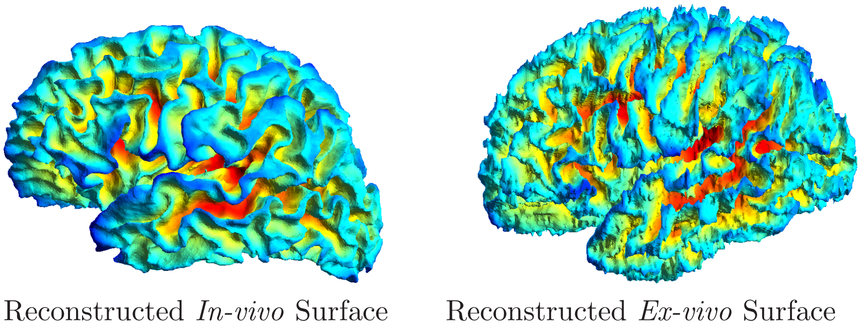

Reconstructed *In-vivo* Surface
Reconstructed *Ex-vivo* Surface

Figure 4-6: Left: example *in-vivo* surface used in the parcellation study. Right: example *ex-vivo* surface used in the Brodmann area study.

ner40" atlas, constructed from 40 *in-vivo* subjects, that is distributed with the FreeSurfer software. Once again, we use either Spherical Demons or FreeSurfer for the registration. We refer to the co-registration using Spherical Demons and FreeSurfer as SD40 and FS40 respectively.

To measure the quality of alignment of the Brodmann areas after cortical registration, we use an adaptation [182] of the modified Hausdorff distance [45]. For each pair of registered subjects, we project each Brodmann area from the first subject onto the second subject and compute the mean distance between the boundaries, measured on the original cortical surface of the second subject. We obtain a second measurement by projecting each Brodmann area from the second subject onto the first subject. Since we have 10 surfaces, we get 90 ordered pairs and 90 alignment errors for each Brodmann area.

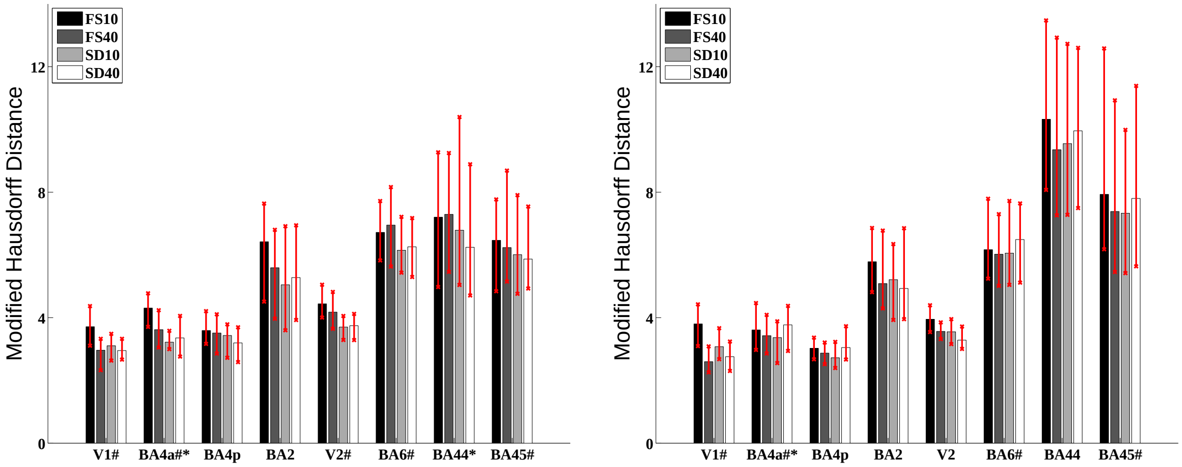
Table 4.2 reports the mean alignment errors for each Brodmann area and for each method. The lowest errors for each Brodmann area are shown in **bold**. We see that for almost all Brodmann areas, the best alignment come from SD10 or SD40. Similarly, Fig. 4-7 shows the median alignment error for each Brodmann area. The error bars indicate the lower and upper quartile alignment errors.

We use permutation testing to evaluate statistical significance of the results. We cannot use the t-test because the 90 alignment errors are correlated - since the subjects are co-registered together, good alignment between subjects 1 and 2 and between subjects 2 and 3 necessarily implies a higher likelihood of good alignment between subjects 1 and 3.

The tests show that SD10 is better than FS10 and SD40 is slightly better than FS40. SD10 and SD40 are comparable. Compared with FS10, SD10 improves the median alignment errors of 5 (4) Brodmann areas on the right (left) hemisphere (FDR = 0.05) and no structure gets worse. Compared with FS40, SD40 statistically improves the alignment of 2 (1) Brodmann areas on the right (left) hemisphere (FDR = 0.05) with no structure getting worse. Permutation tests on the mean alignment errors show similar results, except that FS40 performs better than SD40 for BA4p on the left hemisphere when using the mean statistic. These results suggest that the Spherical Demons algorithm is at least as accurate

Table 4.2: Mean alignment errors of Brodmann areas in *mm* for the four registration methods. Lowest errors are shown in **bold**. SD refers to Spherical Demons; FS refers to FreeSurfer.

	Right Hemisphere							
	V1	BA4a	BA4p	BA2	V2	BA6	BA44	BA45
FS10	3.8	4.4	3.8	6.3	4.6	7.0	7.4	6.8
FS40	**2.9**	3.8	3.6	5.6	4.2	7.1	7.6	6.9
SD10	3.1	**3.3**	3.3	**5.4**	**3.7**	**6.4**	7.7	6.4
SD40	3.0	3.4	**3.2**	5.5	3.8	**6.4**	**6.8**	**6.3**
	Left Hemisphere							
	V1	BA4a	BA4p	BA2	V2	BA6	BA44	BA45
FS10	3.8	3.8	3.1	5.9	4.0	6.5	11.5	9.9
FS40	**2.7**	3.6	2.9	5.7	3.6	**6.3**	10.5	9.2
SD10	3.2	**3.4**	**2.8**	**5.5**	3.5	6.4	**10.4**	**8.6**
SD40	2.8	3.8	3.7	5.6	**3.4**	6.6	10.7	9.0



(a) Right Hemisphere

(b) Left Hemisphere

Figure 4-7: Median alignment errors of Brodmann areas in *mm* for the four registration methods. The error bars indicate the upper and lower quartile alignment errors. "#" indicates that the median errors of SD10 are statistically lower than those of FS10 (FDR = 0.05). "*" indicates SD40 outperforms FS40. For no Brodmann area does FreeSurfer outperform Spherical Demons.

as FreeSurfer in aligning cortical folds and Brodmann areas.

4.4 Discussion

The Demons algorithms [152, 168] discussed in Section 4.1 and the Spherical Demons algorithm proposed in this chapter use a regularization term that modulates the final deformation. Motivated by [29, 32], the Diffeomorphic Demons algorithm [168] admits a fluid prior on the velocity fields. This involves smoothing the velocity field updates *before* computing the exponential map to obtain the displacement field updates to be composed with the current transformation. The resulting algorithm is very similar to the fast implementation [29] of Christensen's well-known fluid registration algorithm [34], except that Christensen's algorithm does not employ a higher-order update method like Gauss-Newon. The Spherical Demons algorithm can similarly incorporate a fluid prior by smoothing the velocity field $\vec{v}^{(t)}$ in Eq. (4.28) before computing the exponential map to obtain the displacement updates $\exp(\vec{v}^{(t)})$.

An alternative interpretation of the smoothing implementation of Christensen's algorithm comes from choosing a different metric for computing the gradient from the derivatives [21]. The choice of the metric also arises in our problem when computing the gradient as discussed in Appendix B.1.3. This suggests that the Spherical Demons algorithm can incorporate a fluid prior by modifying the Gauss-Newton update step Eq. (4.28). Unfortunately, this process introduces coupling among the vertices resulting in the loss of the speed-up advantage of Spherical Demons (see for example the derivations of [74]). The exploration of the performance of the different fluid prior implementations is outside the scope of this thesis.

Because the tools of differential geometry are general, the Spherical Demons algorithm can be in principle extended to arbitrary manifolds, besides the sphere. One challenge is that the definition of coordinate charts for an arbitrary manifold is more difficult than that for the sphere. Approaches of utilizing the embedding space [33] or the intrinsic properties of manifolds [94] are promising avenues of future work.

4.5 Summary

In this chapter, we presented the fast Spherical Demons algorithm for registering spherical images. We showed that the two-step optimization of the Demons algorithm can also be applied on the sphere. By utilizing the one parameter subgroup of diffeomorphisms, the resulting deformation is invertible. A clear future challenge is to take into account the original metric properties of the cortical surface in the registration process, as demonstrated in previously proposed registration methods [59, 154]. We tested the algorithm extensively in two different applications and showed that the accuracy of the algorithm compares favorably with the widely used FreeSurfer registration algorithm [59] while offering more than one order of magnitude speedup.

While we have greatly speeded up registration, the registration parameters still have to be manually tuned. In the next chapter, we will present a framework that automatically learns the parameters of registration cost functions. Leveraging on the one parameter subgroup of diffeomorphisms presented in this chapter allows the framework to be tractable in practice.

Chapter 5

Learning Task-Optimal Registration Cost Functions

In medical image analysis, registration is rarely the final goal, but instead the resulting alignment is used in other tasks, such as image segmentation or group analysis. The parameters of the registration cost function – for example, the tradeoff between the image similarity and regularization terms – are typically determined manually through inspection of the image alignment and then fixed for all applications. However, it is unclear that the same parameters are optimal for different applications.

In this chapter, we propose a principled approach to leveraging the application context to effectively regularize the ill-posed problem of image registration. Our method learns the parameters of any smooth family of registration cost functions with respect to a specific task. The key idea is to introduce a second layer of optimization over and above the usual registration. This second layer of optimization traverses the space of local minima, selecting registration parameters that result in good local minima as measured by the performance of the specific application in a training data set. We instantiate the framework for the alignment of hidden labels whose extent is not necessarily well-predicted by local image features. We apply the resulting algorithm to localize cytoarchitectural and functional regions based only on macroanatomical cortical folding information and achieve state-of-the-art localization results in both histological and functional Magnetic Resonance Imaging (fMRI) data sets.

This chapter is organized as follows. In the next section, we introduce the task-optimal registration framework. We specialize the framework to align hidden labels in Section 5.2. We present localization experiments in Section 5.3 and discuss outstanding issues in Section 5.4. This chapter extends a previously presented conference article [181] and contains detailed derivations, discussions and experiments that were left out in the conference version.

5.1 Task-Optimal Framework

In this section, we present the task-optimal registration framework for learning the parameters of a registration cost function. Given an image I, let $f(w, \Gamma)$ denote a smooth registration cost function, with parameters w and a transformation Γ. For example,

$$f(w = \{\lambda, T\}, \Gamma) = \lambda \, \mathrm{Dissim}(T, I \circ \Gamma) + \mathrm{Reg}(\Gamma), \tag{5.1}$$

where T is the template image and λ is the tradeoff between the image dissimilarity measure and the regularization on the transformation Γ. f is therefore also a function of the image I, which we suppress for conciseness. The optimal transformation Γ^* minimizes the cost function for a given set of parameters w:

$$\Gamma^*(w) = \operatorname*{argmin}_{\Gamma} f(w, \Gamma). \tag{5.2}$$

We emphasize that Γ^* is a function of w since a different set of parameters w will result in a different solution to Eq. (5.2) and thus will effectively define a different image coordinate system.

The resulting deformation Γ^* is used to warp the input image or is itself used for further tasks, such as image segmentation or voxel-based morphometry. We assume that the task performance can be measured by a smooth cost function g, so that a smaller value of $g(\Gamma^*(w))$ corresponds to better task performance. g is typically a function of additional input data associated with a subject (e.g., manual segmentation labels if the task is automatic segmentation), although we suppress this dependency in the notation for conciseness. This auxiliary data is only available in the training set; g cannot be evaluated for the new image.

Given a set of training subjects, let $\Gamma_n^*(w)$ denote the solution of Eq. (5.2) for training subject n for a fixed set of parameters w and $g_n(\Gamma_n^*(w))$ denote the task performance for training subject n using the deformation $\Gamma_n^*(w)$ and other information available for the n-th training subject. A different set of parameters w would lead to different task performance $g_n(\Gamma_n^*(w))$. We seek the parameters w^* that generalize well to a new subject: registration of a new subject with w^* yields the transformation $\Gamma^*(w^*)$ with a small task-specific cost $g(\Gamma^*(w^*))$. To solve this functional approximation problem [52], we use regularized risk minimization. Let $\mathrm{Reg}(w)$ denote regularization on w. We define

$$G(w) \triangleq \sum_{n=1}^{N} g_n(\Gamma_n^*(w)) + \mathrm{Reg}(w) \tag{5.3}$$

and seek

$$w^* = \operatorname*{argmin}_w G(w). \tag{5.4}$$

The optimization is difficult because evaluating the cost function G for a particular set of parameters w requires performing N different registrations!

5.1.1 Characterizing the Space of Local Minima

In this section, we provide theoretical characterizations of the optimization problem in Eq. (5.3). If $\Gamma^*(w)$ is defined strictly to be the global registration optimum, then $\Gamma^*(w)$ is clearly not a smooth function of w, since a small change in w can result in a large change in the global registration optimum. This definition is also impractical, since the global optima for most nonlinear optimization problem cannot be guaranteed in practice. Instead, we relax the definition of $\Gamma^*(w)$ to be a local minimum of the registration cost function for fixed values of w. Here, we derive conditions under which $\Gamma^*(w)$ is a locally smooth function of w, so we can employ gradient descent to optimize Eq. (5.3) with respect to w.

Let $\Gamma^*(w_0)$ denote a local minimum of the registration cost function for a fixed $w = w_0$. Suppose we perturb w by an infinitestimally small δw so that $\Gamma^*(w_0)$ is no longer the registration local minimum for $w = w_0 + \delta w$. We consider two representations of deformation in the optimization and provide the analysis of the change in local minimum for both representations.

Additive deformation models arise when the space of deformations is a vector space, such as the space of displacement fields or positions of B-spline control points. At each iteration of the registration algorithm, deformation updates are *added* to the current deformation estimates. The additive model is general and applies to many non-convex, smooth optimization problems outside of registration. Most registration algorithms can in fact be modeled with the additive framework.

In some registration algorithms, including the one used in this chapter, it is more natural to represent deformation changes through composition rather than addition [17, 147, 168]. For example, in the diffeomorphic variants of the Demons algorithm [168, 183, 186] discussed in the previous chapter, the diffeomorphic transformation Γ is represented as a dense displacement field. At each iteration, the transformation update is restricted to be a member of the one parameter subgroup of diffeomorphisms parameterized by a stationary velocity field. The diffeomorphic transformation update is then *composed* with, rather than added to, the current estimate of the transformation, to ensure that the resulting transformation is diffeomorphic.

(1) ADDITION. Let $\Gamma^*(w_0 + \delta w) = \Gamma^*(w_0) + \delta\Gamma^*(w_0, \delta w)$ denote the new locally optimal deformation for the new set of parameters $w_0 + \delta w$. The following proposition characterizes the existence and uniqueness of $\delta\Gamma^*(w_0, \delta w)$ as δw is varied. In particular, we show that under some mild conditions, $\delta\Gamma^*(w_0, \delta w)$ is a well-defined smooth function in the neighborhood of $(w_0, \Gamma^*(w_0))$. In the remainder, we use ∂_x, ∂_x^2, and $\partial_{x,y}^2$ to denote the corresponding partial derivatives.

Proposition 1. *If the Hessian*[1] $\partial_\Gamma^2 f(w_0, \Gamma)$ *is positive definite at* $\Gamma = \Gamma^*(w_0)$*, then there exists an* $\epsilon > 0$*, such that for all* δw*,* $\|\delta w\| < \epsilon$*, a unique continuous function* $\delta\Gamma^*(w_0, \delta w)$ *exists such that* $\delta\Gamma^*(w_0, 0) = 0$*. Furthermore,* $\delta\Gamma^*$ *has the same order of smoothness as* f*.*

Proof. We define $h(w, \Gamma) \triangleq \partial_\Gamma f(w, \Gamma)$. Since $\Gamma^*(w_0)$ is a local minimum of $f(w_0, \Gamma)$, we have

$$h(w, \Gamma)\Big|_{w_0, \Gamma^*(w_0)} = \partial_\Gamma f(w, \Gamma)\Big|_{w_0, \Gamma^*(w_0)} = 0. \tag{5.5}$$

At $(w_0, \Gamma^*(w_0))$, the Hessian matrix $\partial_\Gamma^2 f(w_0, \Gamma) = \partial_\Gamma h(w, \Gamma)$ is positive definite by the assumption of the proposition and is therefore invertible. By the Implicit Function Theorem [134], there exists an $\epsilon > 0$, such that for all δw, $\|\delta w\| < \epsilon$, there is a unique continuous function $\delta\Gamma^*(w_0, \delta w)$, such that $h(w_0 + \delta w, \Gamma^*(w_0) + \delta\Gamma^*(w_0, \delta w)) = 0$ and $\delta\Gamma^*(w_0, 0) = 0$. Furthermore, $\delta\Gamma^*(w_0, \delta w)$ has the same order of smoothness as h.

Because the Hessian of f is smooth and the eigenvalues of a matrix depend continuously on the matrix [162], there exists a small neighborhood around $(w_0, \Gamma^*(w_0))$ in which the eigenvalues of $\partial_\Gamma^2 f(w, \Gamma)$ are all greater than 0. Since both sufficient conditions for a local minimum are satisfied (zero gradient and positive definite Hessian), $\Gamma^*(w_0) + \delta\Gamma^*(w_0, dw)$ is indeed a new local minimum close to $\Gamma^*(w_0)$. □

At $(w_0, \Gamma^*(w_0))$, the Hessian $\partial_\Gamma^2 f(w_0, \Gamma)$ is positive semi-definite, so the positive definite condition in Proposition 1 is not too restrictive. Unfortunately, degeneracies may arise for local minima with a singular Hessian. For example, let Γ be the 1×2 vector $[a\ b]$ and $f(\Gamma, w) = (ab - w)^2$. Then for any value of w, there is an infinite number of local minima $\Gamma^*(w)$ corresponding to $ab = w$. Furthermore, the Hessian at *any* local minimum is singular. In this case, there is an infinite number of local minima near the current local minimum $\Gamma^*(w_0)$, i.e., $\delta\Gamma^*(w_0, \delta w)$ is not a well-defined function and the gradient is not defined. Consequently, the parameters w of local registration minima whose Hessians are singular are also local minima of the task-optimal cost function Eq. (5.3).

(2) COMPOSITION. Let $\Gamma^*(w_0)$ be the registration local minimum at w_0 and $\delta\Gamma(v)$

[1] Here, we assume that the transformation Γ is finite dimensional, such as the parameters of affine transformations, positions of spline control points or dense displacement field defined on the voxels or vertices of the image domain.

denote an update transformation parameterized by v, so that $\delta\Gamma(0)$ corresponds to the identity transform. For example, v could be a stationary [168, 183, 186] or non-stationary [21] velocity field parameterization of diffeomorphism or positions of spline control points [135]. In the composition model, $\Gamma^*(w_0)$ is a local minimum if and only if there exists an $\epsilon > 0$, such that $f(w_0, \Gamma^*(w_0)) < f(w_0, \Gamma^*(w_0) \circ \delta\Gamma(v))$ for all values of $\|v\| < \epsilon$.

Let $\Gamma^*(w_0) \circ \delta\Gamma(v^*(w_0, \delta w))$ denote the new locally optimal deformation for the new parameters $w_0 + \delta w$. In general, there might not exist a single update transformation $\delta\Gamma(v^*(w_0, \delta w))$ that leads to a new local minimum under a perturbation of the parameters w, so that there is no correponding version of Proposition 1 for the general composition model. However, in the special case of the composition of diffeomorphisms model [168, 183, 186] employed in this and the previous chapters, the following proposition characterizes the existence and uniqueness of $v^*(w_0, \delta w)$ as δw is varied.

Proposition 2. *If the Hessian $\partial_v^2 f(w_0, \Gamma^*(w_0) \circ \delta\Gamma(v))$ is positive definite at $v = 0$, then there exists an $\epsilon > 0$, such that for all δw, $\|\delta w\| < \epsilon$, a unique continuous function $v^*(w_0, \delta w)$ exists, such that $v^*(w_0, \delta w)$ is the new local minimum for parameters $w_0 + \delta w$ and $v^*(w_0, 0) = 0$. Furthermore, $v^*(w_0, \delta w)$ has the same order of smoothness as f.*

Proof. The proof is similar to that of Proposition 1, but with a twist. The details are found in Appendix C.1. □

Just like in the case of the additive deformation model, local registration minima that do not satisfy the conditions of Proposition 2 are also local minima of the regularized task performance G. In the next section, we derive exact and approximate gradients of G.

5.1.2 Optimizing Registration Parameters w

We now discuss the optimization of the regularized task performance G.

(1) ADDITION. In the previous section, we have shown that at $(w_0, \Gamma^*(w_0))$ with a positive definite Hessian, $\delta\Gamma^*(w_0, \delta w)$ is a smooth well-defined function such that $\Gamma^*(w_0) + \delta\Gamma^*(w_0, \delta w)$ is the new local minimum at $w_0 + \delta w$. Therefore, we can compute the derivatives of $\Gamma^*(w)$ with respect to w at w_0, allowing us to traverse a curve of local optima, finding values of w that improve the task-specific cost function for the training images. We first perform a Taylor expansion of $\partial_\Gamma f(w, \Gamma)$ at $(w_0, \Gamma^*(w_0))$:

$$\partial_\Gamma f(w, \Gamma)\Big|_{w_0+\delta w, \Gamma^*(w_0)+\delta\Gamma} = \Big[\partial_\Gamma^2 f(w, \Gamma)\delta\Gamma + \partial_{w,\Gamma}^2 f(w, \Gamma)\delta w + O(\delta w^2, \delta\Gamma^2)\Big]\Big|_{w_0, \Gamma^*(w_0)}, \tag{5.6}$$

where we dropped the term $\partial_\Gamma f(w,\Gamma)\Big|_{w_0,\Gamma^*(w_0)} = 0$. For $\delta\Gamma = \delta\Gamma^*(w_0,\delta w)$, the left-hand side is equal to 0 and we can write

$$\delta\Gamma^*(w_0,\delta w) = \Big[-\left(\partial^2_\Gamma f(w,\Gamma)\right)^{-1}\partial^2_{w,\Gamma} f(w,\Gamma)\delta w + O(\delta w^2,\delta\Gamma^2)\Big]\Bigg|_{w_0,\Gamma^*(w_0)}. \tag{5.7}$$

Therefore by taking the limit $\delta w \to 0$, we get

$$\partial_w\Gamma^*(w_0) \triangleq \partial_{(\delta w)}\delta\Gamma^*(w_0,\delta w)\Bigg|_{\delta w=0} = -\left(\partial^2_\Gamma f(w,\Gamma)\right)^{-1}\partial^2_{w,\Gamma} f(w,\Gamma)\Bigg|_{w_0,\Gamma^*(w_0)}. \tag{5.8}$$

Eq. (5.8) tells us the direction of change of the local minimum at $(w_0,\Gamma^*(w_0))$. In practice, the matrix inversion in Eq. (5.8) is computationally prohibitive for high-dimensional warps Γ. Here, we consider a simplification of Eq. (5.8) by setting the Hessian to be the identity:

$$\partial_w\Gamma^* \approx -\partial^2_{w,\Gamma} f(w,\Gamma)\Big|_{w_0,\Gamma^*(w_0)}. \tag{5.9}$$

Since $-\partial_\Gamma f$ is the direction of gradient descent of the cost function Eq. (5.2), we can interpret Eq. (5.9) as approximating the new local minimum to be in the same direction as the *change* in the direction of gradient descent as w is perturbed.

Differentiating the cost function in Eq. (5.3), using the chain rule, we obtain

$$\begin{aligned}
\partial_w G &= \partial_w\left(\sum_{n=1}^N g_n(\Gamma^*_n(w)) + \mathrm{Reg}(w)\right) && (5.10)\\
&= \sum_{n=1}^N \left[\partial_{\Gamma^*_n} g_n\right]\left[\partial_w \Gamma^*_n\right] + \partial_w \mathrm{Reg}(w) && (5.11)\\
&= -\sum_{n=1}^N \left[\partial_{\Gamma^*_n} g_n\right]\partial^2_{w,\Gamma} f_n(w,\Gamma)\Big|_{w,\Gamma^*_n(w_0)} + \partial_w \mathrm{Reg}(w). && (5.12)
\end{aligned}$$

We note the subscript n on f indicates the dependency of the registration cost function on the n-th training image.

(2) COMPOSITION. In the previous section, we have shown that at $(w_0,\Gamma^*(w_0))$, assuming the conditions of Proposition 2 are true, $v^*(w_0,\delta w)$ is a smooth well-defined function such that $\Gamma^*(w_0)\circ\delta\Gamma(v^*(w_0,\delta w))$ is the new local minimum. Therefore, we can compute

```
Algorithm 3: Task-Optimal Registration
  Data: A set of training images $\{I_n\}$
  Result: Parameters $w$ that minimizes the regularized task performance $G$ (see
          Eq. (5.3))
  Initialize $w^0$.
  repeat
      Step 1. Given current values of $w$, estimate $\Gamma_n^*(w) = \text{argmin}_{\Gamma_n} f_n(w, \Gamma_n)$, i.e.,
      perform registration of each training subject $n$.
      Step 2. Given current estimates $(w, \{\Gamma_n(w)\})$, compute the gradient $\partial_w G$ using
      either
          1. Eq. (5.12) via $\partial_w \Gamma^*$ in Eq. (5.9) for the addition model or
          2. Eq. (5.16) via $\partial_w v^*$ in Eq. (5.14) for the composition model.
      Step 3. Perform line search in the direction opposite to $\partial_w G$ [125].
  until convergence ;
```

the derivatives of v^*. As before, by performing a Taylor expansion, we obtain

$$\partial_w v^* = -\left(\partial^2_{v_1,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2))\right)^{-1} \partial^2_{w,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_2))\Big|_{w_0, v_1=0, v_2=0} \tag{5.13}$$

$$\approx -\partial^2_{w,v} f(w, \Gamma^* \circ \delta\Gamma(v))\Big|_{w_0, v=0}. \tag{5.14}$$

Appendix C.2 provides the detailed derivations. Differentiating the cost function in Eq. (5.3), using the chain rule, we get

$$\frac{\partial G}{\partial w} = \sum_{n=1}^{N} \left[\frac{\partial g_n(\Gamma_n^* \circ \delta\Gamma(v^*))}{\partial v^*}\right] \left[\frac{\partial v^*}{\partial w}\right]\Bigg|_{v^*=0} + \frac{\partial}{\partial w}\text{Reg}(w) \tag{5.15}$$

$$= -\sum_{n=1}^{N} \left[\frac{\partial g_n(\Gamma_n^* \circ \delta\Gamma(v^*))}{\partial v^*}\right] \partial^2_{w,v} f_n(w, \Gamma_n^* \circ \delta\Gamma(v))\Big|_{w, v=v^*=0} + \frac{\partial}{\partial w}\text{Reg}(w). \tag{5.16}$$

Once again, the subscript n on f indicates the dependency of the registration cost function on the n-th training image.

Algorithm 3 summarizes the method for learning the task-optimal registration parameters. Each line search involves evaluating the cost function G multiple times, which in turn requires registering the training subjects, resulting in a computationally intensive process. However, since we are initializing from a local optimum, for a small change in w, each registration converges quickly.

Since nonlinear registration is dependent on initialization, the current estimates $(w, \Gamma^*(w))$, which were initialized from previous estimates, might not be achievable when initializing the registration with the identity transform. The corresponding parameters w might therefore *not* generalize well to a new subject, which are typically initialized with the identity transform. To avoid this problem, after every few iterations, we re-register the training images initialized with the identity transform and check that G is better than the current best value of G computed with initialization from the identity transform.

5.2 Learning wSSD for Hidden Label Alignment

We now instantiate the task-optimal registration framework for localizing hidden labels in images. We demonstrate two schemes for either (1) learning the weights of the weighted Sum of Squared Differences (wSSD) family of registration cost functions or (2) estimating an optimal template for localizing these hidden labels. We emphasize that the optimal template is not necessarily the average of the training images, since the goal is not to align image intensity across subjects, but to localize hidden labels.

As before, we instantiate the framework for spherical images that represent the cortex. Suppose we have a set of spherical training images $\{I_n\}$ with some underlying ground truth structure manually labeled or obtained from another imaging modality. From these training images and labels, we seek to estimate a registration cost function and an image template T with corresponding training labels. We define our task as aligning a new image to the image template and predicting the boundary of the hidden structure in the new subject by transferring the training labels to the new subject.

5.2.1 Instantiating Registration Cost Function f

To register a given image I_n to the template image T, we define the following cost function:

$$\begin{aligned} & f_n(w = \{\{\lambda_i\}, T\}, \Gamma_n) \\ & = \sum_i \lambda_i^2 \left[T(x_i) - I_n(\Gamma_n(x_i))\right]^2 + \sum_i \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \left(\frac{\|\Gamma_n(x_i) - \Gamma_n(x_j)\| - d_{ij}}{d_{ij}} \right)^2, \end{aligned} \quad (5.17)$$

where transformation Γ_n maps a point x_i on the sphere S^2 to another point $\Gamma_n(x_i) \in S^2$. The first term corresponds to the wSSD image similiarity. The second term is a regularization that penalizes relative metric distortion in the transformation Γ_n. $\mathcal{N}_i$ is a predefined neighborhood around vertex i and d_{ij} is the original distance between the neighbors $d_{ij} = \|x_i - x_j\|$ [180]. The weights $\{\lambda_i\}$ are generalizations of the tradeoff parameter λ, allowing for a spatially-varying tradeoff between the image dissimilarity term and regularization: a higher weight λ_i^2 corresponds to placing more emphasis on matching the template image at spatial location x_i relative to the regularization.

In this work, we consider either learning the weights λ_i^2 or the template T for localizing

Brodmann Area (BA) labels or functional labels by aligning cortical folding pattern. Optimizing λ_i^2 leads to placing spatially varying importance on matching different cortical folds with the aim of aligning the underlying cytoarchitectonics or function. Optimizing T corresponds to learning a cortical folding template that is optimal for localizing the underlying cytoarchitectonics or functional labels of the training subjects.

We choose to represent the transformation Γ_n as a composition of diffeomorphic warps $\{\Phi_k\}$ parameterized by a stationary velocity field $\vec{v}$, so that $\Gamma_n = \Phi_1 \circ \cdots \circ \Phi_K$ [168, 183, 186]. We note that our choice of regularization is different from the implicit hierarchical regularization used in Spherical Demons [183] since the Demons regularization is not compatible with our derivations from the previous section. Instead of the efficient 2-Step Spherical Demons algorithm, we will use steepest descent. The resulting registration algorithm is still relatively fast, requiring about 15 min for registering full-resolution meshes with more than 100k vertices, compared with 5 min of computation for Spherical Demons on a Xeon 2.8GHz single processor machine. Just like in the previous chapter, we denote $\Phi(x) = \exp(\vec{v})(x)$.

5.2.2 Optimizing Registration Cost Function f

To register subject n to the template image T for a fixed set of parameters w, let Γ_n^0 be the current estimate of Γ_n^*. We seek an update transformation $\exp(\vec{v})$ parameterized by a stationary velocity field $\vec{v}$:

$$\begin{aligned} f_n(w, \Gamma_n^0 \circ \exp(\vec{v})) = {} & \sum_i \lambda_i^2 \left[T(x_i) - I_n(\Gamma_n^0 \circ \exp(\vec{v})(x_i)) \right]^2 \\ & + \sum_i \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} \left(\frac{\|\Gamma_n^0 \circ \exp(\vec{v})(x_i) - \Gamma_n^0 \circ \exp(\vec{v})(x_j)\| - d_{ij}}{d_{ij}} \right)^2 . \end{aligned} \tag{5.18}$$

Let $\vec{v}_i$ be the velocity vector tangent to vertex x_i, and $\vec{v} = \{\vec{v}_i\}$ be the entire velocity field. We adopt the techniques in the Spherical Demons algorithm [183] described in the previous chapter, to differentiate Eq. (5.18) with respect to $\vec{v}$, evaluated at $\vec{v} = 0$. Using the fact that the differential of $\exp(\vec{v})$ at $\vec{v} = 0$ is the identity [115], i.e., $[D \exp(0)]\, \vec{v} = \vec{v}$, we conclude that a change in velocity $\vec{v}_i$ at vertex x_i does not affect $\exp(\vec{v})(x_n)$ for $n \neq i$ up to the first order derivatives. Defining $\nabla I_n(\Gamma_n^0(x_i))$ to be the 1×3 spatial gradient of the warped image $I_n(\Gamma_n^0(\cdot))$ at x_i and $\nabla \Gamma_n^0(x_i)$ to be the 3×3 Jacobian matrix of Γ_n^0 at x_i, we get the 1×3 derivative

$$\begin{aligned} \partial_{\vec{v}_i} f_n(w, \Gamma_n^0 \circ \exp(\vec{v}))\Big|_{\vec{v}=0} = {} & -2\lambda_i^2 \left[T(x_i) - I_n\left(\Gamma_n^0(x_i)\right) \right] \left[\nabla I_n \left(\Gamma_n^0(x_i) \right) \right] \\ & + 2 \sum_{j \in \mathcal{N}_i} \left(\frac{1}{|\mathcal{N}_i|} + \frac{1}{|\mathcal{N}_j|} \right) \left(\frac{\|\Gamma_n^0(x_i) - \Gamma_n^0(x_j)\| - d_{ij}}{d_{ij}^2 \|\Gamma_n^0(x_i) - \Gamma_n^0(x_j)\|} \right) \left[\Gamma_n^0(x_i) - \Gamma_n^0(x_j) \right]^T \nabla \Gamma_n^0(x_i). \end{aligned} \tag{5.19}$$

We can perform gradient descent of the registration cost function f_n using Eq. (5.19) to obtain Γ_n^*, which can be used to evaluate the regularized task performance G to be described in the next section. We also note that Eq. (5.19) instantiates $\partial_{\vec{v}} f_n$ within the task-optimal gradient Eq. (5.16) for this application.

5.2.3 Instantiating Regularized Task Performance G

We represent the hidden labels in the training subjects as signed distance transforms on the sphere $\{L_n\}$ [95]. We consider a pairwise approach, where we assume that the template image T has a corresponding labels with distance transform L_T and set the task-specific cost function to be

$$g_n(\Gamma_n^*) = \sum_i \left[L_T\left(x_i\right) - L_n\left(\Gamma_n^*\left(x_i\right)\right)\right]^2. \tag{5.20}$$

A low value of g_n indicates good alignment of the hidden label maps between the template and subject n, suggesting good prediction of the hidden label.

We experimented with a prior that encourages spatially constant weights and template, but did not find that the regularization lead to improvements in the localization results. In particular, we consider the following smoothness regularization on the registration parameters depending on whether we are optimizing for the weights λ_i or the template T:

$$\mathrm{Reg}(\{\lambda_i\}) = \sum_i \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (\lambda_i^2 - \lambda_j^2)^2, \tag{5.21}$$

$$\mathrm{Reg}(T) = \sum_i \frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i} (T(x_i) - T(x_j))^2. \tag{5.22}$$

One possible reason for this lack of improvement is that the re-registration after every few line searches helps to regularize against bad parameter values. In the experiments that follow, we will discard the regularization term of the registration parameters (i.e., set $\mathrm{Reg}(w) = 0$), although designing a more useful regularization could potentially improve the results.

5.2.4 Optimizing Task Performance G

To optimize the task performance G over the set of parameters w, we have to instantiate the task-optimal gradient specified in Eq. (5.16). We first compute the derivative of the task-specific cost function with respect to the optimal update $\vec{v}^*$. Once again, we represent $\vec{v}^*$ as the collection $\{\vec{v}_i^*\}$, where $\vec{v}_i^*$ is a velocity vector at vertex x_i. Defining $\nabla L_n(\Gamma_n^*(x_i))^T$ to be the 1×3 spatial gradient of the warped distance transform of the n-th subject $L_n(\Gamma_n^*(\cdot))$

at x_i, we get the 1×3 derivative

$$\partial_{\vec{v}_i^*} g_n(\Gamma_n^* \circ \exp(\vec{v}^*))\Big|_{\vec{v}^*=0} = -2 \left[L_T\left(x_i\right) - L_n\left(\Gamma_n^*\left(x_i\right)\right)\right] \left[\nabla L_n\left(\Gamma_n^*\left(x_i\right)\right)\right]. \tag{5.23}$$

Weight Update. To update the weights $\{\lambda_j\}$ of the wSSD, we compute the derivative of the registration local minimum update $\vec{v}^*$ with respect to the weights. Using the approximation in Eq. (5.14), we obtain the 3×1 derivative of the velocity update $\vec{v}$ with respect to the weights of the wSSD cost function:

$$\partial_{\lambda_k} \vec{v}_i^* \approx -\partial^2_{\lambda_k, \vec{v}_i} f_n(\{\lambda_j\}, \Gamma_n^* \circ \exp(\vec{v}))\Big|_{\{\lambda_j\}, \vec{v}=0} \tag{5.24}$$

$$= -\partial_{\vec{v}_i} \partial_{\lambda_k} f_n(\{\lambda_j\}, \Gamma_n^* \circ \exp(\vec{v}))\Big|_{\{\lambda_j\}, \vec{v}=0} \tag{5.25}$$

$$= -\partial_{\vec{v}_i} 2\lambda_k \left[T(x_k) - I_n(\Gamma_n^* \circ \exp(\vec{v})(x_k))\right]^2 \Big|_{\{\lambda_j\}, \vec{v}=0} \tag{5.26}$$

$$= 4\lambda_k \left[T(x_k) - I_n(\Gamma_n^*(x_k))\right] \nabla I_n\left(\Gamma_n^*\left(x_k\right)\right) \delta(k, i). \tag{5.27}$$

Here $\delta(k, i) = 1$ if $k = i$ and is zero otherwise. Since Eq. (5.27) is in the same direction as the first term of the gradient descent direction of registration (negative of Eq. (5.19)), increasing λ_k^2 will improve the intensity matching of vertex x_k of the template. Substituting Eq. (5.27) and Eq. (5.23) into Eq. (5.16) provides the gradient for updating the weights of the wSSD cost function.

Template Update. To update the template T used for registration, we compute the 3×1 derivative of the velocity update $\vec{v}$ with respect to the template T:

$$\partial_{T(x_k)} \vec{v}_i^* \approx -\partial^2_{T(x_k), \vec{v}_i} f_n(T, \Gamma_n^* \circ \exp(\vec{v}))\Big|_{T, \vec{v}=0} \tag{5.28}$$

$$= -\partial_{\vec{v}_i} \partial_{T(x_k)} f_n(T, \Gamma_n^* \circ \exp(\vec{v}))\Big|_{T, \vec{v}=0} \tag{5.29}$$

$$= -2\partial_{\vec{v}_i} \lambda_k^2 \left[T(x_k) - I_n(\Gamma_n^* \circ \exp(\vec{v})(x_k))\right] \Big|_{T, \vec{v}=0} \tag{5.30}$$

$$= 2\lambda_k^2 \left[T(x_k) - I_n(\Gamma_n^*(x_k))\right] \nabla I_n\left(\Gamma_n^*\left(x_k\right)\right) \delta(k, i). \tag{5.31}$$

Since Eq. (5.31) is in the same direction as the first term of the gradient descent direction of registration (negative of Eq. (5.19)), when $T(x_k)$ is larger than $I_n(\Gamma_n^*(x_k))$, increasing the value of $T(x_k)$ will warp vertex x_k of the template further along the direction of increasing intensity in the subject image. Conversely, if $T(x_k)$ is smaller than $I_n(\Gamma_n^*(x_k))$, decreasing the value of $T(x_k)$ will warp vertex x_k of the template further along the direction of decreasing intensity in the subject image. Substituting Eq. (5.31) and Eq. (5.23) into Eq. (5.16) provides the gradient for updating the template used for registration.

We note that we can in principle optimize both the weights $\{\lambda_i\}$ and the template T. However, in practice, we find that this does not lead to better localization possibly because of too many degrees of freedom, suggesting the need to design better regularization of the parameters or better optimization.

5.3 Experiments

We now present experiments on localizing Brodmann Areas (BAs) and fMRI-defined MT+ (V5) using macro-anatomical cortical folding in two different data sets. For both experiments, we compare the framework with using uniform weights [86, 168] and FreeSurfer [59].

5.3.1 Brodmann Area (BA) Localization

In this study, we consider the localization of Brodmann Areas (BAs) in the cortex of the 10 *ex-vivo* subjects described in Chapter 1.2.2. For an illustration of the BAs on the cortex, we refer the readers to Fig. 1-2(c). Here, we do not consider BA4a, BA4p and BA6 because they were only present in eight of the ten subjects.

As illustrated in Fig. 1-2(c) and discussed in multiple studies [4, 5, 56], we note that V1, V2 and BA2 are well-predicted by local cortical geometry, while BA44, BA45 and MT are not. For all the BAs however, a spherical morph of cortical folding was shown to improve their localization compared with only Talairach or nonlinear spatial normalization in the Euclidean 3D space [56]. Even though each subject has multiple BAs, we focus on each structure independently. This allows for an easier interpretation of the estimated parameters, such as the optimal template example we provide in Section 5.3.1.

We compare the following algorithms:

(a) **Task-Optimal.** We perform leave-two-out cross-validation to predict BA location. For each test subject, we use one of the remaining 9 subjects as the template subject and the remaining 8 subjects for training. When learning the weights of the wSSD, the weights $\{\lambda_j\}$ are globally initialized to 1 and the template T is fixed to the geometry of the template subject. When learning the cortical folding template T, the template is initialized to that of the template subject and the weights $\{\lambda_j\}$ are globally set to 1.

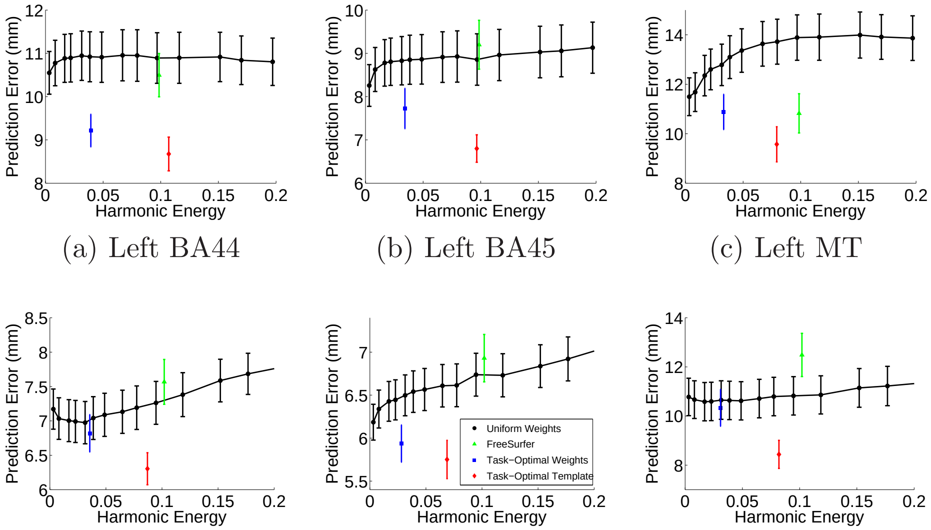
Once the weights or template are learned, we use them to register the test subject and predict the BA of the test subject by transferring the BA label from the template to the subject. We compute the symmetric mean Hausdorff distance between the boundary of the true BA and the predicted BA on the cortical surface of the test subject – smaller Hausdorff distance corresponds to better localization [45]. We consider all 90 possibilities of selecting the test subject and template, resulting in a total of 90 trials and 90 mean Hausdorff distances for each BA and for each hemisphere.

(b) **Uniform-Weights.** We repeat the process for the uniform-weight method that fixes the template T to the geometry of the template subject, and sets all the weights $\{\lambda_j\}$ to a global fixed value λ without training. We explore 14 different values of global weight λ, chosen such that the deformations range from rigid to flexible warps. For each BA and each hemisphere, we pick the *best* value of λ leading to the lowest mean Hausdorff distances. Because there is no cross-validation in selecting the weights, the uniform-weight method is using an unrealistic oracle-based version of the strategy proposed in [180].

(c) **FreeSurfer.** Finally, we use FreeSurfer [59] to register the 10 *ex-vivo* subjects to the FreeSurfer Buckner40 atlas, constructed from the MRI of 40 *in-vivo* subjects [62]. Once registered into this *in-vivo* atlas space, for the same 90 pairs of subjects, we can use the BAs of one *ex-vivo* subject to predict another *ex-vivo* subject. We note that FreeSurfer also uses the wSSD cost function, but a more sophisticated regularization that penalizes both metric and areal distortion. For a particular tradeoff between the similarity measure and regularization, the Buckner40 template consists of the empirical mean and variance of the 40 *in-vivo* subjects registered to template space. We use the reported FreeSurfer tradeoff parameters that were used to produce prior state-of-the-art BA alignment [56].

We note that both the task-optimal and uniform-weights methods use a pairwise registration framework, while FreeSurfer uses an atlas-based registration framework. Under the atlas-based framework, all the *ex-vivo* subjects are registered to an atlas. Therefore to use the BA of one subject to predict a second subject, we have to compose the deformations of the second subject to the atlas with the inverse deformation of the first subject to the atlas. Despite this additional source of error from composing two warps, it has been shown that with carefully constructed atlases, using the atlas-based strategy leads to better registration because of possible bias of the template in the pairwise registration framework [16, 65, 69, 86, 100].

We run the task-optimal and uniform-weights methods on a low-resolution subdivided icosahedron mesh containing 2,562 vertices, whereas FreeSurfer results were computed on high-resolution meshes of more than 100k vertices. In our implementation, training on 8 subjects takes on average 4hrs on a standard PC (AMD Opteron, 2GHz, 4GB RAM). Despite the use of the low-resolution mesh, we achieve state-of-the-art localization accuracy. We also emphasize that while training is computationally intensive, registration of a new subject only requires one minute of processing time since we are working with low-resolution meshes.



(a) Left BA44 (b) Left BA45 (c) Left MT

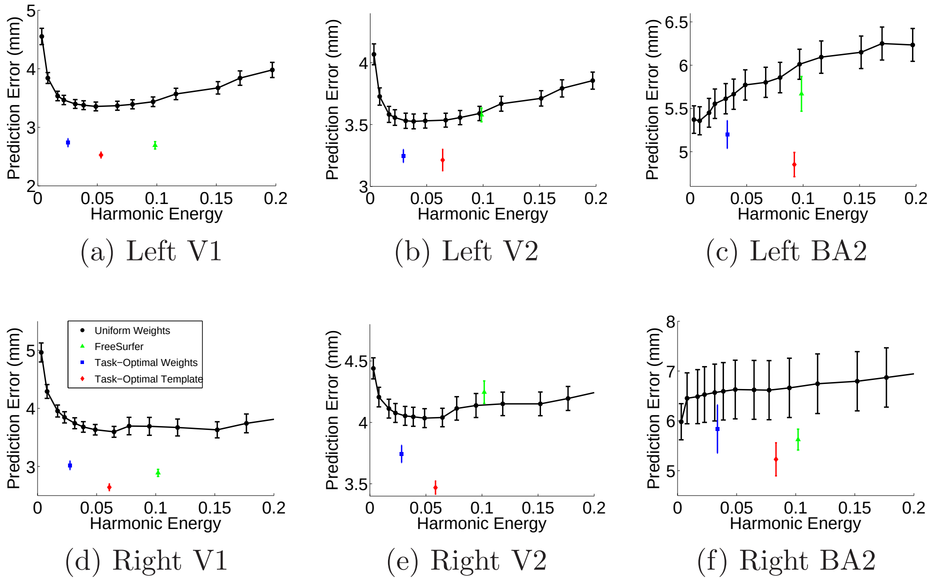
(d) Right BA44 (e) Right BA45 (f) Right MT

Figure 5-1: Mean Hausdorff distances over an entire range of harmonic energy for BA44, BA45 and MT. First row corresponds to left hemisphere. Second row corresponds to right hemipshere.

Quantitative Results

Fig. 5-1 displays the mean and standard errors from the 90 trials of leave-two-out. On average, task-optimal template performs the best, followed by task-optimal weights. The best uniform-weights and FreeSurfer have about the same performance. For the Broca's areas (BA44 and BA45) and MT, this is not surprising. Since local geometry poorly predicts these regions, by taking into account the final goal of aligning BAs instead of blindly aligning the cortical folds, our method achieves better BA localization. FreeSurfer and the uniform-weights method have similar performance because a better alignment of the cortical folds on a finer resolution mesh does not necessary improve the alignment of these areas.

Since local cortical geometry is predictive of V1, V2 and BA2, we expect the advantages of our framework to vanish. Surprisingly, as shown in Fig. 5-2, task-optimal template again achieve significant improvement in BAs alignment over the uniform-weights method and FreeSurfer. Task-optimal weights is also significantly better than the uniform-weights method, but only slightly better than FreeSurfer. This suggests that even when local geometry is predictive of the hidden labels and anatomy-based registration achieves reasonable localization of the labels, tuning the registration cost function can further improve the task performance. We also note that in this case, FreeSurfer performs better than the uniform-weights method on average. Since local cortical folds are predictive of these areas, aligning cortical folds on a higher resolution mesh yields more precise alignment of the cortical geometry and of the BAs.



(a) Left V1 (b) Left V2 (c) Left BA2

(d) Right V1 (e) Right V2 (f) Right BA2

Figure 5-2: Mean Hausdorff distances over an entire range of harmonic energy for V1, V2 and BA2. First row corresponds to left hemisphere. Second row corresponds to right hemipshere.

Qualitative Results

Fig. 5-3 illustrates representative localization of the BAs for FreeSurfer and task-optimal template. We note that the task-optimal boundaries (red) tend to be in better visual agreement with the ground truth (yellow) boundaries, such as the right hemisphere BA44 and BA45.

Interpreting the Template

Fig. 5-4 illustrates an example of learning a task-optimal template for localizing BA2. Fig. 5-4(a) shows the initial cortical geometry of a template subject with its corresponding BA2 in black outline. In this particular subject, the postcentral sulcus is more prominent than the central sulcus. Fig. 5-4(b) shows the cortical geometry of a test subject together with its BA2. In this subject, the central sulcus is more prominent than the postcentral sulcus. Consequently, in the uniform-weights method, the central sulcus of the test subject is incorrectly mapped to the postcentral sulcus of the template, so that BA2 is misregistered. Fig. 5-4(a) shows the BA2 of the test subject (green) overlaid on the cortical geometry of the template subject after registration to the initial template geometry. During task-optimal training, our method interrupts the geometry of the postcentral sulcus in the template because the uninterrupted postcentral sulcus in the template is inconsistent with localizing BA2 in the training subjects. The final template is shown in Fig. 5-4(c). We see that the BA2 of the subject (green) and the task-optimal template (black) are well-aligned, although there still exists localization error in the superior end of BA2.

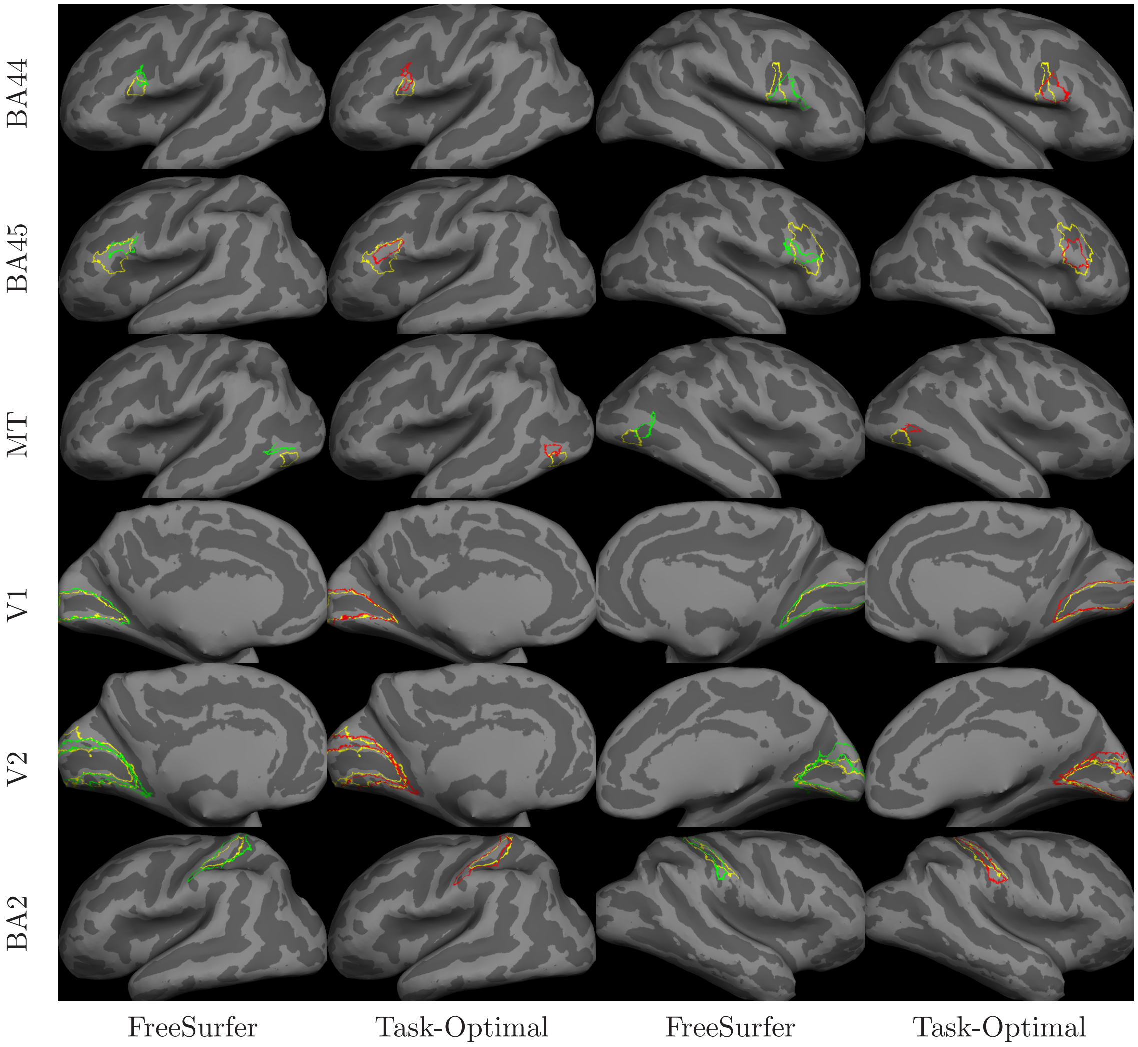
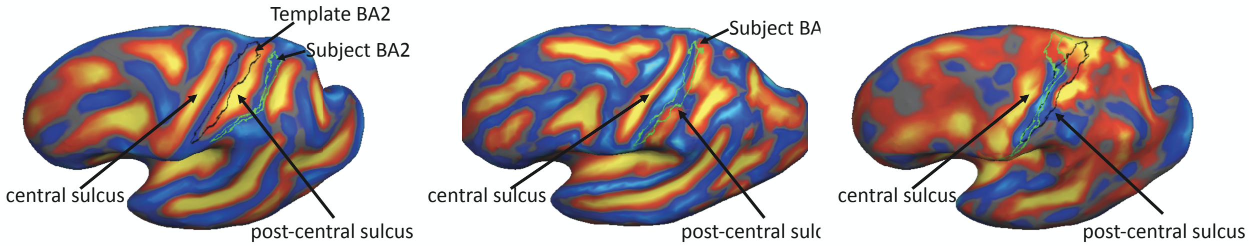


Figure 5-3: Representative BA localization in 90 trials of leave-two-out for FreeSurfer and task-optimal template. Yellow indicates ground truth boundary. Green indicates FreeSurfer prediction. Red indicates Task-Optimal prediction. The representative samples were selected by finding subjects whose localization errors are close to the mean localization errors for each BA. Furthermore, for a given BA, the same subject was selected for both methods to simplify the comparison.

In the next section, we turn our attention to a functional Magnetic Resonance Imaging (fMRI) data set. Since the task-optimal template performed better than the task-optimal weights, we will focus on the comparison between the task-optimal template and FreeSurfer henceforth.

5.3.2 fMRI-MT+ Localization

We now consider the localization of fMRI-defined MT+ in 42 *in-vivo* subjects described in Chapter 1.2.2.



(a) Initial Template (b) Test Subject (c) Task-Optimal Template

Figure 5-4: Template estimation in the task-optimal framework improves localization of BA2. (a) Initial cortical geometry of template subject with corresponding BA2 (in black). (b) Cortical geometry of test subject with corresponding BA2 (in green). In (a), we also show the BA2 of the test subject (in green) after registration to the intial template. (c) Final cortical geometry of template subject after task-optimal training. BA2 of the test subject (in green) after registration to the task-optimal template demonstrates significantly better alignment with the BA2 of the template subject.

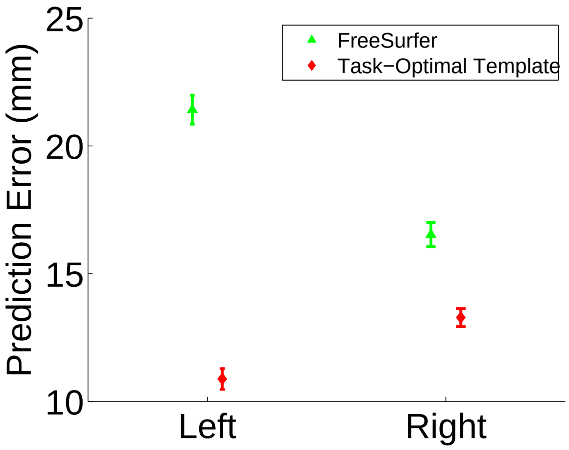


Figure 5-5: Mean Hausdorff distances using *ex-vivo* MT to predict MT+ in *in-vivo* scans.

Ex-vivo MT Prediction of *In-vivo* MT+

In this experiment, we use each of the 10 *ex-vivo* subjects as a template and the remaining 9 subjects for training a task-optimal template for localizing MT. We then register each task-optimal template to each of the 42 *in-vivo* subjects and use the template subject's MT to predict that of the test subjects' MT+. The results are 420 Hausdorff distances for each hemisphere. For FreeSurfer, we align the 42 *in-vivo* subjects to the Buckner40 atlas. Once registered in this space, we can use MT of the *ex-vivo* subjects to predict MT+ of the *in-vivo* subjects.

Fig. 5-5 reports the mean and standard errors of the Hausdorff distances for both methods on both hemispheres. Once again, we find that the task-optimal template significantly outperforms the FreeSurfer template. We note that the errors in the *in-vivo* subjects (Fig. 5-5) are significantly worse than those in the *ex-vivo* subjects (Fig. 5-1). This is not surprising since functionally defined MT+ is slightly different from cytoarchitectonically defined MT. Furthermore, the *ex-vivo* surfaces tend to be noisier and less smooth than those acquired from *in-vivo* subjects [183]. Since our framework attempts to leverage domain specific knowledge about MT from the *ex-vivo* data, one would expect these mismatches between the data sets to be highly deterimental to our framework. Instead, FreeSurfer appears to suffer more than our framework.

In-vivo MT Prediction of *In-vivo* MT+

To understand the effects of the training set size on localization accuracy, we perform cross-validation within the fMRI data set. For each randomly selected template subject, we consider 9, 19 or 29 training subjects. The resulting task-optimal template is used to register and localize MT+ in the remaining 32, 22 or 12 test subjects respectively. The cross-validation trials were repeated 100, 200 and 300 times respectively, resulting in a total of 3,200, 4,400 and 3,600 Hausdorff distances. This constitutes thousands of hours of computation time. For FreeSurfer, we perform a pairwise prediction of MT+ among the *in-vivo* subjects after registration to the Buckner40 atlas, resulting in 1,722 Hausdorff distances per hemisphere.

Fig. 5-6 reports the mean and standard errors of the Hausdorff distances for FreeSurfer and task-optimal template on both hemispheres. We see that the FreeSurfer alignment errors are now commensurate with the *ex-vivo* results (Fig. 5-1). However, the task-optimal template still outperforms FreeSurfer. We also note that the accuracy of MT+ localization improves with the size of the training set. The resulting localization error with a training set of 29 subjects is less than 7mm for both hemispheres. For all training set sizes, the localization errors are also better than the *ex-vivo* MT experiment (Fig. 5-1).

5.4 Discussion and Future Work

Since our measure of localization accuracy uses the mean Hausdorff distance, ideally we should incorporate it into our task-specific objective function instead of the SSD on the distance transform representing the BA. Unfortunately, the resulting derivative is difficult to compute. Furthermore, the gradient will be zero everywhere except at the BA boundaries, causing the optimization to proceed slowly. On the other hand, it is unclear how aligning the distance transform values far from the boundary helps to align the boundary. Since distance transform values far away from the boundary are larger, they can dominate the task-specific objective function g. Consequently, we utilize the distance transform over the entire surface to compute the gradient, but only consider the distance transform within the

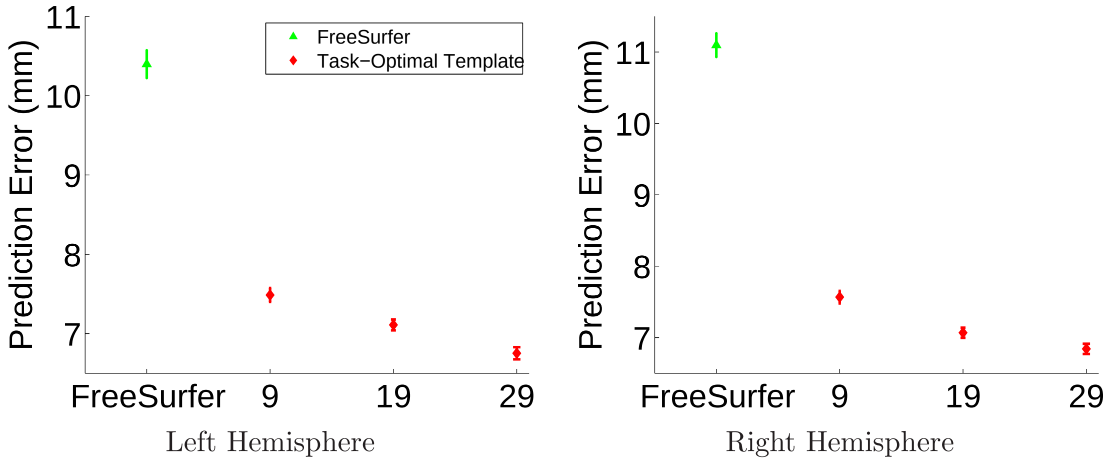


Figure 5-6: Plot of mean hausdorff errors for MT+ from cross-validation of the fMRI data set using either FreeSurfer or *in-vivo* trained task-optimal template. For the task-optimal framework, we tried different number of training subjects. Test errors decrease as we go from 9 to 19 to 29 training subjects.

boundary of the template BA when evaluating the task performance criterion g.

The idea of using multiple atlases for segmentation has gained recent popularity [72, 131, 132, 136, 139]. While we have focused on building a single optimal template, our method can complement the multi-atlas approach. For example, one could simply fuse the results of multiple individually-optimal templates for image segmentation. A more ambitious task would be to optimize for multiple jointly-optimal templates for segmentation.

In this work, we select one of the training subjects as the template subject and use the remaining subjects for training. The task-specific cost function g evaluates the localization of the hidden labels via the template subject. Appendix C.3 describes an alternative groupwise formulation, where the task-specific cost function g is modified to minimize the variance of the distance transforms across training subjects after registration. In this case, a reasonable template initialization is the average geometry of the training subjects. However, our initial experiments in the *ex-vivo* data set do not seem to suggest an improvement in task performance even in the training set beyond the initial average template.

There are two possible reasons for this. First, it may be that only stable cortical folds are predictive of cytoarchitectonics. Therefore, a simple average template could be sufficiently optimal for localizing Brodmann areas. If true, this is a non-trivial neuroscientific finding. Furthermore, previous arguments for unbiased anatomical templates are based on the goal of reducing anatomical bias towards any particular subject [16, 86, 180]. Our finding indicates that anatomical bias leads to poor BA and functional overlap and could provide further impetus for the use of unbiased anatomical templates. A second reason might come from the fact that we are only using an approximate gradient rather than the true gradient for gradient descent. Previous work [186] has shown that while using an approximate gradient can lead to reasonable solutions, using the exact gradient can lead to substantially better local minima. Computing the exact gradient is a challenge in our framework. We leave exploration of efficient means of computing better approximations of the gradient to future work.

While this chapter focuses mostly on localization of hidden labels, different instantiations of the task-specific cost function can lead to other applications. For example, in group analysis, the task-specific cost function could maximize differences between diseased and control groups, while minimizing intra-group differences, similar to a recent idea proposed for discriminative Procrustes alignment [98].

5.5 Summary

In this chapter, we presented a framework for optimizing the parameters of any smooth family of registration cost functions, such as the image dissimilarity-regularization tradeoff, with respect to a specific task. The only requirement is that the task performance can be evaluated by a smooth cost function. We demonstrated state-of-the-art localization of Brodmann areas and fMRI-defined functional regions by optimizing the weights of the wSSD image-similarity measure and estimating an optimal cortical folding template. We believe this work presents an important step towards the automatic selection of parameters in image registration. The generality of the framework also suggests potential applications to other problems in science and engineering formulated as optimization problems.

Chapter 6

Conclusions

In this thesis, we argued that the notion of "correct registration" makes sense only within the context of an application. Furthermore, we demonstrated that by taking into account the application when performing registration, we can improve the application performance.

In Chapter 3, we presented a generative model for joint registration and segmentation images. The atlas construction process naturally arises as estimation of the model parameters. The model allows for computation of unbiased atlases, as well as the joint registration and segmentation of new images consistent with the atlas construction. We demonstrated improvement in segmentation accuracy over the widely used FreeSurfer software that employs sequential registration and segmentation to parcellate the cerebral cortex into different gyral units.

In Chapter 4, we developed the Spherical Demons algorithm for fast registration of spherical images. By improving on warp parameterization and regularization, Spherical Demons is easily an order of magnitude faster than FreeSurfer while being as accurate for registering spherically parameterized cortical surfaces.

The fast nature of the Spherical Demons algorithm allows for its (partial) incorporation into the meta-registration algorithm developed in Chapter 5. The task-optimal registration framework learns the parameters of registration cost function specific to a particular application. The algorithm allows for the optimization of registration parameters to improve the alignment of a new image as measured by an application-specific performance measure.

Traditionally, registration has been considered a pre-processing step. Much work has been devoted to developing generic registration algorithms, which are then specialized to particular imaging modalities (e.g., MR, CT, histology, etc.), particular imaging targets (e.g., cardiac, brain, etc.) and particular post-registration analysis (e.g., segmentation, voxel-based morphometry, functional group analysis, etc.). The task-optimal framework provides a principled method for adapting these generic algorithms to specific applications. For example, we have demonstrated the estimation of weights or template of the generic weighted Sum of Squared Differences (wSSD) dissimilarity measure that are optimal for localizing underlying cytoarchitecture and functional regions of the cerebral cortex.

6.1 Future Directions

Throughout the thesis, we discussed specific avenues of future research. We now sketch out broad outlines of two future directions of research:

1. **Generative Modeling.** In the past five years, there has been a flurry of activities in the development of generative models for medical image registration beyond the simple deformable template with Gaussian noise model [2, 12, 136, 165, 195]. These include our work on joint registration and segmentation using single [180] or multiple templates [138, 139]. Compared with other models, generative models have the advantages of explicit declaration of prior assumptions and beliefs.

   While this thesis has focused on neuroanatomical registration, we would like to turn our attention to the development of template construction algorithms in other "less developed" imaging modalities (e.g., DTI or rest-state fMRI) and multimodal image registration (e.g., jointly modeling both macro-anatomy and DTI, both fMRI and DTI, both fMRI and genetics, etc.). Each of these areas have common and unique challenges that we believe probabilistic modeling has a significant possibility of resolving.

2. **Task-Optimal Applications.** Two unresolved theoretical issues in the task-optimal framework (Chapter 5) are the characterization of local minima and the generalization properties of the training. As discussed in Chapter 5.1, local minima with singular Hessian present a problem for our gradient descent algorithm. Perhaps a different theoretical characterization of such local minima may result in a better algorithm. Secondly, according to learning and regularization theory [52, 166], an important condition for training to generalize to testing is the smoothness of the underlying space. However, the space of local minima is only smooth at local minima with positive definite Hessian. Therefore, although we do observe generalization in practice, it is unclear theoretically why the parameters learned during training should generalize to new images.

   An important practical issue is the use of the approximate gradient for optimizing the task-optimal learning algorithm. While the approximate gradient appears mostly effective in optimizing the task-optimal cost function, experiments in Appendix C suggest that the approximation might not be sufficiently accurate for certain instantiations of the cost function. Furthermore, our previous work on DTI registration suggests that in practice, using an exact gradient can lead to much better local mimima than an approximate gradient [185, 186]. Therefore, there is a need to carefully investigate the use of the actual gradient, despite its computational complexities. An alternative is to consider other practical optimization algorithms that do not require the computation of a gradient.

Despite the need for further research, we believe that we have amply demonstrated the promises of the task-optimal framework. We are interested in applying the algorithm to application other than segmentations. For example, it might be useful to estimate an optimal template to discriminate two population groups. The template can then be applied to characterize group differences on a new data set. Because the framework applies to generic nonlinear optimization problems, we are also interested in exploring its applications in other engineering and scientific domains.

Appendix A

Joint Registration-Segmentation Appendix

A.1 Mean Field Derivation

The mean-field approximation uses a variational formulation [79], where we seek to minimize the KL-divergence (denoted by $D(\cdot||\cdot)$) between $q(\widehat{L}) = \prod_i \mathbf{b}_i(\widehat{L}_i)$ and $p(\widehat{L}|I(R^{(t)}(\{x_i\})); A_\alpha)$:

$$\{\mathbf{b}_i^*\} = \underset{\{\mathbf{b}_i\}}{\operatorname{argmin}}\, D(q(\widehat{L})||p(\widehat{L}|I(R^{(t)}(\{x_i\})); A_\alpha)) \tag{A.1}$$

This results in a fixed-point iterative solution. Since this is a fairly standard derivation [79], we only provide the final update:

$$\mathbf{b}_i(m) \propto e^{U_i(m)+\log p(I(R^{(t)}(x_i))|\widehat{L}_i=m;A_\alpha)+\sum_{j\in\mathcal{N}_i}\sum_{\widehat{L}_j}\mathbf{b}_j(\widehat{L}_j)[V(m,\widehat{L}_j)+V(\widehat{L}_j,m)]} \tag{A.2}$$

where $\mathbf{b}_i$ is normalized to be a valid probability mass function.

A.2 Implementation Details

We now present some implementation details for completeness.

We first discuss the estimation of the atlas A_α defined by $\{U_i, V, W_i\}$ in Eq. (3.20) and Eq. (3.21) from the maximum likelihood function Eq. (3.7). In our model, estimating U_i and V is hard in practice, since evaluating Eq. (3.7) requires computing the NP-hard partition function. Instead, we use frequency counts to estimate the clique potentials, similar to FreeSurfer [60].

- In our implementation, the singleton potential U_i is a row vector of length M and L_n

is a column indicator vector. We set

$$U_i = \log \frac{1}{N} \sum_n L_n^T(R_n(x_i)), \tag{A.3}$$

where $(\cdot)^T$ indicates transpose.

- The pairwise potential V is a $M \times M$ matrix. Following [60], we set

  $$V = \log \frac{1}{2NE} \sum_n \sum_i \sum_{j \in \mathcal{N}_i} L_n(R_n(x_i)) L_n^T(R_n(x_i)), \tag{A.4}$$

  where E is the number of edges in the atlas mesh. More rigorous methods of optimizing the clique potentials through iterative proportional fitting [83] would further improve the clique potential estimates.

- The likelihood potential W_i is a row vector of length M defined at each vertex. The m-th entry of W_i corresponds to the likelihood of observing a particular image intensity or vectors of intensity (in our case, the local surface geometries) at location $R(x_i)$ given a particular label m. While we might be observing multiple geometric features at any vertex, the likelihood of these features is combined into the row vector W_i. We use maximum likelihood estimates of the mean and variance of the Gaussian distribution of cortical geometries conditioned on the segmentation labels to parameterize this distribution. In this work, we use the mean curvature of the original cortical surface and average convexity, which is a measure of sulcal depth [59], as intensity features.

  At spatial locations where there is no training data for a particular label, it is unclear what the value of the entry in W should be since it is spatially varying. We simply assume a mean of zero and a large variance, essentially being agnostic to the value of intensity we observe. A more sophisticated method would involve the use of priors on the atlas parameters, so that the atlas parameters become random. In that case, when there are no observations, the maximum likelihood estimates of the atlas parameters become the priors.

Secondly, we discuss the registration of a training image to an atlas (Eq. (3.22)) and the new image registration (Eq. (3.24)).

- The registration warp R is a map from a 2-Sphere to a 2-Sphere. We represent R as a dense displacement field. In particular, each point x_i has an associated displacement vector u_i tangent to the point x_i on the sphere. $R(x_i)$ maps x_i to $x_i + u_i$ normalized to be a point on the sphere.

- To interpolate (for example the mean curvature of a cortical surface onto $R(x_i)$), we first find the intersection between the vector $R(x_i)$ and each planar triangle of the spherically mapped cortical surface. We then use barycentric interpolation to interpolate the values at the vertices of the mesh onto $R(x_i)$.

- The above two bullets completely specify the computation of the atlas co-registration objective function Eq. (3.22) and the new subject registration function Eq. (3.24). This allows us to compute the gradients of the objective function via the chain rule.
- We use conjugate gradient ascent with parabolic line search [125] on a coarse-to-fine grid. The coarse-to-fine grid comes from the representation of the atlas as a subdivided icosahedron mesh.
- The final segmentation is obtained by selecting for each vertex the label with the highest posterior probability.
- To satisfy $F(R)$, the regularization that induces invertibility, we ensure that no step in the line search results in folded triangles. Unfortunately, in practice, this results in many small steps. It is much more efficient to perform the line search without considering $F(R)$, and then unfold the triangles using $F(R)$ after the line search. In general, we find that after unfolding, the objective function is still better than the previous iteration. This unfolding process can be expensive for small smoothness parameter S ($S \leq 0.1$), resulting in long run times of about 1.5 hours per subject per atlas for $S \leq 0.1$.

Appendix B

Spherical Demons Appendix

B.1 Step 1 Gradient Derivation

In this appendix, we provide details on the computation of the spatial gradient of the warped moving image $M \circ \Upsilon^{(t)}$ and the Jacobian of the deformation $\Upsilon^{(t)}$. We also compute the gradients of the demons cost function using the derivatives computed in Eq. (4.20) and Eq. (4.21), assuming the l_2 inner product space for vector fields and the canonical metric.

B.1.1 Computing Spatial Gradient of $M \circ \Upsilon^{(t)}$

In this appendix, we discuss the computation of m_n^T, the spatial gradient of the warped moving image $M \circ \Upsilon^{(t)}$ at the point x_n. We can think of $M \circ \Upsilon(\cdot)$ as an image $M_\Upsilon \triangleq M \circ \Upsilon$ defined on the mesh vertices $\{x_n\}$. This image is made continuous by the choice of an interpolation method. In this work, we assume that we are working with a triangular mesh. To evaluate M_Υ at a point $x \in S^2$, we first find the triangle that contains the intersection between the vector representing the point x (i.e., the vector between the center and the point x of the sphere) and the mesh. The image value at x is then given by the barycentric interpolation of the image values at the intersection point. Mathematically, we can write

$$M_\Upsilon(x) = I(p(x)), \tag{B.1}$$

where $p(x)$ is the intersection point and $I(\cdot)$ is the barycentric interpolation. Let p_1, p_2, p_3 denote the vertices of the triangle containing $p(x)$ and $\vec{n}$ denote the 3×1 normal vector to the triangle. Since $p(x) = \alpha x$ for some α and $\langle p(x) - p_1, \vec{n} \rangle = 0$, we can write

$$p(x) = \frac{\langle p_1, \vec{n} \rangle}{\langle x, \vec{n} \rangle} x \tag{B.2}$$

and

$$I(p) = \frac{A_1(p)M_\Upsilon(p_1) + A_2(p)M_\Upsilon(p_2) + A_3(p)M_\Upsilon(p_3)}{A}, \tag{B.3}$$

where $A_1(p)$, $A_2(p)$ and $A_3(p)$ are the areas of the triangles $\triangle pp_2p_3$, $\triangle pp_1p_3$ and $\triangle pp_1p_2$ respectively. Note that $A = A_1(p) + A_2(p) + A_3(p)$. $M_\Upsilon(p_1), M_\Upsilon(p_2)$ and $M_\Upsilon(p_3)$ are the image values at the mesh vertices p_1, p_2 and p_3 respectively.

Computing the derivative of the image value at x follows easily from the chain rule:

$$\frac{\partial p(x)}{\partial x} = \frac{\langle p_1, \vec{n}\rangle}{\langle x, \vec{n}\rangle} I_{3\times 3} - \frac{\langle p_1, \vec{n}\rangle}{\langle x, \vec{n}\rangle^2} x\vec{n}^T \tag{B.4}$$

$$\frac{\partial I(p)}{\partial p} = \frac{\nabla_p A_1(p)M_\Upsilon(p_1) + \nabla_p A_2(p)M_\Upsilon(p_2) + \nabla_p A_3(p)M_\Upsilon(p_3)}{A}, \tag{B.5}$$

where $\nabla_p A_i(p)$ is the derivative of the triangle area A_i. For example, $\nabla_p A_1(p)$ is a 1×3 vector in the plane of the triangle pp_2p_3, perpendicular and pointing to the edge p_2p_3, with magnitude half the length of p_2p_3. Combining Eq. (B.4) and Eq. (B.5) gives the spatial gradient of the warped moving image.

A complication arises when x corresponds to one of the mesh vertices, since the spatial gradient is not defined in this case. The same problem arises in Euclidean space with linear interpolation and the spatial gradient is typically defined via finite central difference. It is unclear what the equivalent definition on a mesh is. Here, for a mesh vertex x, we compute the spatial gradient for each of the surrounding triangles and linearly combine the spatial gradients using weights corresponding to the areas of the triangles.

B.1.2 Computing the Jacobian of Deformation $\Upsilon^{(t)}$

In this appendix, we discuss the computation of S_n^T, the Jacobian of the deformation $\Upsilon^{(t)}$ at x_n. We can think of $\Upsilon^{(t)}$ as a vector function on S^2 that maps each mesh vertex $\{x_n\}$ to a new point on the sphere. This vector image is made continuous by the choice of an interpolation method. We use the same interpolation as in Appendix B.1.1, except we need to normalize the barycentric interpolation so that the interpolated point is constrained to be on the sphere:

$$\Upsilon^{(t)}(x) = I(p(x)) \tag{B.6}$$

where $p(x)$ is the same as in the previous section and

$$I(p) = \frac{A_1(p)\Upsilon^{(t)}(p_1) + A_2(p)\Upsilon^{(t)}(p_2) + A_3(p)\Upsilon^{(t)}(p_3)}{\|A_1(p)\Upsilon^{(t)}(p_1) + A_2(p)\Upsilon^{(t)}(p_2) + A_3(p)\Upsilon^{(t)}(p_3)\|} \tag{B.7}$$

The Jacobian is computed via chain rule, just like in the previous section.

B.1.3 Computing the Gradients from the Derivatives

In this appendix, we seek to compute the gradients of $f_n(\vec{z}) \triangleq F(x_n) - M \circ \{\Upsilon^{(t)} \circ \exp(\{E_n\vec{z}_n\})\}(x_n)$ and $g_n(\vec{z}) \triangleq \vec{\Upsilon}_n^{(t)} + G_n^2\{\Upsilon^{(t)} \circ \exp(\{E_n\vec{z}_n\})\}(x_n)$, assuming a l_2 inner product for vector fields and the canonical metric R for S^2. These assumptions imply that the inner product of two vector fields $\vec{z}^1 = \{\vec{z}_k^1\}$ and $\vec{z}^2 = \{\vec{z}_k^2\}$ are given by

$$\langle \vec{z}^1, \vec{z}^2 \rangle_{l_2} = \langle \{E_k\vec{z}_k^1\}, \{E_k\vec{z}_k^2\} \rangle_{l_2} \tag{B.8}$$

$$= \sum_{k=1}^{N} \langle E_k\vec{z}_k^1, E_k\vec{z}_k^2 \rangle_R \tag{B.9}$$

$$= \sum_{k=1}^{N} \langle \vec{z}_k^1, \vec{z}_k^2 \rangle_R, \tag{B.10}$$

where

- Eq. (B.8) follows from the equivalence of the tangent bundles $T\mathbb{R}^2$ and TS^2 induced by the coordinate charts $\{\Psi_n\}$.

- Eq. (B.9) is the result of the l_2 assumption that the inner product of vector fields is given by the sum of the inner product of individual vectors.

- Because we assume the canonical metric, each term in the inner product in Eq. (B.9) is simply the usual inner product between 3×1 vectors $E_k\vec{z}_k^1$ and $E_k\vec{z}_k^2$. Since the columns of E_k are orthonormal with respect to the usual inner product and using linearity of the inner product, Eq. (B.9) implies Eq. (B.10), i.e., the inner product $\langle \vec{z}^1, \vec{z}^2 \rangle_{l_2}$ can be computed by the sum of the usual inner product between 2×1 tangent vectors $\vec{z}_k^1$ and $\vec{z}_k^2$.

Let $df_n(\vec{z})$ be the directional derivative of f_n for any $\vec{z} = \{\vec{z}_k\}$. The directional derivative is *independent* of the choice of metric. Since the derivative of $f_n(\vec{z})$ with respect to $\vec{z}_k$ is a 1×2 vector $-\vec{m}_n^T E_n \delta(k, n)$ (Eq. (4.20)), we get

$$df_n(\vec{z}) = -\vec{m}_n^T E_n \vec{z}_n. \tag{B.11}$$

Recall that the gradient $\nabla_{l_2} f_n$ of $f_n(\vec{z})$ is defined to be a tangent vector field such that $df_n(\vec{z}) = \langle \nabla_{l_2} f_n, \vec{z} \rangle_{l_2}$ for any $\vec{z} = \{\vec{z}_k\}$. The gradient is therefore *dependent* on the choice of the inner product. From Eq. (B.10) and Eq. (B.11), we can write

$$-\vec{m}_n^T E_n \vec{z}_n = df_n(\vec{z}) \tag{B.12}$$

$$= \langle \nabla_{l_2} f_n, \vec{z} \rangle_{l_2} \tag{B.13}$$

$$= \sum_{k=1}^{N} \langle \nabla_{l_2} f_n(x_k), \vec{z}_k \rangle_R \ . \tag{B.14}$$

Therefore, the gradient $\nabla_{l_2} f_n$ can be written as a $2N \times 1$ vector consisting of N blocks of 2×1 vectors, where all the blocks are zeros, except the n-th block is equal to $-E_n^T \vec{m}_n$.

Similarly, we denote the gradient of $g_n(\vec{z})$ as $\nabla_{l_2} g_n(j)$ for $j = 1, 2, 3$ corresponding to the 3 output components of $g_n(\vec{z})$. The derivative of ∇g_n with respect to $\vec{z}_k$ is a 3×2 matrix $G_n^2 S_n^T E_n \delta(k,n) \triangleq [\vec{a}_{1n} \ \vec{a}_{2n} \ \vec{a}_{3n}]^T \delta(k,n)$ (Eq. (4.21)), where $\vec{a}_{jn}^T \delta(k,n)$ is a 1×2 vector corresponding to the derivative of the j-th component of g_n with respect to $\vec{z}_k$. Using the same derivation as before, we can show that $\nabla_{l_2} g_n(j)$ can be written as a $2N \times 1$ vector consisting of N blocks of 2×1 vectors, where all the blocks are zeros, except the n-th block is equal to $\vec{a}_{jn}$.

B.2 Approximating Spline Interpolation with Iterative Smoothing

In this appendix, we demonstrate empirically that iterative smoothing provides a good approximation of spherical vector spline interpolation for a relatively uniform distribution of points corresponding to those of the subdivided icosahedron meshes used in this work. Once again, we work with spheres that are normalized to be of radius 100.

Recall that we seek $\{\vec{\Upsilon}_n\} = \{E_n \Upsilon_n\}$, which is a smooth approximation of the input vector field $\{\vec{\Gamma}_n\} = \{E_n \Gamma_n\}$. The solution of the spherical vector spline interpolation problem is given in Eq. (4.32) as

$$\hat{\Upsilon} = K \left(\frac{\sigma_x^T}{\sigma_T^2} I_{2N \times 2N} + K \right)^{-1} \hat{\Gamma}, \tag{B.15}$$

where K is a $2N \times 2N$ matrix consisting of $N \times N$ blocks of 2×2 matrices: the (i, j) block corresponds to $k(x_i, x_j) T_{x_i, x_j}$. T_{x_i, x_j} is the parallel transport operator from x_i to x_j. $k(x_i, x_j)$ is a non-negative scalar function uniquely determined by the choice of the energetic norm that monotonically decreases as a function of the distance between x_i and x_j.

In constrast, the iterative smoothing approximation we propose can be formalized as follows:

$$\hat{\Upsilon} = (K')^m \hat{\Gamma} \tag{B.16}$$

where m is a positive integer and K' is a $2N \times 2N$ matrix consisting of $N \times N$ blocks of 2×2 matrices: the (i, j) block corresponds to $\lambda(x_i, x_j) T_{x_i, x_j}$ if x_i and x_j are neighboring vertices and is a zero matrix otherwise. $\lambda(x_i, x_i) = \frac{1}{1+|N_i| \exp(-\frac{1}{2\gamma})}$ and $\lambda(x_i, x_j) = \frac{\exp(-\frac{1}{2\gamma})}{1+|N_i| \exp(-\frac{1}{2\gamma})}$ for $i \neq j$, where $|N_i|$ is the number of neighboring vertices of x_i.

B.2.1 Reverse Engineering the Kernel

We now demonstrate empirically that for a range of values of γ, iterations m and the relatively uniform distribution of points corresponding to those of the subdivided icosahedron mesh, there exist kernels $k(x_i, x_j)$ that are well approximated by iterative smoothing. Technically, the resulting $k(x_i, x_j)$ might not correspond to a true choice of the energetic norm. However, in practice, this does not appear to be a problem.

More specifically, given a configuration of mesh points, iterations m and value of γ, we seek $\tilde{k}(x_i, x_j)$, such that $K \left(\frac{\sigma_x^T}{\sigma_T^2} I_{2N \times 2N} + K \right)^{-1}$ is "close" to $(K')^m$. We propose a two-stage estimation of $\tilde{k}(x_i, x_j)$:

1. In the first stage, we seek $k^*(x_i, x_j)$ that is *not* constrained to be a function of the distance between x_i and x_j, such that

$$K \left(\frac{\sigma_x^T}{\sigma_T^2} I_{2N \times 2N} + K \right)^{-1} - (K')^m \approx 0 \tag{B.17}$$

   Rearranging the terms, we get

$$\left(I_{2N \times 2N} - (K')^m \right)^{-1} (K')^m \frac{\sigma_x^2}{\sigma_T^2} \approx K \tag{B.18}$$

   To make the "≈" concrete, we optimize for

$$k^* = \underset{k}{\operatorname{argmin}} \left\| K - \left(I_{2N \times 2N} - (K')^m \right)^{-1} (K')^m \frac{\sigma_x^2}{\sigma_T^2} \right\|_F^2 \tag{B.19}$$

   where $\| \cdot \|_F$ is the Frobenius norm.

   The cost function Eq. (B.19) can be optimized componentwise, i.e., we can solve for $k^*(x_i, x_j)$ for each pair x_i, x_j. For $\gamma = 1$, $m = 10$ and a subdivided icosahedron mesh with 642 vertices, we plot the resulting $k^*(x_i, x_j)$ as a function of the geodesic distance between x_i and x_j in Fig. B-1.

2. In the second stage, we perform a least-squares fit of a b-spline function to the estimated $k^*(x_i, x_j)$ to obtain the final estimate of $\tilde{k}(x_i, x_j)$. Fig. B-1 illustrates an example kernel $\tilde{k}(x_i, x_j)$ we obtain (c.f., the kernel illustrated in [67]). We note that an alternative to b-spline interpolation is to fit the coefficients of the general kernel

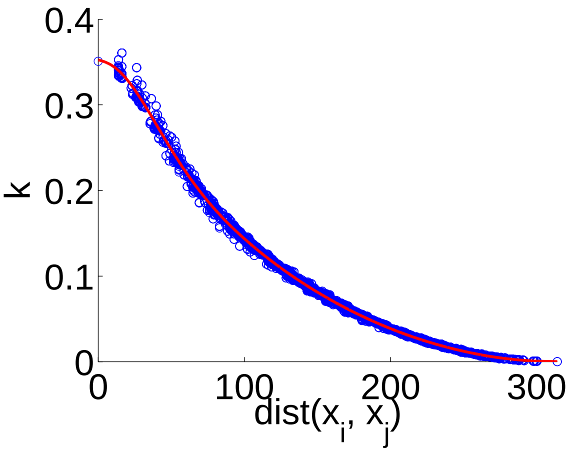


Figure B-1: Approximating the kernel function $k(x_i, x_j)$. The scattered points corresponds to the estimation of $k^*(x_i, x_j)$ via Eq. (B.19). The red curve corresponds to fitting the scattered points so that $\tilde{k}(x_i, x_j)$ is strictly a function of the geodesic distance between x_i and x_j.

function suggested in Appendix A of [67]. This will guarantee that the estimated kernel corresponds to an energetic norm. We leave exploring this direction to future work.

B.2.2 Evaluating Approximation

We now investigate the quality of the estimate $\tilde{k}(x_i, x_j)$ by computing:

$$\left\| K \left(\frac{\sigma_x^T}{\sigma_T^2} I_{2N \times 2N} + K \right)^{-1} - (K')^m \right\|_2^2 \tag{B.20}$$

where $\| \cdot \|_2$ is the l_2 matrix operator norm. The difference metric Eq. (B.20) measures the *maximum* l_2 difference between smoothed vector fields obtained from iterative smoothing and spherical vector spline interpolation for *any* possible input vector field $\{\vec{\Gamma}_n\}$ of unit l_2 norm, i.e., $\sum_n \|\vec{\Gamma}_n\|^2 = 1$. We note that $\tilde{k}(x_i, x_j)$ can be in principle estimated by minimizing Eq. (B.20) instead of the proposed 2-stage process. However, the optimization is difficult since evaluating the cost function itself requires finding the largest singular value of a large, non-sparse matrix.

Fig. B-2 displays the difference metric we obtained with different values of γ and iterations m for meshes ic2, ic3, ic4 and ic5. Each of the meshes is obtained from recursively subdividing a lower resolution mesh: ic2 indicates that the mesh was obtained from subdividing an icosahedron mesh twice. The number of vertices quadruples with each subdivision, so that ic5 corresponds to 10,242 vertices.

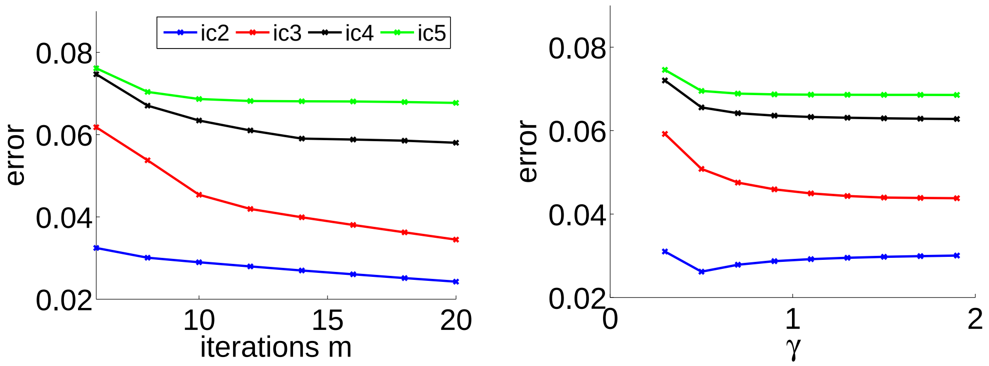


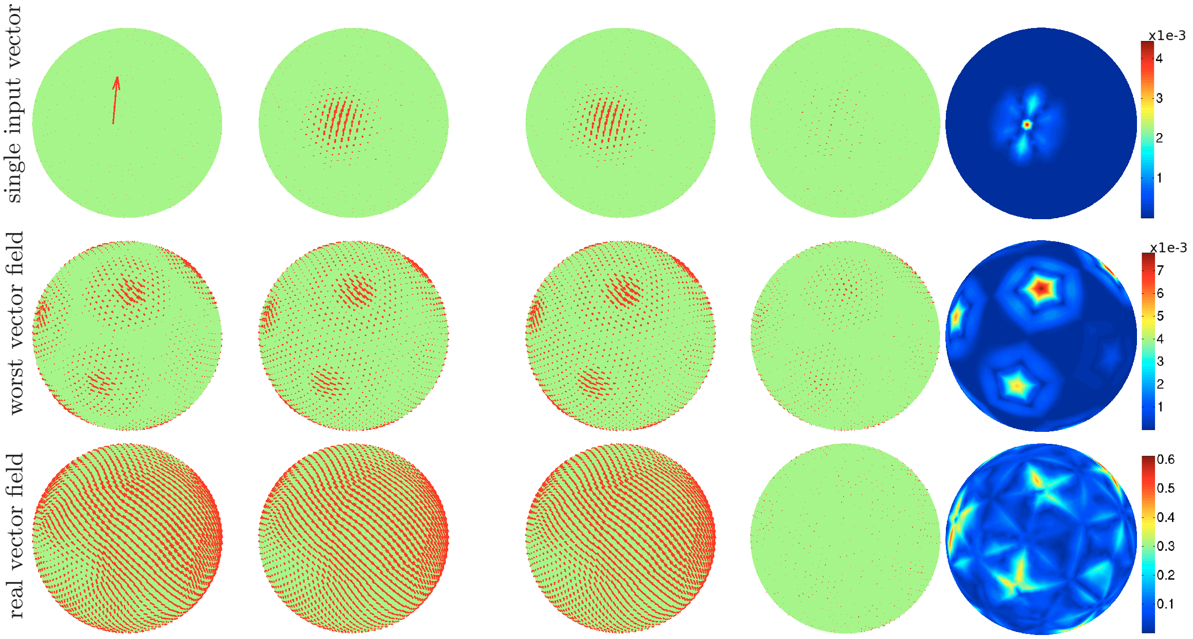
Figure B-2: Difference metric as a function of the number of iterations m and value of γ.

We conclude from the figure that the differences between the two smoothing methods are relatively small and increase with mesh resolution. As discussed in the next section, we run Spherical Demons on different mesh resolutions, including ic7. Unfortunately, because of the large non-sparse matrices we are dealing with, we were only able to compute the differences up to ic5. Computing the difference metric for ic5 took an entire week on a machine with 128GB of RAM. However, the plots in Fig. B-2 indicate that the differences appear to have converged by ic5.

To better understand the incurred differences, Fig. B-3 illustrates the outputs and differences of the two smoothing methods for different inputs on ic4. The first row illustrates an input vector field which is zero everywhere except for a single tangent vector of unit norm. The results of spline interpolation and iterative smoothing correspond to our intuition that smoothing a single tangent vector propagates tangent vectors of smaller magnitudes to the surrounding areas. The two methods also produce almost identical results as shown by the clean difference image in the fourth column.

The second row of Fig. B-3 demonstrates the worst unit norm input vector field as measured by the difference metric Eq. (B.20). This worst unit norm input vector field corresponds to the largest eigenvector in Eq. (B.20). The pattern of large differences correspond to the original 12 vertices of the uniform icosahedron mesh. These original 12 vertices are the only vertices in the subdivided icosahedron meshes with five, instead of six neighbors, as shown by the pentagon pattern. The fact that these 12 vertices are local maxima of differences suggest that these vertices are treated differently by the two smoothing techniques.

The last row of Fig. B-3 demonstrates an input vector field that represents the deformation of an actual registration performed in Section 4.3. The norm of the input vector field is 700 times that in the first two rows, but the discrepancies between spline interpolation and iterative smoothing are less than expected. The differences of 90% of the vectors are less than 0.2mm, with larger differences in the neighborhoods of the 12 vertices identified



(a) Input (b) Spline Interpolation (c) Iterative Smoothing (d) Difference (e) l_2 norm

Figure B-3: Comparison of spline interpolation and iterative smoothing ($m = 10, \gamma = 1$). (a) Input vector field (b) Spline Interpolation Output (c) Iterative Smoothing Output (d) Difference between the second and third columns (e) l_2 norm of the difference. The first row uses an input vector field which is zero everywhere except for a single tangent vector of unit norm. Second row illustrates the worst unit norm input as measured by the difference metric Eq. (B.20). The third row uses a vector field from an actual warped image from the experimental section. Note that the input vector field in the first two rows are scaled down for the purpose of display. The vector fields in the entire third row are of the same scale, but are scaled down relative to the first two rows, since the vector field from the warped image is substantially larger in magnitude than the unit norm inputs of the first two rows. This explains the substantially larger difference metric on the third row.

previously. Since we conclude previously that the difference metric appears to have converged after ic4, the discrepancies are likely to be acceptable at ic7, whose mesh resultion is 1mm.

We should emphasize that the discrepancies between spline interpolation and iterative smoothing do not necessarily imply registration errors. The differences only indicate the deviations of the deformations from true local optima of the Demons registration cost function Eq. (4.1) assuming the estimated kernel. Approximating smoothing kernels by iterative smoothing is an active area of research in medical imaging [35, 70]. Future work would involve understanding the interaction between the number of smoothing iterations m and the choice of the weights $\exp(-\frac{1}{2\gamma})$ on the quality of the spherical registration.

B.3 Atlas-Based Spherical Demons

In this section, we demonstrate how an atlas consisting of a mean image and a standard deviation image can be incorporated into the Spherical Demons algorithm. The standard deviation image replaces Σ in Eq. (4.1). We first discuss a probabilistic interpretation of the Demons objective function and its relationship to atlases. We then discuss the optimization of the resulting probabilistic objective function.

B.3.1 Probabilistic Demons Objective Function

The Demons objective function reviewed in Section 4.1 is defined for the pairwise registration of images. To incorporate a probabilistic atlas, we now reformulate the objective function. Consider the following Maximum-A-Posteriori objective function:

$$
\begin{aligned}
(\Upsilon^*, \Gamma^*) &= \underset{\Upsilon,\Gamma}{\operatorname{argmax}} \log p(\Gamma, \Upsilon, F, M) && \text{(B.21)}\\
&= \underset{\Upsilon,\Gamma}{\operatorname{argmax}} \log p(F, M|\Gamma, \Upsilon) p(\Gamma|\Upsilon) p(\Upsilon) && \text{(B.22)}\\
&= \underset{\Upsilon,\Gamma}{\operatorname{argmax}} \log p(F, M \circ \Gamma|\Gamma) + \log p(\Gamma|\Upsilon) + \log p(\Upsilon). && \text{(B.23)}
\end{aligned}
$$

Assuming a Gaussian noise model, we define

$$
\begin{aligned}
p(F, M \circ \Gamma) &= \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}(\sigma(\Gamma, x_n))} \exp\left(-\frac{(F(x_n) - M \circ \Gamma(x_n))^2}{(\sqrt{2}\sigma(\Gamma, x_n))^2} \right), && \text{(B.24)}\\
\log p(\Gamma|\Upsilon) &= \frac{1}{Z(\sigma_x^2)} \exp\left(-\frac{1}{\sigma_x^2} \sum_{n=1}^{N} \|\vec{\Upsilon}_n - \vec{\Gamma}_n\|^2 \right), && \text{(B.25)}\\
p(\Upsilon) &= \frac{1}{Z(\sigma_T^2)} \exp\left(-\frac{1}{\sigma_T^2} \operatorname{Reg}(\Upsilon) \right), && \text{(B.26)}
\end{aligned}
$$

where $\operatorname{Reg}(\Upsilon)$ is defined via the energetic norm as discussed in Section 4.2.3 and for reasons that will soon be clear, we are being purposefully agnostic about the form of $\sigma(\Gamma, x_n)$. The objective function in Eq. (B.23) becomes

$$
(\Upsilon^*, \Gamma^*) = \underset{\Upsilon,\Gamma}{\operatorname{argmin}} \sum_{n=1}^{N} \frac{(F(x_n) - M \circ \Gamma(x_n))^2}{(\sqrt{2}\sigma(\Gamma, x_n))^2} + \frac{1}{\sigma_x^2} \sum_{n=1}^{N} \|\vec{\Upsilon}_n - \vec{\Gamma}_n\|^2 + \frac{1}{\sigma_T^2} \operatorname{Reg}(\Upsilon) + \sum_{n=1}^{N} \log \sigma(\Gamma, x_n) \ , \tag{B.27}
$$

which is the instantiation of the Demons objective function Eq. (4.1), except for the extra term $\sum_{n=1}^{N} \log \sigma(\Gamma, x_n)$. Note that we have omitted the partition functions $Z(\sigma_x^2)$ and $Z(\sigma_T^2)$ because σ_x and σ_T are constant with respect to the deformations Γ and Υ. In this probabilistic interpretation, the two regularization terms $p(\Gamma|\Upsilon)$ and $p(\Upsilon)$ act as a hierarchical prior on Γ, with the hidden transformation Υ as a hyperparameter.

As before, $\sigma(\Gamma, x_n)$ is the standard deviation of the intensity at vertex n. Given a set of co-registered images, we can create an atlas by computing the mean intensity and standard deviation at each vertex. To incorporate the atlas, we need to make the choice of treating the atlas as the fixed or moving image. If we treat the atlas as the fixed image, then we set F to be the mean image and σ to be the standard deviation. In this case, we do not need to interpolate the mean or standard deviation image. Consequently, $\sigma(\Gamma, x_n) = \sigma(x_n)$ and $\log \sigma(\Gamma, x_n)$ can be omitted from the optimization. The registration becomes identical to the Spherical Demons algorithm for two images.

However, recent work [2, 140] suggests that treating the atlas as a moving image might be more correct theoretically. This involves setting the moving image M to be the mean image. In this case, $\sigma(\Gamma, x_n) = \sigma(\Gamma(x_n))$ is a function of Γ and we must include $\log(\sigma(\Gamma(x_n)))$ in the optimization. We performed experiments for both choices and found the results from interpolating the atlas, i.e., treating it as a moving image, to be only slightly better than interpolating the subject. However, interpolating the subject results in a faster algorithm, whose computational time is less than 3 minutes. We report the results of interpolating the atlas in the experimental section.

B.3.2 Optimization of Atlas-Based Spherical Demons

We now discuss the optimization in Eq. (B.27). Note that the introduction of the new term $\sum_{n=1}^{N} \log \sigma(\Gamma, x_n)$ only affects Step 1 of the Spherical Demons algorithm. By parameterizing $\Gamma^{(t)} = \Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})$, we get

$$\begin{aligned}\{\vec{z}_n^{(t)}\} = \underset{\vec{z}_n}{\operatorname{argmin}} \sum_{n=1}^{N} & \frac{(F(x_n) - M \circ \{\Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})\}(x_n))^2}{\left(\sqrt{2}\sigma \circ \{\Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})\}(x_n)\right)^2} \\ & + \frac{1}{\sigma_x^2} \sum_{n=1}^{N} \left\| \vec{\Upsilon}_n^{(t)} + G_n^2 \{\Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})\}(x_n) \right\|^2 \\ & + \sum_{n=1}^{N} \log \sigma \circ \{\Upsilon^{(t)} \circ \exp(\{E_n \vec{z}_n\})\}(x_n) \qquad \text{(B.28)}\end{aligned}$$

$$\triangleq \underset{\vec{z}_n}{\operatorname{argmin}} \sum_{n=1}^{N} f_{n1}^2(\vec{z}) + f_{n2}^2(\vec{z}) + \log f_{n3}(\vec{z}). \qquad \text{(B.29)}$$

The second term is the same as before, while the first term has become more complicated. Using the product rule and the techniques described in Appendix B.1, we can find the first derivatives of the first and second terms and estimate their second derivatives using the Gauss-Newton method. The difficulty lies in the third term, which is not quadratic and is even strictly concave, so we have to make further approximations.

Consider the problem of optimizing a one-dimensional function $f(x)$. Let the current estimate of x be x_0. Newton's optimization [125] involves the following update:

$$\triangle x = -(f'')^{-1}(x_0) f'(x_0), \tag{B.30}$$

where $f'(x_0)$ and $f''(x_0)$ are the gradient and the Hessian of f evaluated at x_0 respectively. When f'' is negative (positive), the update $\triangle x$ increases (decreases) the objective function, regardless of whether one is attempting to increase or decrease the objective function! The Gauss-Newton approximation of the Hessian for minimizing non-linear quadratic functions actually stabilizes the Newton's method by ensuring the estimated Hessian is positive.

To optimize Eq. (B.29) with Newton's method, we need to compute the gradient and the Hessian. Because we are using the l_2 inner product and the canonical metric (see Appendix B.1.3), the gradient and the Hessian correspond to the first and second derivatives. The first derivative or gradient corresponds to

$$\frac{\partial f}{\partial \vec{z}_k} = 2 f_{n1}(\vec{z}) \frac{\partial f_{n1}}{\partial \vec{z}_k} + 2 f_{n2}(\vec{z}) \frac{\partial f_{n2}}{\partial \vec{z}_k} + \frac{1}{f_{n3}} \frac{\partial f_{n3}}{\partial \vec{z}_k} \tag{B.31}$$

and the second derivative corresponds to

$$\frac{\partial^2 f}{\partial \vec{v}_k^2} = 2\left(\frac{\partial f_{n1}}{\partial \vec{z}_k}\right)^2 + 2 f_{n1}(\vec{z}) \frac{\partial^2 f_{n1}}{\partial \vec{v}_k^2} + 2\left(\frac{\partial f_{n2}}{\partial \vec{z}_k}\right)^2 + 2 f_{n2}(\vec{z}) \frac{\partial^2 f_{n2}}{\partial \vec{v}_k^2} - \left(\frac{\partial f_{n3}}{\partial \vec{z}_k}\right)^2 + \frac{1}{f_{n3}} \frac{\partial^2 f_{n3}}{\partial \vec{v}_k^2} \tag{B.32}$$

$$\approx 2\left(\frac{\partial f_{n1}}{\partial \vec{z}_k}\right)^2 + 2\left(\frac{\partial f_{n2}}{\partial \vec{z}_k}\right)^2 - \left(\frac{\partial f_{n3}}{\partial \vec{z}_k}\right)^2. \tag{B.33}$$

where the last approximation was made using the Gauss-Newton method. Not surprisingly, the third term corresponding to log is negative, which can introduce instability in the Gauss-Newton update. Consequently, we drop the last term, resulting in:

$$\frac{\partial^2 f}{\partial \vec{v}_k^2} \approx 2\left(\frac{\partial f_{n1}}{\partial \vec{z}_k}\right)^2 + 2\left(\frac{\partial f_{n2}}{\partial \vec{z}_k}\right)^2. \tag{B.34}$$

Note that the resulting update Eq. (B.30) is always in the direction of descent since the estimated second derivative is always positive. Theoretically, it is necessary to do a line search along the Gauss-Newton update direction to ensure convergence. In practice, we find that the objective function decreases reliably for each full Newton's step.

B.4 Numerics of Diffeomorphism

While v and $\Phi_v(x) = \exp(v)(x)$ are technically defined on the entire continuous image domain, in practice, v and u are represented by vector fields defined on a discrete set of points in the image, such as at each pixel [152, 168] or control points [9, 21] or in our case, vertices of a spherical mesh. From the theories of ODEs [26], we know that the integral curves or trajectories $u(t) = \Phi_{tv}(\cdot)$ of a velocity field $v(x,t)$ exist and are unique if $v(x,t)$ is Lipschitz continuous in x and continuous in t. This is true in both Euclidean spaces and on manifolds. Uniqueness means that the trajectories do not cross, implying that the deformation is invertible. Furthermore, we know from the theories of ODEs that a C^r continuous velocity field v produces a C^r continuous deformation field. Therefore, a sufficiently smooth velocity field results in a diffeomorphic transformation.

Since the velocity field v is stationary in the case of the one parameter subgroup of diffeomorphisms, v is clearly continuous (and in fact C^∞) in t. A smooth interpolation of v is continuous in the spatial domain and is Lipschitz continuous if we consider a compact domain, which holds since we only consider images that are closed and bounded.

To compute the final deformation of an image, we have to estimate $\exp(v)$ at least at the set of image grid points. We can compute $\exp(v)$ by numerically integrating the smoothly interpolated velocity field v with Euler integration. In this case, the estimate becomes arbitrarily close to the true $\exp(v)$ as the number of integration time steps increases. With a sufficiently large number of integration steps, we expect the estimate to be invertible and the resulting transformation to be diffeomorphic.

The parameterization of diffeomorphisms by a stationary velocity field is popular due to the "scaling and squaring" approach [8] for computing $\exp(v)$. Instead of Euler integration, the "scaling and squaring" approach iteratively composes displacement fields. Because we are working on the sphere S^2, the "scaling and squaring" procedure discussed in [8] has to be slightly modified:

$$\begin{aligned}
\Phi_{\frac{1}{2^K}v}(x) &= \left\{\Psi_n\left(E_n\frac{1}{2^K}v\left(x_n\right)\right)\right\}_{n=1,\cdots,N} && \text{(B.35)}\\
\Phi_{\frac{1}{2^{K-1}}v}(x) &= \Phi_{\frac{1}{2^K}v}\left(\Phi_{\frac{1}{2^K}v}\left(x\right)\right)\\
&\vdots\\
\Phi_{v}(x) &= \Phi_{\frac{1}{2}v}\left(\Phi_{\frac{1}{2}v}\left(x\right)\right), && \text{(B.36)}
\end{aligned}$$

where Ψ_n is the local coordinate chart defined in Eq. (4.10), such that $\Psi_n(0) = x_n$. Eq. (B.35) differs from "scaling and squaring" in Euclidean space. $E_n\frac{1}{2^K}v\left(x_n\right)$ is the velocity vector at the origin of $\mathbb{R}^2$ corresponding to the velocity vector $\frac{1}{2^K}v\left(x_n\right)$ at x_n. For large enough K, we can approximate a particle at the origin to move to position $E_n\frac{1}{2^K}v\left(x_n\right)$ via the flow of $E_n\frac{1}{2^K}v\left(x_n\right)$. Finally, the coordinate chart Ψ_n maps the point $E_n\frac{1}{2^K}v\left(x_n\right)$ back to the sphere. The correctness of this process follows from the fact that the solution trajectories of the ODEs of a vector field can be consistently transformed via the coordinate charts.

While "scaling and squaring" converges to the true answer as K approaches ∞ in the continuous case, in the discrete case, composition of the displacement fields requires interpolation of displacement fields, introducing errors in the process. In particular, suppose $\Phi_{t_0 v}(x)$ and $\Phi_{2t_0 v}(x)$ are the true trajectories found by performing an accurate Euler integration up to time t_0 and $2t_0$ respectively. Then, there does not exist a trivial interpolation scheme, so that $\Phi_{2t_0 v}(x) = \Phi_{t_0 v}(\Phi_{t_0 v}(x))$. In practice however, it is widely reported that in $\mathbb{R}^2$ and $\mathbb{R}^3$, "scaling and squaring" tends to preserve invertibility even with rather large deformations [9, 168].

As discussed in Appendix B.1.2, we employ barycentric interpolation, followed by normalization to ensure the warp stays on the unit sphere. In practice, we find that the resulting transformation is indeed diffeomorphic. Technically speaking, since we use linear interpolation for the displacement field, the transformation is only homeomorphic rather than diffeomorphic. This is because the transformation is continuous, but not differentiable across mesh edges. However, we follow the convention of [8, 9, 168] who call their transformation diffeomorphic even though they are homeomorphic.

Appendix C

Task-Optimal Appendix

C.1 Proof of Proposition 2

In this appendix, we prove Proposition 2 in Chapter 5. In the next section, we first prove that the Hessian $\partial^2_{v_1} f(w_0, \Gamma^*(w_0) \circ \delta\Gamma(v_1))\Big|_{v_1=0}$ is equal to the mix-derivatives matrix $\partial^2_{v_1,v_2} f(w_0, \Gamma^*(w_0) \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2))\Big|_{v_1=v_2=0}$ under the composition of diffeomorphisms model [168, 183, 186]. We then complete the proof of Proposition 2.

C.1.1 Proof of the Equivalence between the Hessian and Mix-Derivatives Matrix for the Composition of Diffeomorphisms Model

We will only provide the proof for when the image is defined in $\mathbb{R}^3$ so as not to obscur the main ideas behind the proof. To extend the proof to a manifold (e.g., S^2), one simply need to extend the notations and bookkeeping by the local parameterizing the velocity fields v_1 and v_2 using coordinate charts. The same proof follows.

Let us define some notations. Suppose the image and there are M voxels. Let $\vec{x}$ be the $\mathbb{R}^{3M}$ rasterized coordinates of the M voxels. For conciseness, we define for the fixed parameters w_0,

$$p(\vec{x}) \triangleq f(w_0, \Gamma^*(w_0)(\vec{x})) \tag{C.1}$$

Therefore p is a function from $\mathbb{R}^{3M}$ to $\mathbb{R}$. Under the composition of diffeomorphisms model, $\delta\Gamma(v)$ is the diffeomorphism parameterized by the stationary velocity field v defined on the M voxels, so that $\delta\Gamma(v)(\cdot)$ is a function from $\mathbb{R}^{3M}$ to $\mathbb{R}$. To make the dependence of $\delta\Gamma(v)$ on v explicit, we define

$$\Upsilon(v, \vec{x}) \triangleq \delta\Gamma(v)(\vec{x}), \tag{C.2}$$

and so Υ is a function from $\mathbb{R}^{3M} \times \mathbb{R}^{3M}$ to $\mathbb{R}^{3M}$. In other words, we can rewrite

$$\partial^2_{v_1} f(w_0, \Gamma^*(w_0) \circ \delta\Gamma(v_1)) = \partial^2_{v_1} p(\Upsilon(v_1, \vec{x})) \tag{C.3}$$

and

$$\partial^2_{v_1,v_2} f(w_0, \Gamma^*(w_0) \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2)) = \partial^2_{v_1,v_2} p(\Upsilon_1(v_1, \Upsilon_2(v_2, \vec{x}))). \tag{C.4}$$

Now that we have gotten the notations out of the way, we will now show that

$$\partial^2_{v_1} p(\Upsilon(v_1, \vec{x}))\Big|_{v_1=0} = \partial^2_{v_1,v_2} p(\Upsilon_1(v_1, \Upsilon_2(v_2, \vec{x})))\Big|_{v_1=v_2=0} = \partial^2_{\vec{x}} p(\vec{x}) \tag{C.5}$$

Hessian. We first compute the $1 \times 3M$ Jacobian via the chain rule

$$\partial_{v_1} p(\Upsilon(v_1, \vec{x})) = (\partial_\Upsilon p)\,(\partial_{v_1} \Upsilon)\,. \tag{C.6}$$

From the above equation, we can equivalently write down the j-th component of the $1 \times 3M$ Jacobian:

$$\partial_{v_1} p(\Upsilon(v_1, \vec{x}))\,(j) = \sum_n (\partial_{\Upsilon^n} p)\,\left(\partial_{v_1^j} \Upsilon^n\right), \tag{C.7}$$

where Υ^n and v_1^j denote the n-th and j-th components of Υ and v_1 respectively. Now, we compute the (i,j)-th component of the $3M \times 3M$ Hessian using the product rule

$$\partial^2_{v_1} p(\Upsilon(v_1, \vec{x}))\Big|_{v_1=0} (i,j)$$

$$= \partial_{v_1^i} \sum_n (\partial_{\Upsilon^n} p)\,\left(\partial_{v_1^j} \Upsilon^n\right)\Big|_{v_1=0} \tag{C.8}$$

$$= \sum_n \left[\left(\partial^2_{v_1^i,\Upsilon^n} p\right)\left(\partial_{v_1^j} \Upsilon^n\right) + (\partial_{\Upsilon^n} p)\left(\partial^2_{v_1^i v_1^j} \Upsilon^n\right)\right]\Big|_{v_1=0} \tag{C.9}$$

$$= \sum_{n,k} \left(\partial^2_{\Upsilon^k \Upsilon^n} p\right)\left(\partial_{v_1^i} \Upsilon^k\right)\left(\partial_{v_1^j} \Upsilon^n\right)\Big|_{v_1=0} + \sum_n (\partial_{\Upsilon^n} p)\left(\partial^2_{v_1^i v_1^j} \Upsilon^n\right)\Big|_{v_1=0} \tag{C.10}$$

Because $\partial_{v_1} \Upsilon\Big|_{v_1=0}$ is the identity matrix and the $1 \times 3M$ Jacobian $\partial_{v_1} p(\Upsilon(v_1, \vec{x}))\Big|_{v_1=0} = (\partial_\Upsilon p)\,(\partial_{v_1} \Upsilon)\Big|_{v_1=0} = 0$ (because derivative is zero at local minimum), we get $\partial_\Upsilon p\Big|_{v_1=0} = 0$, and so the second term in Eq. (C.10) is zero.

To simplify the first term of Eq. (C.10), we once again use the fact that $\partial_{v_1} \Upsilon\Big|_{v_1=0}$ is the identity matrix, and so the summand is zero unless $k = i$ and $n = j$. Consequently,

Eq. (C.10) simplifies to

$$\partial^2_{v_1} p(\Upsilon(v_1,\vec{x}))\Big|_{v_1=0}(i,j) = \partial^2_{\Upsilon^i\Upsilon^j} p \tag{C.11}$$

or equivalently,

$$\partial^2_{v_1} p(\Upsilon(v_1,\vec{x}))\Big|_{v_1=0} = \partial^2_{\vec{x}} p(\vec{x}) \tag{C.12}$$

Mix-Derivatives Matrix. We first compute the $1\times 3M$ Jacobian via the chain rule

$$\begin{align}
\partial_{v_2} p(\Upsilon_1(v_1,\Upsilon_2(v_2,\vec{x})))\Big|_{v_2=0} &= (\partial_{\Upsilon_1} p)\,(\partial_{\Upsilon_2}\Upsilon_1)\,(\partial_{v_2}\Upsilon_2)\Big|_{v_2=0} \tag{C.13}\\
&= (\partial_{\Upsilon_1} p)\,(\partial_{\vec{x}}\Upsilon_1(v_1,\vec{x}))\,. \tag{C.14}
\end{align}$$

From the above equation, we can equivalently write down the j-th component of the $1\times 3M$ Jacobian:

$$\partial_{v_2} p(\Upsilon_1(v_1,\Upsilon_2(v_2,\vec{x})))\Big|_{v_2=0}(j) = \sum_n \left(\partial_{\Upsilon_1^n} p\right)\,\left(\partial_{\vec{x}^j}\Upsilon_1^n\right), \tag{C.15}$$

Now, we compute the (i,j)-th component of the $3M\times 3M$ mix-derivatives matrix using the product rule

$$\begin{align}
&\partial^2_{v_1,v_2} p(\Upsilon_1(v_1,\Upsilon_2(v_2,\vec{x})))\Big|_{v_1=v_2=0}(i,j) \notag\\
&= \partial_{v_1^i}\sum_n \left(\partial_{\Upsilon_1^n} p\right)\,\left(\partial_{\vec{x}^j}\Upsilon_1^n\right)\Big|_{v_1=v_2=0} \notag\\
&= \sum_n \left[\left(\partial^2_{v_1^i,\Upsilon_1^n} p\right)\left(\partial_{\vec{x}^j}\Upsilon_1^n\right) + \left(\partial_{\Upsilon_1^n} p\right)\left(\partial^2_{v_1^i,\vec{x}^j}\Upsilon_1^n\right)\right]\Big|_{v_1=v_2=0} \tag{C.16}\\
&= \sum_{n,k}\left(\partial^2_{\Upsilon_1^k,\Upsilon_1^n} p\right)\left(\partial_{v_1^i}\Upsilon_1^k\right)\left(\partial_{\vec{x}^j}\Upsilon_1^n\right) + \sum_n \left(\partial_{\Upsilon_1^n} p\right)\left(\partial^2_{v_1^i,\vec{x}^j}\Upsilon_1^n\right)\Big|_{v_1=v_2=0} \tag{C.17}
\end{align}$$

Like before, we have $\partial_{\Upsilon} p\Big|_{v_1=v_2=0} = 0$, and so the second term is zero. Because $\partial_{v_1}\Upsilon\Big|_{v_1=0}$ is the identity, $\partial_{v_1^i}\Upsilon_1^n$ is zero unless $k=i$. Since $\Upsilon_1^n(v_1=0,\vec{x})=\vec{x}$, $\partial_{\vec{x}^j}\Upsilon_1^n$ is also equal to zero unless $n=j$. Therefore, we get

$$\partial^2_{v_1,v_2} p(\Upsilon_1(v_1,\Upsilon_2(v_2,\vec{x})))\Big|_{v_1=v_2=0}(i,j) = \partial^2_{\Upsilon_1^i,\Upsilon_1^j} p \tag{C.18}$$

or equivalently,

$$\partial^2_{v_1,v_2} p(\Upsilon_1(v_1,\Upsilon_2(v_2,\vec{x})))\Big|_{v_1=v_2=0} = \partial^2_{\vec{x}} p(\vec{x}) \tag{C.19}$$

C.1.2 Completing the Proof of Proposition 2

We now complete the proof of Proposition2. Let $h(w, v_1) \triangleq \partial_{v_2} f(w, \Gamma^*(w_0) \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2))\Big|_{v_2=0}$. Since $\delta\Gamma(0) = \mathrm{Id}$, we have

$$\begin{aligned}
h(w, v_1)\Big|_{w_0,0} &= \partial_{v_2} f(w, \Gamma^*(w_0) \circ \delta\Gamma(0) \circ \delta\Gamma(v_2))\Big|_{v_2=0} && \text{(C.20)}\\
&= \partial_{v_2} f(w, \Gamma^*(w_0) \circ \delta\Gamma(v_2))\Big|_{v_2=0} && \text{(C.21)}\\
&= 0 && \text{(C.22)}
\end{aligned}$$

where the last equality comes from the definition of $\Gamma^*(w_0)$ being a local minimum for the composition model.

Since the mix-derivatives matrix $\partial_{v_1} h(w, v_1)\big|_{v_1=0}$ is invertible by the positive-definite assumption of this proposition, by the Implicit Function Theorem, there exists an $\epsilon > 0$, such that for all δw, $\|\delta w\| < \epsilon$, there is a unique continuous function $v^*(w_0, \delta w)$, such that $h(w_0 + \delta w, v^*(w_0, \delta w)) = 0$ and $v^*(w_0, 0) = 0$. Furthermore, $v^*(w_0, \delta w)$ has the same order of smoothness as f.

Let $k(w, v_1) = \partial^2_{v_2} f(w, \Gamma^*(w_0) \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2))\Big|_{v_2=0}$. Then $k(w_0, 0)$ is positive definite at $v_1 = 0$ by the assumption of the proposition. By the smoothness of derivatives and continuity of eigenvalues, there exists a small neighborhood around $(w_0, v_1 = 0)$ in which the eigenvalues of $k(w, v_1)$ are all greater than zero. Therefore $\Gamma^*(w_0) \circ \delta\Gamma(v^*(w_0, \delta w))$ does indeed define a new local minimum close to $\Gamma^*(w_0)$. □

C.2 Computing the derivative $\partial_w v^*$

To compute $\partial_w v^*$, we perform a Taylor expansion:

$$\partial_{v_2} f(w, \Gamma^* \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2))\Bigg|_{w_0+\delta w, v_1, v_2=0}$$

$$= \Big[\partial^2_{v_1,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2)) v_1 + \partial^2_{w,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2)) \delta w + O(\delta w^2, v_1^2)\Big]\Bigg|_{w_0, v_1=0, v_2=0} \tag{C.23}$$

$$= \Big[\partial^2_{v_1,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2)) v_1 + \partial^2_{w,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_2)) \delta w + O(\delta w^2, v_1^2)\Big]\Bigg|_{w_0, v_1=0, v_2=0} \tag{C.24}$$

and rearranging the terms for $v_1 = v^*$, we get

$$\partial_w v^* = -\left(\partial^2_{v_1,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_1) \circ \delta\Gamma(v_2))\right)^{-1} \partial^2_{w,v_2} f(w, \Gamma^* \circ \delta\Gamma(v_2))\Bigg|_{w_0, v_1=0, v_2=0} \tag{C.25}$$

C.3 Groupwise Formulation of Task Performance G

In this appendix, we present an alternative groupwise formulation of the task performance G and experiments on localizing Brodmann Areas (BAs). Here, our task is to define a common coordinate system, such that the hidden labels of subjects aligned to this coordinate system are tightly aligned. Consequently, we define the following task-performance that minimizes the variance of the distance transforms across training subjects after registration:

$$G(\{\Gamma_k^*(w)\}) = \sum_n g_n(\Gamma_k^*(w)) = \frac{1}{N-1} \sum_n \sum_{k \neq n} \sum_i \left[L_n\left(\Gamma_n^*(x_i)\right) - L_k\left(\Gamma_k^*\left(x_i\right)\right)\right]^2. \tag{C.26}$$

Given a test subject, we expect the test subject's hidden label to be well-predicted by those of the training subjects.

C.3.1 Optimizing Groupwise Task Performance G

Notice that in this case, the registration outputs $\{\Gamma_k^*\}$ of the different subjects are coupled across the different task-specific cost functions g_n. Consequently, we have to modify the gradient of the task-performance cost function presented in Eq. (5.16). To compute the derivative of the task performance with respect to the parameters w, we apply the chain rule to get

$$\partial_w G = -\sum_n \partial_{\vec{v}^*} G(\Gamma_n^* \circ \delta\Gamma(\vec{v}^*)) \partial^2_{w,\vec{v}} f_n(w, \Gamma_n^* \circ \delta\Gamma(\vec{v}))\Big|_{w,\vec{v}=\vec{v}^*=0}. \tag{C.27}$$

Since the registration cost function is the same as the pairwise formulation in Chapter 5.2, the approximate derivative of the registration local minimum with respect to the parameters $-\partial^2_{w,\vec{v}} f_n(w, \Gamma_n^* \circ \delta\Gamma(\vec{v}))\Big|_{w,\vec{v}=0}$ is the same as before. We only need to concern ourselves with $\partial_{\vec{v}^*} G(\Gamma_n^* \circ \delta\Gamma(\vec{v}^*))\Big|_{w,\vec{v}^*=0}$:

$$\partial_{\vec{v}^*} G(\Gamma_n^* \circ \delta\Gamma(\vec{v}^*))\Big|_{w,\vec{v}^*=0} = \frac{2}{N-1} \partial_{\vec{v}^*} \sum_{k \neq n} \sum_i \left[L_n\left(\Gamma_n^* \circ \delta\Gamma(\vec{v}^*)(x_i)\right) - L_k\left(\Gamma_k^*\left(x_i\right)\right)\right]^2 \Big|_{w,\vec{v}^*=0} \tag{C.28}$$

Once again, we represent $\vec{v}^*$ as the collection $\{\vec{v}_i^*\}$, where $\vec{v}_i^*$ is a velocity vector at vertex x_i. Defining $\nabla L_n(\Gamma_n^*(x_i))^T$ to be the 1×3 spatial gradient of the warped distance transform of the n-th subject $L_n(\Gamma_n^*(\cdot))$ at x_i, we get the 1×3 derivative

$$\partial_{\vec{v}_i^*} G(\Gamma_n^* \circ \delta\Gamma(\vec{v}^*))\Big|_{w,\vec{v}^*=0} = \frac{4}{N-1} \sum_{k \neq n} \left[L_n\left(\Gamma_n^*(x_i)\right) - L_k\left(\Gamma_k^*\left(x_i\right)\right)\right] \nabla L_n(\Gamma_n^*(x_i))^T \tag{C.29}$$

$$= 4 \left(L_n\left(\Gamma_n^*(x_i)\right) - \frac{1}{N-1} \sum_{k \neq n} L_k\left(\Gamma_k^*\left(x_i\right)\right) \right) \nabla L_n(\Gamma_n^*(x_i))^T \tag{C.30}$$

Observe that Eq. (C.30) is the same (up to a factor of 2) as its pairwise counterpart Eq. (5.23) by replacing the distance transform of the template subject with the average distance transform of the other training subjects ($k \neq n$).

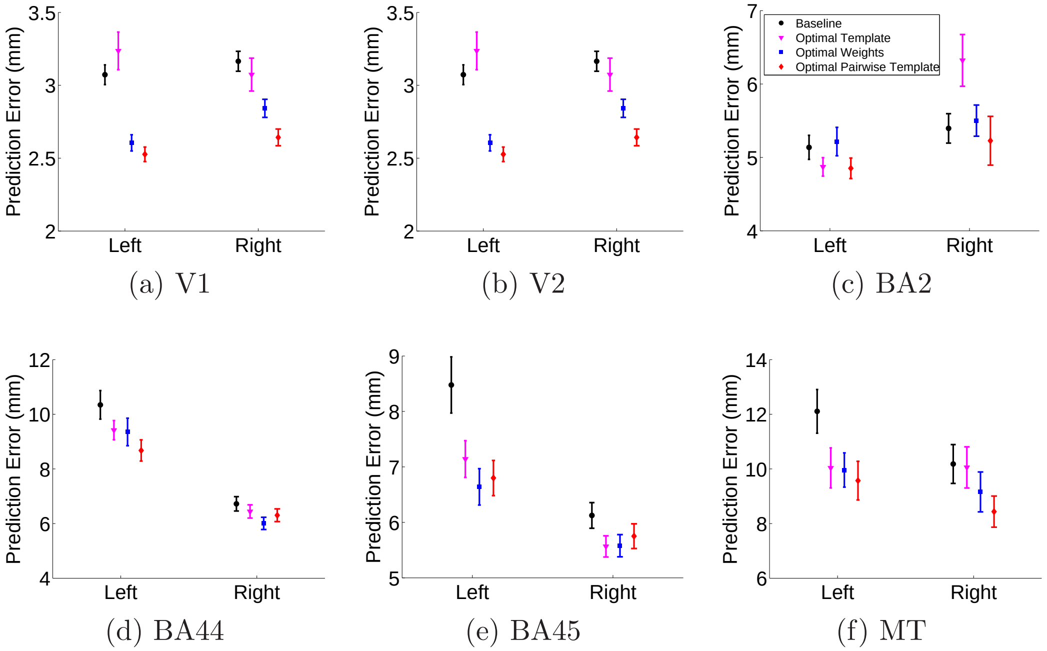
Therefore the derivative of the groupwise task performance G in Eq. (C.26) with respect to the weights $\{\lambda_j\}$ of the wSSD is given by substituting Eq. (C.30) and Eq. (5.27) into Eq. (C.27). The derivative of the groupwise task performance G in Eq. (C.26) with respect to the template T is given by substituting Eq. (C.30) and Eq. (5.31) into Eq. (C.27).

C.3.2 Brodmann Area (BA) Localization

We perform leave-one-out on the ten *ex-vivo* brains (from Chapter 5.3) to predict BA localization. For each BA and each test subject, we use the remaining 9 subjects for learning either the weights λ_j's or the template image T. Once the weights or template is estimated, all the 10 subjects are registered to the learned template space and each of the 9 training subjects is used to predict the BA of the leave-one-out subject. This results in a total of 90 Hausdorff distance per BA, per hemisphere for the entire experiment. Just like in Chapter 5.3, the weights are initialized to 1 and the initial template image is that of average of the 40 *in-vivo* subjects constituting the FreeSurfer Buckner40 atlas.

Fig. C-1 reports the resulting mean and standard errors of the Hausdorff distances of V1, V2, BA2, BA44, BA45 and MT. The magenta and blue correspond to task-optimal groupwise template and weights respectively. Baseline 1 corresponds to the use of global weights of 1 and FreeSurfer Buckner40 template. We see that the task-optimal cost functions are generally better than the baseline.

A second observation is that the optimal groupwise weights performs slightly better than the optimal groupwise template. This is in contrast with the pairwise experiments in Chapter 5.3, where the optimal pairwise template performs better than optimal pairwise weights. Because baseline 1 is in general better than the uniform weights baseline in Chapter 5.3, one might speculate that there is simply not much room for improvements with regards to the template. Indeed, we find that training for the optimal groupwise template tends to terminate more quickly than the training of the optimal groupwise weights. However, when we plot the results of the optimal pairwise template from Chapter 5.3 in red in Fig. C-1, we



(a) V1 (b) V2 (c) BA2

(d) BA44 (e) BA45 (f) MT

Figure C-1: Mean Hausdorff distances for V1, V2, BA2, BA44, BA45 and MT.

find that the optimal pairwise template performs the best. This suggests that the average template is a poor local minimum which our gradient descent algorithm is unable to escape from, possibly because the approximation of the actual derivative in Eq. (5.13) leads to premature termination of the gradient descent algorithm.

Bibliography

[1] D. Alexander and J. Gee. Elastic Matching of Diffusion Tensor Images. *Computer Vision and Image Understanding*, 77:233–250, 2000. 25

[2] S. Allassonniere, Y. Amit, and A. Trouvé. Toward a Coherent Statistical Framework for Dense Deformable Template Estimation. *Journal of the Royal Statistical Society, Series B*, 69(1):3–29, 2007. 27, 29, 33, 45, 98, 114

[3] E. Allgower and K. Georg. *Introduction to Numerical Continuation Methods*. Society for Industrial and Applied Mathematics, 2003. 33

[4] K. Amunts, A. Malikovic, H. Mohlberg, T. Schormann, and K. Zilles. Brodmann's Areas 17 and 18 Brought into Stereotaxic Space - Where and How Variable? *NeuroImage*, 11:66–84, 2000. 17, 20, 88

[5] K. Amunts, A. Schleicher, U. Burgel, H. Mohlberg, H. Uylings, and K. Zilles. Broca's Region Revisited: Cytoarchitecture and Intersubject Variability. *Journal of Comparative Neurology*, 412(2):319–341, 1999. 17, 21, 88

[6] J. Andersson, C. Hutton, J. Ashburner, R. Turner, and K. Friston. Modeling Geometric Deformations in EPI Time Series. *NeuroImage*, 13(5):903–919, 2001. 25

[7] B. Ardekani, A. Bachman, S. Strother, Y. Fujibayashi, and Y. Yonekura. Impact of Inter-subject Image Registration on Group Analysis of fMRI data. In *Proceedings of the International Workshop on Quantitation in Biomedical Imaging with PET and MRI*, volume 1265 of *International Congress Series*, pages 49–59, 2004. 25

[8] V. Arsigny, O. Commowick, X. Pennec, and N. Ayache. A Log-Euclidean Framework for Statistics on Diffeomorphisms. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 4190 of *LNCS*, pages 924–931, 2006. 64, 116, 117

[9] J. Ashburner. A Fast Diffeomorphic Image Registration Algorithm. *NeuroImage*, 38:95–113, 2007. 30, 31, 115, 117

[10] J. Ashburner, J. Andersson, and K. Friston. High-Dimensional Image Registration using Symmetric Priors. *NeuroImage*, 9:619–628, 1999. 25, 30, 44

[11] J. Ashburner and K. Friston. Nonlinear Spatial Normalization using Basis Functions. *Human Brain Mapping*, 7:254–266, 1999. 25

[12] J. Ashburner and K. Friston. Unified segmentation. *NeuroImage*, 26:839–851, 2005. 28, 32, 41, 98

[13] J. Ashburner, C. Hutton, R. Frackowiak, I. Johnsrude, C. Price, and K. Friston. Identifying Global Anatomical Differences: Deformation-based Morphometry. *Human Brain Mapping*, 6:348–357, 1998. 16

[14] J. Augustinack, A. Van der Kouwe, M. Blackwell, D. Salat, C. Wiggins, M. Frosch, G. Wiggins, A. Potthast, L. Wald, and B. Fischl. Detection of Entorhinal Layer ii Using 7 Tesla Magnetic Resonance Imaging. *Annals of Neurology*, 57(4):489–494, 2005. 20

[15] B. Avants, C. Epstein, M. Grossman, and J. Gee. Symmetric Diffeomorphic Image Registration with Cross-Correlation: Evaluating Automated Labeling of Elderly and Neurodegenerative Brain. *Medical Image Analysis*, 12(1):26–41, 2008. 34

[16] B. Avants and J. Gee. Geodesic Estimation for Large Deformation Anatomical Shape Averaging and Interpolation. *NeuroImage*, 23:139–150, 2004. 25, 27, 30, 31, 41, 89, 95

[17] S. Baker and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework. *International Journal on Computer Vision*, 56(3):221–255, 2004. 79

[18] M. Bakircioglu, S. Joshi, and M. Miller. Landmark Matching on Brain Surfaces via Large Deformation Diffeomorphisms on the Sphere. In *Proc. SPIE Medical Imaging*, volume 3661, pages 710–715, 1999. 56

[19] S. Balci, P. Golland, M. Shenton, and W. Wells. Free-Form B-spline Deformation Model for Groupwise Registration. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Statistical Registration Workshop: Pair-wise and Group-wise Alignment and Atlas Formation*, pages 23–30, 2007. 28

[20] P. Bazin and D. Pham. Homeomorphic Brain Image Segmentation with Topological and Statistical Atlases. *Medical Image Analysis*, 12(5):616–625, 2008. 28

[21] M. Beg, M. Miller, A. Trouvé, and L. Younes. Computing Large Deformation Metric Mappings via Geodesic Flows of Diffeomorphisms. *International Journal on Computer Vision*, 61(2):139–157, 2005. 30, 31, 74, 81, 115

[22] Y. Benjamini and Y. Hochberg. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society B*, 57(1):256–300, 1995. 53, 69

[23] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society B*, 48(3):259–302, 1986. 51

[24] J. Bezdek. Review of MR Image Segmentation Techniques using Pattern Recognition. *Medical Physics*, 20(4):1033–1048, 1993. 25

[25] K.K. Bhatia, J.V. Hajnal, B.K. Puri, A.D. Edwards, and D. Rueckert. Consistent groupwise non-rigid registration for atlas construction. In *Proceedings of the International Symposium on Biomedical Imaging: From Nano to Macro*, volume 1, pages 908–911, 2004. 29

[26] G. Birkhoff and G. Rota. *Ordinary Differential Equations*. John Wiley and Sons Inc, 1978. 30, 116

[27] D. Blezek and J. Miller. Atlas Stratification. *Medical Image Analysis*, 11(5):443–457, 2007. 27

[28] M. Boucher, S. Whitesides, and A. Evans. Depth Potential Function for Folding Pattern Representation, Registration and Analysis. *Medical Image Analysis*, 13(2):203–214, 2009. 18

[29] M. Bro-Nielsen and C. Gramkow. Fast Fluid Registration of Medical Images. *Proceedigs Visualization in Biomedical Computing*, 1131:267–276, 1996. 66, 74

[30] K. Brodmann. *Vergleichende Lokalisationslehre der Grohirnrinde in Ihren Prinzipien Dargestellt auf Grund des Zellenbaues*. 1909. 18, 20

[31] L. Brown. A Survey of Image Registration Techniques. *ACM Computing Surveys (CSUR)*, 24(4):325–376, 1992. 15, 25, 34

[32] P. Cachier, E. Bardinet, D. Dormont, X. Pennec, and N. Ayache. Iconic Feature Based Non-Rigid Registration: The PASHA algorithm. *Compute Vision and Image Understanding*, 89(2-3):272–298, 2003. 31, 34, 56, 58, 64, 66, 74

[33] C. Chefd'hotel. A Method for the Transport and Registration of Images on Implicit Surfaces. In *Scale Space and Variational Methods in Computer Vision*, volume 4485 of *LNCS*, pages 860–870, 2007. 74

[34] G. Christensen, R. Rabbit, and M. Miller. Deformable Templates Using Large Deformation Kinematics. *IEEE Transactions on Image Processing*, 5(10):1435–1447, 1996. 74

[35] M. Chung, S. Robbins, K. Dalton, R. Davidson, A. Alexander, and A. Evans. Cortical Thickness Analysis in Autism with Heat Kernel Smoothing. *NeuroImage*, 25(4):1256–1265, 2005. 112

[36] M. Chung, K. Worsley, S. Robbins, T. Paus, J. Taylor, J. Giedd, J. Rapoport, and A. Evans. Deformation-based Surface Morphometry Applied to Gray Matter Deformation. *NeuroImage*, 18(2):198–213, 2003. 16

[37] D. Collins, W. Baaré A. Zijdenbos, and A. Evans. Animal+insect: Improved cortical structure segmentation. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 1613 of *LNCS*, pages 210–223, 1999. 28

[38] D. Collins, AC. Evans, C. Holmes, and T. Peters. Automatic 3D segmentation of neuro-anatomical structures from MRI. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, LNCS, pages 135–152, 1995. 28

[39] O. Commowick, R. Stefanescu, P. Fillard, V. Arsigny, N. Ayache, X. Pennec, and G. Malandain. Incorporating statistical measures of anatomical variability in atlas-to-subject registration for conformal brain radiotherapy. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 3750 of *LNCS*, pages 927–934, 2005. 32, 34

[40] M. De Craene, A. du Bois d'Aische, B. Macq, and S. Warfield. Multi-subject registration for unbiased statistical atlas construction. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 3216 of *LNCS*, pages 655–662, 2004. 29

[41] D. Cremers, M. Rousson, and R. Deriche. A Review of Statistical Approaches to Level Set Segmentation: Integrating Color, Texture, Motion and Shape. *International Journal on Computer Vision*, 72(2):195–215, 2007. 25

[42] A. Dale, B. Fischl, and M. Sereno. Cortical Surface-Based Analysis I: Segmentation and Surface Reconstruction. *NeuroImage*, 9(2):179–194, 1999. 18, 21, 22

[43] R. Desikan, F. Segonne, B. Fischl, B. Quinn, B. Dickerson, D. Blacker, R. Buckner, A. Dale, R. Maguire, B. Hyman, M. Albert, and R. Killiany. An Automated Labeling System for Subdividing the Human Cerebral Cortex on MRI Scans into Gyral Based Regions of Interest. *NeuroImage*, 31(3):968–980, 2006. 19, 28, 37, 43, 51, 54, 68

[44] S. Dinggang and C. Davatzikos. HAMMER: Hierarchical Attribute Matching Mechanism for Elastic Registration. *IEEE Transactions on Medical Imaging*, 21(11):1421–1439, 2002. 29

[45] M. Dubuisson and A. Jain. A Modified Hausdorff Distance for Object Matching. In *Proceedings of the 12th IAPR International Conference on Pattern Recognition*, volume 1, pages 566–568, 1994. 72, 88

[46] S. Durrleman, X. Pennec, A. Trouvé, P., and N. Ayache. Inferring brain variability from diffeomorphic deformations of currents: an integrative approach. *Medical Image Analysis*, 12(5):626–637, 2008. PMID: 18658005. 30, 31, 56

[47] I. Eckstein, A. Joshi, C. Jay Kuo, R. Leahy, and M. Desbrun. Generalized Surface Flows for Deformable Registration and Cortical Matching. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 4791 of *LNCS*, pages 692–700, 2007. 30, 56

[48] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least Angle Regression. *Annals of statistics*, pages 407–451, 2004. 34

[49] S. Eickhoff et al. A new SPM Toolbox for Combining Probabilistic Cytoarchitectonic Maps and Functional Imaging Data. *NeuroImage*, 25(4):1325–1335, 2005. 20

[50] D. Van Essen. A Population-Average, Landmark- and Surface-based (PALS) atlas of human cerebral cortex. *NeuroImage*, 28(3):635–662, 2005. 30

[51] A.C. Evans, D.L. Collins, S.R. Mills, E.D. Brown, R.L. Kelly, and T.M. Peters. 3d Statistical Neuroanatomical Models from 305 MRI Volumes. In *Nuclear Science Symposium and Medical Imaging Conference*, volume 3 of *IEEE Conference Record*, pages 1813–1817, 1993. 28, 37

[52] T. Evgeniou, M. Pontil, and T. Poggio. Regularization networks and support vector machines. In *Advances in Computational Mathematics*, pages 1–50. MIT Press, 2000. 78, 98

[53] P. Fillard, V. Arsigny, X. Pennec, K. Hayashi, P. Thompson, and N. Ayache. Measuring Brain Variability by Extrapolating Sparse Tensor Fields Measured on Sulcal Lines. *Neuroimage*, 34(2):639–650, 2007. 56

[54] B. Fischl and A. Dale. Measuring the Thickness of the Human Cerebral Cortex from Magnetic Resonance Images. *Proceedings of the National Academy of Science*, 97(20):11050–110555, 2000. 30

[55] B. Fischl, A. Liu, and A. Dale. Automated manifold surgery: Constructing geometrically accurate and topologically correct models of the human cerebral cortex. *IEEE Transactions on Medical Imaging*, 20(1):70–80, 2001. 19, 43

[56] B. Fischl, N. Rajendran, E. Busa, J. Augustinack, O. Hinds, B.T.T. Yeo, H. Mohlberg, K. Amunts, and K. Zilles. Cortical Folding Patterns and Predicting Cytoarchictecture. *Cerebral Cortex*, 18(8):1973–1980, 2008. 17, 20, 26, 70, 88, 89

[57] B. Fischl, D. Salat, E. Busa, M. Albert, M. Dieterich, C. Haselgrove, A. van der Kouwe, R. Killiany, D. Kennedy, S. Klaveness, A. Montillo, N. Makris, B. Rosen, and A. Dale. Whole brain segmentation: Automated labeling of neuroanatomical structures in the human brain. *Neuron*, 33(3):341–355, 2002. 28, 29

[58] B. Fischl, M. Sereno, and A. Dale. Surface-Based Analysis II: Inflation, Flattening, and a Surface-Based Coordinate System. *NeuroImage*, 9(2):195–207, 1999. 18, 19, 29, 43, 44

[59] B. Fischl, M. Sereno, R. Tootell, and A. Dale. High-resolution Intersubject Averaging and a Coordinate System for the Cortical Surface. *Human Brain Mapping*, 8(4):272–284, 1999. 18, 22, 25, 26, 27, 30, 33, 56, 67, 68, 74, 88, 89, 102

[60] B. Fischl, A. van der Kouwe, C. Destrieux, E. Halgren, F. Segonne, D. Salat, E. Busa, L. Seidman, J. Goldstein, D. Kennedy, V. Caviness, N. Makris, B. Rosen, and A. Dale. Automatically Parcellating the Human Cerebral Cortex. *Cerebral Cortex*, 14:11–22, 2004. 19, 28, 29, 37, 43, 45, 46, 47, 51, 54, 68, 101, 102

[61] N. Fox, R. Jenkins, S. Leary, V. Stevenson, N. Losseff, W. Crum, R. Harvey, M. Rossor, D. Miller, and A. Thompson. Progressive Cerebral Atrophy in MS: A Serial Study using Registered, Volumetric MRI. *Neurology*, 54:807–812, 2000. 25

[62] *http://surfer.nmr.mgh.harvard.edu/fswiki*. 22, 37, 89

[63] K. Friston, A. Holmes, K. Worsley, J-P. Poline, C. Frith, and R. Frackowiak. Statistical Parametric Maps in Functional Imaging: A General Linear Approach. *Human Brain Mapping*, 2(4):189–210, 1995. 22

[64] J. Gee, D. Alsop, and G. Aguirre. Effect of Spatial Normalization on Analysis of Functional Data. In *Proceedings of SPIE Medical Imaging*, pages 312–322, 1997. 25

[65] X. Geng, G. Christensen, H. Gu, T. Ross, and Y. Yang. Implicit Reference-based Group-wise Image Registration and its Application to Structural and Functional MRI. *NeuroImage*, 47(4):1341–1351, 2009. 89

[66] X. Geng, D. Kumar, and G. Christensen. Transitive Inverse-Consistent Manifold Registration. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 3564 of *LNCS*, pages 468–479, 2005. 30

[67] J. Glaunès, M. Vaillant, and M. Miller. Landmark Matching via Large Deformation Diffeomorphisms on the Sphere. *Journal of Mathematical Imaging and Vision*, 20:179–200, 2004. 30, 31, 44, 56, 64, 65, 109, 110

[68] B. Glocker, N. Komodakis, N. Navab, G. Tziritas, and N. Paragios. Dense Registration with Deformation Priors. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 5636 of *LNCS*, pages 540–551, 2009. 34

[69] A. Guimond, J. Meunier, and J.-P. Thirion. Average brain models: A convergence study. *Computer Vision and Image Understanding*, 77(2):192–210, 2000. 29, 89

[70] D. Hagler, A. Saygin, and M. Sereno. Smoothing and Cluster Thresholding for Cortical Surface-Based Group Analysis for fMRI Data. *NeuroImage*, 33(4):1093–1103, 2006. 112

[71] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu. The Entire Regularization Path for the Support Vector Machine. 5:1391–1415, 2004. 34

[72] R. Heckemann, J. Hajnal, P. Aljabar, D. Rueckert, and A. Hammers. Automatic anatomical brain mri segmentation combining label propagation and decision fusion. *NeuroImage*, 33(1):115–126, 2006. 95

[73] M. Hernandez, M. Bossa, and S. Olmos. Registration of Anatomical Images Using Geodesic Paths of Diffeomorphisms Parameterized with Stationary Velocity Fields. In *Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, International Conference on Computer Vision*, pages 1–9, 2007. 31

[74] M. Hernandez and S. Olmos. Gauss-Newton Optimization in Diffeomorphic Registration. In *Proceedings of the International Symposium on Biomedical Imaging: From Nano to Macro*, pages 1083–1086, 2008. 31, 74

[75] D. Hill, P. Batchelor, M. Holden, and D. Hawkes. Medical Image Registration. *Physics in Medicine and Biology*, 46:R1–R45, 2001. 25

[76] L. Ibanez, M. Audette, B.T.T. Yeo, and P. Golland. Rotational Registration of Spherical Surfaces Represented as QuadEdge Meshes. *Insight Journal*, 2009 January–June (http://hdl.handle.net/10380/3063). 55

[77] L. Ibanez, M. Audette, B.T.T. Yeo, and P. Golland. Spherical Demons Registration of Spherical Surfaces. *Insight Journal*, 2009 July–December (http://hdl.handle.net/10380/3117). 55

[78] L. Ibanez, B.T.T. Yeo, and P. Golland. Iterative Smoothing of Field Data in Spherical Meshes. *Insight Journal*, 2009 July–December (http://hdl.handle.net/10380/3091). 55

[79] T. Jaakkola. *In Advanced Mean Field Methods: Theory and Practice*. MIT Press, 2000. 42, 46, 51, 101

[80] S. Jaume, M. Ferrant, A. Schreyer, L. Hoyte, B. Macq, J. Fielding, R. Kikinis, and S. Warfield. Multiresolution Signal Processing on Meshes for Automatic Pathological Shape Characterization. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 2208 of *LNCS*, pages 1398–1400, 2001. 50

[81] S. Jaume, M. Ferrant, S. Warfield, and B. Macq. Multiresolution Parameterization of Meshes for Improved Surface-based Registration. In *Proceedings of SPIE Medical Imaging*, volume 4322, pages 633–642, 2001. 30

[82] S. Jaume, B. Macq, and S. Warfield. Labeling the Brain Surface Using a Deformable Multiresolution Mesh. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 2488 of *LNCS*, pages 451–458, 2002. 30

[83] R. Jirousek and S. Preucil. On the effective implementation of the iterative proportional fitting procedure. *Computational Statistics and Data Analysis*, 19:177–189, 1995. 102

[84] A. Joshi, R. Leahy, A. Toga, and D. Shattuck. A Framework for Brain Registration via Simultaneous Surface and Volume Flow. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 5636 of *LNCS*, pages 576–588, 2009. 30

[85] A. Joshi, D. Shattuck, P. Thompson, and R. Leahy. Surface Constrained Volumetric Brain Registration Using Harmonic Mappings. *IEEE Transactions on Medical Imaging*, 26(12):1657–1669, 2007. 30

[86] S. Joshi, B. Davis, M. Jomier, and G. Gerig. Unbiased Diffeomorphic Atlas Construction for Computational Anatomy. *NeuroImage*, 23:151–160, 2004. 25, 27, 29, 33, 41, 44, 88, 89, 95

[87] T. Kapur, W.E.L. Grimson, R. Kikinis, and W. Wells. Enhanced spatial priors for segmentation of magnetic resonance imagery. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 1496 of *LNCS*, pages 457–468, 1998. 42, 44, 46, 51

[88] A. Klein and J. Hirsch. Mindboggle: A Scatterbrained Approach to Automate Brain Labeling. *NeuroImage*, 24:261–280, 2005. 19

[89] J. Koenderink and A. van Doorn. Surface Shape and Curvature Scales. *Image and Vision Computing*, 10:557–565, 1992. 18

[90] W. Kühnel. *Differential Geometry: Curves-Surfaces-Manifolds*. American Mathematical Society, second edition, 2006. 56, 60, 61, 63

[91] G. Kuperberg, M. Broome, P. McGuire, A. David, E. Marianna, F. Ozawa, D. Goff, W.C. West, S. Williams, van der Kouwe A, D. Salat, and B. Fischl. Regionally Localized Thinning of the Cerebral Cortex in Schizophrenia. *Archives of General Psychiatry*, 60:878–888, 2003. 30

[92] K. Kwong, J. Belliveau, D. Chesler, I. Goldberg, R. Weisskoff, B. Poncelet, D. Kennedy, B. Hoppe, M. Cohen, R. Turner, H. Cheng, T. Brady, and B. Rosen. Dynamic Magnetic Resonance Imaging of Human Brain Activity During Primary Sensory Stimulation. *Proceedings of the National Academy of Science*, 89:5675–5679, 1992. 21

[93] K. Van Leemput, F. Maes, D. Vandermeulen, and P. Suetens. Automated model-based tissue classification of mr images of the brain. *IEEE Transactions on Medical Imaging*, 18(10):897–908, 1999. 28

[94] J. Lefèvre and S. Baillet. Optical Flow and Advection on 2-Riemannian Manifolds: A Common Framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):1081–1092, 2008. 74

[95] M. Leventon, W. Grimson, and O. Faugeras. Statistical shape inuence in geodesic active contours. In *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1316–1323, 2000. 86

[96] C. Liu, W.T. Freeman, E.H. Adelson, and Y. Weiss. Human-assisted Motion Annotation. *Proceedings of the International Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 35

[97] G. Lohmann and D. Von Cramon. Automatic Labelling of the Human Cortical Surface using Sulcal Basins. *Medical Image Analysis*, 4:179–188, 2000. 19

[98] M. Loog and M. de Bruijne. Discriminative Shape Alignment. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 5636 of *LNCS*, pages 459–466, 2009. 96

[99] P. Lorenzen, M. Prastawa, B. Davis, G. Gerig, E. Bullitt, and S. Joshi. Multi-Modal Image Set Registration and Atlas Formation. *Medical Image Analysis*, 10(3):440–451, 2006. 29

[100] O. Lyttelton, M. Boucher, S. Robbins, and A. Evans. An Unbiased Iterative Group Registration Template for Cortical Surface Analysis. *NeuroImage*, 34(4):1535–1544, 2007. 89

[101] D. MacDonald, N. Kabani, D. Avis, and A.C. Evans. Automated 3D Extraction of Inner and Outer Surfaces of Cerebral Cortex from MRI. *NeuroImage*, 12:340–356, 2000. 28

[102] F. Maes, A. Collignon, D. Vandermeulen, G. Marchal, and P. Suetens. Multimodality Image Registration by Maximization of Mutual Information. *IEEE Transactions on Medical Imaging*, 16(2):187–198, 1997. 34

[103] J. Maintz and M. Viergever. A Survey of Medical Image Registration. *Medical Image Analysis*, 2:1–36, 1998. 15, 25

[104] S. Makrogiannis, R. Verma, B. Karacali, and C. Davatzikos. A joint transformation and residual image descriptor for morphometric image analysis using an equivalence class formulation. In *Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, International Conference on Computer Vision and Pattern Recognition*, 2006. 32

[105] J-F. Mangin, V. Frouin, I. Bloch, J. Régis, and J. Lopez-Krahe. Automatic Construction of an Attributed Relational Graph Representing the Cortex Topography using Homotopic Transformations. *Proceedings of SPIE*, 2299:110–121, 1994. 19

[106] S. Marsland and R. McLachlan. A Hamiltonian Particle Method for Diffeomorphic Image Registration. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 4548 of *LNCS*, pages 396–407, 2007. 30, 31

[107] J. Mazziotta, A. Toga, A. Evans, P. Fox, and J. Lancaster. A probabilistic atlas of the human brain: Theory and rationale for its development the international consortium for brain mapping (icbm). *NeuroImage*, 2(2):89–101, 1995. 29

[108] D. McGonigle, A. Howseman, B. Athwal, K. Friston, R. Frackowiak, and A. Holmes. Variability in fMRI: An Examination of Intersession Differences. *NeuroImage*, 11(6):708–734, 2000. 17

[109] T. McInerney and D. Terzopoulos. Deformable Models in Medical Image Analysis: A Survey. *Medical Image Analysis*, 1(2):91–108, 1996. 25

[110] M. Miller, M. Beg, C. Ceritoglu, and C. Stark. Increasing the Power of Functional Maps of the Medial Temporal Lobe using Large Deformation Metric Mapping. *Proceedings of the National Academy of Science*, 102:9685–9690, 2005. 25

[111] J. Munkres. *Analysis on Manifold*. Westview Press, 1991. 30, 61

[112] D. Nain, S. Haker, A. Bobick, and A. Tannenbaum. Multiscale 3-d Shape Representation and Segmentation using Spherical Wavelets. *IEEE Transactions on Medical Imaging*, 26(4):598–618, 2007. 50

[113] M. Nielsen, L. Florack, and R. Deriche. Regularization, Scale-Space, and Edge Detection Filters. *Journal of Mathematical Imaging and Vision*, 7:291–307, 1997. 58

[114] M. Nielsen, P. Johansen, A.D. Jackson, and B. Lautrup. Brownian Warps: A Least Committed Prior for Non-rigid Registration. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 2489 of *LNCS*, pages 557–264, 2002. 30, 44

[115] P. Olver. *Applications of Lie Groups to Differential Equations*. Springer-Verlag, 2nd edition, 1993. 30, 63, 85

[116] M. Ono, S. Kubick, and C. Abernathey. *Atlas of the Cerebral Sulci*. Georg Thieme Verlag, 1st edition, 1990. 17

[117] N. Pal and S. Pal. A Review on Image Segmentation Techniques. *Pattern Recognition*, 26(9):1277–1294, 1993. 25

[118] H-J. Park, M. Kubicki, M. Shenton, A. Guimond, R. McCarley, S. Maier, R. Kikinis, F. Jolesz, and C-F. Westin. Spatial Normalization of Diffusion Tensor MRI using Multiple Channels. *NeuroImage*, 20(4):1995–2009, 2003. 25

[119] M. Park and T. Hastie. l_1-regularization Path Algorithm for Generalized Linear Models. *Journal of the Royal Statistical Society, Series B*, 69:659–677, 2007. 34

[120] T. Paus, F. Tomaiuolo, N. Otaky, D. MacDonald, M. Petrides, J. Atlas, R. Morris, and A. Evans. Human cingulate and paracingulate sulci: Pattern, variability, asymmetry, and probabilistic map. *Cerebral Cortex*, 6:207–214, 1996. 29

[121] D. Pham, C. Xu, and J. Prince. Current Methods in Medical Image Segmentation. *Annual Review of Biomedical Engineering*, 2:315–337, 2000. 25

[122] J. Pluim, J. Maintz, and M. Viergever. Mutual-Information-Based Registration of Medical Images: A Survey. *IEEE Transactions on Medical Imaging*, 22:986–1004, 2003. 15, 25

[123] K. Pohl, J. Fisher, W.E.L. Grimson, R. Kikinis, and W. Wells. A Bayesian Model for Joint Segmentation and Registration. *NeuroImage*, 31:228–239, 2006. 28, 29, 32, 41

[124] G. Postelnicu, L. Zollei, and B. Fischl. Combined Volumetric and Surface Registration. *IEEE Transactions on Medical Imaging*, 28(4):508–522, 2009. 30

[125] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, second edition, 1992. 64, 83, 103, 115

[126] A. Qiu and M. Miller. Cortical Hemisphere Registration via Large Deformation Diffeomorphic Metric Curve Mapping. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 4791 of *LNCS*, pages 186–193, 2007. 30, 56

[127] P. Rasser, P. Johnston, J. Lagopoulos, P. Ward, U. Schall, R. Thienel, S. Bender, A. Toga, and P. Thompson. Functional MRI BOLD response to Tower of London Performance of First-Episode Schizophrenia Patients using Cortical Pattern Matching. *NeuroImage*, 26(3):941–951, 2005. 25

[128] M. Rettmann, X. Han, C. Xu, and J. Prince. Automated Sulcal Segmentation Using Watersheds on the Cortical Surface. *NeuroImage*, 15:329–344, 2002. 19

[129] F. Richard, A. Samson, and C. Cuenod. A saem algorithm for the estimation of template and deformation parameters in medical image sequences. *preprint MAP5*, Nov 2007. 39

[130] D. Riviere, J-F. Mangin, D. Papadopoulos-Orfanos, J-. Martinez, V. Frouin, and J. Regis. Automatic Recognition of Cortical Sulci of the Human Brain Using a Congregation of Neural Networks. *Medical Image Analysis*, 6:77–92, 2002. 19

[131] T. Rohfling, R. Brandt, R. Menzel, and C. Maurer, Jr. Evaluation of Atlas Selection Strategies for Atlas-based Image Segmentation with Application to Confocal Microscopy Images of Bee Brains. *NeuroImage*, 21(4):1428–1442, 2004. 95

[132] T. Rohfling, D. Russakoff, and C. Maurer. Performance-based Classifier Combination in Atlas-based Image Segmentation Using Expectation-Maximization Parameter Estimation. *IEEE Transactions on Medical Imaging*, 23(8):983–994, 2004. 95

[133] H. Rosas, A. Liu, S. Hersch, M. Glessner, R. Ferrante, D. Salat, A. van der Kouwe, B. Jenkins, A. Dale, and B. Fischl. Regional and Progressive Thinning of the Cortical Ribbon in Huntington's disease. *Neurology*, 58:695–701, 2002. 30

[134] W. Rudin. *Principles of Mathematical Analysis*. 1976. 80

[135] D. Rueckert, L. Sonoda, C. Hayes, D. Hill, M. Leach, and D Hawkes. Non-rigid Registration using Free-Form Deformations: Application to Breast MR Images. *IEEE Transactions on Medical Imaging*, 18(8):712–720, 1999. 81

[136] M.R. Sabuncu, S. Balci, M. Shenton, and P. Golland. Image-driven Population Analysis through Mixture-Modeling. *IEEE Transactions on Medical Imaging*, In Press, 2009. 27, 95, 98

[137] M.R. Sabuncu, B. Singer, B. Conroy, R. Bryan, P. Ramadge, and J. Haxby. Function-based Inter-Subject Alignment of the Cortical Anatomy. *Cerebral Cortex*, In Press, 2009. 35

[138] M.R. Sabuncu, B.T.T. Yeo, K. Van Leemput, B. Fischl, and P. Golland. Nonparameteric Mixture Models for Supervised Image Parcellation. In *Workshop on Probabilistic Models for Medical Image Analysis, Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), In Press*, 2009. 98

[139] M.R. Sabuncu, B.T.T. Yeo, K. Van Leemput, B. Fischl, and P. Golland. Supervised Nonparameteric Image Parcellation. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 5762, pages 1075–1083, 2009. 95, 98

[140] M.R. Sabuncu, B.T.T. Yeo, T. Vercauteren, K. Van Leemput, and P. Golland. Assymetric Image-Template Registration. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 5761, pages 565–573, 2009. 114

[141] D. Salat, R. Buckner, A Snyder, D. Greve, R. Desikan, E. Busa, J. Morris, A. Dale, and B. Fischl. Thinning of the Cerebral Cortex in Aging. *Cerebral Cortex*, 14:721–730, 2004. 30

[142] R. Saxe, M. Brett, and N. Kanwisher. Divide and Conquer: A Defense of Functional Localizers. *NeuroImage*, 30(4):1088–1096, 2006. 35

[143] A. Schleicher, K. Amunts, S. Geyer, P. Morosan, and K. Zilles. Observer-Independent Method for Microstructural Parcellation of Cerebral Cortex: A Quantitative Approach to Cytoarchitectonics. *NeuroImage*, 9:165–177, 1999. 20, 21

[144] T. Schormann and K. Zilles. Three-Dimensional Linear and Non-linear Transformations: An Integration of Light Microscopical and MRI Data. *Human Brain Mapping*, 6:339–347, 1998. 20

[145] F. Ségonne, J. Pacheco, and B. Fischl. Geometrically accurate topology-correction of cortical surfaces using nonseparating loops. *IEEE Transactions on Medical Imaging*, 26:518–529, 2007. 19

[146] Y. Shi, J. Morra, P. Thompson, and A. Toga. Inverse-Consistent Surface Mapping with Laplace-Beltrami Eigen-Features. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 5636, pages 467–478, 2009. 30

[147] H.-Y. Shum and R. Szeliski. Construction of Panoramic Image Mosaics with Global and Local Alignment. *International Journal on Computer Vision*, 16(1):63–84, 2000. 79

[148] X. Tao, J. Prince, and C. Davatzikos. Using a Statistical Shape Model to Extract Sulcal Curves on the Outer Cortex of the Human Brain. *IEEE Transactions on Medical Imaging*, 21:513–524, 2002. 19

[149] B. Thirion, G. Flandin, P. Pinel, A. Roche, P. Ciuciu, and J-B. Poline. Dealing with the Shortcomings of Spatial Normalization: Multi-Subject Parcellation of fMRI Datasets. *Human Brain Mapping*, 27:678–693, 2006. 17

[150] B. Thirion, P. Pinel, S. Mériaux, A. Roche, S. Dehaene, and J-B. Poline. Analysis of a Large fMRI Cohort: Statistical and Methodological Issues for Group Analyses. *NeuroImage*, 35:105–120, 2007. 17

[151] B. Thirion, P. Pinel, A. Tucholka, A. Roche, P. Ciuciu, J-F Mangin, and J. Poline. Structural Analysis of fMRI Data Revisited: Improving the Sensitivity and Reliability of fMRI Group Studies. *IEEE Transactions on Medical Imaging*, 26(9):1256–1269, 2007. 17

[152] J. Thirion. Image Matching as a Diffusion Process: an Analogy with Maxwell's Demons. *Medical Image Analysis*, 2(3):243–260, 1998. 31, 56, 57, 64, 66, 73, 115

[153] P. Thompson and A. Toga. A Surface-Based Technique for Warping 3-Dimensional Images of the Brain. *IEEE Transactions on Medical Imaging*, 15(4):1–16, 1996. 30, 56

[154] P. Thompson, R. Woods, M. Mega, and A. Toga. Mathematical/Computational Challenges in Creating Deformable and Probabilistic Atlases of the Human Brain. *Human Brain Mapping*, 9(2):81–92, 2000. 30, 56, 74

[155] R. Tibshirani. Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996. 34

[156] R. Tootell and J. Taylor. Anatomical Evidence for MT and Additional Cortical Visual Areas in Humans. *Cerebral Cortex*, 5:39–55, 1995. 22

[157] D. Tosun and J. Prince. Cortical Surface Alignment Using Geometry Driven Multispectral Optical Flow. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 3565 of *LNCS*, pages 480–492, 2005. 30, 56

[158] A. Trouvé. Diffeomorphisms, Groups and Pattern Matching in Image Analysis. *International Journal on Computer Vision*, 28(3):213–221, 1998. 64, 65

[159] Z. Tu, S. Zheng, A. Yuille, A. Reiss, R. Dutton, A. Lee, A. Galaburda, I. Dinov, P. Thompson, and A. Toga. Automated Extraction of the Cortical Sulci Based on a Supervised Learning Approach. *IEEE Transactions on Medical Imaging*, 26(4):541–552, 2007. 19

[160] C. Twining, T. Cootes, S. Marsland, V. Petrovic, R. Schestowitz, and C. Taylor. A Unified Information-Theoretic Approach to Groupwise Non-rigid Registration and

Model Building. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 3565 of *LNCS*, pages 1611–3349, 2005. 29, 33

[161] C. Twining, R. Davies, and C. Taylor. Non-Parametric Surface-Based Regularisation for Building Statistical Shape Models. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 4584 of *LNCS*, pages 738–750, 2007. 32

[162] E. Tyrtyshnikov. *A Brief Introduction to Numerical Analysis*. Birkhäuser Boston, 1997. 80

[163] D. Van Essen, H. Drury, S. Joshi, and M. Miller. Functional and Structural Mapping of Human Cerebral Cortex: Solutions are in the Surfaces. *Proceedings of the National Academy of Sciences*, 95(3):788–795, 1996. 25, 30, 56

[164] K. Van Leemput. Probabilistic Brain Atlas Encoding Using Bayesian Inference. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 4190 of *LNCS*, pages 704–711, 2006. 39

[165] K. Van Leemput. Encoding Probabilistic Brain Atlases Using Bayesian Inference. *IEEE Transactions on Medical Imaging*, 28(6):822–837, 2009. 29, 33, 98

[166] V.P. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1998. 98

[167] T. Vercauteren, X. Pennec, A. Perchant, and N. Ayache. Symmetric log-domain diffeomorphic registration: A demons-based approach. 5241:754–761, 2008. 30

[168] T. Vercauteren, X. Pennec, A. Perchant, and N. Ayache. Diffeomorphic demons: Efficient non-parametric image registration. *NeuroImage*, 45(1, Supp.1):S61–S72, March 2009. PMID: 19041946. 30, 31, 33, 56, 57, 58, 66, 73, 74, 79, 81, 85, 88, 115, 117, 119

[169] N. Weisenfeld and S. Warfield. Simultaneous alignment and central tendency estimation for brain atlas construction. In *Workshop on Statistical Registration: Pair-wise and Group-wise Alignment and Atlas Formation*, 2007. 29

[170] W. Wells, W.E.L. Grimson, R. Kikinis, and F. Jolesz. Adaptive Segmentation of MRI Data. *IEEE Transactions on Medical Imaging*, 4:429–442, 1996. 28

[171] W. Wells, P. Viola, H. Atsumi, S. Nakajima, and R. Kikinis. Multi-modal Volume Registration by Maximization of Mutual Information. *Medical Image Analysis*, 1(1):35–51, 1996. 25, 34

[172] T. White, D. O'Leary, V. Magnotta, S. Arndt, M. Flaum, and N. Andreasen. Anatomic and Functional Variability: The Effects of Filter Size in Group fMRI Data Analysis. *NeuroImage*, 13(4):577–588, 2001. 17

[173] P. Wyatt and A. Noble. MAP MRF Joint Segmentation and Registration. *Medical Image Analysis*, 7(4):539–552, 2003. 32, 41

[174] C. Xiaohua, M. Brady, and D. Rueckert. Simultaneous Segmentation and Registration for Medical Image. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 3216 of *LNCS*, pages 663–670, 2004. 32, 41

[175] C. Xiaohua, M. Brady, and D. Rueckert. Simultaneous Segmentation and Registration of Contrast-Enhanced Breast MRI. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 3565 of *LNCS*, pages 126–137, 2005. 32, 41

[176] J. Xiong, S. Rao, P. Jerabek, F. Zamarripa, M. Woldorff, J. Lancaster, and P. Fox. Intersubject Variability in Cortical Activations during a Complex Language Task. *NeuroImage*, 12(3):326–339, 2000. 17

[177] B.T.T. Yeo, W. Ou, and P. Golland. Invertible Filter Banks on the 2-Sphere. In *Proceedings of the International Conference on Image Processing*, pages 2161–2164, 2006. 18

[178] B.T.T. Yeo, W. Ou, and P. Golland. On the Construction of Invertible Filter Banks on the 2-Sphere. *IEEE Transactions on Image Processing*, 17(3):283–300, 2008. 18

[179] B.T.T. Yeo, M. Sabuncu, R. Desikan, B. Fischl, and P. Golland. Effects of Registration Regularization and Atlas Sharpness on Segmentation Accuracy. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 4791 of *LNCS*, pages 683–691, 2007. 22, 38, 41, 42

[180] B.T.T. Yeo, M. Sabuncu, R. Desikan, B. Fischl, and P. Golland. Effects of Registration Regularization and Atlas Sharpness on Segmentation Accuracy. *Medical Image Analysis*, 12(5):603–615, 2008. 22, 30, 32, 38, 56, 68, 84, 89, 95, 98

[181] B.T.T. Yeo, M. Sabuncu, P. Golland, and B. Fischl. Task-Optimal Registration Cost Functions. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 5761 of *LNCS*, pages 598–606, 2009. 23, 70, 77

[182] B.T.T. Yeo, M. Sabuncu, H. Mohlberg, K. Amunts, K. Zilles, P. Golland, and B. Fischl. What Data to Co-register for Computing Atlases. In *Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, International Conference on Computer Vision*, 2007. 29, 70, 72

[183] B.T.T. Yeo, M. Sabuncu, T. Vercauteren, N. Ayache, B. Fischl, and P. Golland. Spherical Demons: Fast Surface Registration. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 5241 of *LNCS*, pages 745–753, 2008. 22, 55, 79, 81, 85, 94, 119

[184] B.T.T. Yeo, M.R. Sabuncu, T. Vercauteren, N. Ayache, B. Fischl, and P. Golland. Spherical Demons: Fast Diffeomorphic Landmark-Free Surface Registration. *IEEE Transactions on Medical Imaging*, In Press, 2009. 22, 55

[185] B.T.T. Yeo, T. Vercauteren, P. Fillard, X. Pennec, P. Golland, N. Ayache, and O. Clatz. DTI Registration with Exact Finite-Strain Differential. In *Proceedings of the International Symposium on Biomedical Imaging: From Nano to Macro*, pages 700–703, 2008. 25, 57, 98

[186] B.T.T. Yeo, T. Vercauteren, P. Fillard, X. Pennec, P. Golland, N. Ayache, and O. Clatz. DT-REFinD: Diffusion Tensor Registration with Exact Finite-Strain Differential. *IEEE Transactions on Medical Imaging*, In Press, 2009. 25, 57, 79, 81, 85, 95, 98, 119

[187] B.T.T. Yeo, P. Yu, P.E. Grant, B. Fischl, and P. Golland. Shape Analysis with Overcomplete Spherical Wavelets. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 5241 of *LNCS*, pages 468–476, 2008. 18

[188] A. Yezzi, L. Zollei, and T. Kapur. A Variational Framework for Integrating Segmentation and Registration through Active Contours. *Medical Image Analysis*, 7(2):171–185, 2003. 32

[189] P. Yu, P.E. Grant, Y. Qi, X. Han, F. Segonne, R. Pienaar, E. Busa, J. Pacheco, N. Makris, R.L. Buckner, P. Golland, and B. Fischl. Cortical Surface Shape Analysis Based on Spherical Wavelets. *IEEE Transactions on Medical Imaging*, 26(4):582–597, 2007. 50

[190] P. Yu, B.T.T. Yeo, P.E. Grant, B. Fischl, and P. Golland. Cortical Folding Development Study based on Over-complete Spherical Wavelets. In *Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, International Conference on Computer Vision*, 2007. 18, 50

[191] H. Zhang, B. Avants, P. Yuschkevich, J. Woo, S. Wang, L. McCluskey, L. Elman, E. Melhem, and J. Gee. High-Dimensional Spatial Normalization of Diffusion Tensor Images Improves the Detection of White matter Differences: An Example Study Using Amyotrophic Lateral Sclerosis. *IEEE Transactions on Medical Imaging*, 26(11):1585–1597, 2007. 25

[192] H. Zhang, P. Yushkevich, D. Alexander, and J. Gee. Deformable Registration of Diffusion Tensor MR Images with Explicit Orientation Optimization. *Medical Image Analysis*, 10(5):764–785, 2006. 25

[193] K. Zilles, A. Schleicher, N. Palomero-Gallagher, and K. Amunts. *Quantitative Analysis of Cyto- and Receptor Architecture of the Human Brain*. Elsevier, 2002. 20, 21

[194] B. Zitová and J. Flusser. Image Registration Methods: A Survey. *Image and Vision Computing*, 21(11):977–1000, 2003. 15, 25

[195] L. Zollei, M. Jenkinson, S. Timoner, and W. Wells. A Marginalized MAP Approach and EM Optimization for Pair-wise Registration. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, volume 4584 of *LNCS*, pages 662–674, 2007. 98

[196] L. Zollei, E. Learned-Miller, E. Grimson, and W. Wells. Efficient Population Registration of 3D Data. In *Proceedings of Computer Vision for Biomedical Image Applications, International Conference on Computer Vision*, volume 3765 of *LNCS*, pages 291–301, 2007. 25, 27

[197] G. Zou, J. Hua, and O. Muzik. Non-rigid Surface Registration using Spherical Thin-Plate Splines. In *Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, volume 4791 of *LNCS*, pages 367–374, 2007. 31