DIFFERENTIAL EQUATIONS
WITH APPLICATIONS IN MEDICAL IMAGE ANALYSIS

By

HONGYU GUO

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2005
Copyright 2005
by
Hongyu Guo
To my daughter Alicia.
ACKNOWLEDGMENTS

I will be forever indebted to my advisor, Dr. Anand Rangarajan, for his invaluable guidance and support. I am very lucky to have the opportunity to work with him. I am very grateful for his appreciation and confidence in my mathematical talent. Although he is an outstanding researcher in the computer vision and medical imaging field with many distinguished contributions, he is very humble and down to earth. During the years of my study, I could knock at his door any time I needed him and he would put down his work and discuss sparkling ideas with me. Working with him has been truly a wonderful and rewarding experience.

I would like to express deepest thanks to Dr. Arunava Banerjee, Dr. Jorg Peters and Dr. Baba Vemuri of the Department of CISE, Dr. Yunmei Chen of the Department of Mathematics and Dr. Haldun Aytug of the Department of Decision and Information Sciences for serving on my supervisory committee. I thank them for their invaluable criticism and advice.

I wish to extend my gratitude to Dr. Sarang Joshi at the University of North Carolina at Chapel Hill and Dr. Laurent Younes at Johns Hopkins University. They are the coauthors of a book chapter and several papers that I have written and this work in my dissertation is greatly benefitted from their previous work on diffeomorphic landmark matching.

I would like to thank Dr. Eric Grimson, the director of MIT AI Lab, for the helpful discussion and valuable comments when he visited the University of Florida. I wish to extend special thanks to Dr. Alper for an enlightening discussion on triangulation and surface reconstruction.
My thanks also go to everyone of my fellow students at the Center for Computer Vision, Graphic and Medical Imaging at the University of Florida, particularly, Bing Jian, Santhosh Kodipaka, Jie Zhang, Adrian Peter, Zhizhou Wang, Eric Spellman, Fei Wang, Tim McGraw and Xiaobin Wu. I get constant help from them every day.

I am indebted to Dr. John M. Sullivan at the University of Illinois at Urbana-Champaign for granting permission to use his graphics image of the Klein bottle.

I would like to thank my wife Yanping for her love and support and I thank my daughter Alicia for being a wonderful daughter. Alicia and I enjoyed working together on many weekends. While she was fascinated by the rotating shapes of computer graphics I displayed with MATLAB, I admired her beautiful and creative colorings and drawings.

This research is supported in part by the National Science Foundation grant IIS 0307712.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>SECTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>iv</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>viii</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>ix</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>xi</td>
</tr>
<tr>
<td>CHAPTER</td>
<td></td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2 PREVIOUS WORK</td>
<td>17</td>
</tr>
<tr>
<td>2.1 Thin Plate Splines (TPS)</td>
<td>18</td>
</tr>
<tr>
<td>2.2 Bookstein’s Application to 2D Landmark Warping</td>
<td>19</td>
</tr>
<tr>
<td>2.3 More on Splines</td>
<td>21</td>
</tr>
<tr>
<td>2.4 Reproducing Kernel Hilbert Space (RKHS) Formulation</td>
<td>23</td>
</tr>
<tr>
<td>2.5 The Folding Problem of Splines</td>
<td>24</td>
</tr>
<tr>
<td>2.6 Imposing Restriction on the Jacobian</td>
<td>25</td>
</tr>
<tr>
<td>2.7 The Flow Approach</td>
<td>25</td>
</tr>
<tr>
<td>2.8 Correspondence and Softassign</td>
<td>28</td>
</tr>
<tr>
<td>2.9 Distance Transforms</td>
<td>28</td>
</tr>
<tr>
<td>2.10 Implicit Correspondence</td>
<td>29</td>
</tr>
<tr>
<td>2.11 Shape Context</td>
<td>31</td>
</tr>
<tr>
<td>2.12 Active Shape Models</td>
<td>31</td>
</tr>
<tr>
<td>2.13 Deterministic Annealing Applied to EM Clustering</td>
<td>32</td>
</tr>
<tr>
<td>2.14 Statistical Shape Analysis on Differentiable Manifolds</td>
<td>34</td>
</tr>
<tr>
<td>2.15 Distance Measures from Information Theory</td>
<td>36</td>
</tr>
<tr>
<td>3 DIFFEOMORPHIC POINT MATCHING</td>
<td>42</td>
</tr>
<tr>
<td>3.1 Existence of a Diffeomorphic Mapping in Landmark Matching</td>
<td>42</td>
</tr>
<tr>
<td>3.2 Symmetric Matching due to Time Reversibility</td>
<td>44</td>
</tr>
<tr>
<td>3.3 A Theoretical Framework for Diffeomorphic Point Matching</td>
<td>46</td>
</tr>
<tr>
<td>3.4 A Diffeomorphic Point Matching Algorithm</td>
<td>49</td>
</tr>
<tr>
<td>3.5 Applications to 2D Corpus Callosum Shapes</td>
<td>53</td>
</tr>
<tr>
<td>3.6 Applications to 3D Shapes</td>
<td>55</td>
</tr>
<tr>
<td>3.6.1 Experiments on Synthetic Data</td>
<td>59</td>
</tr>
<tr>
<td>3.6.2 Experiments on Real Data</td>
<td>61</td>
</tr>
<tr>
<td>Table</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>3-1</td>
<td>Modified Hausdorff distance of the matching point sets.</td>
</tr>
<tr>
<td>3-2</td>
<td>Geodesic distances and modified Hausdorff distances</td>
</tr>
<tr>
<td>3-3</td>
<td>Matching errors on synthetic data with different noise levels</td>
</tr>
<tr>
<td>3-4</td>
<td>Limiting value of $\sigma$ determined by the number of clusters</td>
</tr>
<tr>
<td>3-5</td>
<td>Jensen-Shannon divergence for various pairs of shapes</td>
</tr>
<tr>
<td>3-6</td>
<td>Hausdorff and modified Hausdorff distance for various pairs of shapes</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–1</td>
<td>Grids in $\mathbb{R}^2$ (a) before transformation and (b) after transformation</td>
<td>9</td>
</tr>
<tr>
<td>1–2</td>
<td>Klein bottle immersed in $\mathbb{R}^3$</td>
<td>9</td>
</tr>
<tr>
<td>1–3</td>
<td>(a) The template image and (b) The reference image</td>
<td>10</td>
</tr>
<tr>
<td>1–4</td>
<td>Space deformations</td>
<td>10</td>
</tr>
<tr>
<td>1–5</td>
<td>Ambiguity of landmark correspondence</td>
<td>13</td>
</tr>
<tr>
<td>1–6</td>
<td>Two shapes of a ribbon and the associated landmarks</td>
<td>15</td>
</tr>
<tr>
<td>1–7</td>
<td>Misinterpretation of the shape</td>
<td>16</td>
</tr>
<tr>
<td>2–1</td>
<td>Deformation of the thin plate</td>
<td>38</td>
</tr>
<tr>
<td>2–2</td>
<td>Landmark displacements</td>
<td>38</td>
</tr>
<tr>
<td>2–3</td>
<td>Thin-plate Spline interpolation</td>
<td>39</td>
</tr>
<tr>
<td>2–4</td>
<td>Diffeomorphic interpolation</td>
<td>39</td>
</tr>
<tr>
<td>2–5</td>
<td>Two distance transformed images of three landmarks</td>
<td>40</td>
</tr>
<tr>
<td>2–6</td>
<td>Level sets of two distance transformed images</td>
<td>41</td>
</tr>
<tr>
<td>3–1</td>
<td>Existence of a diffeomorphic mapping</td>
<td>44</td>
</tr>
<tr>
<td>3–2</td>
<td>Asymmetry of the matching</td>
<td>45</td>
</tr>
<tr>
<td>3–3</td>
<td>Point sets of nine corpus callosum images</td>
<td>56</td>
</tr>
<tr>
<td>3–4</td>
<td>Clustering of the two point sets</td>
<td>56</td>
</tr>
<tr>
<td>3–5</td>
<td>Diffeomorphic mapping of the space</td>
<td>56</td>
</tr>
<tr>
<td>3–6</td>
<td>Matching between the two point sets</td>
<td>57</td>
</tr>
<tr>
<td>3–7</td>
<td>Overlay of the after-images of eight point sets with the ninth set</td>
<td>57</td>
</tr>
<tr>
<td>3–8</td>
<td>Two point sets of hippocampal shapes</td>
<td>59</td>
</tr>
<tr>
<td>3–9</td>
<td>Matching errors on synthetic data for different number of clusters</td>
<td>65</td>
</tr>
<tr>
<td>3–10</td>
<td>Two hippocampal shapes</td>
<td>66</td>
</tr>
</tbody>
</table>
Abstract of Dissertation Presented to the Graduate School of the University of Florida in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy

DIFFEOMORPHIC POINT MATCHING WITH APPLICATIONS IN MEDICAL IMAGE ANALYSIS

By

Hongyu Guo

August 2005

Chair: Dr. Anand Rangarajan
Major Department: Computer and Information Science and Engineering

Diffeomorphic matching of unlabeled point sets is very important to non-rigid registration and many other applications but it has never been done before. It is a very challenging problem because we have to solve for the unknown correspondence between the two point sets. In this work we propose a joint clustering method to solve for a simultaneous estimation of the correspondence and the diffeomorphism in space. The cluster centers in each point set are always in correspondence by virtue of having the same index. During clustering, the cluster center counterparts in each point set are linked by a diffeomorphism and hence are forced to move in lock-step with one another. We devise an objective function and design an algorithm to find the minimizer of the objective function. We apply the algorithm to 2D and 3D shapes in medical imaging. We further propose to use a graph representation for the shape topology information. Results are given for prescribed topologies like chain topology, ring topology – which are very common in dealing with 2D contour shapes – and genus zero closed surface topology in 3D. We also investigate the topology problem in general and the learning of topology with a nearest neighbor graph.
CHAPTER 1
INTRODUCTION

This work is a first time effort to study the diffeomorphic point matching problem with unknown correspondence. In this chapter we introduce the need for an effective diffeomorphic point matching algorithm and the background knowledge of the shape matching field. We give some fundamental definitions and theorems from differential topology which are essential in the development of the following chapters. We analyze the nature and difficulty of the correspondence problem.

Shape analysis using tools of modern differential geometry and statistical theory has far and wide applications in computer vision, image processing, biology, morphometrics, computational anatomy, biomedical imaging, and image guided surgery as well as archeology and astronomy. Shapes play a fundamental role in computer vision and image analysis and understanding. In general terms, the shape of an object, a data set, or an image can be defined as the total of all information that is invariant under certain spatial transformations [66]. Shape matching and correspondence problems arise in various application areas such as computer vision, pattern recognition, machine learning and especially in computational anatomy and biomedical imaging. Shape matching becomes an indispensable part of many biomedical applications like medical diagnosis, radiological treatment, treatment evaluation, surgical planning, image guided surgery and pathology research.

Shapes may have many different representations. They can be represented with the intensities of pixels of an image, which is a function defined in a region of 2D or 3D space, or they can be represented with point sets, curves or surfaces. In this dissertation, we focus on the point representation of shapes. Point representation of image data is widely used in all areas and there is a huge amount of
point image data acquired in various modalities, including optical, MRI, computed
tomography and diffusion tensor images [18, 24].

The advantage of point set representation of shapes, as opposed to curve and
surface representations is multifarious and that is why we focus on point shape
matching, instead of curve or surface matching [64, 68, 67]. The point set repre-
sentation is compact in storage. The computational time when using point set
representation is dramatically reduced as opposed to when image intensities are
used. The point set representation of shapes is universal and homogeneous. It does
not require the prior knowledge about the topology of shapes. It has the capability
to fuse different types of features into a global, uniform and homogeneous repre-
sentation. A point set representation of shapes is especially useful when feature
 grouping (into curves and the like) cannot be assumed. Statistical analysis on point
set shapes is straightforward, as demonstrated in Cootes et al. [18] using active
shape models. The recent work of Glaunes et al. [27] is a bold step forward to
generalize the point set shapes to general Radon measures and distributions in
the sense of Schwartz generalized functions, in order to model shapes represented
by a mixture of points and submanifolds of different dimensions (curves and sur-
faces). Although this is viable in theory, it becomes unpractical when it comes to
applications. In their experiments dealing with a mixture of points and a curve,
they used the technique of resampling the curve. That comes back to the point set
representation of the curve itself. Moreover, by doing so, new free parameters, i.e.,
the relative weight of measure between points and the curves, is introduced, which
can be arbitrary.

Point shape matching is ubiquitous in medical imaging and in particular, there
is a real need for a turnkey, non-rigid point shape matching algorithm [18, 24, 19].
Point shape matching in general is a difficult problem because, as with many
other problems in computer vision, like image registration and segmentation, it
is often ill-posed. We try to make abstractions out of the practical problems to formulate precise mathematical models of the problem. Point matching can be viewed in the context, or out of the context of image registration. In the context of image registration, the points are viewed as feature points in the image. The correspondence can be known or unknown. When the cardinality of the two point sets is the same and when the correspondence is known, we call this the landmark matching problem. The number of feature points may also be unequal. In that case we are dealing with outliers. The difficulty increases dramatically when the correspondence is not known and/or when there are outliers. The point matching problem can also exist out of the context of image registration. In this case, the points are samples from a shape and we have a point representation of the shape. When we have two such shapes represented by points, usually the number of points in the two shapes is different and there is no point-wise correspondence. We want to find the correspondence between the two shapes. We call this the point shape matching problem. In the following, we will first address the landmark matching problem and then the point shape matching problem.

We assume the image domain is $d$-dimensional Euclidean space $\mathbb{R}^d$. Usually $d = 2$ or $d = 3$. Suppose we have two images, $I_1 : \Omega_1 \rightarrow \mathbb{R}$ and $I_2 : \Omega_2 \rightarrow \mathbb{R}$, where $\Omega_1 \subseteq \mathbb{R}^d$ and $\Omega_2 \subseteq \mathbb{R}^d$. The image registration problems can be classified into two categories: intensity based registration and feature based registration [52]. In the intensity based image registration, we need to find a map $f : \Omega_1 \rightarrow \Omega_2$ such that $\forall x \in \Omega_1$, $I_1(x) = I_2(f(x))$. This is the ideal registration problem. In feature based registration, we suppose we have two corresponding sets of feature points, or landmarks, \{\pi \in \Omega_1 | i = 1, 2, ..., n\} and \{\qi \in \Omega_2 | i = 1, 2, ..., n\}. We need to find a transformation $f : \Omega_1 \rightarrow \Omega_2$ such that $\forall i = 1, 2, ..., n$, $f(p_i) = q_i$.

In many applications, we are required to find the transformation within some restricted groups, like rigid transformations, similarity transformations, affine
transformations, projective transformations, polynomial transformations, B-spline transformations and “non-rigid” transformations. Different transformation groups have different degrees of freedom, the number of parameters needed to describe a transformation in the group. This also determines the number of landmark pairs that the transformation can exactly interpolate. Let us look at some examples. In the two dimensional space, where \( d = 2 \), a rigid transformation, which preserves Euclidean distance, defined by 1.1 has 3 degrees of freedom \((\varphi, x_0, y_0)\) and cannot interpolate arbitrary landmark pairs.

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} = \begin{bmatrix}
  \cos \varphi & -\sin \varphi \\
  \sin \varphi & \cos \varphi
\end{bmatrix}\begin{bmatrix}
  x \\
  y
\end{bmatrix} + \begin{bmatrix}
  x_0 \\
  y_0
\end{bmatrix}
\] (1.1)

The landmark pairs to be matched must be subject to some constraints. That is, they have to have the same Euclidean distance. A similarity transformation defined by 1.2 has 4 degrees of freedom \((k, \varphi, x_0, y_0)\) and can map any 2 points to any 2 points.

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} = k\begin{bmatrix}
  \cos \varphi & -\sin \varphi \\
  \sin \varphi & \cos \varphi
\end{bmatrix}\begin{bmatrix}
  x \\
  y
\end{bmatrix} + \begin{bmatrix}
  x_0 \\
  y_0
\end{bmatrix}
\] (1.2)

An affine transformation defined by 1.3 has 6 degrees of freedom \((a_{11}, a_{12}, a_{21}, a_{22}, x_0, y_0)\) and can map any 3 non-degenerate points to any 3 non-degenerate points.

\[
\begin{bmatrix}
  x' \\
  y'
\end{bmatrix} = \begin{bmatrix}
  a_{11} & a_{12} \\
  a_{21} & a_{22}
\end{bmatrix}\begin{bmatrix}
  x \\
  y
\end{bmatrix} + \begin{bmatrix}
  x_0 \\
  y_0
\end{bmatrix}
\] (1.3)

A projective transformation defined by 1.4 in term of non-homogeneous coordinates has 8 degrees of freedom and can map any 4 non-degenerate points to any 4 non-degenerate points.
In three dimensional space, where $d = 3$, we write the transformation in a more general form,

\[
\begin{pmatrix}
x' \\
y' \\
z'
\end{pmatrix} = \begin{pmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23} \\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
\begin{pmatrix}
x \\
y \\
z
\end{pmatrix} + \begin{pmatrix}
x_0 \\
y_0 \\
z_0
\end{pmatrix},
\] (1.5)

where

\[
A = \begin{pmatrix}
a_{11} & a_{12} & a_{13} \\
a_{21} & a_{22} & a_{23} \\
a_{31} & a_{32} & a_{33}
\end{pmatrix}
\]
is a $3 \times 3$ matrix, which represents a linear transformation. A rigid transformation requires $A$ to be orthogonal and has 6 degrees of freedom. A similarity transformation requires $A$ to be a similar matrix and has 7 degrees of freedom. An affine transformation allows $A$ to be the most general form and has 12 degrees of freedom and can map any 4 non-degenerate points to any 4 non-degenerate points. A projective transformation in 3D as defined by 1.6 has 15 degrees of freedom and can map any 5 non-degenerate points to any 5 non-degenerate points.

\[
\begin{pmatrix}
x' \\
y' \\
z'
\end{pmatrix} = \begin{pmatrix}
a_{11}x + a_{12}y + a_{13}z + a_{14} \\
a_{21}x + a_{22}y + a_{23}z + a_{24} \\
a_{31}x + a_{32}y + a_{33}z + a_{34}
\end{pmatrix}
\] (1.6)

The term “non-rigid” transformation is often used in a narrower sense. Although similarity, affine and projective transformations do not preserve Euclidean distance, they all have finite degrees of freedom. In the literature, “non-rigid”
transformation usually refers to a transformation of infinite degrees of freedom, which can potentially map any finite number of points to the same number of points. So we immediately see a big difference between finite degree of freedom transformations and non-rigid transformations. Given a fixed number of landmark pairs to be interpolated, the former is easily over constrained; the latter is always under constrained. This is one of the reasons that the non-rigid point matching problem is much more difficult. To find a unique non-rigid transformation, we need further constraints. We call this regularization.

Two desirable properties of non-rigid transformations are smoothness and topology preserving. Let \( \Omega_1 \subseteq \mathbb{R}^d \) and \( \Omega_2 \subseteq \mathbb{R}^d \). A transformation \( f : \Omega_1 \rightarrow \Omega_2 \) is said to be smooth if all partial derivatives of \( f \), up to certain orders, exist and are continuous. If transformation \( f : \Omega_1 \rightarrow \Omega_2 \) preserves the topology, then \( \Omega_1 \) and \( \text{Img}(f) = \{ p_2 \in \Omega_2 | \exists p_1 \in \Omega_1, p_2 = f(p_1) \} \) have the same topology. A transformation that preserves topology means we will not have tears in space or in the image. A transformation that preserves topology is called a homeomorphism and its formal definition is:

**Definition 1.** Let \( \Omega_1 \) and \( \Omega_2 \) be two topological spaces. A map \( f : \Omega_1 \rightarrow \Omega_2 \) is a homeomorphism if

- \( f \) is a bijection;
- \( f \) is continuous;
- the inverse \( f^{-1} \) is continuous.

A smooth transformation \( f : \Omega_1 \rightarrow \Omega_2 \) may not preserve the topology. Namely, a smooth map may not be a homeomorphism. It is easy to see this because a smooth map even may not be a bijection. On the other hand, a homeomorphism may not be smooth because in the definition, we only require continuity in both \( f \) and its inverse but we do not require differentiability. It is strongly desirable that
the transformation is both smooth and topology preserving. What we want is a
diffeomorphism which is defined as follows.

**Definition 2.** Let $M_1$ and $M_2$ be differentiable manifolds. A map $f : M_1 \to M_2$ is
a diffeomorphism if

- $f$ is a bijection;
- $f$ is differentiable;
- the inverse $f^{-1}$ is differentiable.

Because the concept of diffeomorphism is essential to this work, we would
have a little more discussion here in order to clarify some common misconceptions
about this concept. It is obvious that a diffeomorphism is both a smooth map and
a homeomorphism from the definitions. However, what is not so obvious is that by
requiring a diffeomorphism, we are asking for more than something that is both
smooth and homeomorphism. Let us look at some of following counterexamples
and we will learn what factors may contribute to make the transformation fail to be
a diffeomorphism.

First, a smooth bijection is not necessarily a diffeomorphism. Let $M_1 \subset \mathbb{R}$
be $[0, 2\pi)$ and $M_2 = S^1$ the unit circle. $f : M_1 \to M_2$ defined by $f(\varphi) =
(\cos(\varphi), \sin(\varphi))$ is smooth and bijective, but not a diffeomorphism because the
inverse $f^{-1}$ is not continuous, and hence not differentiable. In fact, $f$ is not a
homeomorphism.

Second, a smooth homeomorphism is not necessarily a diffeomorphism.
Consider $f : \mathbb{R} \to \mathbb{R}$ with $f(x) = x^3$. It is smooth and it is a homeomorphism but
the inverse $f^{-1}$ is not differentiable at $x = 0$.

A related concept is **local diffeomorphism** and it is defined as:

**Definition 3.** A differentiable map $f : M_1 \to M_2$ is a local diffeomorphism if for
each $x \in M_1$ there exists a neighborhood $U$ of $x$ such that $f|_U : U \to f(U)$ is a
diffeomorphism.
Furthermore, it is useful that we list without proof some known facts in differential topology \([9, 8, 50, 10, 31]\) about diffeomorphism and local diffeomorphism.

**Theorem 1.** A map \( f : M_1 \to M_2 \) is a local diffeomorphism if and only if its tangent map is an isomorphism.

With some simple facts in linear algebra, the above theorem can be rewritten as the following.

**Theorem 2.** A map \( f : M_1 \to M_2 \) is a local diffeomorphism if and only if its Jacobian of the tangent map is nowhere equal to zero.

**Theorem 3.** A map \( f : M_1 \to M_2 \) is a diffeomorphism if and only if it is a bijection and a local diffeomorphism.

With the help of these theorems we can visualize more situations when a smooth map fails to be diffeomorphism. One situation is when the tangent map fails to be an isomorphism everywhere. Namely, the Jacobian is zero at some points. In that case, the smooth map even fails to be a local diffeomorphism. The other situation is that the smooth map is a local diffeomorphism but not a global diffeomorphism. The following example demonstrates this.

Consider \( f : \mathbb{R}^2 \to \mathbb{R}^2 \) with \( f(x, y) = (e^x \cos y, e^x \sin y) \). The Jacobian is

\[
J = \begin{vmatrix}
\frac{\partial f_x}{\partial x} & \frac{\partial f_x}{\partial y} \\
\frac{\partial f_y}{\partial x} & \frac{\partial f_y}{\partial y}
\end{vmatrix} = \begin{vmatrix}
e^x \cos y & -e^x \sin y \\
e^x \sin y & e^x \cos y
\end{vmatrix} = e^{2x} \neq 0 \ \forall (x, y) \in \mathbb{R}^2.
\]

So \( f \) is a local diffeomorphism. However, \( f \) is not a diffeomorphism as can be seen in Figure. 1–1. Notice that because the function is periodic in \( y \), in the codomain, the image of the function has infinitely many sheets overlaid. Another related example is the self intersection of the immersion of the Klein bottle in \( \mathbb{R}^3 \) as shown in Figure. 1–2.

Now let us look at an example of another smooth transformation, namely, the thin-plate spline (TPS) \([70]\).
Figure 1–1: Grids in $\mathbb{R}^2$ (a) before transformation and (b) after transformation.

Figure 1–2: Klein bottle immersed in $\mathbb{R}^3$.
Figure 1–3: (a) The template image and (b) The reference image

Figure 1–4: Space deformations (a) obtained with thin-plate spline interpolation. The folding of space is illustrated by the deformation of the grid lines. (b) The desired space transformation: a diffeomorphism, which eliminates the space folding problem.
Figure 1–3a shows a template image. Figure 1–3b is the reference image obtained by warping the template. Some landmarks are selected and shown in the images. Figure 1–4a demonstrates the transformation of space by showing the deformation of the rectangular grid. We can see the folding of space. This is the drawback of the thin-plate spline interpolation. Due to the folding of space, features in the template may be smeared in the overlapping regions. And furthermore, the transformation is not invertible. A diffeomorphic transformation is strongly desirable, which preserves the features, the topology and which is smooth as shown in Figure 1–4b.

We still face the unknown correspondence problem, which is a difficult problem. There are two scenarios in which we encounter the correspondence problem. The first scenario is when we do landmark based image registration. If the landmarks are selected by hand, we do know the correspondence. However, the process of hand picking landmarks is painstaking and it requires expert knowledge about the image and the subject area and it may involve human error. The complete automation of landmark selection is still not achieved at present but there is good progress towards the goal. If the landmarks are automatically selected, then the correspondence is unknown and the correspondence needs to be automatically obtained. The second scenario is when we have a point set representation of the shapes. Each shape is represented by a large point set (point cloud) and we have no knowledge about the correspondence between the two point sets.

What makes the unknown correspondence problem more difficult is that it is ill-posed. The correspondence between two point sets is a very intuitive concept which everyone seems to understand. However, as is well known within the medical image analysis community, it is very difficult to define correspondence precisely. This creates a problem for validation since an irreducibly subjective factor seems to be present in deciding what is a good correspondence.
We continue with a short discussion aimed at reaching a better understanding of point correspondence. Suppose we have two point sets $S_1 = \{ p_i \in \Omega_1 | i = 1, 2, ..., n \}$ and $S_2 = \{ q_i \in \Omega_2 | i = 1, 2, ..., n \}$ where $\Omega_1 \subseteq \mathbb{R}^d$ and $\Omega_2 \subseteq \mathbb{R}^d$, usually with $d = 2$, or $d = 3$. When $S_1$ and $S_2$ are point sets of equal cardinality with points randomly distributed in space, what is the correspondence between $S_1$ and $S_2$? There are $n!$ bijections, or permutations between the two sets. We have no *a priori* knowledge allowing us to judge that one correspondence is better than the other. Finite point sets have no extra structure over and beyond their discrete structure. So when we talk about the correspondence between point sets, we always imply that the point set is the representation of some underlying shape, which is a topological space. And, as is fairly standard in the literature, we can bypass discussing point correspondence by focusing on the correspondence of the underlying topological space, which is a homeomorphism, and most desirably a diffeomorphism. Even so, the homeomorphism between the two topological spaces is not unique. Here we make our first assumption: the optimal correspondence between the two point sets is the one that induces the least deformation of space. Here we understand the notion of space deformation intuitively but we will carefully define it later. It is easy to realize that this assumption is not always true because we are only given the point set and we have no explicit knowledge of the underlying shape.

This is easily illustrated with a simple example. In Figure 1–5a and 1–5b we have two point sets. It is not clear at all what the correspondence is between the two point sets. Figure 1–5c and 1–5d put the two point sets on top of two underlying images, the images of two women. Figure 1–5c is the drawing *My Wife and My Mother-in-law* published in 1915 by the cartoonist W. E. Hill. Figure 1–5d is the image of *Head of a Woman in Profile* (cropped) by Jean-Pierre David. If you interpret Figure 1–5c as the image of a young girl, namely the point C as the
Figure 1-5: Ambiguity of landmark correspondence
chin and H as the ear, then we may have the correspondence between the two point sets from Figure 1–5a to 1–5b: A→A, B→B, C→C, D→D, E→E, F→F, G→G and H→H. From another perspective, if we view the image as that of an old woman, namely point C as the tip of the nose, E the chin and H the left eye, we may have the correspondence: A→A, B→B, C→I, D→J, E→C, F→K, G→L and H→M.

Let us look at another example. Figure 1–6a and Figure 1–6b are two images of a black strip with the latter non-rigidly deformed. We select four landmarks A1 through A4 for shape A and four landmarks B1 through B4 for shape B. Figure 1–6c shows the landmarks for shape A without the underlying ribbon shape. Figure 1–6d shows the landmarks for shape B without the underlying ribbon shape.

Given the fact that we can see the shapes, we know the correspondences are $A_1 \rightarrow B_1$, $A_2 \rightarrow B_2$, $A_3 \rightarrow B_3$, $A_4 \rightarrow B_4$. Figure 1–7a shows this landmark correspondence and Figure 1–7b shows the deformation of space for this landmark correspondence. However, if we only have the two sets of points as shown in Figure 1–6c and Figure 1–6d, without the knowledge of the underlying shapes, the above correspondence is not the one that we will find since it does not give the least deformation of space. In fact, another correspondence, $A_1 \rightarrow B_1$, $A_2 \rightarrow B_3$, $A_3 \rightarrow B_2$, $A_4 \rightarrow B_4$ shown in Figure 1–7d gives a smaller deformation solution as shown in Figure 1–7e. This is in fact a misinterpretation of the shape, with Figure 1–7c as the original shape and Figure 1–7f the misinterpreted shape, due to the lack of information regarding the underlying shapes.

Consequently, by making the above assumption of least space deformation, we are really assuming that the points are dense enough such that the underlying shape is well represented by the points. Adding more points can help resolve this ambiguity and hence keep the above assumption—that the correct correspondence gives the smallest deformation of space—approximately valid.
Figure 1–6: Two shapes of a ribbon and the associated landmarks. (a) Shape A, (b) Shape B, (c) Landmarks of shape A without the underlying shape, (d) Landmarks of shape B without the underlying shape
The rest of the dissertation is organized as follows. In Chapter 2 we briefly review the previous related work in the past ten years. In Chapter 3 we describe our theory of diffeomorphic point matching, develop an algorithm to solve the problem and apply the algorithm to 2D and 3D medical imaging applications. Chapter 4 is an extension to the work in Chapter 3 and in it we describe a method of topological clustering and matching. In Chapter 5 we summarize our contributions and point out the directions of future work.
CHAPTER 2
PREVIOUS WORK

In this chapter, we give a brief account of the previous research that is related to our work. Most of the work was developed in the past ten years but some work may be traced back to the 1970s. We start with thin-plate splines, which play an important role in shape matching, and all other splines work with the same principle. The Reproducing Kernel Hilbert Space formulation gives us a great insight on the nature of the entire class of spline warping problems. We then discuss the folding problem with splines and various approaches to overcome this, including imposing constraints on the Jacobian and the flow approaches. We also discuss the correspondence problem. In general there are two classes of approaches to tackle the correspondence problem. One is to confront it directly while the other is to try to circumvent it. With the second class of approaches, people have used implicit correspondence by using a shape distance that is the function of the two point sets, instead of depending on the point-to-point correspondence, and there are also some methods that transform the point matching problem to image matching problem through distance transform. With the first class of approaches, the direct approach, there are methods that treat the correspondence and matching separately, like softassign and there are methods that handles both at the same time like the joint clustering and matching (JCM) algorithms. Shape context method makes use of extra local context information to help better resolve the ambiguous point correspondence problem. Active shape models play as a bridge between the landmark based methods and image intensity based methods, and also a bridge between rigid matching and non-rigid matching. We spend a section discussing the deterministic annealing method for EM clustering, which is an
effective method to avoid local minima and we adopt such an annealing approach in our algorithm. Also closely related is the work of statistical shape analysis on differentiable manifold and some distance measures from Fisher information theory.

2.1 Thin Plate Splines (TPS)

A spline is a long strip of wood or metal that is fixed at a number of points. This has long been used as a drafting tool to draw smooth curves that are required to pass certain points (called “ducks,” or “dogs,” or “rats”). These wooden strips are not only used in drafting, but also used in constructions. The Wright brothers used one to shape their wings. In the old days splines were used in shipbuilding. They are also used for bending the wood for musical instruments like pianos, violins, violas, etc. In the modern days, splines are used to model the body of automobiles. In all those uses, it is actually the physical realization of some smooth curves. In physics, the shape of the wood strip has to take the form of such a curve that the bending energy of the strip is minimized. Sheoenerg [63, 62] is credited to be the first to study spline functions started with one dimensional problems, the cubic splines. The thin plate spline is the natural generalization of the cubic spline. In the drafting practice, the wooden strip is long but very thin and narrow, so that it is abstracted as a one dimensional curve. In the thin plate case, we have a plate which is very thin, so the problem is two dimensional. In both cases, the same physical bending energy of the strip or the thin plate in minimized.

The problem of thin plate spline interpolation is stated as follows. Given \( \{p_i \in \mathbb{R}^2\mid i = 1, 2, \ldots, n\} \) and \( \{z_i \in \mathbb{R}\mid i = 1, 2, \ldots, n\} \), we want to find a function \( f : \mathbb{R}^2 \to \mathbb{R}^2 \) such that it interpolates the points with \( f(p_i) = z_i \) \( i = 1, 2, \ldots, n \) and minimizes the thin plate bending energy

\[
E(f) = \int \int_{\mathbb{R}^2} \left[ \left( \frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 f}{\partial y^2} \right)^2 \right] \, dx \, dy. \tag{2.1}
\]
If we interpret the \( \{ z_i \in \mathbb{R} | i = 1, 2, ..., n \} \) as the displacements of the thin plate in the \( z \) direction at points \( \{ p_i \in \mathbb{R}^2 | i = 1, 2, ..., n \} \) in the \( x - y \) plane, we can easily see the physical intuition of this interpolation problem, as shown in Figure 2-1.

Sometimes it is desirable that we sacrifice the exact interpolation. We only seek an approximation of the interpolation as a trade off of less deformation of the thin plate. So the thin plate smoothing problem is defined as given \( \{ p_i \in \mathbb{R}^2 | i = 1, 2, ..., n \} \) and \( \{ z_i \in \mathbb{R} | i = 1, 2, ..., n \} \), we want to find a function \( f : \mathbb{R}^2 \to \mathbb{R}^2 \) such that it minimizes the cost function

\[
\frac{1}{n} \sum_{i=1}^{n} (z_i - f(p_i))^2 + \lambda E(f). \tag{2.2}
\]

### 2.2 Bookstein’s Application to 2D Landmark Warping

Bookstein [4, 5] applied thin plate splines to the landmark interpolation problem. For simplicity, we discuss the problem in 2D space. Everything in the 2D formulation easily applies to 3D except we have a different kernel for 3D. The goal is to find a smooth transformation \( f : \Omega_1 \to \Omega_2 \) that interpolates \( n \) pairs of landmarks \( \{ p_i \in \Omega_1 | i = 1, 2, ..., n \} \) and \( \{ q_i \in \Omega_2 | i = 1, 2, ..., n \} \). It is obvious that such a smooth transformation is not unique. We will give it further constraints. One choice is to require the transformation minimize some functional, in this case

\[
E = \sum_{h=1}^{2} \int_{\mathbb{R}^2} \left[ \left( \frac{\partial^2 f_h}{\partial x^2} \right)^2 + 2 \left( \frac{\partial^2 f_h}{\partial x \partial y} \right)^2 + \left( \frac{\partial^2 f_h}{\partial y^2} \right)^2 \right] \, dx \, dy, \tag{2.3}
\]

where \( f_1 \) and \( f_2 \) are the \( x \) and \( y \) components of the mapping,

\[
f(x, y) = (f_1(x, y), f_2(x, y)). \tag{2.4}
\]

If we interpret each of \( f_1 \) and \( f_2 \) as the bending in the \( z \) direction of a metal sheet, or thin plate, extending in the \( x - y \) plane, the energy in (2.3) is the analogy of the
thin plate bending energy. The problem can be solved using a standard Green’s function method. The Green’s function for this problem is the function satisfying the equation

\[
\Delta^2 U(\vec{x}, \vec{x}’) = \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) \left( \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right) U = \delta(\vec{x}, \vec{x}’) \tag{2.5}
\]

where \(\delta(\vec{x}, \vec{x}’)\) is the Dirac delta function and \(\vec{x} = (x, y)\) and \(\vec{x}’ = (x’, y’) \in \mathbb{R}^2\).

This can be solved with the solution

\[
U(r) = r^2 \log r^2, \tag{2.6}
\]

where \(r\) is the distance \(\sqrt{x^2 + y^2}\).

Because an affine transformation has a zero contribution to the bending energy, the transformation allows a free affine transformation. Define the matrices

\[
K = \begin{bmatrix}
0 & U(r_{12}) & \ldots & U(r_{1n}) \\
U(r_{21}) & 0 & \ldots & U(r_{2n}) \\
\vdots & \vdots & \ddots & \vdots \\
U(r_{n1}) & U(r_{n2}) & \ldots & 0
\end{bmatrix}, \text{ which is } n \times n; \tag{2.7}
\]

\[
P = \begin{bmatrix}
1 & x_1 & y_1 \\
1 & x_2 & y_2 \\
\vdots & \vdots & \vdots \\
1 & x_n & y_n
\end{bmatrix}, \text{ which is } 3 \times n; \tag{2.8}
\]

and

\[
L = \begin{bmatrix}
K & P \\
PT & O
\end{bmatrix}, \text{ which is } (n + 3) \times (n + 3), \tag{2.9}
\]

where the symbol \(T\) is the matrix transpose operator and \(O\) is a \(3 \times 3\) matrix of zeros.
Let \( V = (v_1, ..., v_n) \) be any \( n \)-vector and write \( Y = (V | 000)^T \). Define the vector \( W = (w_1, ..., w_n) \) and the coefficients \((a_1, a_x, a_y)\) by the equation

\[
L^{-1}Y = (W | a_1 a_x a_y)^T.
\]

Finally we have the solution

\[
f(x, y) = a_1 + a_x x + a_y y + \sum_{i=1}^{n} w_i U(|P_i - (x, y)|).
\] (2.11)

2.3 More on Splines

Bookstein’s seminal work applying thin-plate splines to the landmark problems in 2D \([4, 5]\) is well known in the computer vision and imaging community. What is less known is the theoretical foundations for the thin-plate spline laid by Duchon \([21, 22, 23]\) and Meinguet \([48]\) in the 1970s.

This solution is correct and simple. However it is not numerically stable because it involves the inverse of the large kernel matrices. A numerically stable solution is given by Wahba \([70]\) using the QR decomposition. Wahba has an account for a more generalized formulation for \(d\)-dimensional space and the energy involving \(m^{th}\) order partial derivatives. Here we only discuss the special thin plate spline with \(d = 2\) and \(m = 2\). Later in our work in 3D situations, 3D thin plate splines are used where \(d = 3\) but it is straightforward to generalize from 2D to 3D, while the kernel in 3D is different than that for 2D.

Let \(f, g \in W^{k,2}(\Omega)\), where \(W^{k,2}(\Omega)\) is the Sobolev space. The inner product of \(f\) and \(g\) is defined as

A thin-plate smoothing problem in 2D is to find \(f \in W^{k,2}(\Omega)\) to minimize the functional

\[
\frac{1}{n} \sum_{i=1}^{n} ||q_i - f(p_i)||^2 + \lambda E(f).
\] (2.12)
The null space is spanned by

$$\phi_1(t) = 1, \phi_2(t) = x, \phi_3(t) = y,$$

which is an affine space.

Duchon [23] proved that if \( \{p_i \in \Omega_1 | i = 1, 2, ..., n\} \) are such that least squares regression on \( \phi_1, \phi_2, \phi_3 \) is unique, then 2.12 has a unique minimizer \( \hat{f} \), with the representation

$$\hat{f}(t) = \sum_{\nu=1}^{3} d_{\nu}\phi_{\nu}(t) + \sum_{i=1}^{n} c_{i}K(t, p_i),$$

where \( K \) is a Green’s function for the double iterated Laplacian. \( K \) has the form

$$K = K(|s - t|),$$

and

$$K(\tau) = \tau^2 \log |\tau|.$$  

(2.15)

(2.16)

By using integration by part, we obtain that \( c, d \) are the minimizers of

$$\frac{1}{n}||y - Td - Kc||^2 + \lambda c'Kc$$

subject to \( T'c = 0 \), where \( K \) is a \( n \times n \) matrix with \( ij \)th entries \( K(p_i, p_j) \). Let us do a QR decomposition on \( T \)

$$T = (Q_1 : Q_2) \begin{pmatrix} R \\ 0 \end{pmatrix}$$

(2.18)

where \( (Q_1, Q_2) \) is orthogonal and \( R \) is lower triangular. \( Q_1 \) is \( n \times 3 \) and \( Q_2 \) is \( n \times (n - 3) \). Since \( T'c = 0 \), \( c \) must be in the column space of \( Q_2 \), with \( c = Q_2\gamma \) for some \( n - 3 \) vector \( \gamma \). Substituting in 2.17 and the energy is in the form of
\[ \frac{1}{n}||Q_2'y - Q_2'KQ_2\gamma||^2 + \frac{1}{n}||Q_1'y - Rd - Q_1'KQ_2\gamma||^2 + \lambda\gamma'Q_2'KQ_2\gamma. \]  

(2.19)

The solutions are

\[ d = R^{-1}(y - KQ_2\gamma) \]
\[ c = Q_2\gamma, \]  

(2.20)

with

\[ \gamma = (Q_2'KQ_2 + n\lambda I)^{-1}Q_2'y \]  

(2.21)

where \( I \) is the identity matrix.

2.4 Reproducing Kernel Hilbert Space (RKHS) Formulation

It is possible to come to the same result for the minimizer using reproducing kernels and from this point of view it is easier to understand the existence and uniqueness of the solution \([70, 7]\).

Let \( ||f||^2 = E = \int_\Omega (f_{xx}^2 + 2f_{xy}^2 + f_{yy}^2)dx\,dy \), where \( ||f|| \) is the norm of \( f \) in \( W^{k,2}(\Omega) \). Since \( W^{k,2}(\Omega) \) is a Hilbert space, from Riesz representation theorem, for any \( x \in \Omega \), the evaluation linear functional

\[ \delta_x : W^{k,2}(\Omega) \to R, \]
\[ \delta_x(f) = f(x) \]  

(2.22)

has a representer \( u_x \in W^{k,2}(\Omega) \) such that

\[ f(x) = <u_x, f>. \]  

(2.23)

Now the problem is transformed to a new problem: find a function \( f \in W^{k,2}(\Omega) \) with minimal norm \( ||f|| \), subject to constraints
\[ <u_{X_i}, f> = v_i, \ i = 1, 2, \ldots, n. \quad (2.24) \]

For \( X_a, X_b \in \Omega, u(X_a, X_b) = u_{X_a}(X_b) \) is the kernel of the reproducing kernel Hilbert space.

Let \( T \) be the linear subspace spanned by \( u_{x_i}, i = 1, 2, \ldots, n \). Any function \( f \in W^{k,2}(\Omega) \) can be decomposed into \( f = f_T + f_\perp \) where \( f_T \in T \) and \( f_\perp \) is in the orthogonal complement of \( T \) and hence \( <u_{X_i}, f_\perp> = 0 \). We know if \( f_T \) satisfies (2.24), then \( f \) also satisfies (2.24) only with \( ||f|| > ||f_T|| \) if \( f_\perp \neq 0 \). So we only need to search for the solution in \( T \). The general solution thus can be written as

\[
f(X) = a_0 + a_1 x + a_2 y + \sum_{i=1}^{n} w_i u(X_i, X),
\]

where functions of the form \( a_0 + a_1 x + a_2 y \) span the null space. With this form, \( E \) can be rewritten as

\[
\sum_{i=1}^{n} w_i U_{ij} w_j = WW^+. \quad (2.25)
\]

### 2.5 The Folding Problem of Splines

This is all very nice except as mentioned above there is no mechanism to guarantee a bijection in order to be homeomorphic or diffeomorphic. Intuitively this problem is known as the folding of space.

Figure 2-2 shows the displacement of landmarks. Figure 2-3 is the thin-plate spline interpolation. We can see the folding of space. This is the drawback of thin-plate spline interpolation. Due to the folding of space, features in the template may be smeared in the overlapping regions. And furthermore, the transformation is not invertible. A diffeomorphic transformation is strongly desirable, since it preserves the features, the topology and is smooth as shown in Figure 2-4.
2.6 Imposing Restriction on the Jacobian

One straightforward approach to find a diffeomorphism for practical use is to make some remedy on the thin-plate spline. We can restrict our search space to the set of diffeomorphisms and the ideal one should minimize the thin-plate energy. We make the observation that if the Jacobian of the transformation $f$ changes sign at a point, then there is folding. We put the constraint requiring the Jacobian is always positive. There is some literature on this approach but most of these approaches does not guarantee that the transformation is smooth everywhere [38, 11].

2.7 The Flow Approach

Another approach is to utilize a flow field[28, 39, 49]. We introduce one parameter, the time $t$ into the diffeomorphism. Let $\phi_t : \Omega \rightarrow \Omega$ be the diffeomorphism from $\Omega$ to $\Omega$ at time $t$. A point $x$ is mapped to the point $\phi_t(x)$. Sometimes we also denote this as $\phi(x, t)$. It is easy to verify that for all the values of $t$, $\phi_t$ form a one parameter diffeomorphism group. If $x$ is fixed, then $\phi(x, t)$ traces a smooth trajectory in $\Omega$. The interpolation problem becomes: find the one parameter diffeomorphic group $\phi(\cdot, t) : \Omega \rightarrow \Omega$ such that given $p_i \in \Omega$ and $q_i \in \Omega \forall i = 1, 2, ..., n$, $\phi(x, 0) = x$ and $\phi(p_i, 1) = q_i$. We introduce the velocity field $v(x, t)$ and construct a dynamical system using the transport equation

$$\frac{\partial \phi(x, t)}{\partial t} = v(\phi(x, t), t). \tag{2.26}$$

The integral form of the relation between $\phi(x, t)$ and $v(x, t)$ is

$$\phi(x, 1) = x + \int_0^1 v(\phi(x, t), t)dt. \tag{2.27}$$

Obviously, such a $\phi(x, t)$ is not unique and there are infinitely many such solutions. With the analogy to the thin-plate spline, it is natural that we require the desirable diffeomorphism results in minimal space deformation. Namely we require the deformation energy
\[ \int_0^1 \int_\Omega ||Lv(x,t)||^2 dxdt \tag{2.28} \]
to be minimized, where \( L \) is a given linear differential operator.

The following theorem [39] states the existence of such a velocity field and shows a way to solve for it.

**Theorem 4.** Let \( p_i \in \Omega \) and \( q_i \in \Omega \) \( \forall i = 1, 2, ..., n \). The solution to the energy minimization problem

\[ \hat{v}(\cdot) = \arg \min \int_0^1 \int_\Omega ||Lv(x,t)||^2 dxdt \tag{2.29} \]

subject to

\[ \phi(p_i, 1) = q_i, \quad \forall i = 1, 2, ..., n \tag{2.30} \]

where

\[ \phi(x, 1) = x + \int_0^1 v(\phi(x,t), t) dt \tag{2.31} \]
exists and defines a diffeomorphism \( \phi(\cdot, 1) : \Omega \to \Omega \). The optimum velocity field \( \hat{v} \) and the diffeomorphism \( \hat{\phi} \) are given by

\[ \hat{\nu}(x,t) = \sum_{i=1}^n K(\phi(x,t), x) \sum_{j=1}^n (K(\phi(t))^{-1})_{ij} \hat{\phi}(x_j, t) \tag{2.32} \]

where

\[ K(\phi(t)) = \begin{pmatrix} K(\phi(p_1, t), \phi(p_1, t)) & \cdots & K(\phi(p_1, t), \phi(p_n, t)) \\ & \ddots & \vdots \\ & \vdots & \ddots \end{pmatrix} \tag{2.33} \]
with \( (K((\phi(t))_{ij} \text{ denoting the } ij, \ 3 \times 3 \text{ block entry } (K(\phi(t))_{ij} = K(\phi(p_i, t), \phi(p_j, t)), \)
and

\[
\hat{\phi}(p_n, \cdot) = \arg \min_{\phi(p_n, \cdot)} \int_0^1 \sum_{ij} \phi(p_i, t)^T (K(\phi(t))^{-1})_{ij} \phi(p_j, t) dt \tag{2.34}
\]

subject to \( \phi(p_i, 1) = q_i, \ i = 1, 2, ..., N \) with the optimal diffeomorphism given by

\[
\hat{\phi}(x, 1) = x + \int_0^1 \hat{v}(\phi(x, \hat{t}), t) dt . \tag{2.35}
\]

The proof [39] is omitted here. With this theorem, we can convert the original optimization problem looking for the vector field \( \hat{v}(x, t) \) to a problem of finite dimensional optimal control with end point conditions.

This problem is called the exact matching problem because we required the images of the given points \( p_i, i = 1, 2, ..., n \) are exactly another set of given points \( q_i, i = 1, 2, ..., n \). The exact matching problem is symmetric with respect to two sets of landmarks or two point shapes. Swapping the two point sets \( \{p_i \in \Omega_1 | i = 1, 2, ..., n\} \) and \( \{q_i \in \Omega_2 | i = 1, 2, ..., n\} \), results in the new optimal diffeomorphism to be the inverse of the old diffeomorphism.

The exact matching problem can be generalized to the inexact matching problem. In the inexact matching problem, we do not require the points exactly match. Instead, we seek a compromise between the closeness of the matching points and the deformation of space. We minimize

\[
\int_0^1 \int_{\Omega} ||Lv(x, t)||^2 dx dt + \lambda \sum_{i=1}^n ||q_i - \phi(p_i, 1)||^2 , \tag{2.36}
\]

which can be similarly solved.

As seen from the formulation of the problem, there may be infinitely many diffeomorphisms that interpolate the two sets of landmarks. A usual way to find a particular desirable diffeomorphism is to require the diffeomorphism to minimize a
certain objective function. Camion and Younes [7] proposed a different objective function in the form of

\[
E(v, q) = \int_0^1 \int_\Omega ||Lv(x,t)||^2dxdt + \sum_{i=1}^n \left( \frac{dq_i(t)}{dt} - v(q_i(t), t) \right)^2 dt
\]  

over all time dependent velocities \( v(x,t) \) on \( \Omega \) and over all trajectories \( q_1(t), ..., q_n(t) \). This can be interpreted as a geodesic distance between two points on the configuration manifold. This is generalized to exact matching by Marsland et al. [46].

### 2.8 Correspondence and Softassign

So far all work has assumed landmark matching with a known correspondence. When the correspondence is unknown, the problem is dramatically complicated. One approach is the softassign [54, 43] method, which solves for the correspondence as a permutation problem using linear programming in the space of doubly stochastic matrices [65, 3]. The problem of finding the correspondence between two sets of points can be formulated as finding the permutation matrix.

This approach is expensive in computational time. Chui et al. [12, 17, 14, 13, 15] adopted the joint clustering scheme, in which the correspondence and space deformation is estimated simultaneously. However, in all the work, splines are used and a potential drawback is that a diffeomorphic mapping in space is not guaranteed.

### 2.9 Distance Transforms

An unusual way of getting around the point correspondence problem is to use distance transforms to convert the point matching problem to an image matching problem. The distance transform was first introduced by Rosenfeld and Pfaltz [60], and it has a wide range of applications in image processing, robotics, pattern recognition and pattern matching. Paragios et al. [53] give one example of using distance transforms to establish local correspondences for compact representations.
of anatomical structures. Distance transform applies to binary images, as well as point sets, which can be thought as a special case of a binary image. Suppose we have a domain \( \Omega \), and a point set \( S \subset \Omega \). For each point \( x \in \Omega \), we assign to it a non-negative real number, which is the shortest distance from \( x \) to all the points in \( S \). This way we obtain a scalar field in domain \( \Omega \). We treat this scalar field as a gray scale image, we call it the distance transformed image of the point set. If we have two point sets, we can first perform the distance transform on the two sets respectively and later register the two distance transformed images. However, there are problems with this approach. If the matching of the two distance transformed images is intensity based, then the original points may not be matched exactly. Even in the inexact matching case, the optimization for the distance transformed images is over the entire image region and that may not be optimal for the original point set. If the matching of the two distance transformed images is level set based, the level sets in the two images may not be topologically equivalent, as shown in Figure 2–5 and Figure 2–6. In Figure 2–5, there are two distance transformed images, with three points each in the image. Figure 2–6 shows the level sets of the two distance transformed images. Hence this method cannot guarantee to obtain a diffeomorphism.

Due to the indirect approach of transforming point sets into distance transforms, this method has not seen wide applicability for point sets.

2.10 Implicit Correspondence

It is possible to define distance measures between two shapes, with the shapes viewed as point sets, without the knowledge of correspondence. Hausdorff distance between two point sets is a well-known distance measure for such purposes. Huttenlocher et al. [37] develop methods comparing shapes using Hausdorff distance. Glaunes et al. [27] use another method to circumvent the correspondence problem by introducing a distance measure of the two shapes using an external
Hilbert space structure. If the two shapes are point sets, the distance measure between two point sets used in Glaunes et al. [27] is

\[ \frac{1}{\sigma^2_R} \left| \phi_{\mu} - v \right|^2, \]

(2.38)

where \( |\mu|^2 \) is the norm squared in some Hilbert space \( I^* \) and

\[ |\mu|^2 = \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} c_i c_j k_I(x_i, y_j) \]

(2.39)

where \( c_i, c_j \) are constant coefficient. \( x_i \) are the points in the first sets and \( y_j \) are the points in the second sets. \( k_I \) is some kernel. In their experiments, they used radial basis function kernels

\[ k_I(x, y) = f_I \left( \frac{|x - y|^2}{\sigma^2_I} \right), \]

(2.40)

with \( f_I(u) = e^{-u} \) and \( f_I(u) = \frac{1}{1+u} \).

The idea of distance measure without explicit correspondence actually goes back to Grimson et al. [29] and Lu and Mjolsness [45] in 1994. The distance measure used is

\[ \sum_i \sum_j e^{-\frac{|T_{ij} - m_{ij}|^2}{2\sigma^2}}. \]

(2.41)

As pointed out in Rangarajan et al. [55] this is equivalent to using

\[ E(M, T) = \alpha \sum_{ij} M_{ij} D_{ij}(T) + \sum_{ij} M_{ij} \log M_{ij}, \]

(2.42)

where \( M_{ij} \) is the explicit correspondence and \( T \) is the space transformation and \( D_{ij} \) is the distance measure between point pairs that are in correspondence. Guo et al. suggested a joint clustering algorithm for solving 2D diffeomorphic point matching problems, with unknown correspondence, using explicit correspondences [33, 32].

The different formulations with explicit correspondence and implicit correspondence
are closely related to each other, through a Legendre transform, as pointed out by Mjolsness and Garrett [51].

2.11 Shape Context

In the approach of shape context [2], the correspondence is expressed explicitly and the correspondence problems is tackled directly. Moreover, the correspondence is solved separately from the space transformation with the help of shape context. Shape context is the local shape information at each point. For each point \( q \) and all other points \( p_i \), we can draw a vector from \( q \) to \( p_i \), that is \( p_i - q \). The local shape context information is all stored in this set of \( n - 1 \) vectors. This information is rich and in practice the distribution of these \( n - 1 \) vectors gives us more robust, compact and discriminative descriptor. For each point \( p_i \), we compute a histogram \( h_i \) of these \( n - 1 \) vectors

\[
    h_i(k) = \# \{ q \neq p_i \mid (q - p_i) \in \text{bin}(k) \}. \tag{2.43}
\]

The histogram is defined to be the shape context of point \( p_i \). Bins that are uniform in log-polar space are used to make the nearby context points more important the the far away context points. The cost for the correspondence between two shapes is the sum of the cost of corresponding point pairs

\[
    H(\pi) = \sum_i C(p_i, q_{\pi(i)}), \tag{2.44}
\]

which is a function of the permutation \( \pi \), and the cost with each individual pair of points is defined as

\[
    C_{ij} = C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^{K} \frac{(h_i(k) - h_j(k))^2}{h_i(k) + h_j(k)}. \tag{2.45}
\]

2.12 Active Shape Models

Cootes et al. propose a method, which they call Active Shape Models [18] to locate objects in images, with the help of model shapes and the training of
these model shapes. This approach plays as a bridge between the landmark based methods and image intensity based methods, and also a bridge between rigid matching and non-rigid matching. They use landmark points on the models but not on the test images. The landmarks are hand picked and hand labeled. The automatic correspondence problem is circumvented by the human involvement of the landmark picking and labeling process. The deformation model of the template is similarity, which is very close to rigid, allowing translation, rotation plus a scaling. What makes it applicable to non-rigid deformation is that they learn the statistics of the model shapes through a set of training samples and find the mean and variance in higher dimensional space and apply the PCA analysis with the variances. When it applies to locating the model in the image, they use the snake model, which is non-rigid in nature, with the help and restriction of the statistics of the model shapes.

2.13 Deterministic Annealing Applied to EM Clustering

Deterministic annealing is an effective technique used in the clustering problems. The clustering problem is a non-convex optimization problem. The traditional clustering techniques use descent based algorithms and they tend to get trapped in a local minimum. Rose et al. [59] proposed an annealing approach using analogies to statistical physics.

The clustering problem is to partition a set of data points, \( \{x_i \in \mathbb{R}^d | i = 1, 2, ..., n \} \) into \( K \) clusters \( C_1, C_2, ..., C_K \), with the centers \( r_1, r_2, ..., r_K \) respectively. In the fuzzy clustering literature, we call the probability of each point \( p_i \) belonging to each cluster \( C_j \) the fuzzy membership. Hard clustering is a marginal special case, where each point is deterministically associated with a single cluster. Let \( E_j(x_i) \) denote the energy associated with assigning a data point \( x_i \) to cluster \( C_j \).
The average total energy is

\[< E > = \sum_i \sum_j P(x_i \in C_j) E_j(x_i). \quad (2.46)\]

Since we do not make any assumption about the data distribution, we apply the principle of maximum entropy. It is well known from statistical physics, the association probabilities \( P(x_i \in C_j) \) that maximize the entropy under constraint (2.46) are Gibbs canonical distributions,

\[P(x_i \in C_j) = \frac{1}{Z_i} e^{-\beta E_j(x_i)}, \quad (2.47)\]

where \( Z_i \) is the partition function

\[Z_i = \sum_k e^{-\beta E_k(x_i)}. \quad (2.48)\]

The parameter \( \beta \) is the Lagrange multiplier determined by the given value of \(< E > \) in (2.46). In the analogy in statistical physics, \( \beta \) is inversely proportional to the temperature \( T \). We have assumed that we have a fixed set of clusters. We want to extend this to include optimization over the number of clusters as well. The optimal solution then is the one that minimizes the free energy

\[F = -\frac{1}{\beta} \ln Z. \quad (2.49)\]

The set of cluster centers are the ones that satisfy

\[\frac{\partial F}{\partial r_j} = 0, \quad \forall j. \quad (2.50)\]

And the solution is

\[r_j = \frac{\sum_i x_i P(x_i \in C_j)}{\sum_i P(x_i \in C_j)}. \quad (2.51)\]

This procedure determines a set of cluster centers \( \{r_j \in \mathbb{R}^d | j = 1, 2, ..., n\} \) for each fixed \( \beta \). Generally, changing \( K \), the imposed number of clusters, will modify the positions of the set of cluster centers. However, there exists some \( n_c \) such that
for all $K > n_c$, one gets only $n_c$ distinct cluster centers while the remaining $K - n_c$ cluster centers are repetitions from this set. Thus at each given $\beta$ we get at most $n_c$ clusters. Here we assume $K \geq n_c$ at a given $\beta$ and we only consider them without repetitions.

The free energy $F$ and $\beta$ are Legendre transform images of each other. Fixing one of them determines the other. For $\beta = 0$, each data point is uniformly associated with all clusters and all the centers have the same location, the centroid of the data. Clearly, for $\beta = 0$ we have a single minimum, which is the global minimum, for $F$, and the entire data set is interpreted as one cluster. At higher $\beta$, the free energy may have many local minima, and the concept of annealing emerges here can be viewed as tracking the global minimum while gradually increasing $\beta$. Moreover, at $\beta = 0$ there is only one cluster ($n_c = 1$), but at some positive $\beta$ we shall have $n_c > 1$. In other words, this cluster will split into smaller clusters, and will thus undergo a phase transition. The first phase transition occurs at a critical value for $\beta$

$$\beta_c = \frac{1}{2\lambda_{\text{max}}}, \quad (2.52)$$

where $\lambda_{\text{max}}$ is the largest eigenvalue of $Cxx$, the covariance matrix of the data set. Given $\beta = 1/T$ and for Gaussian mixture clustering $T = 2\sigma^2_T$, it is quite understandable that the critical value $\sigma_c$ for the prescribed $\sigma_T$ is $\sigma_c = \sqrt{\lambda_{\text{max}}}$. 

2.14 Statistical Shape Analysis on Differentiable Manifolds

David Kendall first introduced the idea of representing shapes in complex projective spaces. The idea is developed to represent shapes on general differentiable manifolds, or shape spaces [66]. A shape is constructed from a sequence of landmarks $p_1, p_2, ... p_n$ with $p_i \in \mathbf{R}^2$. Namely each landmark is a point in 2-dimensional Euclidean space. The two sequences of landmarks in $\mathbf{R}^2$ are considered of the same shape if they differ only by a similarity transformation in $\mathbf{R}^2$. A shape then is
defined as an equivalent class of these landmark sequences and the shape space is
carved out of the quotient space defined by the equivalent relation.

One way to distill the shape information out of the landmark sequence is to
remove the location, scale and orientation. To remove the location, we can define
\[
    r_i = x_i - \bar{x},
\]
where
\[
    \bar{x} = \frac{1}{n} \sum_{j=1}^{n} x_j
\]
is the mean or centroid of the landmarks. That is, to remove the location, we make
the landmarks mean zero. To remove the scale we can make the variance of the
landmarks as one. So we define
\[
    \tau_i = \frac{x_i - \bar{x}}{\sqrt{\sum_{j=1}^{n} ||x_j - \bar{x}||^2}}.
\]
We refer the vector $\tau$ as the pre-shape of the landmarks. The pre-shape space is the
intersection of the $(n - 2)$-dimensional subspace
\[
    F^{2n-2} = \{(x_1, \ldots, x_n) \in \mathbb{R}^{2n} \mid \sum_{j=1}^{n} x_j = 0\}
\]
with the unit sphere
\[
    S^{2n-1} = \{(x_1, \ldots, x_n) \in \mathbb{R}^{2n} \mid \sum_{j=1}^{n} ||x_j||^2 = 1\}.
\]
The intersection
\[
    S^{2n-3} = F^{2n-2} \cap S^{2n-1}
\]
is a $(2n - 3)$-dimensional sphere within the ambient Euclidean space $\mathbb{R}^{2n}$.

It is more difficult to remove the orientation of the shape. To do this, we
define the orbit of a pre-shape $\tau \in S^{2n-3}$ is the circle
\[
    O(\tau) = \{\theta(\tau) \mid 0 \leq \theta < 2\pi\} \subset S^{2n-3}.
\]
Two pre-shapes are of the same shape if they are on the same orbit. A shape is defined as the equivalent class of the pre-shapes. If \( \tau_1 \) and \( \tau_2 \) are two preshapes, then the great circle distance between \( \tau_1 \) and \( \tau_2 \) is given by

\[
d(\tau_1, \tau_2) = \cos^{-1}(\langle \tau_1, \tau_2 \rangle).
\] (2.60)

The induced metric on \( \Sigma^m_2 \) is then defined as

\[
d[O(\tau_1), O(\tau_2)] = \inf\{d([\theta_1(\tau_1), \theta_2(\tau_2)]) | 0 \leq \theta_1, \theta_2 < 2\pi\}. 
\] (2.61)

This is called Procrustean distance. It can be proved that if \( \tau_1 \) and \( \tau_2 \) are two representatives of two shapes, the procrustean distance between the two shapes can be expressed as

\[
d(\sigma_1, \sigma_2) = \cos_1(| \sum_{k=1}^{n} \tau_{1k}\tau_{2k}^* |).
\] (2.62)

The standard statistics like means and variances can be performed on the shape manifold.

One drawback of this approach is that the shape is treated as the sequence of landmarks instead of sets. That means if we two identical sets of landmarks and only label them differently, this theory treats them as two distinct shape. It does not consider to make an equivalent class from the permutation of landmark points.

2.15 Distance Measures from Information Theory

Information geometry is an emerging discipline that studies the probability and information by way of differential geometry. In information geometry, every probability distribution is a point in some space. A family of distributions corresponds to points on a differentiable manifold.

Endres et al. [25] proposed a metric for two distributions. Given two probability distributions \( P \) and \( Q \), and \( R = \frac{1}{2}(P + Q) \), we can define a distance \( D_{PQ} \)
between the two distributions by

\[ D_{PQ}^2 = 2H(R) - H(P) - H(Q) \]

\[ = D(P||R) + D(Q||R) \]

\[ = \sum_{i=1}^{N} \left( p_i \log \frac{2p_i}{p_i + q_i} + q_i \log \frac{2p_i}{p_i + q_i} \right). \] (2.63)

This metric can also be interpreted as the square root of an entropy approximation to the logarithm of an evidence ratio when testing if two samples have been drawn from the same underlying distribution. \( \frac{1}{2}D_{PQ}^2 \) is named Jensen-Shannon divergence, which is defined as

\[ D_{\lambda}(P, Q) = \lambda D(P||R) + (1 - \lambda)D(Q||R) \] (2.64)

\[ R = \lambda P + (1 - \lambda)Q \]

and therefore

\[ \frac{1}{2}D_{PQ}^2 = D_{\frac{1}{2}}(P, Q). \] (2.65)

With probability distribution \( P(x|\theta) \) where \( \theta \) is a set of parameters \( \theta_1, ..., \theta_n \), the Fisher information is defined as [26]

\[ G_{ij}(\theta) = -E \left[ \frac{\partial^2 \log p(x|\theta)}{\partial \theta_i \partial \theta_j} \right] = -\int p(x|\theta) \frac{\partial^2 \log p(x|\theta)}{\partial \theta_i \partial \theta_j} dx. \] (2.66)

C. R. Rao [57] suggested this is a metric. In fact, it is the only suitable metric in parametric statistics and it is called Fisher-Rao metric.
Figure 2–1: Deformation of the thin plate

Figure 2–2: Landmark displacements
Figure 2–3: Thin-plate Spline interpolation

Figure 2–4: Diffeomorphic interpolation
Figure 2–5: Two distance transformed images of three landmarks
Figure 2–6: Level sets of two distance transformed images
CHAPTER 3
DIFFERENTIAL POINT MATCHING

In this chapter we investigate the differential point matching theory and apply the theory to shapes in medical imaging. In Section 3.1 we prove a theorem about the existence of a differential mapping matching the landmarks. In Section 3.2, we prove another theorem about the symmetric nature in the case of exact landmark matching. In Section 3.3 we formulate a theory of differential point matching with unknown correspondence and devise an objective function. In Section 3.4 we design an algorithm to solve the problem using joint clustering and deterministic annealing. In Section 3.5 we apply the algorithm to 2D corpus callosum shapes. In Section 3.6 perform matching on 3D hippocampus shapes. Both corpus callosum and hippocampus are parts in the human brain and the matching of these shapes have great significance in medical treatment and medical research.

3.1 Existence of a Differential Mapping in Landmark Matching

We have discussed in Chapter 1 that in 2D a similarity transformation can map exactly 2 given points to 2 given points. A affine transformation can map exactly 3 given points to 3 given points. A projective transformation can map exactly 4 given points to 4 given points. Now given \( n \) arbitrary distinct points \( \{p_i \in \mathbb{R}^2 | i = 1, 2, \ldots, n\} \) and another set of \( n \) arbitrary distinct points \( \{q_i \in \mathbb{R}^2 | i = 1, 2, \ldots, n\} \), we want to find a differentialism \( f : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) such that \( f(p_i) = q_i \). It is natural to ask the question, does such a differential mapping always exist? Our intuition is it exist and there are infinitely many such differentialisms. This is stated as our first theorem and the proof follows.
Theorem 5. A diffeomorphic transformation that interpolates arbitrary number of \( n \) pairs of landmarks always exists.

Proof. We show the existence by construction. We construct a simple, although most likely undesirable in most of the applications, diffeomorphism. The intuitive idea is to “dig canals” connecting the landmark pairs. We first choose the first pair of landmarks \( p_1 \) and \( q_1 \). Assume no other landmarks lie on the line connecting \( p_1 \) and \( q_1 \). Establish a coordinate system such that \( p_1 \) and \( q_1 \) are on the \( x \) axis. Let the signed distance from \( p_1 \) to \( q_1 \) be \( d \). Construct the transformation \( f_1 : \Omega_1 \to \Omega_2 \) such that \( f_1(x, y) = (x', y') \),

\[
\begin{align*}
x' &= x + de^{-v^2} \\
y' &= y
\end{align*}
\]

where \( v = \tan\left(\frac{\pi}{2} \epsilon\right) \), for any arbitrarily small \( \epsilon \). It is easy to show that \( f_1 \) is a diffeomorphism and it maps \( p_1 \) to \( q_1 \) and keeps all other landmarks \( q_2, \ldots, q_n \) fixed. This is very much like the flow of viscous fluid in a tube. Similarly we can construct diffeomorphism \( f_i \) that maps \( p_i \) to \( q_i \) and keeps all other landmarks fixed, for \( i = 1, 2, \ldots, n \). The composition of this series of diffeomorphisms

\[
f = f_n \circ \ldots f_2 \circ f_1
\]

is also a diffeomorphism and obviously \( f \) maps \( p_i \) to \( q_i \), for \( i = 1, 2, \ldots, n \).

If some landmark \( q_k \) lies on the line of \( p_i \) and \( q_i \), we can find such a direction such that we draw a line \( l_k \) through \( q_k \) and there are no other landmarks on the line. Then we make a diffeomorphism \( h \) transporting \( q_k \) to a nearby point \( q_k' \) along the line without moving any other landmarks, using the same canal of viscous fluid technique. Then we make diffeomorphism \( f_i \) as described before. After that, we move landmark \( q_k' \) back to the old position with the inverse of \( h^{-1} \). So we use \( F_i = h^{-1} f_i h \) in place of \( f_i \).
3.2 Symmetric Matching due to Time Reversibility

Asymmetry exist in many image and shape matching situations. Suppose we have a point set \( \{p_i \in \mathbb{R}^2 | i = 1, 2, ... n\} \) and another point set \( \{q_i \in \mathbb{R}^2 | i = 1, 2, ... n\} \), we find the diffeomorphic mapping \( f : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) which minimize the energy functional \( E(f) \) subject to the constraints \( f(p_i) = q_i \). Now we define a reverse problem, namely to find a mapping \( g : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) which minimize the same energy functional \( E(f) \) subject to the constraints \( f(p_i) = q_i \) as shown in Figure 3–2. In general, depending on the objective function to minimize, \( g \neq f^{-1} \). This is called asymmetry of the matching. In some cases we do have \( g = f^{-1} \). The matching is called symmetric then and this is a nice property to have. The following theorem states that for the exact diffeomorphic landmark matching case, the matching is symmetric due to the time reversibility of the flow.
Figure 3-2: Asymmetry of the matching

**Theorem 6.** If $\phi(x_k, 1) = y_k$ and $\phi(x, t)$ and $v(x, t)$ minimize the energy

$$E = \int_0^1 \int_\Omega ||Lv(x, t)||^2 dx dt,$$

then the inverse mapping maps the landmarks backward $\phi^{-1}(y_k, 1) = x_k$ and $\phi^{-1}(x, t)$ and $-v(x, -t)$ also minimize the energy $E$.

**Proof.** First, from the known property of the diffeomorphism group of such a dynamical system, $\phi(x, t_1 + t_2) = \phi(\phi(x, t_1), t_2)$, it is easy to show that $\phi^{-1}(x, t) = \phi(x, -t)$. This is because $\phi(., -t) \circ \phi(., t)(x) = \phi(., t) \circ \phi(., -t)(x) = \phi(\phi(x, t), -t) = \phi(x, t + (-t)) = \phi(x, 0) = x$. And $\phi(x, -t)$ and $-v(x, -t)$ also satisfy the transport equation $\frac{\partial \phi(x, -t)}{\partial t} = -v(\phi(x, -t), -t)$. Suppose $\phi(x, t)$ and $v(x, t)$ minimize the energy $E = \int_0^1 \int_\Omega ||Lv(x, t)||^2 dx dt$, but $\phi^{-1}(x, t) = \phi(x, -t)$ and $-v(x, -t)$ do not minimize the energy $E = \int_0^1 \int_\Omega ||Lv(x, t)||^2 dx dt$. Let the minimizer be $\psi(x, t)$ and $u(x, t)$ such that $\forall k, \psi(y_k) = x_k$ and $\int_0^1 \int_\Omega ||Lu(x, t)||^2 dx dt < \int_0^1 \int_\Omega ||Lv(x, t)||^2 dx dt$.

Then, we can construct $\psi^{-1}(x, t) = \psi(x, -t)$ such that $\psi^{-1}(x, t)$ and $-u(x, -t)$ satisfy the transport equation and $\psi^{-1}(x_k, 1) = y_k$. However $\int_0^1 \int_\Omega ||Lu(x, t)||^2 dx dt < \int_0^1 \int_\Omega ||Lv(x, t)||^2 dx dt$ contradicts the assumption that $v(x, t)$ is the minimizer of the energy $E$. 

\qed
3.3 A Theoretical Framework for Diffeomorphic Point Matching

There are different ways to solve for the unknown point correspondences [56, 14, 17, 16]. Essentially, within the framework of explicit point correspondences—as opposed to the distance function framework of implicit correspondence—we have a choice between i) solving for an optimal permutation and ii) letting corresponding “labeled” points discover their optimal locations. We opt for the latter in this work because of its simplicity. The clustering in fact serves two purposes. First, it is the method to find the unknown correspondence. We initialize the two sets of cluster centers around the centroids of their data points, respectively. The cluster centers are labeled with identical labels in the two sets denoting correspondence. The cluster centers evolve during the iterations of an incremental EM algorithm and they are linked by a diffeomorphism and are forced to move in lock-step with one another. Second, clustering is the modeling of the real data sets, with noise and/or outliers because with two shapes represented by point samples, we cannot assume a point-wise correspondence. The correspondence is only between the two shapes and clustering is a useful way to model the shapes.

We use a Gaussian mixture model to describe the clustering of the point sets. For more details on this approach along with justifications for the use of this model, please see [15, 17]. The Gaussian mixture probability density is

\[ p(x|r, \sigma_T) = \frac{1}{N} \sum_{k=1}^{N} \frac{1}{(2\pi\sigma^2_T)^{d/2}} \exp\left(-\frac{1}{2\sigma^2_T}||x - r_k||^2\right) \]  

(3.3)

with \( x \) being a point in \( \mathbb{R}^d \), \( r \) as the collective notation of a set of cluster centers and \( \sigma^2_T \) as the variance of each Gaussian distribution. The reason for the notation of subscript \( T \) will be discussed in Section 3.4 in the context of annealing. (\( T \) will be the temperature. \( \sigma_T \) and \( T \) are related by \( T = 2\sigma^2_T \).) Here we just understand \( \sigma^2_T \) as the prescribed variance in the Gaussian mixture model as opposed to the actual measured variance \( \sigma^2 \) from data fitting.
The clustering process is the estimation of the parameters $r$ that leads to the maximum log-likelihood of the observed sample

$$\log p(x|r, \sigma_T) = \sum_{i=1}^{N_1} \log \sum_{k=1}^{N} \exp(-\frac{1}{2\sigma_T^2}||x_i - r_k||^2). \quad (3.4)$$

The solution can be found by applying the EM algorithm. As pointed out by Hathaway [34], in the mixture model context, the EM algorithm maximizing (3.4) can be viewed as an alternative maximization of the following objective

$$F(M, r) = -\frac{1}{2\sigma_T^2} \sum_{i=1}^{N_1} \sum_{k=1}^{N} M_{ik}^x ||x_i - r_k||^2$$

$$- \sum_{i=1}^{N_1} \sum_{k=1}^{N} M_{ik}^x \log M_{ik}^x. \quad (3.5)$$

This is equivalent to minimizing

$$E(M, r) = -F(M, r) \quad (3.6)$$

$$= \frac{1}{2\sigma_T^2} \sum_{i=1}^{N_1} \sum_{k=1}^{N} M_{ik}^x ||x_i - r_k||^2$$

$$+ \sum_{i=1}^{N_1} \sum_{k=1}^{N} M_{ik}^x \log M_{ik}^x$$

with simplex constraints on $M$.

The clustering of the other point set is identical. For the joint clustering and diffeomorphism estimation, we put together the clustering energy of the two point sets and the diffeomorphic deformation energy induced in space giving us an
In the above objective function, the cluster membership matrices satisfy

\[ M^x_{ik} \in [0, 1], \forall i k \quad M^y_{jk} \in [0, 1], \forall j k \quad \text{and} \quad \sum_{k=1}^{N} M^1_{ik} = 1, \sum_{k=1}^{N} M^2_{jk} = 1. \]

The matrix entry \( M^x_{ik} \) is the membership of data point \( x_i \) in cluster \( k \) whose center is at location \( r_k \). The matrix entry \( M^y_{jk} \) is the membership of data point \( y_j \) in cluster \( k \) whose center is at position \( s_k \).

The diffeomorphic deformation energy in \( \Omega \) is induced by the landmark displacements from \( r \) to \( s \), where \( x \in \Omega \) and \( \phi(x, t) \) is a one parameter diffeomorphism: \( \Omega \to \Omega \). Since the original point sets differ in point count and are unlabeled, we cannot immediately use the diffeomorphism objective functions as in Joshi and Miller [39] or Camion and Younes [7] respectively. Instead, the two point sets are clustered and the landmark diffeomorphism objective is used between two sets of cluster centers \( r \) and \( s \) whose indices are always in correspondence. The diffeomorphism \( \phi(x, t) \) is generated by the flow \( v(x, t) \). \( \phi(x, t) \) and \( v(x, t) \) together satisfy the transport equation

\[ \frac{\partial \phi(x,t)}{\partial t} = v(\phi(x,t), t) \]

and the initial condition \( \forall x, \phi(x,0) = x \) holds. This is in the inexact matching form and the displacement term \( \sum_{k=1}^{N} \| s_k - \phi(r_k, 1) \|^2 \) plays an important role here as the bridge between the two systems. This is also the reason why we prefer the deformation energy in this form because the coupling of the two sets of clusters appear naturally through the inexact matching term and we don’t have to introduce external coupling terms as
in Guo et al. [33]. Another advantage of this approach is that in this dynamic system described by the diffeomorphic group $\phi(x,t)$, the landmarks trace a trajectory exactly on the flow lines dictated by the field $v(x,t)$. Also, the feedback coupling is no longer needed as in the previous approach because with this deformation energy described above, if $\phi(x,t)$ is the minimizer of this energy, then $\phi^{-1}(x,t)$ is the backward mapping which also minimizes the same energy.

The $\sigma_T$ in (3.7) is a fixed parameter. It is not a variable during the minimization. It is an attribute of the point set and can be a priori estimated for the point shapes. The reason that we have coefficient $\frac{1}{2\sigma_T^2}$ in front of the term $\sum_{k=1}^N ||s_k - \phi(r_k, 1)||^2$, instead of another free parameter is that $\sigma_T$ is a natural unit of measurement for distant discrepancies in clustering and it does not make sense to make this coefficient too big or too small. Since $\sigma_T$ is a constant, we multiply the objective function by a constant $2\sigma_T^2$ and we get the final form of the objective function as

$$E(M^x, M^y, r, s, v, \phi) \quad (3.8)$$

$$= \sum_{i=1}^{N_1} \sum_{k=1}^N M^x_{ik} ||x_i - r_k||^2 + 2\sigma_T^2 \sum_{i=1}^{N_1} \sum_{k=1}^N M^x_{ik} \log M^x_{ik}$$

$$+ \sum_{k=1}^N \sum_{j=1}^{N_2} M^y_{jk} ||y_j - s_k||^2 + 2\sigma_T^2 \sum_{j=1}^{N_2} \sum_{k=1}^N M^y_{jk} \log M^y_{jk}$$

$$+ \sum_{k=1}^N ||s_k - \phi(r_k, 1)||^2 + 2\sigma_T^2 \lambda \int_0^1 \int \Omega ||Lv(x,t)||^2 dx dt.$$

### 3.4 A Diffeomorphic Point Matching Algorithm

Our joint clustering and diffeomorphism estimation algorithm has two components: i) clustering and ii) diffeomorphism estimation.

For the clustering part, we use the deterministic annealing approach. The clustering problem is a non-convex optimization problem. The traditional clustering
techniques use descent based algorithms and they tend to get trapped in a local minimum. Rose et al. [59] proposed an annealing approach using analogies to statistical physics. The clustering cost function is seen as the free energy of a Gibbs canonical distribution. The minimization of clustering cost function is seen as the simulation of a physical annealing process in which free energy is minimized.

Let $T$ be the temperature of the system and in the clustering system $T = 2\sigma_T^2$.

Let $\beta = 1/T$ be the reciprocal temperature. Initially let $\beta = 0$. We have a single minimum for the energy in (3.6), which is the global minimum and all the cluster centers are located at the same point, which is the center of mass of all the data points and each data point is uniformly associated with all clusters. In the numerical implementation, we initialize all the cluster centers on a sphere of a very small radius and there are no data point within the sphere. When the temperature is lowered gradually, at a certain critical value of temperature, the clusters will split into smaller clusters and a phase transition occurs. At lower $T$, the free energy may have many local minima but the annealing process is able to track the global minimum.

For the diffeomorphism estimation, we expand the flow field in term of the kernel $K$ of the $L$ operator

$$v(x, t) = \sum_{k=1}^{N} \alpha_k(t)K(x, \phi_k(t))$$ (3.9)

where $\phi_k(t)$ is notational shorthand for $\phi(r_k, t)$ and we also take into consideration the affine part of the mapping (not written out in the above equation) when we use the thin-plate kernel with matrix entry $K_{ij} = r_{ij}^2 \log r_{ij}$ for 2D and $K_{ij} = -r_{ij}$ for 3D, with $r_{ij} = \| x_i - x_j \|$. After discretizing in time $t$, the objective in (3.7) is expressed as
\[ E(M^x, M^y, r, s, \alpha(t), \phi(t)) \] (3.10)
\[
= \sum_{i=1}^{N_1} \sum_{k=1}^{N} M^x_{ik} \|x_i - r_k\|^2 + T \sum_{i=1}^{N_1} \sum_{k=1}^{N} M^x_{ik} \log M^x_{ik} \\
+ \sum_{j=1}^{N_2} \sum_{k=1}^{N} M^y_{jk} \|y_j - s_k\|^2 + T \sum_{j=1}^{N_2} \sum_{k=1}^{N} M^y_{jk} \log M^y_{jk} \\
+ \sum_{k=1}^{N} \|s_k - r_k - \sum_{l=1}^{N} \sum_{t=0}^{S} [P(t)d_l(t) + \alpha_l(t)K(\phi_k(t), \phi_l(t))]\|^2 \\
+ \lambda T \sum_{k=1}^{N} \sum_{l=1}^{N} \sum_{t=0}^{S} <\alpha_k(t), \alpha_l(t)> K(\phi_k(t), \phi_l(t))
\]

where

\[
P(t) = \begin{bmatrix}
1 & \phi_1^1(t) & \phi_1^2(t) & \phi_1^3(t) \\
& \cdot & \cdot & \cdot \\
& \cdot & \cdot & \cdot \\
& \cdot & \cdot & \cdot \\
1 & \phi_N^1(t) & \phi_N^2(t) & \phi_N^3(t)
\end{bmatrix}
\] (3.11)

and \( d \) is the affine parameter matrix. After we perform a QR decomposition on \( P \),

\[
P(t) = (Q_1(t) : Q_2(t)) \begin{pmatrix} R(t) \\ 0 \end{pmatrix}.
\] (3.12)

We iteratively solve for \( \alpha_k(t) \) and \( \phi_k(t) \) using an alternating algorithm. When \( \phi_k(t) \)

is held fixed, we solve for \( \alpha_k(t) \). The solutions are

\[
d(t) = R^{-1}(t) [Q_1(t)\phi(t + 1) - Q_1(t)K(\phi(t))Q_2(t)\gamma(t)]
\] (3.13)

\[
\alpha(t) = Q_2(t)\gamma(t)
\] (3.14)
where $K(\phi(t))$ denotes the thin-plate kernel matrix evaluated at
\[ \phi(t) \overset{\text{def}}{=} \{ \phi(r_k, t) | k = 1, \ldots, N \} \]
and
\[ \gamma(t) = (Q_2^T(t)K(\phi(t))Q_2(t) + \lambda T)^{-1}Q_2^T(t)\phi(t + 1). \] (3.15)

When $\alpha_k(t)$ is held fixed, we use gradient descent to solve for $\phi_k(t)$:
\[ \frac{\partial E}{\partial \phi_k(t)} = 2 \sum_{l=1}^{N} < \alpha_k(t), \alpha_l(t) - 2W_l > - \nabla_1 K(\phi_k(t), \phi_l(t)) \] (3.16)
where $W_l = s_l - r_l - \sum_{m=1}^{N} \int_{0}^{1} \alpha_m(t) K(\phi_m(t), \phi_l(t)) dt$.

The clustering of the two point sets is handled by a deterministic annealing EM algorithm which iteratively estimates the cluster memberships $M^x$ and $M^y$ and the cluster centers $r$ and $s$. The update of the memberships is the very standard E-step of the EM algorithm \[17\] and is performed as shown below.

\[ M^x_{ik} = \frac{\exp(-\beta \|x_i - r_k\|^2)}{\sum_{l=1}^{N} \exp(-\beta \|x_i - r_l\|^2)}, \forall ik \] (3.17)
\[ M^y_{jk} = \frac{\exp(-\beta \|y_j - s_k\|^2)}{\sum_{l=1}^{N} \exp(-\beta \|y_j - s_l\|^2)}, \forall jk. \] (3.18)

The cluster center update is the M-step of the EM algorithm. This step is not the typical M-step. We use a closed-form solution for the cluster centers which is an approximation. From the clustering standpoint, we assume that the change in the diffeomorphism at each iteration is sufficiently small so that it can be neglected.

After making this approximation, we get
\[ r_k = \frac{\sum_{i=1}^{N_1} M^x_{ik}x_k + s_k - \sum_{i=1}^{N_1} \int_{0}^{1} \alpha_i(t) K(\phi_i(t), \phi_k(t)) dt}{1 + \sum_{i=1}^{N_1} M^x_{ik}}, \] (3.19)
\[ s_k = \frac{\sum_{j=1}^{N_2} M^y_{jk}y_j + \phi(r_k, 1)}{1 + \sum_{j=1}^{N_2} M^y_{jk}}, \forall k. \] (3.20)

The overall algorithm is described below.
Initialization: Initial temperature

\[ T = 0.5(\max_i \|x_i - x_c\|^2 + \max_j \|y_j - y_c\|^2) \]

where \(x_c\) and \(y_c\) are the centroids of \(X\) and \(Y\) respectively.

Begin A: While \(T > T_{\text{final}}\)

Step 1: Clustering

Update memberships according to (3.17), (3.18).

Update cluster centers according to (3.19), (3.20).

Step 2: Diffeomorphism

Update \((\phi, v)\) by minimizing

\[
E_{\text{diff}}(\phi, v) = \sum_{k=1}^{C} \|s_k - \phi(r_k, 1)\|^2 + \lambda T \int_0^1 \int_{\Omega} \|Lv(x, t)\|^2 dx dt
\]

according to (3.13)(3.14) and (3.16).

Step 3: Annealing. \(T \leftarrow \gamma T\) where \(\gamma < 1\).

End

3.5 Applications to 2D Corpus Callosum Shapes

We applied the algorithm to nine sets of 2D corpus callosum slices. The feature points were extracted with the help of a neuroanatomical expert. Figure 3–3 shows the nine corpus callosum 2D images, labeled CC1 through CC9. In our experiments, we first did the simultaneous clustering and matching with the corpus callosum point sets CC5 and CC9. The clustering of the two point sets is shown in Figure 3–4. There are 68 cluster centers. The circles represent the centers and the dots are the data points. The two cluster centers induce the diffeomorphic mapping of the 2D space. The warping of the 2D grid under this diffeomorphism is shown in Figure 3–5. Using this diffeomorphism, we calculated the after-image of original data points and compared them with the target data points. Due to the
large number of cluster centers, the cluster centers nearly coincide with the original data points and the warping of the original data points is not shown in the figure. The correspondences (at the cluster level) are shown in Figure 3–6. The algorithm allows us to simultaneously obtain the diffeomorphism and the correspondence.

Using our formulation, we are able to calculate the geodesic distances between the two sets of cluster centers. This is done on the shape manifold. Each point $q$ on the shape manifold $M$ represents a set of $N$ cluster centers $x_1, x_2, \ldots, x_N \in \mathbb{R}^2$ and has a coordinate $q = (x_1^1, x_1^2, x_2^1, x_2^2, \ldots, x_N^1, x_N^2)$ where $x_i = (x_i^1, x_i^2), i = 1, 2, \ldots, N$.

Let $q(t)$ be the geodesic path connecting two points $q_1$ and $q_2$ on the manifold. Using the norm defined for the tangent vector in Camion and Younes [7], the geodesic distance between $q_1$ and $q_2$ is

$$D_{\text{geodesic}}(q_1, q_2) = \int_0^1 \sqrt{\lambda q^T Q_2 (Q_2^T K(q) Q_2 + \lambda)^{-1} Q_2^T q} \, dt. \quad (3.21)$$

where $K(q)$ is the kernel of the $L$ operator evaluated at $q(t)$ and as mentioned previously, the thin-plate spline kernel is used. $Q_2$ comes from the QR decomposition of $P$

$$P = (Q_1 : Q_2) \begin{pmatrix} R \\ 0 \end{pmatrix} \quad (3.22)$$

and

$$P = \begin{pmatrix} 1 & x_1^1 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_N^1 & x_N^2 \end{pmatrix} \quad (3.23)$$

We also experimented with different number of cluster centers. Table 3–1 shows a modified Hausdorff distance as first introduced [20] between the image
set of points CC5 after diffeomorphism and the target set of points CC9 when the number of clusters vary. The reason for using the modified Hausdorff distance instead of the Hausdorff distance is that the latter is too sensitive to outliers. The definition of the modified Hausdorff distance is

\[
H_{\text{mod}}(A, B) = \max(h_{\text{mod}}(A, B), h_{\text{mod}}(B, A)),
\]

where \(A\) and \(B\) are finite point sets and

\[
h_{\text{mod}}(A, B) = \frac{1}{|A|} \sum_{a \in A} \min_{b \in B} \| a - b \|
\]

is the average of the minimum distances instead of the maximum of the minimum distances. It is easy to see that when the number of clusters increases, the matching improves as the modified Hausdorff distance decreases.

In the third column in Table 3-2, we list the geodesic distances between the two sets of cluster centers after pair-wise warping and clustering all the pair of corpus callosum point sets. Using the cluster centers as landmarks, a diffeomorphic mapping of the space is induced. With this induced diffeomorphism, we mapped the original data sets and compared the image of the original point set under the diffeomorphism and the target point set using the modified Hausdorff distance. The modified Hausdorff distances between the pairs are listed in the fourth column in Table 3-2. Finally, from the original nine corpus callosum point sets, we warped the first eight point sets onto the ninth set and Figure 3-7 displays the overlay of all point sets after diffeomorphic warping.

3.6 Applications to 3D Shapes

We applied our formulation and algorithm to the 3D point data of hippocampal shapes. We first applied the algorithm to synthetic data, where we have the
Figure 3-3: Point sets of nine corpus callosum images.

Figure 3-4: Clustering of the two point sets.

Figure 3-5: Diffeomorphic mapping of the space.
Figure 3–6: Matching between the two point sets.

Figure 3–7: Overlay of the after-images of eight point sets with the ninth set.

Table 3–1: Modified Hausdorff distance of the matching point sets.

<table>
<thead>
<tr>
<th>Number of Clusters</th>
<th>Modified Hausdorff Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.0082</td>
</tr>
<tr>
<td>20</td>
<td>0.0082</td>
</tr>
<tr>
<td>30</td>
<td>0.0057</td>
</tr>
<tr>
<td>40</td>
<td>0.0050</td>
</tr>
<tr>
<td>50</td>
<td>0.0043</td>
</tr>
<tr>
<td>60</td>
<td>0.0035</td>
</tr>
<tr>
<td>68</td>
<td>0.0027</td>
</tr>
</tbody>
</table>
Table 3–2: Geodesic distances between two sets of cluster centers and modified Hausdorff distances of matching points.

<table>
<thead>
<tr>
<th>From</th>
<th>To</th>
<th>Geodesic</th>
<th>M.H.</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC1</td>
<td>CC2</td>
<td>0.0264</td>
<td>0.0055</td>
</tr>
<tr>
<td>CC1</td>
<td>CC3</td>
<td>0.0132</td>
<td>0.0014</td>
</tr>
<tr>
<td>CC1</td>
<td>CC4</td>
<td>0.0289</td>
<td>0.0048</td>
</tr>
<tr>
<td>CC1</td>
<td>CC5</td>
<td>0.0269</td>
<td>0.0056</td>
</tr>
<tr>
<td>CC1</td>
<td>CC6</td>
<td>0.0250</td>
<td>0.0097</td>
</tr>
<tr>
<td>CC1</td>
<td>CC7</td>
<td>0.0323</td>
<td>0.0054</td>
</tr>
<tr>
<td>CC1</td>
<td>CC8</td>
<td>0.0256</td>
<td>0.0043</td>
</tr>
<tr>
<td>CC1</td>
<td>CC9</td>
<td>0.0241</td>
<td>0.0041</td>
</tr>
<tr>
<td>CC2</td>
<td>CC3</td>
<td>0.0277</td>
<td>0.0059</td>
</tr>
<tr>
<td>CC2</td>
<td>CC4</td>
<td>0.0342</td>
<td>0.0063</td>
</tr>
<tr>
<td>CC2</td>
<td>CC5</td>
<td>0.0308</td>
<td>0.0057</td>
</tr>
<tr>
<td>CC2</td>
<td>CC6</td>
<td>0.0211</td>
<td>0.0100</td>
</tr>
<tr>
<td>CC2</td>
<td>CC7</td>
<td>0.0215</td>
<td>0.0040</td>
</tr>
<tr>
<td>CC2</td>
<td>CC8</td>
<td>0.0271</td>
<td>0.0044</td>
</tr>
<tr>
<td>CC2</td>
<td>CC9</td>
<td>0.0258</td>
<td>0.0093</td>
</tr>
<tr>
<td>CC3</td>
<td>CC4</td>
<td>0.0443</td>
<td>0.0059</td>
</tr>
<tr>
<td>CC3</td>
<td>CC5</td>
<td>0.0294</td>
<td>0.0047</td>
</tr>
<tr>
<td>CC3</td>
<td>CC6</td>
<td>0.0181</td>
<td>0.0032</td>
</tr>
<tr>
<td>CC3</td>
<td>CC7</td>
<td>0.0256</td>
<td>0.0060</td>
</tr>
<tr>
<td>CC3</td>
<td>CC8</td>
<td>0.0153</td>
<td>0.0018</td>
</tr>
<tr>
<td>CC3</td>
<td>CC9</td>
<td>0.0305</td>
<td>0.0044</td>
</tr>
<tr>
<td>CC4</td>
<td>CC5</td>
<td>0.0231</td>
<td>0.0046</td>
</tr>
<tr>
<td>CC4</td>
<td>CC6</td>
<td>0.0304</td>
<td>0.0056</td>
</tr>
<tr>
<td>CC4</td>
<td>CC7</td>
<td>0.0324</td>
<td>0.0056</td>
</tr>
<tr>
<td>CC4</td>
<td>CC8</td>
<td>0.0311</td>
<td>0.0054</td>
</tr>
<tr>
<td>CC4</td>
<td>CC9</td>
<td>0.0434</td>
<td>0.0090</td>
</tr>
<tr>
<td>CC5</td>
<td>CC6</td>
<td>0.0266</td>
<td>0.0056</td>
</tr>
<tr>
<td>CC5</td>
<td>CC7</td>
<td>0.0325</td>
<td>0.0053</td>
</tr>
<tr>
<td>CC5</td>
<td>CC8</td>
<td>0.0225</td>
<td>0.0037</td>
</tr>
<tr>
<td>CC5</td>
<td>CC9</td>
<td>0.0305</td>
<td>0.0069</td>
</tr>
<tr>
<td>CC6</td>
<td>CC7</td>
<td>0.0244</td>
<td>0.0050</td>
</tr>
<tr>
<td>CC6</td>
<td>CC8</td>
<td>0.0186</td>
<td>0.0026</td>
</tr>
<tr>
<td>CC6</td>
<td>CC9</td>
<td>0.0274</td>
<td>0.0056</td>
</tr>
<tr>
<td>CC7</td>
<td>CC8</td>
<td>0.0212</td>
<td>0.0044</td>
</tr>
<tr>
<td>CC7</td>
<td>CC9</td>
<td>0.0241</td>
<td>0.0102</td>
</tr>
<tr>
<td>CC8</td>
<td>CC9</td>
<td>0.0196</td>
<td>0.0050</td>
</tr>
</tbody>
</table>
Figure 3-8: Two point sets of hippocampal shapes. The set with crosses is the original set and the set with dots is the one after GRBF warping.

knowledge of ground truth and this serves as the validation of the algorithm. We then experimented with real data and evaluated the results using various measures.

3.6.1 Experiments on Synthetic Data

We selected one hippocampal point set and warped it with a known diffeomorphism using the Gaussian Radial Basis Function (GRBF) kernel. We choose $\sigma = 60$ for the GRBF because with this large value of $\sigma$, we are able to generate a more global warping.

Figure 3-8 shows the two point sets of hippocampal shapes. The set with crosses is the original set and the set with dots is the one after GRBF warping.

First, we have no noise added. We used the TPS kernel to recover the diffeomorphism via joint clustering using our algorithm. The reason we use different kernels for warping and recovering is the objectiveness. It is trivial to recover the
deformation that is warped with the same kernel. Since the reference data are synthesized, we know the ground truth and we are able to compare our result with the ground truth. After unwarping the point set with our recovered diffeomorphism, we find the squared distances between the corresponding data points, and find the average and then take the square root. This is the standard error for our recovered diffeomorphism. We have two free parameters, $\lambda$ and $T_{final}$. $T_{final}$ is determined by the limiting value of $\sigma_T$ which is in turn determined by the number of clusters. We choose a $\lambda$ value such that the whole optimization process is stable in the temperature range from initial $T$ to $T_{final}$. We experimented with different numbers of clusters and listed the corresponding standard errors in the first row of Table 3–3. It is easy to see that the standard error goes down as the number of clusters goes up from 100 to 300 and goes up again when the number of clusters increases further. This is because when we have too few clusters, the points are not well represented by the cluster centers. On the other hand, if we have too many clusters, the variance between the two shapes is too big and the deformation increases dramatically. For the standard error, there is an optimal number of clusters and in this case we find it to be 300. We need to estimate the macroscopic and microscopic dimensions of the shape in order to see how big the standard error is. We calculated the covariance matrix of the original data set. We find their
Table 3–4: Limiting value of $\sigma$ determined by the number of clusters

<table>
<thead>
<tr>
<th>Number of clusters</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit $\sigma$</td>
<td>3.3</td>
<td>1.9</td>
<td>1.2</td>
<td>0.9</td>
<td>0.6</td>
</tr>
</tbody>
</table>

eigenvalues to be [48.1, 11.8, 4.1]. This gives us an estimate of the macroscopic dimensions to be about 100, 24 and 8, namely twice the eigenvalues. We then find out the average distance between the nearest neighbors to be 2.65. This gives us the microscopic dimension of the shape. As we can see from the table, our matching is very accurate.

Next we add noise to the warped data and test the robustness of our algorithm to noise. After GRBF warping, we add Gaussian noise to the warped data with different variances $\sigma$. We experimented with ten trials for each noise level from 0.1 to 1.0 and for each cluster level from 100 to 500. The standard errors and the deviation are shown in Table 3–3. We can see the standard error increase with the increasing noise level but it approximately stays in the range of the noise. Stronger noise does not increase the matching error dramatically and this shows the algorithm is robust against noise. This is easier to see when plotted in Figure 3–9 with error bars. We split the five levels of clusters into two plots because it looks messy if they were put together in a single plot. Figure 3–9(a) has the errors for 100, 200 and 300 clusters and Figure 3–9(b) has the errors for 300, 400 and 500 clusters. We can see that at the 300 cluster level, we obtain the best matching.

3.6.2 Experiments on Real Data

We applied the algorithm on different real hippocampal data sets. Figure 3–10 shows two hippocampal shapes. Figure 3–11 shows the annealing process. The $x$ axis is the iteration step. The dashed line is the scaled temperature $\sqrt{T/2}$ or $\sigma_T$. The solid line is the actual variance $\sigma$. We can see when the temperature goes down, it drives the $\sigma$ down. We observe a phase transition at temperature
$T = 3.37 \times 10^3$. We observe that there is a lower limit for $\sigma$. When the temperature gets very low, the $\sigma$ becomes a constant, which is 1.2, and no matter how much lower the temperature gets, the $\sigma$ stays constant. This constant is determined by the number of clusters. In Figure 3–12a through Figure 3–12e, we show how this limit changes with the number of clusters. When the number of clusters equals or exceeds the number of data points, the limit approaches zero. Table 3–4 displays the limits of $\sigma$ changing with the number of clusters. Because of noise and sampling error, we should not allow this limit to go to zero. Again we observe when we have 300 clusters, we have a reasonable $\sigma = 1.2$ as we recall the average distance between the nearest neighbors is about 2.65.

Figure 3–13 shows the clustering of the two shapes. We then did the matching for all the pairs out of ten hippocampal shapes. Table 3–5 and Table 3–6 shows three measures for the matching results with different clusters: Jensen-Shannon divergence, Hausdorff distance and modified Hausdorff distance.

The Jensen-Shannon divergence (a special case with $\lambda = 1/2$) is defined as [25]

$$D = \int_{\Omega} (p(x) \log \frac{2p(x)}{p(x) + q(x)} + q(x) \log \frac{2q(x)}{p(x) + q(x)}) dx,$$

where $x$ is the random variable while $p(x)$ and $q(x)$ are the two probability densities. Notice this measure is highly non-linear. When $p(x)$ and $q(x)$ are completely independent, namely in our matching case, when the two shapes are completely different, $D$ has a maximum of $2 \log 2 = 1.39$. In practice, we use the following technique to compute $D$. We observe in (3.26), the integral can be expressed as the expectation values of some functions under two different probability distributions:

$$D = < \log \frac{2p(x)}{p(x) + q(x)} >_p + < \log \frac{2q(x)}{p(x) + q(x)} >_q,$$

(3.27)
where \( < \log \frac{2p(x)}{p(x)+q(x)} >_p \) is the expectation value of function \( \log \frac{2p(x)}{p(x)+q(x)} \) under probability distribution \( p(x) \) and \( < \log \frac{2q(x)}{p(x)+q(x)} >_q \) is the expectation value of function \( \log \frac{2q(x)}{p(x)+q(x)} \) under probability distribution \( q(x) \). In our Gaussian mixture model, we see the data points as samples from a Gaussian mixture probability distribution with known cluster centers. Here

\[
p(x) = \frac{1}{N} \frac{1}{(2\pi\sigma^2)^{3/2}} \sum_{k=1}^{N} \exp\left(-\frac{||x - r_k||^2}{2\sigma^2}\right), \quad (3.28)
\]

and

\[
q(x) = \frac{1}{N} \frac{1}{(2\pi\sigma^2)^{3/2}} \sum_{k=1}^{N} \exp\left(-\frac{||x - s_k||^2}{2\sigma^2}\right), \quad (3.29)
\]

where \( x \) is the random variable, namely the space location and \( \{r_k\} \) is the first set of cluster centers and \( \{s_k\} \) is the second set of cluster centers. We use the average of finite samples as an approximation of the expectation values and we have

\[
D = \frac{1}{N_1} \sum_{i=1}^{N_1} \log \frac{2p(x_i)}{p(x_i) + q(x_i)} + \frac{1}{N_2} \sum_{j=1}^{N_2} \log \frac{2q(y_j)}{p(y_j) + (y_j)}, \quad (3.30)
\]

where \( \{x_i\} \) is the first set of data points; \( N_1 \) is the number of points in the first set; \( \{y_j\} \) is the second set of data points; and \( N_2 \) is the number of points in the second set. We have seen that Jensen-Shannon divergence is very useful in estimating the validity of the registration of two point shapes without knowing the ground truth.

The Hausdorff distance is defined as

\[
H(A, B) = \max(h(A, B), h(B, A)), \quad (3.31)
\]

where \( A \) and \( B \) are finite point sets and

\[
h(A, B) = \max_{a \in A} \min_{b \in B} \| a - b \|. \quad (3.32)
\]

The Hausdorff distance measures the worst case difference between the two point sets. From Table 3-5 and Table 3-6 we can see that when we have 300
clusters, we have the minimum Jensen-Shannon divergence and the Hausdorff distance. However, the Hausdorff distance is too sensitive to outliers. We also calculated the modified Hausdorff distance as first introduced in Dubuisson and Jain [20]. The definition of the modified Hausdorff distance was given before in (3.24) and (3.25). It is the average of the minimum distances instead of the maximum of the minimum distances. It is easy to see that when the number of clusters increases, the modified Hausdorff distance decreases.
Figure 3–9: Matching errors on synthetic data for different number of clusters
Figure 3–10: Two hippocampal shapes
Figure 3–11: Deterministic annealing in the clustering process: the dashed line is the scaled temperature $\sqrt{T}/2$ or $\sigma_T$. The solid line is the actual variance $\sigma$. When the temperature goes down, it drives the $\sigma$ down. There is a phase transition at temperature $T = 3.37 \times 10^3$ and there exists a lower limit 1.2 for $\sigma$. 
Figure 3-12: Limiting value of \( \sigma \) determined by the number of clusters
(a) Clustering of the template hippocampal shape

(b) Clustering of the reference hippocampal shape

Figure 3–13: Clustering of the two hippocampal shapes
Table 3-5: Jensen-Shannon divergence for various pairs of shapes

<table>
<thead>
<tr>
<th>Trial no.\No. clusters</th>
<th>Jensen-Shannon div.</th>
<th>100</th>
<th>200</th>
<th>300</th>
<th>400</th>
<th>500</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.87</td>
<td>0.31</td>
<td>0.03</td>
<td>0.13</td>
<td>0.21</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.93</td>
<td>0.62</td>
<td>0.47</td>
<td>0.05</td>
<td>0.24</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.76</td>
<td>0.27</td>
<td>0.04</td>
<td>0.16</td>
<td>0.32</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.98</td>
<td>0.52</td>
<td>0.34</td>
<td>0.09</td>
<td>0.45</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.69</td>
<td>0.41</td>
<td>0.14</td>
<td>0.18</td>
<td>0.36</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.57</td>
<td>0.23</td>
<td>0.43</td>
<td>0.78</td>
<td>0.97</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.66</td>
<td>0.21</td>
<td>0.05</td>
<td>0.14</td>
<td>0.30</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.99</td>
<td>0.70</td>
<td>0.25</td>
<td>0.19</td>
<td>0.63</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.85</td>
<td>0.42</td>
<td>0.11</td>
<td>0.68</td>
<td>0.74</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.97</td>
<td>0.62</td>
<td>0.10</td>
<td>0.18</td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.70</td>
<td>0.33</td>
<td>0.06</td>
<td>0.13</td>
<td>0.26</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1.02</td>
<td>0.64</td>
<td>0.08</td>
<td>0.44</td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0.89</td>
<td>0.54</td>
<td>0.20</td>
<td>0.31</td>
<td>0.65</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>0.57</td>
<td>0.09</td>
<td>0.15</td>
<td>0.66</td>
<td>0.80</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.88</td>
<td>0.30</td>
<td>0.05</td>
<td>0.29</td>
<td>0.36</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>0.90</td>
<td>0.75</td>
<td>0.12</td>
<td>0.17</td>
<td>0.44</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0.61</td>
<td>0.16</td>
<td>0.28</td>
<td>0.53</td>
<td>0.72</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0.91</td>
<td>0.37</td>
<td>0.18</td>
<td>0.40</td>
<td>0.88</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>1.12</td>
<td>0.80</td>
<td>0.47</td>
<td>0.09</td>
<td>0.28</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.96</td>
<td>0.54</td>
<td>0.33</td>
<td>0.60</td>
<td>0.74</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>0.65</td>
<td>0.23</td>
<td>0.51</td>
<td>0.78</td>
<td>1.04</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>0.93</td>
<td>0.46</td>
<td>0.22</td>
<td>0.51</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>0.92</td>
<td>0.60</td>
<td>0.28</td>
<td>0.15</td>
<td>0.34</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>0.80</td>
<td>0.26</td>
<td>0.57</td>
<td>0.69</td>
<td>0.86</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>1.10</td>
<td>0.62</td>
<td>0.44</td>
<td>0.78</td>
<td>0.97</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>0.90</td>
<td>0.39</td>
<td>0.05</td>
<td>0.21</td>
<td>0.47</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>0.58</td>
<td>0.07</td>
<td>0.20</td>
<td>0.56</td>
<td>0.77</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>0.93</td>
<td>0.51</td>
<td>0.09</td>
<td>0.40</td>
<td>0.63</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>0.99</td>
<td>0.26</td>
<td>0.18</td>
<td>0.37</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0.60</td>
<td>0.06</td>
<td>0.17</td>
<td>0.54</td>
<td>0.57</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>0.83</td>
<td>0.19</td>
<td>0.08</td>
<td>0.37</td>
<td>0.76</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>1.22</td>
<td>0.42</td>
<td>0.57</td>
<td>0.70</td>
<td>0.95</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>0.80</td>
<td>0.59</td>
<td>0.30</td>
<td>0.86</td>
<td>0.92</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>0.89</td>
<td>0.76</td>
<td>0.35</td>
<td>0.28</td>
<td>0.67</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>1.05</td>
<td>0.42</td>
<td>0.37</td>
<td>0.81</td>
<td>1.13</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>0.92</td>
<td>0.25</td>
<td>0.31</td>
<td>0.60</td>
<td>0.85</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>0.79</td>
<td>0.35</td>
<td>0.08</td>
<td>0.24</td>
<td>0.40</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>0.90</td>
<td>0.42</td>
<td>0.16</td>
<td>0.35</td>
<td>0.68</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>0.86</td>
<td>0.27</td>
<td>0.38</td>
<td>0.50</td>
<td>0.71</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0.55</td>
<td>0.04</td>
<td>0.19</td>
<td>0.36</td>
<td>0.67</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>1.02</td>
<td>0.30</td>
<td>0.47</td>
<td>0.81</td>
<td>0.98</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>0.43</td>
<td>0.07</td>
<td>0.22</td>
<td>0.56</td>
<td>0.86</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>0.78</td>
<td>0.56</td>
<td>0.18</td>
<td>0.39</td>
<td>0.61</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>0.61</td>
<td>0.09</td>
<td>0.25</td>
<td>0.70</td>
<td>0.82</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>0.44</td>
<td>0.15</td>
<td>0.26</td>
<td>0.53</td>
<td>0.93</td>
<td></td>
</tr>
</tbody>
</table>
Table 3–6: Hausdorff and modified Hausdorff distance for various pairs of shapes

<table>
<thead>
<tr>
<th>Trial no.\No. clusters</th>
<th>Hausdorff distance</th>
<th>modified Hausdorff</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100</td>
<td>200</td>
</tr>
<tr>
<td>1</td>
<td>7.1 7.4 5.7 6.2 7.3</td>
<td>2.8 2.0 1.4 1.2 1.1</td>
</tr>
<tr>
<td>2</td>
<td>9.3 8.9 7.2 8.3 8.7</td>
<td>3.5 3.1 2.8 2.4 2.3</td>
</tr>
<tr>
<td>3</td>
<td>7.2 6.1 4.9 5.6 6.4</td>
<td>2.0 1.7 1.4 1.3 1.2</td>
</tr>
<tr>
<td>4</td>
<td>8.4 7.8 7.2 5.2 6.5</td>
<td>2.7 2.4 2.3 1.7 1.4</td>
</tr>
<tr>
<td>5</td>
<td>9.6 9.7 8.0 8.4 8.9</td>
<td>3.9 3.6 3.1 2.8 2.7</td>
</tr>
<tr>
<td>6</td>
<td>9.2 6.3 7.1 7.8 8.6</td>
<td>3.1 2.8 2.5 2.2 2.1</td>
</tr>
<tr>
<td>7</td>
<td>6.9 5.8 4.4 6.0 7.3</td>
<td>2.4 2.2 2.1 1.7 1.5</td>
</tr>
<tr>
<td>8</td>
<td>8.9 8.5 7.0 6.4 8.2</td>
<td>3.0 2.6 2.4 2.2 1.9</td>
</tr>
<tr>
<td>9</td>
<td>9.3 8.0 5.9 7.6 9.1</td>
<td>2.9 2.7 2.3 2.1 1.6</td>
</tr>
<tr>
<td>10</td>
<td>7.8 7.3 4.7 6.7 8.1</td>
<td>3.2 2.8 2.3 1.8 1.4</td>
</tr>
<tr>
<td>11</td>
<td>8.7 7.7 5.8 7.4 9.0</td>
<td>2.5 2.1 1.6 1.4 1.2</td>
</tr>
<tr>
<td>12</td>
<td>9.1 8.3 6.1 7.4 8.6</td>
<td>3.3 3.0 2.5 2.2 2.0</td>
</tr>
<tr>
<td>13</td>
<td>9.3 8.4 6.5 7.0 8.7</td>
<td>3.6 3.4 3.1 2.4 2.2</td>
</tr>
<tr>
<td>14</td>
<td>7.4 5.1 5.5 6.8 8.3</td>
<td>3.0 2.7 2.3 2.0 1.8</td>
</tr>
<tr>
<td>15</td>
<td>8.8 7.1 4.9 6.3 7.8</td>
<td>2.6 2.0 1.5 1.3 1.2</td>
</tr>
<tr>
<td>16</td>
<td>9.4 9.0 6.1 7.3 8.5</td>
<td>3.2 3.0 2.4 2.1 1.9</td>
</tr>
<tr>
<td>17</td>
<td>8.6 6.8 7.9 9.0 9.9</td>
<td>3.4 3.5 3.1 2.7 2.4</td>
</tr>
<tr>
<td>18</td>
<td>9.5 8.2 6.5 7.4 8.0</td>
<td>3.7 3.2 2.9 2.6 2.1</td>
</tr>
<tr>
<td>19</td>
<td>9.2 7.8 7.2 5.1 6.4</td>
<td>2.8 2.6 2.3 2.2 2.0</td>
</tr>
<tr>
<td>20</td>
<td>9.6 8.0 7.3 8.7 9.3</td>
<td>3.9 3.5 3.3 3.1 2.7</td>
</tr>
<tr>
<td>21</td>
<td>8.4 6.1 6.9 7.8 9.5</td>
<td>3.3 3.1 2.8 2.4 2.3</td>
</tr>
<tr>
<td>22</td>
<td>9.7 8.5 7.0 8.1 9.0</td>
<td>2.9 2.7 2.6 2.3 2.1</td>
</tr>
<tr>
<td>23</td>
<td>9.6 8.2 7.3 6.5 7.7</td>
<td>2.4 2.1 1.7 1.6 1.5</td>
</tr>
<tr>
<td>24</td>
<td>7.8 6.6 7.2 8.9 9.6</td>
<td>3.1 2.7 2.5 2.2 1.9</td>
</tr>
<tr>
<td>25</td>
<td>9.8 7.9 7.6 8.8 9.2</td>
<td>3.8 3.4 3.0 2.8 2.7</td>
</tr>
<tr>
<td>26</td>
<td>9.0 7.3 5.8 7.0 8.7</td>
<td>2.9 2.4 2.0 1.4 1.2</td>
</tr>
<tr>
<td>27</td>
<td>7.8 6.0 6.5 7.2 8.3</td>
<td>3.2 2.9 2.5 2.1 1.9</td>
</tr>
<tr>
<td>28</td>
<td>9.5 8.1 6.1 7.4 8.8</td>
<td>3.0 2.8 2.5 2.1 1.8</td>
</tr>
<tr>
<td>29</td>
<td>9.7 8.3 6.7 7.0 8.5</td>
<td>3.4 2.7 2.2 1.9 1.7</td>
</tr>
<tr>
<td>30</td>
<td>7.1 5.6 6.2 7.3 7.8</td>
<td>2.5 2.2 1.8 1.3 1.2</td>
</tr>
<tr>
<td>31</td>
<td>9.3 6.7 5.8 7.5 8.4</td>
<td>2.7 2.4 2.2 1.6 1.4</td>
</tr>
<tr>
<td>32</td>
<td>9.9 7.2 7.8 8.5 9.2</td>
<td>3.5 3.0 2.8 2.5 2.3</td>
</tr>
<tr>
<td>33</td>
<td>8.9 8.0 6.3 7.1 8.7</td>
<td>3.3 3.1 2.6 2.4 2.3</td>
</tr>
<tr>
<td>34</td>
<td>8.9 8.5 6.7 6.4 7.0</td>
<td>3.0 2.8 2.5 2.2 2.1</td>
</tr>
<tr>
<td>35</td>
<td>9.6 8.4 7.0 8.1 9.9</td>
<td>3.7 3.4 3.0 2.3 2.0</td>
</tr>
<tr>
<td>36</td>
<td>8.7 5.9 6.2 7.5 8.4</td>
<td>3.2 2.9 2.5 2.3 2.2</td>
</tr>
<tr>
<td>37</td>
<td>7.7 6.3 5.4 6.1 7.6</td>
<td>2.4 2.2 1.9 1.6 1.5</td>
</tr>
<tr>
<td>38</td>
<td>9.5 7.1 5.7 6.6 7.9</td>
<td>2.8 2.4 2.3 2.0 1.8</td>
</tr>
<tr>
<td>39</td>
<td>9.2 7.3 8.2 8.4 9.0</td>
<td>3.2 3.0 2.7 2.5 2.4</td>
</tr>
<tr>
<td>40</td>
<td>6.5 5.2 5.8 6.7 7.3</td>
<td>2.0 1.6 1.3 1.1 0.9</td>
</tr>
<tr>
<td>41</td>
<td>9.4 7.5 7.9 8.6 9.2</td>
<td>3.8 3.2 2.5 2.4 2.3</td>
</tr>
<tr>
<td>42</td>
<td>7.2 5.8 6.7 7.3 7.9</td>
<td>2.3 1.9 1.6 1.5 1.3</td>
</tr>
<tr>
<td>43</td>
<td>8.2 7.4 6.0 6.7 7.8</td>
<td>2.9 2.4 2.2 2.0 1.8</td>
</tr>
<tr>
<td>44</td>
<td>7.1 5.9 6.6 7.5 8.0</td>
<td>2.2 1.9 1.7 1.5 1.4</td>
</tr>
<tr>
<td>45</td>
<td>7.0 6.5 7.2 7.7 9.1</td>
<td>2.5 2.1 1.9 1.7 1.6</td>
</tr>
</tbody>
</table>
CHAPTER 4
TOPOLOGICAL CLUSTERING AND MATCHING

In this chapter we extend our diffeomorphic point matching theory and algorithm to include the known information of the topology of the underlying shapes. In Section 4.1 we provide a brief introduction to the basic concepts of topological spaces. In Section 4.2 we review the Kohonen Self-Organizing Feature Map (SOFM) which was first introduced by Kohonen in the context of neural networks in 1980s. Our topological clustering and matching is related to SOFM because the essence of SOFM is topology preserving. However, our topological clustering and matching is different from SOFM in many ways and as part of Section 4.3 we discuss these differences. In Section 4.3 we discuss the motivation and methods of topological clustering and matching. We do this with graph topology assigned to the set of cluster centers. The graph topology can be prescribed if we have prior knowledge of the shape topology or it can be arbitrary in the case we don’t know the shape topology in advance. In Section 4.5 we present results for clustering and matching with prescribed topology with example of chain topology, ring topology and genus zero closed surface topology, or $S^2$ topology. Section 4.6 describes how we can approximate the topology if we do not have the prior knowledge about the topology of the shape in advance.

4.1 Fundamentals of Topological Spaces

The heart of topology is the concept of “nearness,” described by “neighborhood”. We first introduce the concept of topological space. Readers can confer the Encyclopedic Dictionary of Mathematics compiled by Mathematical Society of Japan and translated by Massachusetts Institute of Technology [47]. We also cite from a book by Bourbaki [6] and a book by Rosenfeld [61].
Felix Hausdorff in his *Foundations of Set Theory* (Grundzüge der Mengenlehre. Leipzig, 1914) [35, 36] defined his concept of a topological space based on the four axioms.

Let $X$ be a set. A **neighborhood system** for $X$ is a function $\mathcal{U}$ that assigns to each point $x$ of $X$, a family $\mathcal{U}(x)$ of subsets of $X$ subject to the following axioms ($\mathcal{U}$):

1. ($\mathcal{U}1$) $x \in U$ for each $U$ in $\mathcal{U}(x)$.
2. ($\mathcal{U}2$) If $U_1, U_2 \in \mathcal{U}(x)$, then $U_1 \cap U_2 \in \mathcal{U}(x)$.
3. ($\mathcal{U}3$) If $U \in \mathcal{U}(x)$ and $U \subseteq V$, then $V \in \mathcal{U}(x)$.
4. ($\mathcal{U}4$) For each $U$ in $\mathcal{U}(x)$, there is a member $W$ of $\mathcal{U}(x)$ such that $U \in \mathcal{U}(y)$ for each $y$ in $W$.

$\mathcal{U}(x)$ is interpreted as the family of all neighborhoods of point $x$. An element $U \in \mathcal{U}(x)$ is called a neighborhood of point $x$. The intuitive translation of the above axioms is as follows.

1. ($\mathcal{U}1$) $x$ is in each neighborhood of $x$.
2. ($\mathcal{U}2$) The intersection of two neighborhoods of $x$ is a neighborhood of $x$.
3. ($\mathcal{U}3$) If a set $V$ contains a neighborhood of $x$, then $V$ is itself a neighborhood of $x$.
4. ($\mathcal{U}4$) For each neighborhood $U$ of $x$, there is another neighborhood $W$ of $x$, such that $U$ is the neighborhood of each point $y$ in $W$.

Pavel Sergeevič Aleksandrov [Alexandroff] proposed in the paper *On the foundation of n-dimensional topology* (Zur Begründung der $n$-dimensionalen Topologie. Leipzig, 1925) [1]:

A **system of open sets** for a set $X$ is a family $\mathcal{O}$ of subsets of $X$ satisfying the following axioms ($\mathcal{O}$):

1. ($\mathcal{O}1$) $X, \emptyset \in \mathcal{O}$.
2. ($\mathcal{O}2$) If $O_1, O_2 \in \mathcal{O}$, then $O_1 \cap O_2 \in \mathcal{O}$.
(O3) If $O_{\lambda} \in \mathfrak{O}$ ($\lambda \in \Lambda$), then $\bigcup_{\lambda \in \Lambda} O_{\lambda} \in \mathfrak{O}$. The elements in $\mathfrak{O}$ are called open sets. An easy intuitive interpretation of this set of axioms is

(O1) The empty set is an open set. The entire space $X$ is an open set.

(O2) The intersection of two open sets is an open set.

(O3) The union of arbitrarily many open sets is an open set.

Using the set complement and DeMorgan’s law we get a system of closed sets. A system of closed sets for a space $X$ is a family $\mathfrak{F}$ of subsets of $X$ satisfying the following axioms ($\mathfrak{F}$):

($\mathfrak{F}1$) $X, \emptyset \in \mathfrak{F}$.

($\mathfrak{F}2$) If $F_1, F_2 \in \mathfrak{F}$, then $F_1 \cup F_2 \in \mathfrak{F}$.

($\mathfrak{F}3$) If $F_{\lambda} \in \mathfrak{F}$ ($\lambda \in \Lambda$), then $\bigcap_{\lambda \in \Lambda} F_{\lambda} \in \mathfrak{F}$.

The elements in $\mathfrak{F}$ are called closed sets. An easy intuitive interpretation of this set of axioms is

($\mathfrak{F}1$) The entire space $X$ is a closed set. The empty set is a closed set.

($\mathfrak{F}2$) The union of two closed sets is a closed set.

($\mathfrak{F}3$) The intersection of arbitrarily many closed sets is a closed set.

Kazimierz Kuratowski in the paper The operation $\bar{A}$ of analysis situs (L’operation $\bar{A}$ de l’analysis situs. Warsaw, 1922) [44] proposed:

A closure operator for a space $X$ is a function that assigns to each subset $A$ of $X$, a subset $A^a$ of $X$ satisfying the following axioms($\mathfrak{C}$):

($\mathfrak{C}1$) $\emptyset^a = \emptyset$.

($\mathfrak{C}2$) $(A \cup B)^a = A^a \cup B^a$.

($\mathfrak{C}3$) $A \subset A^a$.

($\mathfrak{C}4$) $A^a = A^{aa}$.

The intuition of the closure of a set $A$ is the union of $A$ and its boundary. This set of axioms explained in words is as follows.
(C1) The closure of the empty set is the empty set.

(C2) The closure of \((A \cup B)\) is the union of closure of \(A\) and the closure of \(B\).

(C3) The closure of \(A\) contains \(A\) as a subset.

(C4) The closure of the closure of \(A\) is the same as the closure of \(A\). This is saying that the closure operator is idempotent.

Related to the closure operator is an interior operator.

An **interior operator** for a space \(X\) is a function that assigns to each subset \(A\) of \(X\) a subset \(A^i\) of \(X\) satisfying the following axioms (I):

(I1) \(X^i = X\).

(I2) \((A \cap B)^i = A^i \cap B^i\).

(I3) \(A^i \subset A\).

(I4) \(A^{ii} = A^i\).

The intuition of the interior of a set \(A\) is \(A\) minus its boundary. This set of axioms explained in words is as follows.

(I1) The interior of the entire space \(X\) is itself.

(I2) The interior of \((A \cap B)\) is the intersection of the interior of \(A\) and the interior of \(B\).

(I3) The interior of \(A\) is a subset of \(A\).

(I4) The interior of the interior of \(A\) is the same as the interior of \(A\). This is saying that the interior operator is idempotent.

All these systems equivalently define the topological space. We can interpret one easily in the language of another.

We can define open set using neighborhood:

*An set \(A \subset X\) is an open set, if \(\forall x \in A, \) there exists a neighborhood \(U\) of \(x\) such that \(U \subset A\), then \(A\) is an open set.*

Define closed set using open set:

*An set \(A \subset X\) is a closed set, if \(X - A\) is an open set.*
Define closure of $A$ using closed set:

If $A \subseteq X$, the closure of $A$ is the set $A^a \equiv \bigcap_{\lambda} B_{\lambda}$, where $B_{\lambda}$ is a closed set and $A \subseteq B$. In other words, the closure of $A$ is the smallest closed set that contains $A$.

Define interior using closure:

$A^i \equiv X - (X - A)^a$. In other word, for set $A \subseteq X$, first find the complement of $A$, $B = X - A$. Then find $B^a$, the closure of $B$. The the interior of $A$ is the complement of $B^a$.

Define neighborhood using interior:

A set $U \subseteq X$ is the neighborhood of point $x \in X$, if $x \in U^i$, namely $x$ is in the interior of $U$.

Figure 4–1 shows some examples of different topological spaces.

Another importance concept in topological spaces is the separation axioms, dictating the extent the points are separated from each other. Of our interest is one particular space, called Hausdorff space.

**Definition 4.** A topological space is called Hausdorff space if any two distinct points have disjoint neighborhoods.

### 4.2 Kohonen Self-Organizing Feature Map (SOFM)

Kohonen developed the Self-Organizing Feature Map algorithm. He described the SOFM in the context of neural networks with an aim to understand the the brain cortex mapping of sensory organs, like retina [41, 40, 42]. In the Kohonen network, there are two layers of neurons, with the first layer the input and the second layer the output. The output neurons are arranged in a rectangular grid. Each input neuron is connected with each output neuron. Each output neuron is associated with $d$-weights, with $d$ being the number of the input neurons.

Kohonen describes a procedure of the initialization and update of the weights. The important part of the algorithm is that each neuron on the output grid has a neighborhood and when each neuron updates itself, the neurons in the
Figure 4–1: Different topological spaces

neighborhood update themselves accordingly. This is described in many books as “topologically ordered map”, “topology preserving map” or “topographical map”. There are many misconceptions and misnomer here. First, the neural network is not such a map but it simulates such a map because the codomain is a discrete space, namely the rectangular grid. Second, “topographical map” is a misnomer for “topological map.” Third, what the network simulates is a continuous map. That is the exact interpretation of the idea that “if the two outputs are in a neighborhood then the two inputs are also in a neighborhood.” “Topology preserving” is not guaranteed. In fact this is obvious when the input domain is two dimensional and the output domain is one dimensional. What we get is a plain filling Peano curve. As it is well known that a plain region cannot be topologically equivalent to a curve segment and the Peano curve is a continuous map but not a topological map, or homeomorphism.

Kohonen only gives a procedure but does not give an objective function that this procedure optimize. People soon find an interpretation of SOFM in the context
of clustering and dimension reduction of data, with mathematical abstraction. Ritter et al. give an objective function [58]

$$E\{w\} = \frac{1}{2} \sum_{i,k,\mu} M_{\mu k} \Lambda(i, k) ||\xi_k - w_i||^2,$$

where $M_{\mu k}$ is the membership matrix element, equal to 1 if pattern $\xi_\mu$ is in cluster $k$ and 0 otherwise. $\Lambda(i, k)$ is the neighborhood function between cluster $i$ and cluster $k$. $\Lambda(i, k) = 1$ if $i = k$ and falls off with distance $||w_i - w_k||$. A typical choice for $\Lambda(i, k)$ is

$$\Lambda(i, k) = e^{-\frac{||w_i - w_k||^2}{2\sigma^2}}.$$

The Kohonen procedure is actually the clustering process of high dimensional data. It tries to model the dimensionality reduction. The data points are in $h$-dimensional Euclidean space but they may approximately lie on a two dimensional manifold. The clusters are constrained with a 2D rectangular grid. During the clustering it is required that neighboring cluster centers stay close to each other. Thus this procedure provides an approximate discrete 2D patch which is a map from a 2D rectangular region with grid to the 2D manifold embedded in $h$-dimensional Euclidean space, where the data dwell. This provides an intrinsic coordination (2D) for the data and it extracts 2D features from the data. Because of this clustering process, the coordinates are only approximate and the coordinates are discrete (the $i, j$ indices). In the general case, each data point has $h$ coordinate components and thus live in $\mathbb{R}^h$ but the set of data only populate a $l$-dimensional submanifold $M^l$ of $\mathbb{R}^h$. Suppose we have a $k$-dimensional grid of cluster centers, $U \subset \mathbb{R}^k$. Only when $k = l$ can we have a homeomorphism from $U$ to an open neighborhood of $M^l$. One example is that $l = 3$ and $k = 2$. If the embedded manifold is roughly a 2D sheet with some non-negligible thickness in the third
dimension, the clustering result in an approximation in which we represent this 3D curved thick sheet with a 2D thick-less sheet. If instead of a thick sheet, we have a solid 3D manifold and we still want to use a 2D sheet to approximate it, it will result in that the 2D sheet will wrinkle and scramble so that it tries to fill the 3D space. It is easier to visualize when \( l = 2 \) and \( k = 1 \) and then we have a space filling Peano curve, resulting a continuous map but not an homeomorphism.

4.3 Topological Clustering and Matching

4.3.1 Why: the Need for Topology

When the two shape differ by a large deformation, complications do occur. Figure 4-2 shows the contours of two hand shapes. The hand on the left has the thumb and the fore finger pointing out while the hand on the right has the fore finger folded. These two shapes differ by a large deformation. We apply our diffeomorphic point matching algorithm to these two shapes and Figure 4-3 shows the clustering of the two shapes. While the clustering looks pretty good, a closer examination of the correspondence as shown in Figure 4-4 indicates that there are incorrect correspondences. We have discussed in Chapter 1 the issues that there is no way to clearly define what the correct correspondence is, with our visual intuition, we know this is not the correspondence we want. This is the case where nearby points correspond to points that are not nearby, a violation of topological property of the mapping. We want to enforce the constraint that nearby points are mapped to nearby points and introduce the topology constraint to the matching. This has significance in two scenarios:

1. The problem definition of the shape matching does not have topology constraint and the objective function does not have the topology term. However, the numerical procedure of solving this problem may be caught up in a local minimum, which gives incorrect correspondence. In this case,
Figure 4-2: Image contours of two hands

introducing the topological constraint will help avoid the local minimum and
find the correct correspondence as defined by the objective function.

2. The numerical procedure does find the global minimum defined by the ob-
jective function without topological constraint. However, the correspondence
is still not what we intended. In this case, the point set as a set of point
without any topological structure is not sufficient to describe the underlying
shape. For example of the two hand shapes as in Figure 4–2, we know that
the points lie on a contour curve. By adding this topology requirement, we
are actually defining a different problem from that without topology structure
and of course the solutions of the correspondence should be different with and
without topology since they are two different problems.

4.3.2 How: Graph Topology

It is clear that the information of the topological structure of the underlying
shape of the point set helps define and solve for the correct correspondence. Then
it is natural to introduce some topological structure in the set of cluster centers and
a graph is the easiest way to represent this topology. We make the cluster centers
the vertices of the graph and assign edges between the vertices.

An intuitive thinking of the topology construction on the graph may be
that make the set of vertices the support set and make the adjacent vertices
Figure 4–3: Clustering of two hands

Figure 4–4: Correspondence
Let $(X, \mathcal{T})$ be a topological space and $X$ is a finite set. $\mathcal{T}$ is Hausdorff iff $\mathcal{T}$ is discrete topology.

**Proof.** First we prove if $\mathcal{T}$ is discrete topology then $\mathcal{T}$ is Hausdorff.

The proof is trivial because in discrete topology every singleton set of a point is an open set. For any two distinct points $A \in X$ and $B \in X$, the disjoint neighborhood $U_A$ and $U_B$ are $U_A = \{A\}$ and $U_B = \{B\}$.

Next we prove if $\mathcal{T}$ is Hausdorff then $\mathcal{T}$ is discrete topology.

We use proof by contradiction. Now we assume the contrary, namely $\mathcal{T}$ is Hausdorff but $\mathcal{T}$ is a topology other than discrete topology. Then there must exist a point $A \in X$, such that $\{A\}$ is not an open set. Let $U_A$ be the smallest open set that contains $A$:

$$U_A = U_1 \cap U_2 \cap ... \cap U_k,$$

where $U_1, U_2, ..., U_k$ are all the open sets that contains $A$. By assumption, $U_A$ must have at least another point different from $A$. We call this point $B$ as shown in

---

**Figure 4-5: Finite topology**
Figure 4-5. Now it is obvious that there is no open neighborhood of $A$ that does not contain $B$. Hence $\mathcal{T}$ is not Hausdorff. So the original claim is proved.

Since the Euclidean space is a Hausdorff space and the shapes we consider as sub-topological spaces of the Euclidean space are Hausdorff spaces, we are not interested in graph topologies that are not Hausdorff. However, from the above theorem we know that the only Hausdorff topology is discrete topology. The discrete topology is not good here because each point is an open set and each point can have a open set consisting of itself. Each point is completely discrete, meaning isolated and disconnected. So the discrete topology, in some sense, is no topology, no “nearness” or topological structure.

The correct approach is through graph realization or graph embedding. This is the study of a branch of graph theory, topological graph theory [30]. Intuitively, graph realization or graph embedding is to think the vertices of the graph as points in an Euclidean space and the edges of the graphs as the lines or curves in the Euclidean space connecting the vertices. With the graph realization or graph embedding, the graph $G$ is a sub-topological space of the shape $S$ as a topological space. So if the points in the graph $G$ are in a neighborhood, then they are also in a neighborhood in the shape topological space $S$.

In the following of this chapter, we develop topological clustering techniques for the purpose of diffeomorphic point matching. It has many similarities with Kohonen SOFM but there are also many differences. First, the purpose of SOFM is dimensionality reduction while our topological clustering is for point matching. Second, the map in SOFM is local while our topology is global. SOFM only provides a single patch for one open neighborhood of a point on the submanifold while our clustering allows non-trivial topologies which cannot be covered by a single patch. Third, the interaction in SOFM is between one cluster center and
the data points in the neighboring clusters while the interaction in our model in between one cluster center and the neighboring cluster centers.

4.4 Objective Function and the Algorithm

In order to enforce the principle of that points in a neighborhood should stay in the neighborhood, we add the term

$$\sum_{m=1}^{N} \sum_{n=1}^{N} G_{mn} \| r_m - r_n \|^2$$  \hspace{1cm} (4.1)

to the objective function, where $G$ is the adjacency matrix of the graph of the cluster centers of one set. The other set has the similar topological constraints. $G_{mn}$ is 1 if there is an edge between $r_m$ and $r_n$, and 0 otherwise. The new objective function now is

$$E(M^x, M^y, r, s, v, \phi)$$  \hspace{1cm} (4.2)

$$= \sum_{i=1}^{N_1} \sum_{k=1}^{N} M^x_{ik} \| x_i - r_k \|^2 + 2\sigma_T^2 \sum_{i=1}^{N_1} \sum_{k=1}^{N} M^x_{ik} \log M^x_{ik}$$

$$+ \sum_{j=1}^{N_2} \sum_{k=1}^{N} M^y_{jk} \| y_j - s_k \|^2 + 2\sigma_T^2 \sum_{j=1}^{N_2} \sum_{k=1}^{N} M^y_{jk} \log M^y_{jk}$$

$$+ \sum_{k=1}^{N} \| s_k - \phi(r_k, 1) \|^2 + 2\sigma_T^2 \lambda \int_0^1 \int_{\Omega} \| Lv(x, t) \|^2 dx dt$$

$$+ \frac{\tau}{4} \sum_{m=1}^{N} \sum_{n=1}^{N} G_{mn} \| r_m - r_n \|^2 + \frac{\tau}{4} \sum_{q=1}^{N} \sum_{q=1}^{N} G_{pq} \| s_p - s_q \|^2.$$  

The matrix $G$ is a symmetric matrix. The reason for the factor $\frac{1}{2}$ in the last line is that each edge in the graph is counted twice in the summation. $\tau$ is a new parameter describing the strength of the links between the cluster centers.

The algorithm to minimize this energy is very similar to the algorithm introduced in Chapter 3. However, the update equations for the cluster centers (3.19) and (3.20) should be modified accordingly:
\[ r_k = \frac{\sum_{i=1}^{N_1} M_{ik}^x x_k + s_k}{1 + \sum_{i=1}^{N_1} M_{ik}^x + \tau \sum_{m=1}^{N} G_{mk} r_m}, \quad (4.3) \]

\[ s_k = \frac{\sum_{j=1}^{N_2} M_{jk}^y y_j + \phi(r_k, 1) + \tau \sum_{m=1}^{N} G_{mk} s_m}{1 + \sum_{j=1}^{N_2} M_{jk}^y + \tau \sum_{m=1}^{N} G_{mk}}, \quad \forall k. \quad (4.4) \]

The rule of modification is, when updating the cluster center positions for \( r_k \), consider all other cluster centers \( r_m \), \( m = 1, 2, \ldots, N \) in the graph, whenever there is an edge from \( r_m \) to \( r_k \), we add \( \tau r_m \) to the numerator and we add \( \tau \) to the denominator. The modification to the update of \( s_k \) in the second set is similar.

### 4.5 Prescribed Topology

In some situations, the class of shapes have the same topology and the topology is known and the graph representation of the topology is easy. In such cases, we can initialize the graph with the prescribed topology. In the 2D case, when we deal with line contours, two typical situations are the open curves and closed curves. We can use chain topology and ring topology for the graphs.

#### 4.5.1 Chain Topology

We solved the matching problem again with the hand shapes with chain topology. The clustering result is shown in Figure 4–6 and this way we find the correct correspondence in Figure 4–7.

#### 4.5.2 Ring Topology

We applied the topological clustering and matching algorithm to the corpus callosum data, with ring topology. Figure 4–8 shows the topological clustering while Figure 4–9 shows the correspondence.

#### 4.5.3 \( S^2 \) Topology

We know the hippocampus shapes have a \( S^2 \) topology. We initialize the graph as a latitude and longitude grid. Figure 4–11 is the topological clustering of the
Figure 4–6: Topological clustering and matching of two hands

Figure 4–7: Correspondence with topology constraint
Figure 4–8: Topological clustering of corpus callosum shapes
Figure 4-9: Correspondence in topological clustering of corpus callosum shapes
first hippocampus set and Figure 4–12 is the topological clustering of the second hippocampus set.

4.6 Arbitrary Topology

We can see the limitations with prescribed topology. First, the class of shapes may have different topologies. Second, even if the class of the shapes have the same topology, the topology may be unknown before we run the matching algorithm. Third, even if the topology is known, it may be too complicated to construct a graph approximation and the initialization may involve human intervention. Forth, the edges of the graph is not truly topological in a sense they are not indefinitely flexible and stretchable strings. The topological constraints in the objective function 4.2 are actually identical elastic strings. So we see the geometrical or metric factor in the constraints. Sure we can adjust the coefficient
Figure 4–11: Topological clustering of hippocampus with $S^2$ topology: the first set
Figure 4–12: Topological clustering of hippocampus with $S^2$ topology: the second set
\(\tau\) for different stiffness of the strings, but it is difficult to tune the different relative stiffness connecting the cluster centers. Because this elasticity, the graph may have resistance to completely fit the shape. This can be seen in previous hippocampus examples.

The solution is to find the topology of the shape on the fly. First we cluster one of the data set. We then find the N-nearest neighbors of each cluster center. From each cluster center, we draw a directed edge to each of its nearest neighbors. This way we have a directed graph. The adjacency matrix is not symmetric in general. We then symmetrize the adjacency matrix, meaning if we have an edge from node \(i\) to \(j\) but not an edge \(j\) to \(i\), we then add an edge from \(j\) to \(i\). This way we get an undirected graph. We will use this graph topology for the clustering. Since in the matching problem, we assume the two data sets have the same topology, we will use the same graph for both data sets. We shrink the graph to the centroid of the data sets in the initialization and we continue with the joint clustering and matching algorithm. Figure 4–13 is the approximate graph topology of hippocampus data sets with 4 nearest neighbors. Figure 4–14 shows the topological clustering of the first hippocampus set and Figure 4–15 of the second hippocampus set.

Figure 4–16 a and b shows two 2D point shapes. Figure 4–18 is the graph topology we have learned using 4 nearest neighbors after clustering. Figure 4–18 shows the topological clustering of the two shapes.
Figure 4–13: Graph topology of hippocampus shape through learning
Figure 4–14: Topological clustering of hippocampus shape: the first point set
Figure 4–15: Topological clustering of hippocampus shape: the second point set

Figure 4–16: Fish shapes
Figure 4–17: Graph topology for the fish shape with 4 nearest neighbors
Figure 4–18: Topological clustering and matching of the fish shapes
CHAPTER 5
CONCLUSIONS

5.1 Contributions

The need for a good point feature matching algorithm arises in various application areas of medical image analysis. To my knowledge, this is one of the first attempts at diffeomorphic point matching in the circumstances of unknown correspondence. We have designed an objective function and an algorithm to simultaneously find the best clustering of two point sets and a mapping with the least deformation of space. We require the space deformation to be a diffeomorphic mapping because it is smooth and homeomorphic at the same time. The essence of this requirement is that in a homeomorphic mapping, neighboring points are mapped to neighboring points and the same is true for the inverse mapping. A diffeomorphic mapping can preserve the features of shapes. The diffeomorphism parameterization allows us to recover large deformations while simultaneously achieving good correspondence. We have demonstrated a joint clustering and diffeomorphism algorithm and applied it to 2D corpus callosum shapes and 3D hippocampal point sets in medical imaging. After diffeomorphism estimation, the shape distance, defined as the geodesic distance on the shape manifold is computed. Since the point sets have different cardinalities and since the shape distance is only defined w.r.t. the cluster centers, we also computed a modified Hausdorff distance between one original point set and the after-image of a second point set. We conclude that when the number of cluster centers increases, the modified Hausdorff distance decreases. In the process of careful validation, we investigated the role of the different numbers of clusters using Jensen-Shannon divergence in the joint clustering and diffeomorphism optimization process.
We further propose to use a graph representation for the shape topology information. Results are given for prescribed topologies like chain topology, ring topology – which are very common in dealing with 2D contour shapes – and genus zero closed surface topology in 3D. We also investigate the topology problem in general and the learning of topology with a nearest neighbor graph.

5.2 Future Work

In the current formulation, we still have a free parameter $\lambda$ whose value has to be determined. The immediate future goal is to further address (theoretically and experimentally), the role of free parameters. The same framework can be used for atlas estimation. Furthermore, once we have a turnkey 3D diffeomorphic feature matching algorithm, we plan to use it for hippocampal shape classification of epilepsy patients [69]. We realize with topological clustering and matching, it is a compromise and competition between clustering and topology preservation, in the future, we want to separate the two phases of correspondence and diffeomorphism and hope that may further improve the matching accuracy.
REFERENCES


BIOGRAPHICAL SKETCH

Hongyu Guo was born in China. He received BS and MS degree in physics in Nankai University. He received his MS degree in computer science from the University of Florida. Hongyu Guo has taught in a university. He has also worked in software industry in San Francisco, CA. His current research interests are in computer vision and biomedical imaging.