

Linköping Studies in Science and Technology
Thesis No. 1235

Manifold Learning and Representations for Image Analysis and Visualization

Anders Brun



INSTITUTE OF TECHNOLOGY
LINKÖPING UNIVERSITY

LiU-TEK-LIC-2006:16

Department of Biomedical Engineering
Linköpings universitet, SE-581 85 Linköping, Sweden
<http://www.imt.liu.se>

Linköping, February 2006

**Manifold Learning and Representations for Image Analysis and
Visualization**

© 2006 Anders Brun

*Department of Biomedical Engineering
Linköpings universitet
SE-581 85 Linköping
Sweden*

ISBN 91-85497-33-9

ISSN 0280-7971

Printed in Sweden by UniTryck AB, Linköping

“*What a useful thing a pocket-map is!*”

from *Sylvie and Bruno Concluded* by Lewis Carroll

Abstract

A manifold is a mathematical concept which generalizes surfaces to higher dimensions. The values of signals and data are sometimes naturally described as points in manifolds – they are manifold-valued.

In this thesis some recently proposed spectral methods for manifold learning are applied to a visualization problem in medical imaging. 3-D volume data of the human brain, acquired using Diffusion Tensor MRI, is post processed in a novel way in order to represent and visualize the shape and connectivity of white matter fiber bundles.

In addition to this real-world application of manifold learning, the contributions to a generic framework for processing of manifold-valued signals and data consist of the following. 1) The idea of the *diffusion mean*, which is a preliminary result related to the extrinsic and intrinsic means in certain manifolds. 2) A representation for extrinsic manifold-valued signal processing in $SO(3)$, Q , which is useful for linear averaging, filtering and interpolation. 3) A novel class of methods for manifold learning, the *sample logmaps*, which has strong connections to differential geometry. These maps can be used to visualize high-dimensional data and they are potentially useful for performing intrinsic signal processing and statistics of manifold-valued signals and data, in cases where the manifold is only known from samples.

Acknowledgements

The work presented in this thesis would have been impossible without the support of a number of other people.

First of all, I would like to express my deepest gratitude to my supervisor Professor Hans Knutsson. His many contributions to image processing and machine learning led me to this group in the first place, and he is still a major source of inspiration in my research. Thank you for believing in my ideas.

I am very grateful to my co-supervisor Associate Professor Carl-Fredrik Westin, Laboratory of Mathematics in Imaging at Brigham and Women's Hospital and Harvard Medical School, for providing great opportunities, for an excellent introduction to the exciting field of Diffusion Tensor MRI and for guiding me around at various bars and parties in Boston and Cambridge. Also thanks to my second and newest co-supervisor Associate Professor Magnus Herberthson, from the Department of Mathematics at Linköping University, master of orienteering and index gymnastics. Thanks also to Professor Martha E. Shenton, Harvard Medical School, in particular for her advices on grant writing and scientific leadership. Things like that come handy when you least expect it. I would also like to thank my other co-authors Professor Hae-Jeong Park, Assistant Professor Marek Kubicki, Dr. Steven Haker and Dr. Raúl San-José Estépar. As well as Lauren O'Donnell for valuable comments on the manuscript.

A big thank you goes to all friends and colleagues at the Department of Biomedical Engineering in Linköping, in particular all past and present staff in the Medical Informatics group. As well as all the people that I have met at the Laboratory of Mathematics in Imaging and the Surgical Planning Laboratory at Brigham and Women's Hospital and Harvard Medical School in Boston. Also special thanks to Dr. Lisa Jonasson, Dr. Pentti Kanerva and Dr. Karl Krissian.

I acknowledge the support from the Similar Network of Excellence, the Center for Medical Image Science and Visualization (CMIV) and Vetenskapsrådet (the Swedish Research Council).

All my dear friends, in Linköping and worldwide: Thank you for still being my friends.

Finally a warm thank you to my relatives and family, in particular my parents Anita and Paul Brun and to my grandmother Ingrid Erixon. Thank you for your love and support.

Table of Contents

1	Introduction	1
1.1	Motivations	1
1.2	Potential impact	2
1.3	Thesis Overview	2
1.4	Abbreviations	4
1.5	Mathematical Notation	5
2	Manifold-Valued Data	7
2.1	Background	7
2.2	Examples	8
3	Manifold-Valued Signal Processing	11
3.1	Background	11
3.2	Filtering Manifold-Valued Signals	11
3.3	Averaging, Filtering and Interpolation	13
3.4	The Extrinsic and Intrinsic Mean	13
3.5	The Idea of the “Diffusion Mean”	13
3.6	A More Mathematical Version of the Diffusion Mean	15
3.6.1	The Diffusion Mean in \mathbb{R}^n	15
3.6.2	The Diffusion Mean when $t \rightarrow 0$	17
3.6.3	The Diffusion Mean when $t \rightarrow \infty$	20
3.7	A Final Remark	22
4	Dimensionality Reduction and Manifold Learning	25
4.1	Dimensionality Reduction	26
4.2	Manifold Learning	27
4.3	Laplacian Eigenmaps	28
4.4	Isomap – Isometric feature mapping	29
4.5	Sample Logmaps	30
4.6	Some Concluding Remarks	33
5	Diffusion Tensor Magnetic Resonance Imaging	35
5.1	Diffusion Imaging	35
5.1.1	Diffusion	35
5.1.2	Estimating Diffusion Tensors	36
5.1.3	Diffusion in the Human Brain	38
5.1.4	Applications of DT-MRI	38

5.2	Processing Diffusion Tensor Data	39
5.2.1	Scalar Invariants	39
5.2.2	Fiber Tracking	41
5.2.3	Fiber Tract Connectivity	42
5.2.4	Segmentation of White Matter	43
5.3	Visualization of Streamline Data	44
5.3.1	Local and Global Features in DT-MRI	44
5.3.2	Learning and Representations	45
5.3.3	Visualization of Fiber Tract Connectivity	46
6	Review of Papers	49
6.1	Paper I: Coloring of DT-MRI Fiber Traces Using Laplacian Eigenmaps	49
6.2	Paper II: Clustering Fiber Traces using Normalized Cuts	50
6.3	Paper III: A Tensor-Like Representation for Averaging, Filtering and Interpolation of 3-D Object Orientation Data	50
6.4	Paper IV: Fast Manifold Learning Based on Riemannian Normal Coordinates	51
7	Discussion	53
7.1	Manifold Learning in Diffusion Tensor Imaging	53
7.2	Intrinsic vs. Extrinsic Methods for Manifold-Valued Signal Processing	54
7.3	Sample Logmaps – Intrinsic Processing of Empirical Data	55
7.4	Future research	55
8	Coloring of DT-MRI Fiber Traces Using Laplacian Eigenmaps	57
8.1	Introduction	57
8.2	Previous Work	59
8.3	Embedding Fiber Traces – a Motivation	60
8.4	Spectral Clustering and Embedding	61
8.5	Laplacian Eigenmaps	61
8.6	Similarity Through Connectivity	62
8.7	In Vivo DT-MRI Data	63
8.8	Experiments	64
8.9	Discussion	65
8.10	Conclusion	68
9	Clustering Fiber Traces Using Normalized Cuts	69
9.1	Introduction	70
9.1.1	Previous Work	70
9.2	Determining Fiber Similarity	71
9.2.1	Mapping Fiber Traces to an Euclidean Feature Space	72

9.2.2	Using the Gaussian Kernel for Pairwise Comparison	72
9.2.3	Constructing W	73
9.3	Normalized Cut and Clustering	73
9.4	Results	74
9.5	Discussion	76
9.6	Acknowledgments	77
10	A Tensor-Like Representation for Averaging, Filtering and Interpolation of 3-D Object Orientation Data	79
10.1	Introduction	79
10.2	Quaternions	80
10.3	Mapping Orientations to \mathbb{R}^{16}	82
10.4	... and Back Again	82
10.5	Rotation Invariance	83
10.6	Experiments	84
10.7	Discussion	84
11	Fast Manifold Learning Based on Riemannian Normal Coordinates	87
11.1	Introduction	87
11.2	Theory	89
11.3	Method	90
11.4	Experiments	91
11.4.1	The Swiss Roll	91
11.4.2	The Torus	92
11.4.3	The Klein Bottle	92
11.5	Discussion	94
11.6	Acknowledgements	95

1

Introduction

1.1 Motivations

The work presented in this thesis was inspired by recent advances in so called manifold learning and mainly financed by the *Manifold-Valued Signal Processing* project funded by Vetenskapsrådet (the Swedish Research Council). Being a Swedish licentiate thesis, it should be seen as a piece of work in the middle between a masters thesis and a PhD thesis.

The focus on manifold learning is mainly motivated by the need for methods for high-dimensional data analysis and visualization, both in imaging sciences in general and in medicine in particular. Texture, shape, orientation and many other aspects of data need to be quantified and compared, and the mathematical theory of smooth manifolds is a natural approach for many such problems.

In this thesis the use of manifolds and manifold learning, for image analysis and visualization, is explored from three different views.

Dimension reduction Finding a low-dimensional parameterization of manifold-valued data embedded in a high-dimensional space.

Data visualization Visualization of manifold-valued data embedded in a high-dimensional space.

Signal processing Basic signal processing, such as interpolation, smoothing and filtering of manifold-valued time series and volume data.

For the PhD thesis, the goal will be to give a more complete view on manifold learning for data processing and visualization. The exploration done so far, expressed in the three topics discussed in this intermediate thesis, will provide a good roadmap for this future work.

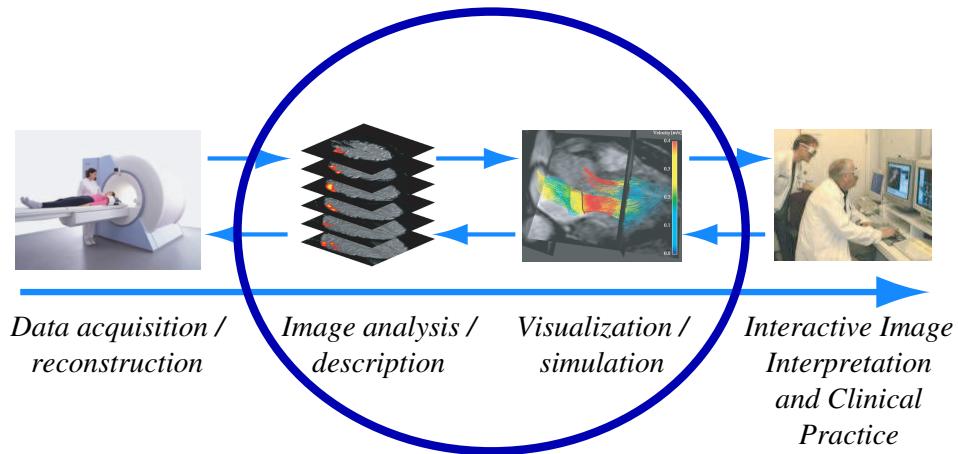


Figure 1.1: Within medical image science, the immediate applications for manifold-valued signal processing are in image analysis and visualization. Promising areas of future research also include the design of intelligent user interfaces using manifold learning. (Image modified from CMIV presentation material, with permission.)

1.2 Potential impact

The outcome of this and future work will be a new set of tools to understand and process manifold-valued signals, embedded in possibly high dimensional data. Increased ability to represent and process features present in medical images, such as shape, texture and organ orientation, will aid in the development of better diagnoses and increase our ability to make demographical studies using data from the imaging sciences. This will be of benefit not only within our field of research, which is medical image analysis, but also for the rest of the signal processing community where there is a need to describe and objectively quantify features that are naturally represented as points in manifolds.

1.3 Thesis Overview

The thesis consists of two parts. The first part (chapters 2–7) contains an introduction to the material and some overall conclusions. The second part (chapters 8–11) consists of four full-length conference papers.

Chapter 2 contains a very short introduction to manifolds and gives some examples of manifold-valued data in real-world applications.

Chapter 3 gives an introduction to manifold-valued signal processing, with some comments on extrinsic and intrinsic means to motivate the representation used in chapter 10.

Chapter 4 is an introduction to dimension reduction and manifold learning, including some comments on *sample logmaps*, a class of methods presented later in chapter 11.

Chapter 5 contains a short introduction to Diffusion Tensor MRI, including some remarks to motivate the methods presented in chapter 8 and 9.

Chapter 6 is a review of the four papers presented in chapter 8 – 11.

Chapter 7 contains a discussion and conclusions for the total work presented in this thesis, including notes on future work.

Chapter 8–11 contains reformatted versions of four full conference papers. Some minor corrections have been made.

1.4 Abbreviations

A list of abbreviations used in the thesis.

ADC	Apparent Diffusion Coefficient
CCA	Canonical Correlation Analysis / Curvilinear Components Analysis
C-Isomap	Conformal Isomap
CSF	Cerebrospinal Fluid
DT-MRI	Diffusion Tensor Magnetic Resonance Imaging
DWI	Diffusion Weighted Imaging
EOF	Empirical Orthogonal Functions
FA	Fractional Anisotropy
FIR	Finite Impulse Response
GTM	Generative Topographic Map
HLLE	Hessian Locally Linear Embedding
ICA	Independent Components Analysis
i.i.d.	independent and identically distributed
Isomap	Isometric Feature Mapping
KPCA	Kernel Principal Components Analysis
L-Isomap	Landmark Isomap
LE	Laplacian Eigenmaps
LLE	Locally Linear Embedding
LSI	Latent Semantic Indexing
LSDI	Line Scan Diffusion weighted Imaging
LTSA	Local Tangent Space Alignment
MDS	Multidimensional Scaling
MR	Magnetic Resonance
MRI	Magnetic Resonance Imaging
Ncut	Normalized Cut
PCA	Principal Components Analysis
PDD	Principal Diffusion Direction
PP	Projection Pursuit
RANSAC	Random Sample Consensus
RGB	Red, Green, Blue
SOM	Self Organizing Maps

1.5 Mathematical Notation

\mathbf{v}	Unspecified vectors
\mathbf{b}_i	A contravariant basis vector
\mathbf{b}^i	A covariant basis vector
v^i	(The coordinates of) a contravariant vector
w_i	(The coordinates of) a covariant vector
g_{ij}	(The components of) the metric tensor
M	A manifold
TM	The tangent bundle of M
T^*M	The cotangent bundle of M
$T_p M$	The tangent space of M at the point p
$T_p^* M$	The cotangent space of M at the point p
V^*	The dual vector space of a vector space V
$\dim V$	The dimensionality of V
$\hat{\mathbf{e}}_i$	A unit basis vector in $T_p M$
\mathbf{g}	A gradient vector in $T_p^* M$
X	A set of data points on M embedded in \mathbb{R}^N
x, y	Points on M embedded in \mathbb{R}^N
p	A point on a manifold
$B_r(p)$	A ball of p with radius r in a set
$N(p)$	A neighborhood of p in a set
$H(t)$	A curve along a geodesic path.
$\exp_p(v)$	The exponential of v at base point p
$\log_p(x)$	The logarithmic of x at base point p
$d(x, y)$	The geodesic distance between x and y
$\Psi(y)$	Maps a point in $B(p)$ to \mathbb{R}^n , which gives a chart
\mathbb{R}	The set of all real numbers
\mathbb{H}	The set of all quaternions
S^1	The 1-sphere, i.e. the circle in a 2-dimensional space
S^2	The 2-sphere, i.e. the sphere in a 3-dimensional space
S^n	The n -sphere, i.e. a sphere in a $(n + 1)$ -dimensional space
\mathbb{RP}^2	The real projective plane
\mathbb{RP}^3	The real projective space
\mathbb{RP}^n	The real projective n -space
$SO(3), SO(3, \mathbb{R})$	The (real) special orthogonal group in 3 dimensions

2

Manifold-Valued Data

Manifolds are generalizations of surfaces. They are topological spaces which are locally topologically equivalent to \mathbb{R}^n for some n . At a larger scale however, a manifold may look different. One example of a manifold is the surface of the sphere in \mathbb{R}^3 , \mathbb{S}^2 , which is locally topologically equivalent to \mathbb{R}^2 but which global properties are quite different to \mathbb{R}^2 . Other examples of manifolds include for instance a circle, \mathbb{S}^1 and a torus $\mathbb{S}^1 \times \mathbb{S}^1$. It is easy to think of a surface, the manifold, *embedded* in an extrinsic space. However, manifolds and operations on points in manifolds can also be defined without the notion of any extrinsic space. They are defined by their intrinsic properties.

2.1 Background

Today there is a rapidly increasing need to process more complex features that are naturally represented as points on a manifold, hidden in high-dimensional signals such as images (Seung and Lee, 2000). Quite often there is a need to quantify various phenomena which are obvious for a human observer, but difficult to describe in mathematical terms. In medical image processing in particular, there is an immediate need for robust methods to characterize shape (Fletcher et al., 2004), texture and object (e.g. anatomical entities) orientation in 3-D. Examples of such applications are analysis of multi-dimensional spatial data such as Diffusion Tensor MRI, where each volume element of data contains an estimate of local anatomical structure, filtering of manifold-valued temporal signals such as 3-D object orientation for tracking of objects during surgical intervention and for diagnostic purposes, lip reading (Bregler and Omohundro, 1994) and recent methods for processing of color spectra (Lenz et al., 2005) using Lie-group theory.

2.2 Examples

Manifold-valued signals may sound rare at a first glance. However, non-trivial examples do arise naturally under some quite common circumstances (Seung and Lee, 2000; Donoho and Grimes, 2005). Also, introducing invariances to a vector-valued signal will often give a new representation which is a manifold. Invariance to signal strength in \mathbb{R}^3 gives a representation equivalent to a unit sphere in \mathbb{R}^3 , i.e. \mathbb{S}^2 . Here are some examples:

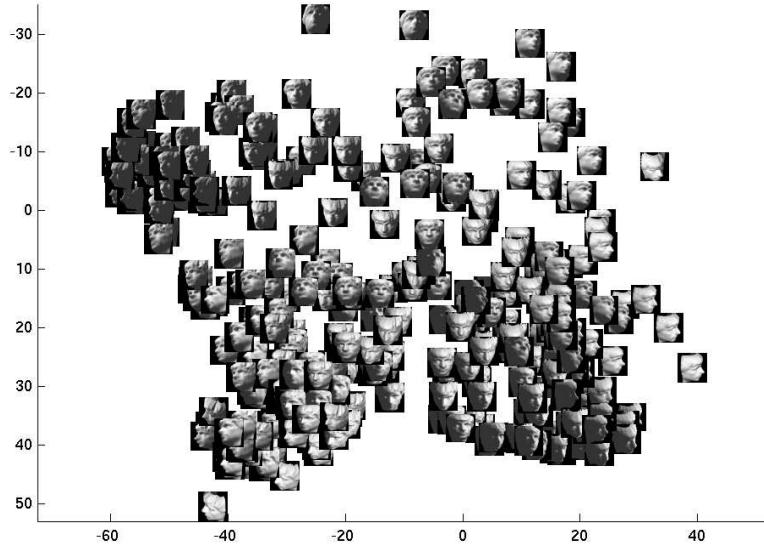


Figure 2.1: The space or manifold of facial expressions in a particular dataset, is naturally parameterized by a low-dimensional manifold.

1. A low-dimensional example of the importance of invariant representations and how they form manifolds is hue. Color can be described for instance by components of red, green and blue, but also using the concepts of hue, saturation and brightness. Hue is a typical example of a non-linear manifold in a representation, invariant to brightness and saturation.
2. Invariance to signal amplitude in a N -dimensional vector space, \mathbb{R}^N , implies a manifold topologically equivalent to the unit sphere in N dimensions, i.e. \mathbb{S}^{N-1} .
3. Continuous representation of line orientations in a N -dimensional space (think of unit vectors invariant to sign flips). This forms a manifold topologically equivalent to a projective space, \mathbb{RP}^{N-1} , which identifies antipodal

points on a sphere, \mathbb{S}^{N-1} , in N dimensions.

4. Signal representations insensitive to noise play an important role in signal processing. The resulting manifold typically has a lower intrinsic dimension than the original vector space, which can be seen in figure 2.2.

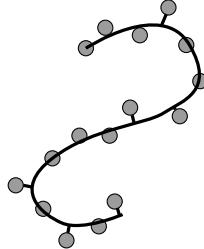


Figure 2.2: Example of a vector space where the original dimensionality of the data-points (o) is two. Invariance to noise is created by projecting the points onto a one-dimensional manifold (the line).

A real-world example of data which is manifold-valued is images of a face, inspired by (Tenenbaum et al., 2000; Seung and Lee, 2000). These images are parameterized by pose and direction of light in the scene. It is natural to describe these parameters using a 3-dimensional manifold (with a border). Figure 2.1 shows how an image of a face varies continuously in two dimensions.

3

Manifold-Valued Signal Processing

Building a completely new framework for identifying and processing of manifold-valued signals is not possible within the scope of this thesis. I will instead touch on a few important aspects of this complex problem and deal with them separately.

Manifold-valued signals arise naturally from invariant representations in signal- and image processing. The specific goal of this thesis is to develop the beginning of a generic framework, motivated by first principles from statistics and differential geometry, to 1) Identify low-dimensional manifold-valued signals in high-dimensional spaces such as images. 2) Perform basic signal processing on such manifold-valued signals, e.g. averaging, filtering and interpolation. Within the project funding large parts of this thesis, we have chosen to call this *manifold-valued signal processing*.

3.1 Background

Traditional signal processing is based on a well-established theory developed for scalar- and vector-valued signals in a statistical framework. A substantial part of the theory deals with operations which may be seen as convolutions. Originally, the theory was developed for one-dimensional signals, but in pace with the development of techniques for acquiring images and other multi-dimensional data, it has been extended to handle two-, three-, and higher-dimensional stochastic scalar fields. In recent years, not only the outer dimensionality of the data has been increasing, but also the inner dimensionality. Perhaps the earliest example of such data is color images, where each pixel needs to be represented by a three component vector, while a more recent example is Diffusion Tensor MRI.

3.2 Filtering Manifold-Valued Signals

Real-, complex- and vector-valued signal processing has reached a level of maturity, where a mathematical framework based on linear algebra and statistics under

Gaussian assumptions forms a solid base. One of the two specific goals of this project is to develop similar basic tools for signal processing of manifold-valued signals. While some concepts in vector spaces have no equivalence on manifolds, others may be translated (Pennec et al., 2005).

One important idea of this thesis is to study manifolds using both intrinsic and extrinsic methods. Working with a manifold using intrinsic methods, there is no need to consider any external coordinate system in which the manifold is embedded. The study of the unit sphere, \mathbb{S}^2 , can for instance be performed without reference to an Euclidean 3-dimensional space in which it may live. Extrinsic methods on the other hand work with manifolds embedded in an extrinsic space and inherit the metric of the embedding space. While intrinsic methods are often believed to be the best way to treat manifolds in a mathematically sound way, extrinsic methods are sometimes easier to implement and give similar results to intrinsic methods in practice. In (Srivastava and Klassen, 2002) the authors have for instance recently described the advantages of using extrinsic estimators on manifolds. Some preliminary results presented in this thesis point towards important special cases, certain manifolds, where both intrinsic and extrinsic methods may be considered optimal for signal processing, depending on the noise model.

Filtering of manifold-valued signals may be performed using local linearizations, when the amount of noise is moderate. It would then be possible to apply for instance a Wiener or Kalman filter locally to the signal in order to perform signal processing. Some of our previous work however, mainly in the field of tensor image processing, are examples of extrinsic methods which use global linearization rather than local linearization (Knutsson, 1989; Granlund and Knutsson, 1995). While this may sound crude, the results of filtering and interpolation are sometimes very similar to intrinsic methods.

One important example of a temporal signal which is manifold-valued, for which filtering has been described in the literature, is 3-D object orientation parameterized by time. This example has important applications, such as filtering of camera orientations from noisy sensor data in video production. In (Lee and Shin, 2002) the authors describe an intrinsic framework for filtering of such orientation data based on linearizations using the exponential map of the Lie-group $SO(3)$ describing rotations in 3-D. In a paper devoted to calculating mean values of orientation data, Moakher has compared intrinsic and extrinsic means of 3-D object orientations (Moakher, 2002). Later in chapter 10 the paper (Brun et al., 2005a) is presented, which describes a special kind of extrinsic averaging on $SO(3)$ using the so called Q representation.

3.3 Averaging, Filtering and Interpolation

A useful framework for processing of manifold-valued signals should be simple, sound and powerful at the same time. Linear filtering and interpolation used in classical signal processing is based on linear estimators of the true signal. Simple averaging, obtaining mean values of a set of points, may also be interpreted as linear estimation. On manifolds however, as opposed to vector spaces, algebraic structures for calculating linear estimators are seldom present.

However, the Gaussian distribution, which is the basis of all linear estimation in \mathbb{R}^n , can be generalized to manifolds. Using this generalized Gaussian, based on Brownian motion, it is sometimes possible to derive an estimate which shares many similarities with linear estimation in \mathbb{R}^n .

3.4 The Extrinsic and Intrinsic Mean

The intrinsic and extrinsic mean are defined as follows. First the intrinsic:

$$\bar{x}_{\text{int}} = \arg \min_{q \in M} \sum_{i=1}^N d_{\text{int}}^2(\mathbf{x}_i, \mathbf{q}) \quad (3.1)$$

i.e. the point which minimizes the sum of squared distances to all other points, measured with intrinsic distances. The extrinsic mean on the other hand is

$$\bar{x}_{\text{ext}} = \arg \min_{q \in M} \sum_{i=1}^N d_{\text{ext}}^2(\mathbf{x}_i, \mathbf{q}) \quad (3.2)$$

This is basically the same as for the intrinsic mean, except that distances are measured using the distance function from the embedding space. It is shown in Srivastava and Klassen (2002) that the extrinsic mean may be calculated by taking the usual mean vector in the embedding space first, and then projecting that *extrinsic* mean vector to the closest point on the manifold M . If the projection back to the manifold is easy to compute, which it is for instance if the manifold is a sphere, then the main advantage of the extrinsic mean is that we can avoid a nonlinear optimization problem constrained on M . See figure 3.1 for a schematic version of the two procedures.

3.5 The Idea of the “Diffusion Mean”

By interpreting the mean value or mean vector in \mathbb{R}^n as a maximum likelihood estimation from a set of stochastic variables which are independent and identically distributed, a concept which is here called the *diffusion mean*, is described and studied for a couple of very trivial cases. This analysis will be far from complete, but it will give the embryo to a motivation of why extrinsic mean values sometimes

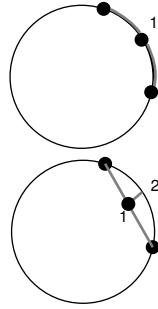


Figure 3.1: A schematic view of how intrinsic mean (top) and extrinsic mean (bottom) are calculated. Step 1 correspond to a minimization procedure. Step 2 (in the extrinsic mean) correspond to a projection.

are optimal. This in turn could serve as a motivation for extrinsic signal processing in general, or at least of a particular class of symmetric manifolds.

The basic idea is simple. We may arrive at the mean value of a set of samples in \mathbb{R}^n by the following procedure:

1. For each sample point \mathbf{x}_i in \mathbb{R}^n , calculate the solution to the diffusion equation at time equal to t for the PDE which has a Dirac function centered at the sample as initial value. In \mathbb{R}^n , this will result in a Gaussian probability distribution spreading out from the sample \mathbf{x}_i . Calculate this solution separate for each of the samples.
2. Multiply the solutions, i.e. the isotropic Gaussians in \mathbb{R}^n . Call the resulting function $P(\mathbf{x})$.
3. Seek the maximum of $P(\mathbf{x})$. This maximum will be located at the mean value of the points \mathbf{x}_i for any choice of $t > 0$.

The above procedure is possible to perform on any manifold, as long as it is possible to solve the diffusion equation for an initial value which is a Dirac function. The diffusion corresponds to an increase of uncertainty in the measurement when $t \rightarrow \infty$. In \mathbb{R}^n , which is a flat manifold without border, the choice of t will not affect the result. For nonlinear manifolds however, the answer will depend on the choice of t or equivalently, the amount of uncertainty in our Brownian motion model for noise. This is also true for manifolds with a boundary, even if they are flat, such as the interval $[0, 1] \in \mathbb{R}$.

In figures 3.2 – 3.5 the *diffusion mean* is studied experimentally on the circle, \mathbb{S}^1 . It turns out that for $t \rightarrow 0$, the diffusion mean approaches the intrinsic mean on the circle. And maybe more surprising, for $t \rightarrow \infty$ the diffusion mean approaches the extrinsic mean on the circle. This speaks in favor of both the intrinsic and the extrinsic mean, as two sides of the same coin, for this particular case.

From the above discussion it is clear that the diffusion mean procedure is defined

for all $t > 0$, but the uniqueness of the solution is not guaranteed in any way. The idea very simple. In fact, it would come as a surprise if this particular idea has not been discovered before. The point to be made however is that for certain manifolds, both the intrinsic and extrinsic mean are optimal, depending on the model of the noise. In the next section the diffusion mean is discussed in a slightly more mathematical way.

3.6 A More Mathematical Version of the Diffusion Mean

While intrinsic approaches to estimation on manifolds may be considered to be the “most natural” from a theoretical point of view, extrinsic methods may be more efficient to use and quite often produce similar results. Preliminary results show that for some highly symmetrical manifolds, certain extrinsic means may even be optimal from a statistical point of view under the appropriate assumptions on noise.

3.6.1 The Diffusion Mean in \mathbb{R}^n

A statistical interpretation of a mean vector $\bar{\mathbf{x}}$ for a set of vectors $\mathbf{x}_i \in R^n$ is the following. Suppose the vectors in the set are measurements of a vector \mathbf{x} corrupted by Gaussian noise. Thus $\mathbf{x}_i = \mathbf{x} + \mathbf{n}_i$, $\mathbf{n}_i \in N(0, \Sigma)$, is a random variable and $\{\mathbf{x}_i\}$ is a set of i.i.d. (independent identically distributed) samples. To estimate the value of \mathbf{x} given the set of samples, the following likelihood function may be used.

$$\begin{aligned} P(\mathbf{x}|\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N) &= \\ P(\mathbf{x}|\mathbf{x}_1)P(\mathbf{x}|\mathbf{x}_2)P(\mathbf{x}|\mathbf{x}_N) &= \\ C_1 \prod_{i=1}^N \exp \left(-\frac{1}{2}(\mathbf{x}_i - \mathbf{x})^T \Sigma^{-1} (\mathbf{x}_i - \mathbf{x}) \right) &= \\ C_1 \exp \left(-\frac{1}{2} \sum_{i=1}^N (\mathbf{x}_i - \mathbf{x})^T \Sigma^{-1} (\mathbf{x}_i - \mathbf{x}) \right) &= \\ C_2 \exp \left(-\frac{1}{2} N(\bar{\mathbf{x}} - \mathbf{x})^T \Sigma^{-1} (\bar{\mathbf{x}} - \mathbf{x}) \right) \end{aligned}$$

From this we see that regardless of Σ , the covariance matrix, the maximum likelihood estimate of \mathbf{x} is $\hat{\mathbf{x}}_{ML} = \bar{\mathbf{x}}$.

The Gaussian distribution in R^n is related to particle diffusion processes and the heat equation. Given a distribution $I(\mathbf{p}, t_0)$, describing the distribution of heat or particles at time t_0 , the heat equation states

$$I_t(\mathbf{p}, t) = D \Delta_p I(\mathbf{p}, t) \quad (3.3)$$

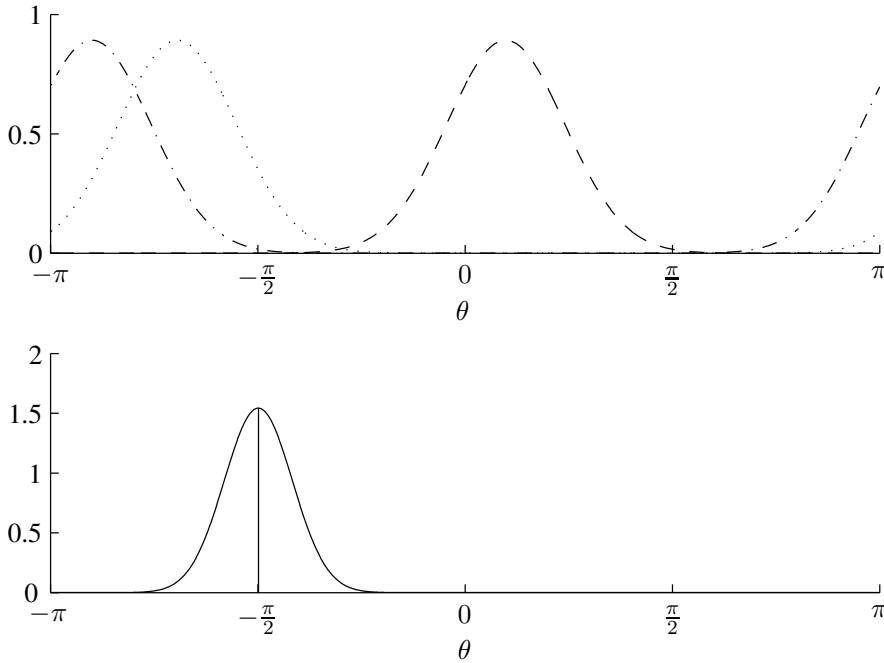


Figure 3.2: The “diffusion mean” for three points on the circle S^1 . **Top:** Three samples have been collected on S^1 , corresponding to positions -2.80 , -2.11 and 0.34 . For $t = 0.1$ their individual likelihood functions look like in the plot. **Bottom:** The total likelihood, regarding the three samples as independent, peaks around -1.52 , which is close to the intrinsic mean:
 $\bar{x}_{\text{int}} = (-2.80 - 2.11 + 0.34)/3 \approx 1.52$.

where D is the diffusion coefficient, I_t is the time derivate and Δ_p is the Laplacian operator acting in the spatial domain. The solution to the heat equation at a time $t_0 + t$ is obtained by convolution in the spatial domain

$$I(\mathbf{p}, t_0 + t) = I(\mathbf{p}, t_0) * K(\mathbf{p}, t) \quad (3.4)$$

where $K(\mathbf{p}, t)$ is the heat kernel in n dimensions, centered at the origin.

$$K(\mathbf{p}, t) = \frac{1}{(4\pi t)^{n/2}} \exp\left(-\frac{|\mathbf{p}|^2}{4Dt}\right) \quad (3.5)$$

To study the behavior of a single particle moving according to a Brownian motion diffusion process, one may choose $I(\mathbf{p}, t_0)$ to be a Dirac function $\delta(\mathbf{p} - \mathbf{x})$. Applying the diffusion process to the initial probability distribution during time t would describe the probability of finding a particle at position \mathbf{p} at time $t_0 + t$ if the position was known to be \mathbf{x} at time t_0 .

The Gaussian distribution may be generalized to non-linear manifolds by using the diffusion equation. In this way the maximum likelihood estimate of a set of

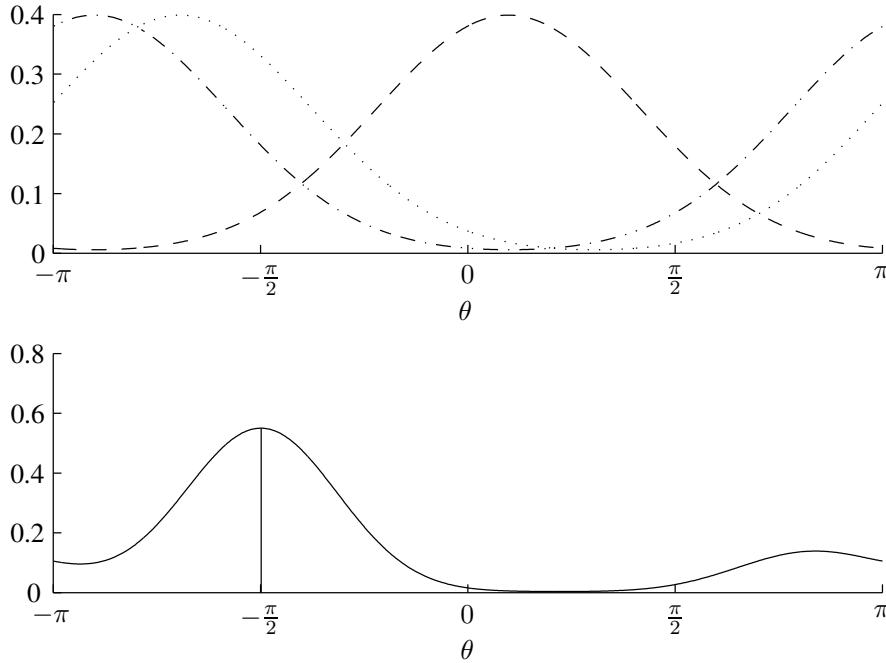


Figure 3.3: The “diffusion mean” for three points on the circle S^1 . **Top:** Three samples have been collected on S^1 , corresponding to positions -2.80 , -2.11 and 0.34 . For $t = 0.5$ their individual likelihood functions look like in the plot. **Bottom:** The total likelihood, regarding the three samples as independent.

measurements on a manifold may also be calculated. Depending on the choice of the time interval t and diffusion coefficient D , an analogy with an isotropic covariance matrix Σ can be made. However, unlike for R^n , the maximum likelihood estimate we call the *diffusion mean* may depend on the uncertainty of the particle distribution, i.e. t and D . Some simple but relevant special cases will now be discussed.

3.6.2 The Diffusion Mean when $t \rightarrow 0$

First an expression for the diffusion mean on general manifolds is derived for the limit $t \rightarrow 0$. This corresponds to measurements of points on a manifolds which have been affected by a Brownian motion, but only very short time. In a signal processing context, this corresponds to measurements with Gaussian distribution and high certainty.

The so called *short time kernel* for the diffusion equation, an approximation to the diffusion equation for short intervals of time, has been studied in (Spira et al., 2003). In their work they only derive a formula for the 2-D case and this is the result used in this thesis as well. For 2-D manifolds, the following formula is

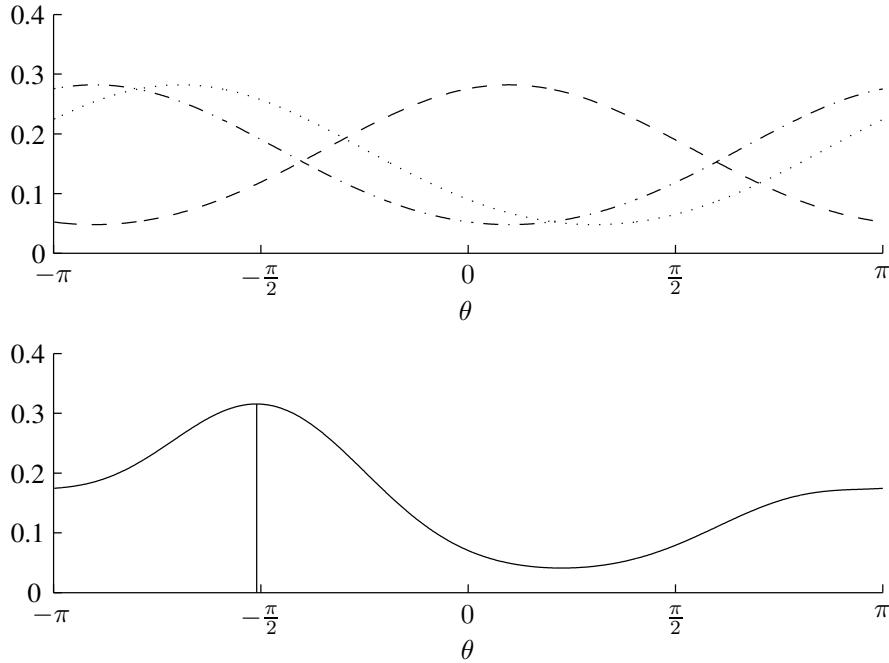


Figure 3.4: The “diffusion mean” for three points on the circle S^1 . **Top:** Three samples have been collected on S^1 , corresponding to positions -2.80 , -2.11 and 0.34 . For $t = 1.0$ their individual likelihood functions look like in the plot. **Bottom:** The total likelihood, regarding the three samples as independent

obtained,

$$K(p, q, t) \approx \frac{H_0}{t} \exp \left(-\frac{d_g^2(p, q)^2}{4t} \right). \quad (3.6)$$

Here p and q are points on the manifold and the kernel K describes the probability of diffusion from p to q during time t . This means that the probability function has a lot of similarities with the ordinary Gaussian in R^n when $t \rightarrow 0^+$. The term $d_g(p, q)^2$ in the exponential function is the squared geodesic distance between p and q . In particular, H_0 is a constant which does not depend on p or q . This is shown in greater detail in (Spira et al., 2003).

An approximation to the particle diffusion probability function may now be written down for the probability of measuring p if the true value is q after time t when $t \rightarrow 0^+$.

$$P(p|q, t) = \frac{H_0}{t} \exp \left(-\frac{d_g^2(p, q)^2}{4t} \right) \quad (3.7)$$

The likelihood $L(q|p, t) = P(p|q, t)$ and the likelihood of q given a set $\{\mathbf{x}_i\}$ of

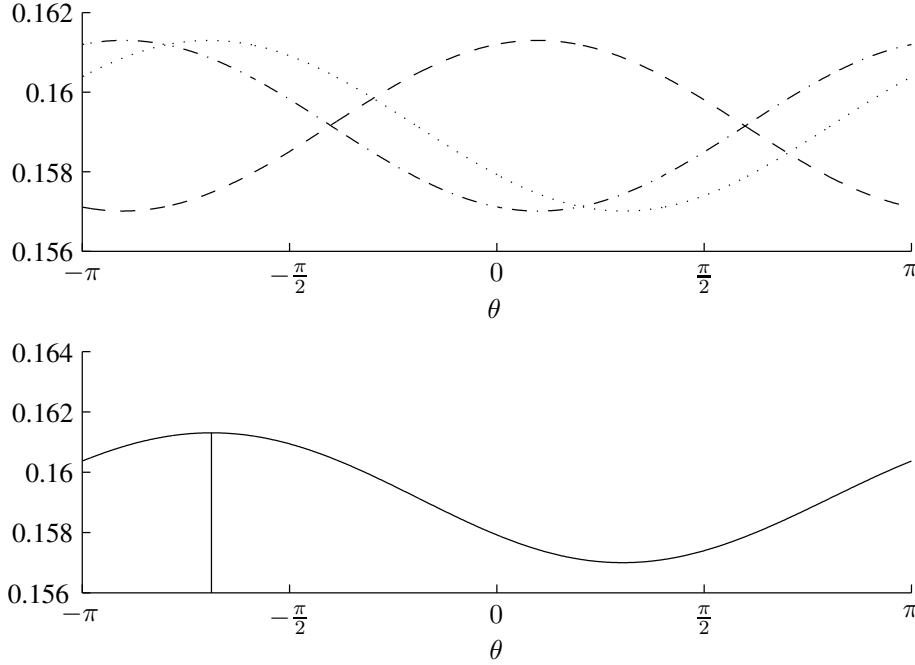


Figure 3.5: The “diffusion mean” for three points on the circle \mathbb{S}^1 . **Top:** Three samples have been collected on \mathbb{S}^1 , corresponding to positions -2.80 , -2.11 and 0.34 . For $t = 5.0$ their individual likelihood functions look like in the plot. **Bottom:** The total likelihood, regarding the three samples as independent, peaks around -2.11 , which is close to the extrinsic mean:

$$\bar{x}_{\text{ext}} = -\pi + \tan^{-1} \frac{(\sin(-2.80) + \sin(-2.11) + \sin(0.34))}{(\cos(-2.80) + \cos(-2.11) + \cos(0.34))} \approx -2.11.$$

i.i.d. samples is

$$\begin{aligned}
 L(q|\{\mathbf{x}_i\}, t) &= \\
 L(q|\mathbf{x}_1, t)L(q|\mathbf{x}_2, t)\dots L(q|\mathbf{x}_N, t) &= \\
 P(\mathbf{x}_1|q, t)P(\mathbf{x}_2|q, t)\dots P(\mathbf{x}_N|q, t) &= \\
 \prod_{i=1}^N \frac{H_0}{t} \exp\left(-\frac{d_g^2(\mathbf{x}_i, q)}{4t}\right) &= \\
 \frac{H_0^N}{t^N} \exp\left(-\frac{1}{4t} \sum_{i=1}^N d_g^2(\mathbf{x}_i, q)\right)
 \end{aligned}$$

Finding the maximum likelihood now yields

$$\begin{aligned}
\hat{\mathbf{x}}_{ML} &= \arg \max_q L(q|\{\mathbf{x}_i\}, t) \\
&= \arg \max_q \frac{H_0^N}{t^N} \exp \left(-\frac{1}{4t} \sum_{i=1}^N (d_g^2(\mathbf{x}_i, q)) \right) \\
&= \arg \min_q \sum_{i=1}^N d_g^2(\mathbf{x}_i, q) \\
&= \bar{\mathbf{x}}_g
\end{aligned}$$

where $\bar{\mathbf{x}}_g$ is known as the intrinsic mean for a set of data points in a manifold. While the intrinsic mean is often considered to be the natural generalization of mean value for points on a manifold, the probabilistic interpretation of the short time diffusion kernel further motivates the use of the $\bar{\mathbf{x}}_g$ to estimate \mathbf{x} . However, this line of reasoning is only valid for samples from a manifold affected by Gaussian noise, when $t \rightarrow 0$.

3.6.3 The Diffusion Mean when $t \rightarrow \infty$

Another interesting case is $t \rightarrow \infty$. What can be said under the assumption that the measurements of \mathbf{x} have been affected by a Brownian motion for a long time and thereby have a very flat Gaussian distribution on the manifold? At present time, the author is not aware of the solution for this in the general case. But for particular manifolds it is possible to derive a formula for the maximum likelihood estimate given the above assumptions.

For a thin rod of length L with insulated ends, the heat equation or equivalently the diffusion equation $I_t(p, t) = D\Delta_p I(p, t)$, $x \in [0, L]$ gives solutions of the form (Strauss, 1992)

$$I(x, t) = \frac{1}{2}A_0 + \sum_{n=1}^{\infty} A_n \exp(-(n\pi/L)^2 Dt) \cos(n\pi x/L) \quad (3.8)$$

if the initial distribution at $t = 0$ is given by the Fourier cosine expansion

$$I(x) = \frac{1}{2}A_0 + \sum_{n=1}^{\infty} A_n \cos(n\pi x/L) \quad (3.9)$$

$$A_n = \frac{2}{L} \int_0^L I(x) \cos(m\pi x/L) dx \quad (3.10)$$

The insulated ends of the rod may also be interpreted in a diffusion setting, as impermeable walls of a container, where particles affected by Brownian motion cannot pass through.

Again, studying the probability density function when observing a particle initiated in a point $x_i \in [0, L]$ corresponds to the diffusion of a Dirac function, $I(x) = \delta(x_i - x)$. The Fourier cosine coefficients are

$$A_n = \frac{2}{L} \int_0^L \delta(x_i - x) \cos(m\pi x/L) dx \quad (3.11)$$

$$= \frac{2}{L} \cos(m\pi x_i/L) \quad (3.12)$$

and the probability density function for observing the particle at a point x at time t is

$$P(x_i|q, t) = \frac{1}{L} + \sum_{n=1}^{\infty} \frac{2}{L} \cos(n\pi q/L) \exp(-(n\pi/L)^2 Dt) \cos(n\pi x_i/L) \quad (3.13)$$

The likelihood for a set of N independent observations x_i having the above distribution now becomes

$$L(q|x_i, t) = \quad (3.14)$$

$$L(q|x_1, t)L(q|x_2, t) \dots L(q|x_N, t) = \quad (3.15)$$

$$P(x_1|x, t)P(x_2|q, t) \dots P(x_N|q, t). \quad (3.16)$$

$$(3.17)$$

When $t \rightarrow \infty$ each of the individual likelihood functions converges to the uniform distribution $L(q|x_i, t \rightarrow \infty) = 1/L$. However, for every $t < \infty$ the function $L(q|x_i, t)$ has a maximum and the same is true for $L(q|\{x_i\}, t)$. If we assume the second term in the Fourier cosine expansion of $\delta(x - x_i)$ is non-zero, $A_1 \neq 0$, for at least one of the likelihood functions, it turns out that these terms will determine which q maximizes the likelihood function when $t \rightarrow \infty$. This maximum will be located either at $q = 0$ or $q = L$ and give very little information regarding the position of the true x . Clearly the diffusion mean is not meaningful in this context, when $t \rightarrow \infty$ and uncertainty increases.

For diffusion on the circle, \mathbb{S}^1 , the diffusion mean procedure will give a more interesting estimate. Using the full Fourier series on the interval $x \in [-L, L[$, a function on a circle with circumference $2L$ may be represented by (Strauss, 1992)

$$I(x) = \frac{1}{2} A_0 + \sum_{n=1}^{\infty} (A_n \cos(n\pi x/L) + B_n \sin(n\pi x/L)) \quad (3.18)$$

$$A_n = \frac{1}{L} \int_{-L}^L I(x) \cos(n\pi x/L) dx \quad (n = 0, 1, 2, \dots) \quad (3.19)$$

$$B_n = \frac{1}{L} \int_{-L}^L I(x) \sin(n\pi x/L) dx \quad (n = 0, 1, 2, \dots). \quad (3.20)$$

Solving $I_t(p, t) = D\Delta_p I(p, t)$ gives the following solution:

$$I(x, t) = \frac{1}{2}A_0 + \sum_{n=1}^{\infty} e^{-(n\pi/L)^2 Dt} (A_n \cos(n\pi x/L) + B_n \sin(n\pi x/L)) \quad (3.21)$$

The probability density function after time t for a particle moving according to Brownian motion on the circle parameterized by $x_i \in [-L, L]$ corresponds to the diffusion of a Dirac function, $I(x) = \delta(x_i - x)$. The Fourier cosine coefficients are

$$A_n = \frac{1}{L} \cos(m\pi x_i/L) \quad (n = 0, 1, 2, \dots) \quad (3.22)$$

$$B_n = \frac{1}{L} \sin(m\pi x_i/L) \quad (n = 0, 1, 2, \dots) \quad (3.23)$$

and the probability density function for observing the particle at a point x at time t is

$$P(x_i|q, t) = \frac{1}{2L} + \frac{1}{L} \sum_{n=1}^{\infty} F_{in} \quad (3.24)$$

where

$$F_{in} = e^{-(n\pi/L)^2 Dt} (\cos(n\pi q/L) \cos(n\pi x_i/L) + \sin(n\pi q/L) \sin(n\pi x_i/L)). \quad (3.25)$$

The likelihood for a set of N independent observations x_i on the circle now gives a distribution

$$L(q|x_i, t) = \quad (3.26)$$

$$L(q|x_1, t)L(q|x_2, t) \dots L(q|x_N, t) = \quad (3.27)$$

$$P(x_1|q, t)P(x_2|q, t) \dots P(x_N|q, t). \quad (3.28)$$

When $t \rightarrow \infty$, every distribution for a diffusion process converges to the uniform distribution, in this case $L(q|x, t \rightarrow \infty) = 1/(2L)$. On the circle, the dominant terms for finding the maximum of the distribution when $t \rightarrow \infty$ are A_1 and B_1 . These two coefficients correspond to the functions $\sin(x)$ and $\cos(x)$, which both are eigenfunctions to the Laplacian operator and have the same eigenvalue. It turns out that the diffusion mean converges to the extrinsic mean in the case of a circle. This is not proved here, but it is at least demonstrated experimentally in figures 3.2 – 3.5.

3.7 A Final Remark

While the results in this section are preliminary, there is still a point in mentioning that it seems like these results are valid not only for the circle but also for a larger

class of compact symmetric spaces such as the torus, the sphere and possibly more. In particular, if future research shows that it is valid for the Q embedding of $SO(3)$, we have an excellent motivation for the representation proposed in chapter 10.

For this reason, the main contribution of this section is the notion of the *diffusion mean* and its connection to manifold-valued signal processing. A full set of rigorous proofs for the behavior of this technique, in different manifolds and for different values of t , is the topic of future research. While the idea is quite simple, it would not be surprising if other people have come to similar conclusions regarding its asymptotic properties, in this or similar contexts.

4

Dimensionality Reduction and Manifold Learning

Visualization, processing and analysis of high-dimensional data such as images often requires some kind of preprocessing to reduce the dimensionality of the data and find a mapping from the original representation to a low-dimensional vector space. The assumption is that the original data resides in a low-dimensional subspace or manifold, embedded in the original space. This topic of research is called dimensionality reduction, non-linear dimensionality reduction or more recently manifold learning.

The class of methods for dimension reduction and manifold learning is quite broad and the criteria for finding a low-dimensional parameterization varies. One of the most well known algorithms is PCA, Principal Components Analysis, which projects data on a n -dimensional linear subspace which maximizes the variance of the data in the new space.



Figure 4.1: *Left – Right: A 1-D manifold embedded in \mathbb{R}^2 . A 1-D manifold immersed in \mathbb{R}^2 . The torus, a 2-D manifold embedded in \mathbb{R}^3 . Boy’s surface, an immersion of the projective plane \mathbb{RP}^2 in \mathbb{R}^3 .*

If the original data points lie on a manifold, the mapping to a new space may give an embedding or an immersion of the original manifold. In differential geometry, an immersion corresponds to a smooth mapping $f(x)$ for which the differential of $f(x)$, $d_x f(x): T_p(M) \rightarrow T_{f(p)}(N)$, is non-singular and injective. When the mapping $f(x)$ itself is also injective, it corresponds to an embedding. An example of an embedding is the mapping of a set of pictures (high-dimensional) of a clock to a representation on the unit circle in \mathbb{R}^2 . An immersion could then be a mapping to a curve in \mathbb{R}^2 shaped like the figure “8”.

4.1 Dimensionality Reduction

Linear methods for dimensionality reduction is a rather mature area of research, starting with PCA, Principal Components Analysis (Pearson, 1901) a.k.a. the Hotelling transform (Hotelling, 1933) and the Karhunen-Loëve Transform (Karhunen, 1947). Variants of PCA include generalizations such as Empirical Orthogonal Functions (Lorentz, 1956) and Kernel Principal Components Analysis (Schölkopf et al., 1998). See figure 4.2 for a schematic view of linear methods for dimension reduction.

The basic idea in PCA is to find a projection of the data that maximizes variance. For a set of vectors $\mathbf{x}_i \in \mathbb{R}^N$, this can be done by the following procedure.

1. Calculate the $N \times 1$ sample mean vector, $\mathbf{u} = \frac{1}{M} \sum_{i=1}^M \mathbf{x}_i$.
2. Subtract mean from the data points $\tilde{\mathbf{x}}_i = \mathbf{x}_i - \mathbf{u}$
3. Organize $\tilde{\mathbf{x}}_i$ into a $N \times M$ matrix \mathbf{X} .
4. Create the sample covariance matrix $\mathbf{C} = \frac{1}{M-1} \tilde{\mathbf{X}} \tilde{\mathbf{X}}^T$.
5. Calculate the K largest eigenvalues of \mathbf{C} and store the corresponding eigenvectors in a $N \times K$ matrix called \mathbf{W} .
6. Projections on the PCA basis may now be calculated as $\mathbf{y}_i = \mathbf{W}^T (\mathbf{x}_i - \mathbf{u})$.

PCA has been widely used; “eigenfaces” (Turk and Pentland, 1991) is one of the more well known applications where it is used to create a low-dimensional linear subspace describing variations in images of human faces. The Karhunen-Loëve transform is also known to be useful to create natural basis functions for image compression in general.

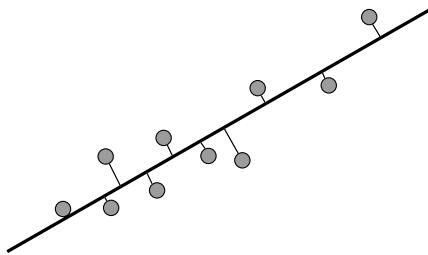


Figure 4.2: A schematic view of the fitting of 1-D linear model to a set of data points embedded in 2-D.

Another well known linear method to find embeddings or immersions of data points, possibly sampled from a manifold, is Multidimensional Scaling (MDS) (Torgerson, 1952; Young and Householder, 1938). Instead of preserving variance in the projection, it strives to preserve all pairwise distances during the projection. Similar to PCA, the basic variant of Multidimensional Scaling is possible to calculate by solving an eigenvalue problem. This is attractive since eigenvalue

problems are optimization problems for which efficient and globally convergent algorithms exist. The classic MDS is stated as a minimization problem of finding new low-dimensional coordinates \mathbf{y}_i for the dataset \mathbf{x}_i given all pairwise Euclidean distances $d(\mathbf{x}_i, \mathbf{x}_j)$. The solution, up to a rotation, is given by

$$\{\mathbf{y}_i\} = \arg \min_{\{\mathbf{y}_i\}} \sum_{i=1}^M (d(\mathbf{x}_i, \mathbf{x}_j)^2 - \|\mathbf{y}_i - \mathbf{y}_j\|^2)^2 \quad (4.1)$$

Important to note is that classical MDS works with quadratic distances, which might seem unnatural but makes it possible to solve the minimization problem by the solution of an eigenvalue problem. If distances correspond to Euclidean distances, classical MDS is equivalent to PCA.

Variants of MDS include non-metric Multidimensional Scaling and weighted MDS. In weighted MDS the objective function is replaced by

$$\arg \min_{\{\mathbf{y}_i\}} \sum_{i=1}^M w_{ij} (d(\mathbf{x}_i, \mathbf{x}_j) - \|\mathbf{y}_i - \mathbf{y}_j\|)^2. \quad (4.2)$$

This objective function differs from classical MDS. It does not fit *squared* distances. As a consequence, this objective function might have several local minima and eigen-decomposition cannot be used to solve the problem in one step. Therefore some strategy for coping with local minima should be employed in the numerical minimization procedure. The benefit of weighted MDS is that uncertainty and missing data can be modeled using appropriate weights.

Other important linear projections of data in vector spaces include Projection Pursuit (Friedman and Tukey, 1974) and Independent Component Analysis (Jutten and Herault, 1991). A well known related example for non-metric data is Latent Semantic Indexing or LSI (Berry et al., 1995). LSI maps document-vectors, describing the occurrences of words in documents, to a low-dimensional vector space.

4.2 Manifold Learning

Recently there has been a great interest in methods for parameterization of data using low-dimensional manifolds as models. Within the neural information processing community this has become known as *manifold learning*. Methods for manifold learning are able to find non-linear manifold parameterizations of data-points residing in high-dimensional spaces, very much like Principal Component Analysis (PCA) is able to learn or identify the most important linear subspace of a set of data points. In two often cited articles in Science, Roweis and Saul introduced the concept of Locally Linear Embedding (Roweis and Saul, 2000) and Tenenbaum et al. introduced the so called Isomap (Tenenbaum et al., 2000).

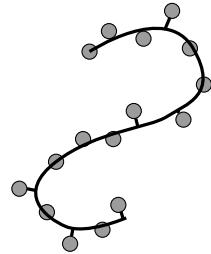


Figure 4.3: A schematic view of the fitting of 1-D non-linear manifold to a set of data points embedded in 2-D.

This seems to have been the start of the most recent wave of interest in manifold learning.

Early work was done by Kohonen with the so called Self-Organizing Maps (SOM) (Kohonen, 1982), in which a topologically constrained manifold is modeled by a grid of points, usually rectangular, which are fitted to the data set. This work was later improved in the Generative Topographic Map (GTM) (Bishop et al., 1998). Bregler and Omohundro were also early in adopting the view of data as points on a non-linear manifold in a vector space, modeling the manifold of lip images (Bregler and Omohundro, 1994). A non-linear variant of PCA, called Kernel Principal Components Analysis (KPCA) (Schölkopf et al., 1998), has also been introduced. In KPCA the input vectors are mapped to a new feature space before applying PCA, a procedure which due to mathematical properties is possible to solve mostly using linear methods. Later, contemporary with Isomap and LLE, Belkin and Niyogi described how approximations to the Laplacian operator and heat equation defined on the manifold (Belkin and Niyogi, 2002) may be used to perform manifold learning by so called Laplacian Eigenmaps (LE).

4.3 Laplacian Eigenmaps

As an example of a method for manifold learning, we first mention Laplacian Eigenmaps (Belkin and Niyogi, 2002). The basic algorithm consists of three steps:

1. First a graph is constructed where each node corresponds to a data point \mathbf{x}_i . Edges are created to each of the K nearest neighbors of \mathbf{x}_i . See figure 4.4.
2. Weights are then assigned to each edge in the graph, for instance using a Gaussian kernel to give strong weight to edges connecting data points which are close in the original space. The weights are collected in a matrix W_{ij} .
3. To find a low-dimensional embedding $\{\mathbf{y}_i\}$ corresponding to $\{\mathbf{x}_i\}$, define an objective function V which has a low value when nodes with a strong

edge are mapped close to each other.

$$V(\{\mathbf{y}_i\}) = \frac{1}{2} \sum_{i,j} ||\mathbf{y}_i - \mathbf{y}_j||^2 W_{ij} \quad (4.3)$$

Define a diagonal matrix D , such that $D_{ii} = \sum_j W_{ij}$ and the Laplacian matrix $L = D - W$. If Y gives the m -dimensional coordinates of \mathbf{y}_i on the i th row of Y , and the constraint $Y^T D Y = I$ is added, the Laplacian eigenmap of dimension m is now found by the solution of the eigenvalue problem $L\mathbf{v} = \lambda D\mathbf{v}$. If the eigenvectors $\{\mathbf{v}^{(0)}, \mathbf{v}^{(1)}, \dots, \mathbf{v}^{(N-1)}\}$ are ordered after the size of the eigenvalues, the first being the smallest (actually equal to 0), then $\hat{Y} = (\mathbf{v}^{(1)}, \mathbf{v}^{(2)}, \dots, \mathbf{v}^{(m)})$ gives the solution for the optimal embedding, minimizing the value of V .

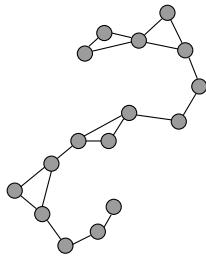


Figure 4.4: A schematic view of the formation of a graph by connecting nearby samples.

The Laplacian Eigenmaps is sometimes referred to as a local method for manifold learning, meaning that it is an attempt to preserve local geometrical properties in the mapping to a low-dimensional space (de Silva and Tenenbaum, 2002).

4.4 Isomap – Isometric feature mapping

An example of a global method for manifold learning is Isomap (Tenenbaum et al., 2000). It tries to preserve the geometry of the data manifold in all scales, mapping nearby points to nearby points and faraway points to faraway points (de Silva and Tenenbaum, 2002). The basic steps of the algorithm are:

1. Create a neighborhood graph G for the dataset $\{\mathbf{x}_i\}$, based for instance on the K nearest neighbors of each point \mathbf{x}_i .
2. For every pair of nodes in the graph, compute the shortest path as an estimate of intrinsic distance within the data manifold. The edges of the graph are weighted according to the Euclidean distance between the corresponding data points.
3. Use the intrinsic distance estimates as input to classical MDS and find an optimal m -dimensional embedding $\{\mathbf{y}_i\}$.

The convergence properties of the estimation procedure for the intrinsic distances is further described in Bernstein et al. (2000).

Computing $N \times N$ pairwise distances is a computationally heavy operation, and so is solving a large eigenvalue problem. In comparison to for instance Laplacian Eigenmaps, the eigenvalue problem in Isomap is not sparse. A variation of Isomap is the L-Isomap, based on the so called Landmark MDS method. It works by first calculating the Isomap embedding for n points, the landmarks, selected *at random*. Then the solution for the rest of the points are computed by an interpolation technique similar to triangulation. This technique is also very similar to the proposed method for calculating the sample logmap, and even though the two approaches are different in philosophy, they share some obvious similarities. The interpolation procedure is the following for a point \mathbf{x}_i which is not a landmark. Let the m -dimensional landmark coordinates be column vectors in a $m \times n$ matrix L . Let Δ_n be the squared distance matrix for all pairs of landmarks and $\overline{\Delta}_n$ the column mean of Δ_n . Let Δ_i be a column vector of all squared distances from \mathbf{x}_i to the landmarks. Also assume that the landmarks are centered. Then the interpolated coordinate is given by

$$\mathbf{y}_i = \frac{1}{2} L^\dagger (\overline{\Delta}_n - \Delta_i) \quad (4.4)$$

where \dagger denotes the Moore-Penrose pseudoinverse. This is basically an estimate of $-1/2$ times the derivative of the squared distance function to \mathbf{x}_i , evaluated at the origin.

4.5 Sample Logmaps

The *sample logmaps* presented in chapter 11 are related to the Isomap algorithm and in particular the variation called Landmark-Isomap or L-Isomap. However, the goal of the *sample logmaps* is fundamentally different in philosophy and the *sample logmaps* should not be seen as variations of Isomap. The *sample logmaps* try to approximate the well known Log map on a manifold, while Isomap is a unique mapping in its own right, trying to preserve all pairwise distances during the mapping.

The function $\log_p(x)$ in a manifold is a mathematically well defined function, which maps points x on the manifold to the tangent space in p , $T_p M$. It is the inverse of the exponential function, $\exp_p \mathbf{x}$, which maps a vector $\mathbf{x} \in T_p M$ to points on the manifold.

One way to see how the logmap may be estimated is to consider some results related to how the intrinsic mean is computed (Karcher, 1977; Fletcher et al., 2004). Let $\{x_i\}$ be N data points in a manifold M and seek the minimizer to the

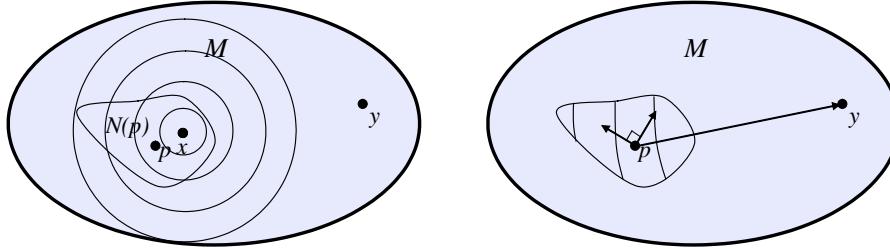


Figure 4.5: A schematic view of the estimated logmap procedure. The manifold is denoted M while X corresponds to a set of samples from M . **Left:** 1. All distances $d(x, y)$, $(x, y) \in (N(p) \times X)$ are calculated. **Right:** 2. For each $y \in X$, gradients at p are calculated using the information from the previous step.

function

$$f(x) = \frac{1}{2N} \sum_{i=1}^N d^2(x, x_i), \quad (4.5)$$

where $d^2(x, y)$ is the squared intrinsic distance between points x and y . It is then shown in Karcher (1977), under the appropriate assumptions of convexity, that the gradient of f is

$$\nabla f(x) = -\frac{1}{N} \sum_{i=1}^N \log_x x_i. \quad (4.6)$$

From this we directly see that for $N = 1$, we have

$$\log_p(y) = -\frac{1}{2} \nabla_p d^2(p, y) \quad (4.7)$$

which shows that the logmap of x_1 calculated at the base point x , is precisely $-1/2$ times the gradient of the squared distance function. The approach for calculating *sample logmaps* in this thesis is based on this result, the squared distance function is calculated using the estimate of intrinsic distances proposed in (Tenenbaum et al., 2000; Bernstein et al., 2000) and the gradient is calculated numerically. It is also proposed to use a robust method to calculate the gradient, while the squared distance function is not smooth for points close to the *cut locus* of the base point x .

The steps in the *sample logmap* algorithm for a single point y are as follows:

1. Select a ball of points around a base point p , $B(p)$.
2. Calculate the coordinates of all points in $B(p)$ in an ON-basis.
3. Estimate the distances from a point y in the manifold to all points in $B(p)$.

4. Estimate the gradient in p of the squared distance function of y , for instance using a non-robust approach such as the one used in L-Isomap.

The similarities with the L-Isomap algorithm also makes it tempting to formulate the *sample logmap* estimation as a minimization problem. The exact details are left for future research, but it is reasonable to assume that *sample logmaps* can be formulated as minimizers of

$$V(\{\mathbf{y}_i\}) = \sum_{i \in I(B(p))} \sum_{j \in I(X)} (d^2(x_i, x_j) - \|\mathbf{y}_i - \mathbf{y}_j\|^2)^2, \quad (4.8)$$

where $I(x)$ is a function which maps points to indices and $d(x, y)$ is the Riemannian metric on M . The way we “solve” this minimization problem at the moment is to first find an embedding of all points in $B(p)$ and then use interpolation to find the optimal coordinates for the rest of points. Considering the structure of the objective function above, the sample logmaps may be seen as a *local-global* approach to manifold learning.

The logmap, $\log_p(x)$ may produce a very distorted mapping of the manifold for points x faraway from p , if the manifold is intrinsically curved. Distances and angles measured at p are however perfectly preserved and geodesics through p are mapped to straight lines. In this aspect the logmap is equal to the map projection called Azimuthal Equidistant Projection for mapping of the Earth, a map projection often used in radio communications, which also happens to be the projection of the earth used in the flag of the United Nations (see figure 4.6).



Figure 4.6: The flag of the United Nations was adopted on October 20, 1947. The emblem of the flag contains an azimuthal equidistant projection of a world map, centered at the North Pole (Wikipedia, 2005). Image from the Open Clip Art Library, <http://www.openclipart.org>.

One goal in manifold-valued signal processing is to represent the signal processing algorithms in a coordinate free way. This means that the operations has a geometric, intrinsic meaning, not relying on any particular coordinate system. For example, using coordinate free methods, operations can be defined on the whole of \mathbb{S}^2 while any coordinate description must have coordinate singularities. In a way, coordinate free approaches actually points towards not using manifold learning to find a low-dimensional parameterization of the data manifold, but instead perform all data- and signal processing intrinsically in the manifold.

Subtraction	$\vec{xy} = y - x$	$\vec{xy} = \log_x(y)$
Addition	$y = x + \vec{xy}$	$y = \exp_x(\vec{xy})$
Distance	$dist(x, y) = y - x $	$dist(x, y) = \vec{xy} _X$
Mean value (implicit)	$\sum_i \vec{x}_i = 0$	$\sum_i \log_{\bar{x}}(x_i) = 0$
Gradient descent	$x_{t+\epsilon} = x_t - \epsilon \nabla C(x_t)$	$x_{t+\epsilon} = \exp_{x_t}(-\epsilon \nabla C(x_t))$
Linear interpolation	$x(t) = x_1 + t \vec{x}_1 \vec{x}_2$	$x(t) = \exp_x(t \vec{x}_1 \vec{x}_2)$

Table 4.1: In (Pennec et al., 2005) the above analogies are made between operations in vector spaces and manifolds.

As pointed out in (Pennec et al., 2005)

“the implementation of \log_x and \exp_x is the basis of any programming on Riemannian manifolds”.

Using *sample logmaps* we now have the basic building block for performing these calculations for signal and data processing in sampled manifolds. Even the \exp_p function is easily evaluated when the manifold is flattened in the point p by interpolation in the resulting mapping. In table 4.5, which is reproduced from (Pennec et al., 2005), some basic operations in vector spaces are compared to analogous operations in manifolds.

4.6 Some Concluding Remarks

A full review of dimension reduction and manifold learning is out of scope for this thesis. The activity in this field is increasing and the following list is a summary which may also serve as a timeline.

- Principal Components Analysis, PCA (Pearson, 1901; Hotelling, 1933; Karhunen, 1947).
- Multidimensional Scaling, MDS (Young and Householder, 1938; Torger-son, 1952)
- Empirical Orthogonal Functions, EOF (Lorentz, 1956)
- Projection Pursuit, PP (Friedman and Tukey, 1974)
- Self Organizing Maps, SOM (Kohonen, 1982)
- Principal Curves (Hastie and Stuetzle, 1989)
- Independent Component Analysis, ICA (Jutten and Herault, 1991).
- Surface Learning with Applications to Lip Reading (Bregler and Omohun-dro, 1994)
- Curvilinear Component Analysis, CCA (Demartines and Herault, 1997)

- Generative Topographic Mapping (Bishop et al., 1998)
- Kernel Principal Components Analysis, KPCA (Schölkopf et al., 1998)
- Isometric feature mapping, Isomap (Tenenbaum et al., 2000) and C-Isomap and L-Isomap (de Silva and Tenenbaum, 2002).
- Locally Linear Embedding, LLE (Roweis and Saul, 2000)
- Laplacian Eigenmaps, LE (Belkin and Niyogi, 2002)
- Local Tangent Space Alignment, LTSA (Zhang and Zha, 2002)
- Hessian Eigenmaps, HLLE (Donoho and Grimes, 2003)
- Diffusion Maps (Nadler et al., 2006)
- Relational Perspective Map, RPM (Li, 2004)
- Sample Logmaps (Brun et al., 2005b)

In general, linear methods for dimension reduction are more stable and more mature. Principal Components Analysis and Multidimensional Scaling are still very popular and have the advantage of being able to learn meaningful relations from few samples. Some of the oldest methods for manifold learning, such as the Self Organizing Feature Maps, have also been used in many applications and may be considered as mature from an application point of view. The more recent methods for manifold learning have mainly two advantages: they are 1) based on global optimization and the solution of eigenvalue problems (unlike SOMs which are sensitive to local minima in the objective function) and 2) they have shown to be efficient for difficult datasets, such as the “Swiss roll” (Tenenbaum et al., 2000; Roweis and Saul, 2000) dataset, where linear methods such as PCA and MDS fail.

Also, while not mentioned here, a lot of work has been done in related topics. One important method for dimensionality reduction is Canonical Correlation Analysis, which finds meaningful relationships and performs dimension reduction between paired datasets. It has also been used to find manifold representations, for instance in image analysis Knutsson et al. (1998); Knutsson and Borga (1999); Knutsson et al. (2000). Another related topic is learning functions on manifolds (Landelius and Knutsson, 1993; Bregler and Omohundro, 1994; Landelius, 1997).

5

Diffusion Tensor Magnetic Resonance Imaging

5.1 Diffusion Imaging

In the physical world, diffusion is the collective process of random motion of particles in a solution or gas. On a macroscopic scale this phenomenon is visible to the eye, for instance by adding a drop of ink to a glass of water and watching it dissolve. The process, also known as Brownian motion, was named after the Scottish botanist Robert Brown who observed the random motion of individual plant spores in a water solution using a microscope. In 1905 Albert Einstein presented a theoretical analysis of Brownian motion and linked it to the Boltzmann constant.

Today diffusion processes are fundamental for the understanding of both physics and mathematics. In Magnetic Resonance Imaging, MRI, it is possible to measure and visualize diffusion of water molecules inside living organisms. The advent of this technology named Diffusion Weighted MRI has today become clinical practice for the diagnosis of for instance stroke. More recent methods, such as Diffusion Tensor MRI combined with so called fiber tractography, are able to *in vivo* infer the anatomy and connectivity of white matter in the human brain. The usefulness of this, for morphological or functional studies of the brain or perform surgical planning prior to the removal of a tumor, is evident.

5.1.1 Diffusion

To get some intuition on diffusion processes, consider the following example of coin flipping.

Let two players, player A and player B, flip a coin. If heads come up, player B gives one dollar to player A. If tails come up, A gives one dollar to B. Call the profit for player A after n turns $a(n) \in [-n, n]$ and let $a(0) = 0$. Each turn of the game, $a(n + 1)$ is either $a(n) + 1$ or $a(n) - 1$, and the variable $a(n)$ perform a random walk in \mathbb{Z} . Whether A or B is the winner after n turns in a particular

game is impossible to say from the beginning, but the variance, $Var(a(n)) = E\{a(n)^2\}$, after many games lasting for n turns is easy to calculate. The variances of n independent variables, each with variance 1, is n . Thus $Var(a(n)) = n$, meaning that the variance of the profit is growing linearly with the respect to the number of turns in the game.

The Diffusion Coefficient

Translating the example of coin flipping to particles performing a random walk in discrete time in one dimension, the variance is growing linearly if the jumps of the particle are according to a set of independent and identically distributed (i.i.d.) variables. Generalizing to continuous time, a natural physical unit to measure the strength of diffusion is m^2/s .

Diffusion in a 3-D isotropic medium is in a similar way characterized by the diffusion coefficient, c . The variance of the distance, $|\mathbf{r}|$, a particle moves by a random walk during time t is $Var(|\mathbf{r}|) = 6ct$. Looking at the individual dimensions, we have $Var(r_x) = Var(r_y) = Var(r_z) = 2ct$.

The diffusion tensor is a generalization of c to account for anisotropic diffusion in three dimensions. It is defined as $\mathbf{D} = \frac{Var(\mathbf{r})}{2t} = \frac{E\{\mathbf{rr}^T\}}{2t}$. Similar to the variance, it is a second order contravariant tensor, described by a symmetric positive semidefinite 3×3 -matrix. Using \mathbf{D} , we may measure the diffusion coefficient along a particular direction $\hat{\mathbf{g}}$ by the formula $c(\hat{\mathbf{g}}) = \hat{\mathbf{g}}^T \mathbf{D} \hat{\mathbf{g}}$. In an isotropic medium the diffusion tensor (in a ON basis) simply becomes

$$\mathbf{D} = \begin{pmatrix} c & 0 & 0 \\ 0 & c & 0 \\ 0 & 0 & c \end{pmatrix} \quad (5.1)$$

The Apparent Diffusion Coefficient

The diffusion coefficient and the diffusion tensor both describe the behavior of unrestricted diffusion. For water molecules in biological tissue, the diffusion is often restricted by for instance cell membranes. For short time intervals, the diffusion of a single molecule is governed by the diffusion tensor or the diffusion coefficient. On a larger time scale however, collisions with boundaries of various kinds will restrict diffusion. This will affect the measurement of diffusion and the term apparent diffusion coefficient (ADC) is used instead.

5.1.2 Estimating Diffusion Tensors

Using diffusion weighted MRI, it is possible to measure the apparent diffusion coefficient in different directions. The Stejskal-Tanner equation relates measure-

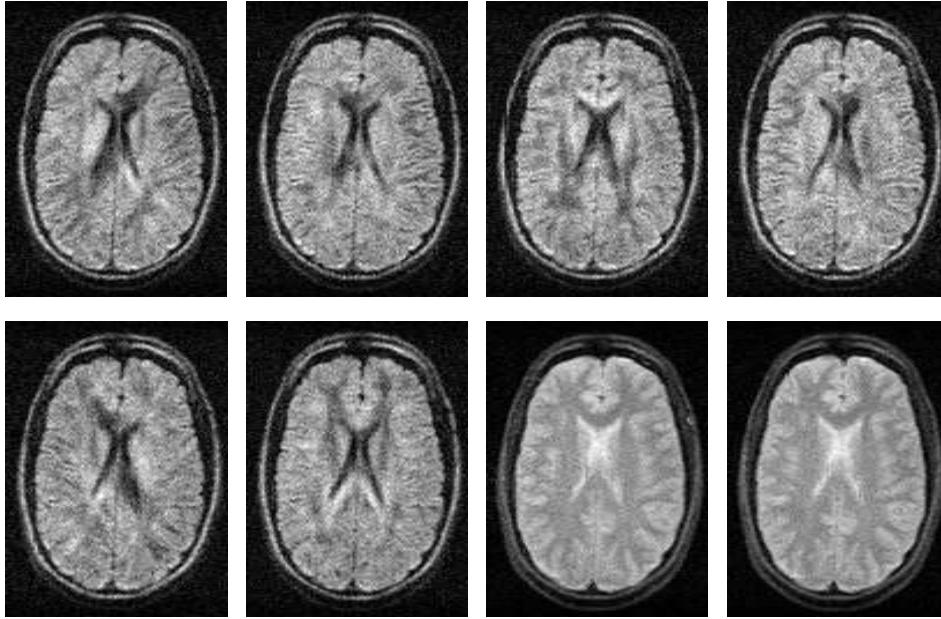


Figure 5.1: A total of eight axial slices of a human brain have been acquired to calculate one slice of diffusion tensors. The six first images are diffusion weighted and have been collected with non-zero gradients in six different gradient directions $\hat{\mathbf{g}}$. The two last images have been collected with zero gradients, $\mathbf{g} = 0$.

ments to ADC values:

$$S_k = S_0 e^{-\gamma^2 \delta^2 [\Delta - (\delta/3)] c} \quad (5.2)$$

A generalization to diffusion tensors \mathbf{D} and gradient directions $\hat{\mathbf{g}}$ is straight forward.

$$S_k = S_0 e^{-\gamma^2 \delta^2 [\Delta - (\delta/3)] \mathbf{g}^T \mathbf{D} \mathbf{g}} \quad (5.3)$$

In the equation above, γ is the proton gyromagnetic ratio (43MHz / Tesla), and \mathbf{g} is the gradient field vector, δ is the duration of the diffusion gradient pulses and Δ is the time between the diffusion gradient RF pulses. The value S_k refers to the measured signal, attenuated by diffusion, and S_0 is the corresponding value obtained when the diffusion gradient strength is zero.

Estimation of D from a series of diffusion weighted measurements is possible, either using a least squares approach (Westin et al., 2002) or using statistical methods. The unknown values are S_0 and \mathbf{D} , containing in total 7 degrees of freedom (due to the symmetry of \mathbf{D}). See figure 5.1 for a set of 8 images used in DT-MRI (two of the images are averaged before estimation begins). The measurements S_k will be affected by Rician distributed noise (Gudbjartsson and Patz, 1995) from the MRI acquisition process.

Eigenvalues of \mathbf{D} ($10^{-6}\text{mm}^2/\text{s}$)	Pyramidal tract (WM)	Splenium of the corpus callosum (WM)	Optic radiation (WM)	Caudate Nucleus (GM)	Cerebrospinal fluid (CSF)
λ_1	1,708 \pm 131	1,685 \pm 121	1,460 \pm 75	783 \pm 55	3,600 \pm 235
λ_2	303 \pm 71	287 \pm 71	496 \pm 59	655 \pm 28	3,131 \pm 144
λ_3	114 \pm 12	109 \pm 26	213 \pm 67	558 \pm 17	2,932 \pm 212

Table 5.1: Typical ADC values found in the human brain, measured in the orientations of the three eigenvectors of \mathbf{D} (Pierpaoli et al., 1996).

5.1.3 Diffusion in the Human Brain

Inside the human brain, the apparent diffusion properties will vary depending of the type of tissue. In table 5.1 some values of ADC is measured for various tissues. The different eigenvalues mentioned will be explained in more detail below, but refers to the fact that diffusion varies in different directions – the diffusion tensor \mathbf{D} is anisotropic – for certain types of tissue, in particular inside white matter (WM).

Close to fiber structures in the brain, the diffusion of water molecules is restricted. The variance of the random walk is attenuated in directions perpendicular to the fibers, while the movement along the fibers is similar to free diffusion. The anisotropy of the apparent diffusion is captured in the diffusion tensor. By studying the main direction of diffusion, derived from the eigenvalues and eigenvectors of the diffusion tensor, it is possible to infer the orientation of fibers going through a voxel. This forms the basis for fiber tracking. Studying the degree of anisotropy of a diffusion tensor also give a lot of information about the organization of tissue within that specific voxel.

5.1.4 Applications of DT-MRI

The applications of DT-MRI in a clinical setting include examples of both quantitative and qualitative methods.

Surgical Planning

During surgical planning involving the brain, knowledge of the location of important fiber bundles may guide the surgeon to avoid damage on important functional parts of the brain. This is particularly important when planning the removal of tumors, while fiber tracts may have been distorted by the growth of the tumor so that experience and prior knowledge of fiber bundles are of little importance in the case at hand.

Clinical Studies

For morphological and functional studies of the human brain, in both healthy populations and patients, diffusion tensor MRI can be useful to derive knowledge related to white matter variations and abnormalities. This includes for instance studies on Schizophrenia and Multiple Sclerosis. With DT-MRI it is also possible to perform non-invasive and harmless experiments on human subjects to find out about pathways in the brain, and confirm hypotheses about the human brain derived from invasive and dangerous studies previously only performed on animals and in particular monkeys.

5.2 Processing Diffusion Tensor Data

Processing and analysis of tensor-valued data in image volumes requires a treatment different from that of scalar data. While image processing for tensor images was available prior to the introduction of DT-MRI, see for instance (Knutsson, 1989; Granlund and Knutsson, 1995), the recent advances in acquisition of tensor-valued data in medicine (Westin et al., 2002) has made this field of research popular again.

5.2.1 Scalar Invariants

Tensors and tensor volumes are more difficult to visualize and analyze than scalars and scalar-valued volumes. For this reason, methods for calculating scalar values derived from tensors are important, in particular methods which yields scalars that are invariant to rotations of the coordinate frame. Three important invariants are the trace, fractional anisotropy and the shape classification of tensors by Westin.

Trace

The trace of the tensor is defined

$$Tr(\mathbf{D}) = \sum_{i=1}^n \mathbf{D}^i_i \quad (5.4)$$

For a mixed second order tensor, the trace is a scalar which is invariant to changes of basis and thereby invariant to rotations. While the diffusion tensor is a contravariant tensor, \mathbf{D}^{ij} , and the trace is only defined for mixed tensors, it is necessary to first transform the diffusion tensor \mathbf{D}^{ij} to a mixed tensor $\mathbf{D}^i_j = \mathbf{D}^{ik}g_{kj}$. Using the trace, a mean diffusion coefficient can be calculated using

$$\bar{c} = \frac{1}{3}Tr(\mathbf{D}^i_j) = \frac{1}{3}Tr(\mathbf{D}^{ik}g_{jk}) = \frac{1}{3}\mathbf{D}^{ik}g_{ik} = \frac{1}{3}\sum_{i=1}^n \sum_{k=1}^n \mathbf{D}^{ik}g_{ik} \quad (5.5)$$

This scalar invariant is formed by letting the metric tensor operate on the diffusion tensor. It is thus dependent of the choice of unit used to define the metric, i.e. whether length one represents one meter, one centimeter or one foot. In most context related to diffusion tensor imaging one simply speaks of the trace of the tensor, indirectly assuming that the tensor is expressed in an ON-basis for which the metric tensor is the identity matrix.

If the eigenvalue equation

$$\mathbf{D}_j^i \mathbf{x}^j = \lambda \mathbf{x}^i \quad (5.6)$$

has $n = \dim V$ non-trivial solutions with corresponding linearly independent eigenvectors \mathbf{e}_i with eigenvalues λ_i , the matrix \mathbf{D}_j^i may be decomposed according to the eigen decomposition theorem as

$$\mathbf{D}_j^i = (\mathbf{PWP}^{-1})_j^i \quad (5.7)$$

where $\mathbf{P} = [\mathbf{e}_1^i, \mathbf{e}_2^i, \dots, \mathbf{e}_n^i]$, $\mathbf{W}_i^j = \lambda_i$ if $i = j$ and $W_i^j = 0$ if $i \neq j$. The eigenvalues may be found by solving the so called characteristic equation

$$\begin{vmatrix} \mathbf{D}_1^1 - \lambda & \mathbf{D}_2^1 & \mathbf{D}_3^1 \\ \mathbf{D}_1^2 & \mathbf{D}_2^2 - \lambda & \mathbf{D}_3^2 \\ \mathbf{D}_1^3 & \mathbf{D}_2^3 & \mathbf{D}_3^3 - \lambda \end{vmatrix} = 0 \quad (5.8)$$

equivalent to

$$A_1 = \mathbf{D}_1^1 + \mathbf{D}_2^2 + \mathbf{D}_3^3 \quad (5.9)$$

$$A_2 = \begin{vmatrix} \mathbf{D}_2^2 & \mathbf{D}_3^3 \\ \mathbf{D}_3^2 & \mathbf{D}_2^3 \end{vmatrix} + \begin{vmatrix} \mathbf{D}_1^1 & \mathbf{D}_2^2 \\ \mathbf{D}_2^1 & \mathbf{D}_1^2 \end{vmatrix} + \begin{vmatrix} \mathbf{D}_1^1 & \mathbf{D}_3^3 \\ \mathbf{D}_3^1 & \mathbf{D}_1^3 \end{vmatrix} \quad (5.10)$$

$$A_3 = \begin{vmatrix} \mathbf{D}_1^1 & \mathbf{D}_2^1 & \mathbf{D}_3^1 \\ \mathbf{D}_2^1 & \mathbf{D}_2^2 & \mathbf{D}_2^3 \\ \mathbf{D}_3^1 & \mathbf{D}_3^2 & \mathbf{D}_3^3 \end{vmatrix} \quad (5.11)$$

$$(5.12)$$

$$\lambda^3 - \lambda^2 A_1 + \lambda A_2 - \lambda A_3 = 0 \quad (5.13)$$

Any invariant which is independent of coordinate system may be written as a function of A_1 , A_2 and A_3 . The left hand side of the last equation is called the characteristic polynomial. Eigenvalues are independent of the choice of coordinate system and for this reason the coefficients in the polynomial are invariant to coordinate changes as well.

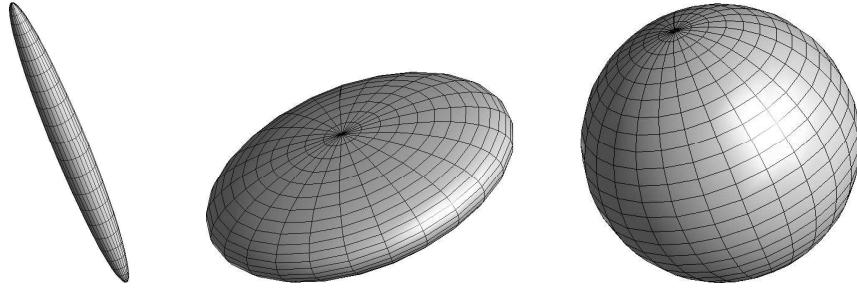


Figure 5.2: Typical linear, planar and spherical tensors.

Fractional Anisotropy

The fractional anisotropy (FA) is a measure explaining how much the norm of the tensor stems from anisotropic contributions.

$$FA = \frac{1}{\sqrt{2}} \frac{\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_1 - \lambda_3)^2}}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \quad (5.14)$$

$$= \frac{\sqrt{3}}{\sqrt{2}} \frac{|\mathbf{D} - \frac{1}{3}Tr(\mathbf{D})\delta_j^i|}{|\mathbf{D}|} \quad (5.15)$$

Due to the properties of the norm and the trace, it is invariant to rotations and scaling. See figure 5.5 for a typical axial slice displayed using FA.

Linear, Planar & Spherical

In (Westin et al., 2002) the following three measures of diffusion tensor shape are defined, corresponding to linear, planar and spherical shape

$$c_l = \frac{\lambda_1 - \lambda_2}{\lambda_1} \quad (5.16)$$

$$c_p = \frac{\lambda_2 - \lambda_3}{\lambda_1} \quad (5.17)$$

$$c_s = \frac{\lambda_3}{\lambda_1} \quad (5.18)$$

See figure 5.2 for an intuitive explanation of the concept.

5.2.2 Fiber Tracking

While scalar invariants have been used widely, both to visualize and obtain quantitative measures of diffusion within the human brain, even more stunning visualizations and analyses of connectivity may be performed using so called fiber tracking algorithms. They release seeds, virtual particles, in the data volume, creating streamlines while following the principal direction of diffusion (PDD). The

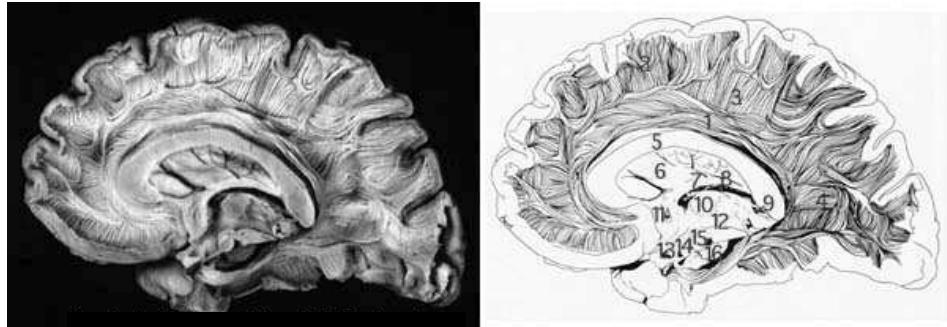


Figure 5.3: A dissection of a real brain showing the structure of white matter (from The Virtual Hospital, University of Iowa).

tracking is usually seeded within white matter and terminates when reaching a gray matter mask or when the FA value becomes too low. See figure 5.6 for an example of fiber tracking.

PDD tracking

The simplest and maybe most widely used kind of fiber tracking is to follow the principal direction of diffusion. Each particle, seeded within white matter, is iteratively propagated along the principal direction of diffusion in the data. Great care should be taken in order to interpolate the tensor field within each voxel in order to obtain smooth fiber traces.

Stochastic Tracking

In stochastic or probabilistic fiber tracking (Brun et al., 2002; Björnemo et al., 2002; Behrens et al., 2003b; Hagmann et al., 2003; Behrens, 2004; Behrens et al., 2003a; Friman and Westin, 2005), particles are propagated in a similar way as in PDD tracking. For each time step, a particle is propagated in a direction taken as a random sample from the estimated probability distribution of the PDD. In this way, uncertainty from the measurements and the model is taken into account. Seeding from a particular voxel A, multiple fiber traces are possible, and a kind of “connectivity estimate” $p(B|A, t)$ may be calculated to measure the proportion of particles starting in a point A and reaching a point B after t time steps.

5.2.3 Fiber Tract Connectivity

Estimation of “connectivity” in the human has been something of a holy grail for the DT-MRI imaging community. Figures 5.3 and 5.4 show a dissection of a real brain, revealing some of the complexity of the human brain white matter architecture. If one can see fiber traces and fiber bundles in DT-MRI and in dissections of real brains, extending the algorithms to give a quantitative measure of connectivity

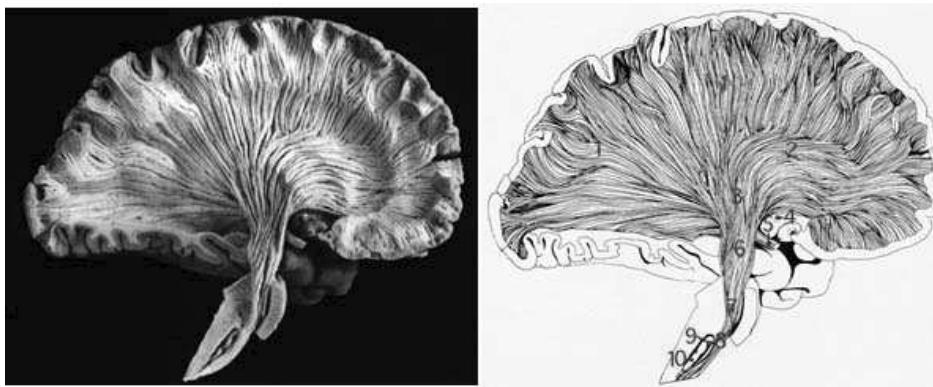


Figure 5.4: A dissection of a real brain showing the structure of white matter (from *The Virtual Hospital*, University of Iowa).

ought to be possible. The probabilistic and stochastic algorithms for fiber tracking give quantitative answers to the question: $p(B|A)$ = “what are the chances of ending of in voxel B if we start in voxel A” but this measure is not the same as $p(A|B)$ which is a somewhat confusing property. Sometimes the connectivity measure is simply made symmetrical by brute force, i.e. $c(A, B) = \frac{1}{2}(p(A|B) + p(B|A))$ (Behrens, 2004).

One way to obtain a symmetric measure of connectivity would be to embed all voxels in a metric space (or even a manifold) in which a short (geodesic) distance $d(A, B)$ means that two points A and B are more connected. In for instance (O'Donnell et al., 2002) the image volume is embedded by warping the metric according to the inverse of diffusion tensors. A problem with this approach could be that the triangle inequality plays a trick. Assume we have three points A, B and C in the brain. A is connected to B and A is also functionally connected to C. However, B and C are not connected at all. The triangle inequality says that $d(B, C) \leq d(A, B) + d(A, C)$ and thus forces the points B and C to be close if A is connected to both B and C.

Apparently some work remains to be done before everybody agree on what kinds of anatomical connectivity there are, to what extent these quantities are possible to measure in DT-MRI and what the exact axiomatic properties, in a mathematical sense, should be for the various kinds of connectivity.

5.2.4 Segmentation of White Matter

Without diffusion weighted imaging, it is difficult to segment fiber bundles in human brain white matter. In other image modalities, voxels within white matter are represented by one single intensity and there is no way to distinguish between different bundles. With DT-MRI on the other hand, voxels in white matter may be segmented depending on what areas of the brain they connect. The same

technique also works for segmenting gray matter into areas related to function (Behrens et al., 2003a).

Virtual dissection (Catani et al., 2002) is one example of how a medical doctor can interactively explore the anatomy of white matter by selecting fiber traces of interest depending on their connectivity. Other examples include automatic Fuzzy C-means (Shimony et al., 2002) clustering and NCut clustering (Brun et al., 2004) of DT-MRI fiber traces.

5.3 Visualization of Streamline Data

The approach for visualization of DT-MRI data, presented in chapter 8 and 9, use methods inspired by dimension reduction and manifold learning in order to enhance the perception of connectivity in DT-MRI data of the human brain. This is different from obtaining quantitative measurements of connectivity and we envision these approaches to be useful for the purpose of interactive visualization and explorative analysis of DT-MRI. The primary goal is to create a visual interface to a complex dataset.

5.3.1 Local and Global Features in DT-MRI

The scalar invariants presented in 5.2.1 are important features of the kind of tensor-valued data obtained from DT-MRI. Using scalar invariants, local features of the data inside a voxel may be visualized using for instance a color map. This is one example of a local feature of the dataset. Other slightly less local features in tensor data include edge information (O'Donnell et al., 2004; Granlund and Knutsson, 1995) For vector-valued velocity data, which is also a kind of tensor data, features based on vortex and convergence/divergence have been proposed (Heiberg, 2001).

Connectivity as a feature

The connectivity of a voxel, for instance defined by streamlines or probabilistic fiber tracking, may also be regarded as a feature of that voxel. This is not a local feature, while the connectivity of one single voxel depends on a spatially distributed set of voxels within the dataset. We call this a macro-feature. Voxels with a similar connectivity profile may be mapped to similar places in a feature space describing connectivity.

Viewing voxels as the atomic unit when visualizing connectivity in DT-MRI is one alternative. The other alternative is to visualize streamlines. The main difference is that a streamline is itself a representation of its connectivity. A streamline also has a simpler connectivity profile, while it connects exactly two endpoints with

each other. A single voxel on the other hand may, through for instance probabilistic fiber tracking, connect to multiple endpoints. One single voxel may also contain several, perhaps crossing, streamlines. This is particularly true if the tracking algorithm or the data is rich enough to cope with crossing fiber bundles.

The shape and position of a streamline reveals its connectivity and in a way also the connectivity of the voxels it goes through. Similar streamlines usually belong to the same fiber bundle

The Fiber Bundle Assumption

Performing fiber tracking can be seen as a kind of feature transform, where the data volume is transformed into a set of feature points. Each voxel inside the white matter in the brain is used for seeding a fiber tracking procedure or performing stochastic fiber tracking. The result is similar to a Hough transform, where each fiber trace is analogous to the line integral of the Hough transform and maps to a specific point in a fiber feature space.

In this fiber feature space we assume there are clusters of points, corresponding to major fiber tracts such as the corpus callosum and the cingulum bundles. These clusters of points live in a high-dimensional space, the fiber feature space, but will intrinsically have only two dimensions corresponding to the cross section of a fiber bundle. Early work on a similar topic may be found in (Westin, 1991).

5.3.2 Learning and Representations

To bring order into the fiber feature space, we propose to utilize methods inspired by dimension reduction, manifold learning and spectral clustering.

Laplacian Eigenmaps

Laplacian Eigenmaps is spectral technique for manifold learning, which maps nearby points on a manifold in a possibly high-dimensional Euclidean feature-space to nearby points a low-dimensional Euclidean space. Using this method, it is possible to map high-dimensional objects such as fiber traces into a 3-dimensional Euclidean space. This mapping is used in this thesis to assign colors to fiber traces, in a way that fiber traces with similar connectivity, shape and position are mapped to similar colors. This greatly enhances the perception of connectivity in the fiber trace dataset.

Normalized Cuts

A recently proposed clustering technique, called Normalized cuts or NCut, make a strong connection between recent spectral methods for manifold learning and certain graph-based methods for data clustering. The eigenvalue problem solved

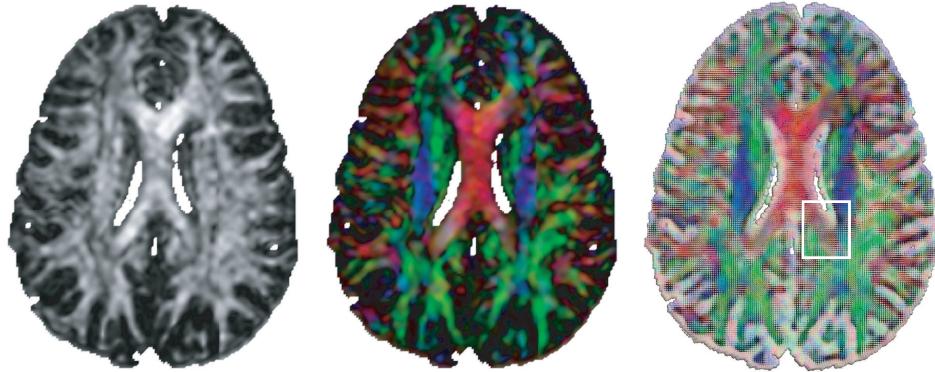


Figure 5.5: An axial slice of a brain. **Left:** Intensity corresponds to fractional anisotropy. **Middle:** Color corresponds to main principal direction of diffusion. Red: left-right, green: anterior-posterior, blue: superior-inferior. **Right:** A rendering using tensor ellipsoid glyphs. Courtesy of Gordon Kindlmann.

in NCut is almost identical to the one solved for creating Laplacian Eigenmaps. Using this method for data clustering, fiber traces are clustered into fiber bundles. In fact any clustering technique would be possible to use for this task, NCut was chosen mainly because of its similarity to the Laplacian Eigenmaps and other spectral methods for manifold learning.

5.3.3 Visualization of Fiber Tract Connectivity

Scalar Invariants

Using the scalar invariants defined in 5.2.1 we may visualize a 2-D slice of a 3-D DT-MRI volume of a human brain. See figure 5.5 for a demonstration of fractional anisotropy.

Glyphs

If the $(2,0)$ or contravariant diffusion tensor is transformed into a $(1,1)$ mixed tensor using the metric g_{ij} , it is possible to interpret it as a linear transformation and a spectral decomposition into eigenvectors and eigenvalues is possible.

In figure 5.6, two variants of tensor glyphs are shown: Ellipsoids and superquadrics (Kindlmann, 2004). Tensor glyphs show the strength, anisotropy and orientation of the diffusion tensors.

Streamlines/Streamtubes

The result of fiber tracking may be visualized using either streamlines or streamtubes. By choosing the appropriate viewpoint, lighting and possibly a selection of

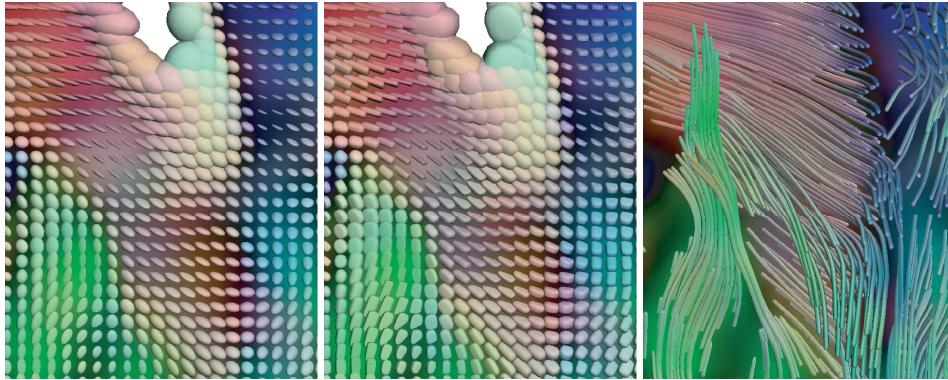


Figure 5.6: A detail of an axial slice of the brain shown in figure 5.5. **Left:** Tensor ellipsoids. **Middle:** Tensor superquadrics (Kindlmann, 2004). **Right:** Streamtubes. Courtesy of Gordon Kindlmann.

a subset of fiber traces to visualize, it is possible to understand the geometry and connectivity of the dataset. See figure 5.6.

Streamtube Coloring

When the set of fiber traces becomes too complex, an enhancement of the perception of connectivity may be created if the fiber traces are colored according to their position, shape and connectivity. Similar colors help the user to mentally group fiber traces into bundles. Fiber traces may also be clustered and colored in very different colors, to emphasize the difference between distinct clusters.

Voxel Coloring

Finally, the result of streamtube coloring may be transformed to voxel space so that each voxel is colored or clustered in the same way as the fiber trace(s) passing through it. Alternatively this may be viewed as coloring each voxel according to its connectivity profile – voxels connecting similar parts of the brain are mapped to similar colors or clusters. See chapter 9 figure 9.4 for a demonstration of the concept.

6

Review of Papers

This chapter provides a short review of the papers included in the second part of this thesis. All papers have been reformatted and minor changes have been made to correct misspellings and typographical errors.

6.1 Paper I: Coloring of DT-MRI Fiber Traces Using Laplacian Eigenmaps

This paper (Brun et al., 2003) was presented in 2003 at the 9th International Workshop on Computer Aided Systems Theory (Eurocast'03) in Las Palmas de Gran Canaria, Spain. The proceedings were published in Springer Lecture Notes in Computer Science. Here the concept of *fiber coloring* or *streamline coloring* is introduced for the first time. The idea is to visualize a set of streamlines, obtained e.g. by tractography in DT-MRI data, by choosing colors for the individual streamlines such that *similar streamlines* are assigned *similar colors*. This greatly enhances the user's perception of *connectivity* within the white matter as well as the separation of fiber traces into *fiber bundles* for the DT-MRI application.

To create the mapping from streamlines to colors, a recently proposed method for manifold learning called Laplacian Eigenmaps (Belkin and Niyogi, 2002) is used. This paper thus shows that dimensionality reduction and manifold learning can be used to solve a medical visualization problem. In principle many other methods for dimensionality reduction could also have been used, for instance Isomap (Tenenbaum et al., 2000) or LLE (Roweis and Saul, 2000).

Since the paper was first presented, the idea of embedding streamlines or fiber traces in a low-dimensional space has been further explored in (O'Donnell and Westin, 2005). Also, the particular measure of fiber similarity described in this paper have been evaluated by others in (Möberts et al., 2005). This method for spectral coloring of fiber traces have been widely appreciated for its aesthetic value. A large poster of colored fiber traces from a human brain has for instance

been displayed for the public at the nationally touring exhibition *Se Hjärnan* (See the Brain) sponsored by Vetenskapsrådet (The Swedish Research Council).

6.2 Paper II: Clustering Fiber Traces using Normalized Cuts

The second paper (Brun et al., 2004) was presented at the Seventh International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'04) in Saint-Malo, France, and published in Springer Lecture Notes in Computer Science. Here a spectral graph theoretical method strongly related to the Laplacian Eigenmaps (Belkin and Niyogi, 2002), called Normalized Cuts (Shi and Malik, 2000) is used to *cluster* fiber trace data in DT-MRI into discrete bundles. Our interest in this method came mainly from the fact that it was highly similar to Laplacian Eigenmaps.

Methods inspired by this paper has already been implemented by other people (Enders et al., 2005), and cited several times (O'Donnell and Westin, 2005; Moberts et al., 2005; Blaas et al., 2005; Enders et al., 2005; Jonasson et al., 2005; Maddah et al., 2005; Kouby et al., 2005; O'Donnell and Westin, 2006). It was the first application of a spectral clustering method to find fiber bundles in Diffusion Tensor MRI fiber trace data. It also featured a novel measure of fiber similarity based on the mean vector and covariance matrix of the points building up the fiber trace.

6.3 Paper III: A Tensor-Like Representation for Averaging, Filtering and Interpolation of 3-D Object Orientation Data

In a paper (Brun et al., 2005a) presented in Genoa at the IEEE International Conference on Image Processing (ICIP'05) an extrinsic method for averaging, filtering and interpolation of data on $SO(3)$ is presented. In the literature, many methods for performing signal processing on $SO(3)$ have been described.

The contribution in this paper is mainly that it points out how algorithms for linear averaging, filtering and interpolation can be directly translated into tools for manifold-valued signal processing, given a suitable extrinsic representation. From the discussion in the previous chapter 3, it is clear that while one common opinion among researchers is that extrinsic methods should be seen as approximations to intrinsic ditto, extrinsic methods may in fact be optimal in some cases.

6.4 Paper IV: Fast Manifold Learning Based on Riemannian Normal Coordinates

In the fourth paper (Brun et al., 2005b), presented at the Scandinavian Conference on Image Analysis (SCIA'05) and published in Springer Lecture Notes in Computer Science, a new kind of method for manifold learning is presented. This method is highly related to differential geometry and the logarithm, the inverse of the exponential map, on the manifold. From an algorithmic point of view this method shares some similarities with Isomap, but is different from most modern methods for manifold learning by not relying on the solution of a large eigenvalue problem. Apart from being useful for visualization and dimensionality reduction of manifold-valued data, the logarithm defined on a manifold is also a fundamental building block in methods for performing *intrinsic* signal processing on manifold valued signals.

The method is called LOGMAP in the paper, but for the rest of this thesis the name *Sample Logmaps* or *S-Logmaps* will be used for the whole class of methods that estimate the log map on a manifold given a set of samples.

7

Discussion

The results presented in this thesis point towards the usefulness of manifolds and manifold learning in image analysis and visualization.

7.1 Manifold Learning in Diffusion Tensor Imaging

The work on Diffusion Tensor MRI should be seen mainly as a proof of concept, indicating that new techniques for nonlinear dimensionality reduction are useful in real applications. It is also important to note that manifold learning gives inspiration to look at data in new ways.

The main contributions in this part of the thesis are:

- The introduction of fiber coloring and voxel coloring, i.e. continuous mapping of position and shape of fiber traces to a color space in order to enhance visualization of connectivity and organization of the white matter in the human brain.
- The first use of the *NCut* criteria to perform fiber trace clustering, and the first use of spectral clustering, to organize fiber traces into fiber bundles. It should be noted that the use of *NCut* for clustering of *voxels* based on connectivity was independently reported in (Behrens, 2004).
- The introduction of two simple and novel ways to measure fiber similarity: The similarity of fiber trace endpoints and the similarity of fiber trace mean vector and covariance matrix.

Mapping of fiber traces to colors, such that *similar fiber traces* are assigned *similar colors*, also works for streamlines in general and the results are therefore not limited to Diffusion Tensor MRI but applicable to all approaches involving streamlines.

So far only Laplacian Eigenmaps have been tested for fiber coloring. It is likely that other methods for manifold learning and even linear dimension reduction

work equally well or even better. The method of *sample logmaps* presented in this thesis however is probably not useful for this particular application, at least not without heavy preprocessing of the data and transformation of fiber traces into a suitable feature space.

While methods such as Laplacian Eigenmaps works fairly well even for data that does not strictly come from a single manifold, but rather from many smaller manifolds scattered in feature space (each corresponding to a fiber bundle), the sample logmaps rely heavily on the assumption of a single manifold. For this reason, the main criticism of the work presented here should be that it exploits the robustness of the Laplacian Eigenmaps, to give reasonable results even for data where the assumption of a single manifold fails.

7.2 Intrinsic vs. Extrinsic Methods for Manifold-Valued Signal Processing

In this part of the thesis, signal processing on a manifold embedded in Euclidean space was explored using the examples of the circle, \mathbb{S}^1 , and the Q -representation for $SO(3)$. Both of these manifolds are examples of symmetric spaces.

The main contributions of the work presented in this part of the thesis are:

- The idea of translating algorithms for linear averaging, filtering and interpolation for 1-D signals to 3-D object orientation data on $SO(3)$. This is not always the best choice, but it may sometimes be convenient from an application point of view.
- The embryo of a motivation for using extrinsic averaging and signal processing in compact symmetric spaces through the *diffusion mean*. This is clearly work that has not been completed, but it is still mentioned in this thesis while the idea might be important for future work. It also serves as a motivation for exploring extrinsic means on $SO(3)$ using the Q -representation.

While our research group has been working with extrinsic signal processing on \mathbb{RP}^n for many years, for instance to represent and filter line- and hyperplane orientations, this piece of research is useful to put some of the earlier work into context.

Future uses of the Q -representation could be for instance to perform template matching in an image volume with a rotating template.

7.3 Sample Logmaps – Intrinsic Processing of Empirical Data

Finally the *sample logmaps* present a novel way to perform nonlinear dimension reduction as well as the beginning of a framework for intrinsic signal processing on manifolds known only from samples.

The main contributions of the work presented in this part of the thesis are:

- A simple and novel way to perform manifold learning and nonlinear dimension reduction using *sample logmaps*.
- A way to translate algorithms for vector spaces to *sampled* manifolds and perform intrinsic manifold-valued signal processing.

Even though the current way of estimating a *sample logmap* could be improved, the theoretical properties of the analytical log map and its importance to manifold-valued signal processing, makes methods for estimation of *sample logmaps* a promising field of research. Important is also to compare the *sample logmaps* with other approaches for manifold learning, a work which has already addressed in part by other researchers in the field (Kayo, 2006).

One last note on sample logmaps is that the terminology used in chapter 11 might not be perfect while “LOGMAP”, “LogMap” or “Logmap” is very similar to “log map” which is usually the analytical log map on the manifold and nothing else. It is more appropriate to talk about “an estimated log map” and name this class of methods “sample logmaps” or “S-Logmaps” for short.

7.4 Future research

The main objective of future research for the PhD thesis is to seek a more unified framework for dealing with manifold-valued data and signals. Here are some directions which could be fruitful:

- Mapping fiber traces to an RGB color space, trying to preserve some kind of shape metric in the RGB space, is far from optimal for human perception. Careful mapping of fiber traces into a CIE Lab or CIE XYZ color system may, at least in theory, create a mapping where perceived color distance corresponds more accurately to distances in the feature space at hand.
- Sampled manifold-valued data is often affected by noise, for instance isotropic Gaussian noise added in the embedding space. This increases the apparent dimensionality of the manifold. It would be useful to have methods to remove this noise from the data before applying manifold learning.
- From a more philosophical point of view, the mapping of fiber traces to a feature space give raise to many new possibilities related to registration

of datasets. One aspect of this has already been explored in (O'Donnell and Westin, 2005) where fiber traces from different brains are mapped to a common feature space using Laplacian Eigenmaps. This allow for new ways to compare connectivity between different brains in a population.

Another aspect related to registration is to actually perform registration of two DT-MRI volumes using features derived from connectivity. Just like scalar invariants may be used to register two DT-MRI datasets, features describing voxel connectivity may also be used to steer the registration. Either features of voxel connectivity which are invariant to rotation or features for which we have transformation laws so that they transform appropriately with the volume. Describing the connectivity of a single voxel by the covariance tensor of the fiber traces passing through the inside of the voxel is one example of a simple but suitable representation for this. It will naturally transform as a contravariant tensor for linear (affine) transformations of the image volume.

- Many things remain to be explored related to connectivity based voxel coloring and direct volume rendering.
- Investigate the properties of the “diffusion mean” in various manifolds.
- Improved estimation of distance functions on sampled manifolds. Apart from the work which is the core of the Isomap algorithm (Bernstein et al., 2000), some recent activity have focused on a more robust estimation of distances on sampled manifolds which are less sensitive to “shortcuts” (Nilsson and Andersson, 2005).
- More applications for manifold learning in medical image analysis.
- The creation of a generic framework for manifold-valued signal-processing, for analytical manifolds as well as sampled and “learned” manifolds.
- Robust gradient Estimation. One of the steps in the current *sample logmap* algorithm is to estimate the gradient. This is difficult in the vicinity of the *cut locus* and there is a clear need for robust methods here.
- Can image processing be applied in manifolds to find edges, corners and other interesting features inside sampled manifolds?
- Are there ways to characterize sampled manifolds with respect to genus and topology?

Inside manifolds, new worlds are waiting to be explored and studied. There is a need for quantitative as well as explorative analysis of manifolds. And there is a great need to standardize methods for working with manifold-valued signal processing.

Coloring of DT-MRI Fiber Traces Using Laplacian Eigenmaps¹

Anders Brun^a, Hae-Jeong Park^b, Hans Knutsson^c and Carl-Fredrik Westin^a

^a Laboratory of Mathematics in Imaging, Brigham and Women's Hospital, Harvard Medical School, Boston, USA, anders@bwh.harvard.edu, westin@bwh.harvard.edu,

^b Clinical Neuroscience Div., Lab. of Neuroscience, Boston VA Health Care System-Brockton Division, Dep. of Psychiatry, Harvard Medical School and Surgical Planning Laboratory, Brigham and Women's Hospital, Harvard Medical School, hjpark@bwh.harvard.edu,

^c Medical Informatics, Linköping University, Inst. för Medicinsk Teknik, Universitetssjukhuset, Linköping, Sweden, knutte@imt.liu.se.

Abstract: We propose a novel post processing method for visualization of fiber traces from DT-MRI data. Using a recently proposed non-linear dimensionality reduction technique, Laplacian eigenmaps (Belkin and Niyogi, 2002), we create a mapping from a set of fiber traces to a low dimensional Euclidean space. Laplacian eigenmaps constructs this mapping so that similar traces are mapped to similar points, given a custom made pairwise similarity measure for fiber traces. We demonstrate that when the low-dimensional space is the RGB color space, this can be used to visualize fiber traces in a way which enhances the perception of fiber bundles and connectivity in the human brain.

8.1 Introduction

Diffusion Tensor MRI (DT-MRI) makes it possible to non-invasively measure water diffusion, in any direction, deep inside tissue. In fibrous tissue such as muscles and human brain white matter, water tend to diffuse less in the directions perpendicular to the fiber structure. This means that despite the fact that spatial resolu-

¹Published in the Proceedings of the 9th International Conference on Computer Aided Systems Theory (EUROCAST'03).

tion in MRI is too low to identify individual muscle fibers or axons, a macroscopic measure of diffusion in a voxel may still reveal information about the fiber structure in it. Using DT-MRI it is therefore possible to infer the direction of the fiber in for instance white matter in the human brain. In particular, it is possible to estimate the direction of the fibers when the fiber organization is coherent within the voxel.

When a whole volume of data is acquired using DT-MRI, each voxel contains information about the local characteristics of diffusion inside that particular voxel. The diffusion is described by a tensor D , a symmetric positive definite 3×3 matrix, which through the Stejskal-Tanner equation (8.1) explains the measurements obtained from the MR scanner

$$S_k = S_0 e^{-b\hat{g}_k^T D \hat{g}_k}. \quad (8.1)$$

Here \hat{g}_k is a normalized vector describing the direction of the diffusion-sensitizing pulse, b is the diffusion weighting factor (Bihan et al., 1986) and S_0 is a non-diffusion weighted measure. In order to estimate a tensor D inside each voxel, at least one non-diffusion weighted image S_0 and six diffusion weighted images with different directions are needed (Westin et al., 1999). The product $\hat{g}_k^T D \hat{g}_k$ is often referred to as the Apparent Diffusion Coefficient, ADC, and describes the amount of diffusion in the gradient direction.

The tensor can be visualized as an ellipsoid, described by the eigenvectors of the diffusion tensor D , scaled with the square root of their respective eigenvalue. This ellipsoid will represent an isosurface of the probability distribution which describes the position of a water molecule, due to diffusion, a short time after it has been placed in the center of the tensor. A spherical ellipsoid therefore corresponds to an isotropic tensor, which describes that water diffusion is equally probable in any direction. When the ellipsoid is more oblate or elongated, it means that water diffuses less or more in a particular direction, and the tensor is therefore referred to as anisotropic. The anisotropy is often characterized using some rotationally invariant and normalized tensor shape measure, for instance the Fractional Anisotropy index (Westin et al., 1999)

$$FA = \frac{1}{\sqrt{2}} \frac{\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_1 - \lambda_3)^2}}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}}. \quad (8.2)$$

One of the most intriguing uses of DT-MRI data is the possibility to follow and visualize fiber pathways in the brain. Traditionally this has been accomplished using fiber tracking algorithms, see for instance (Basser, 1995; Basser et al., 2000; Westin et al., 1999). In these approaches a path originating from a seed point is calculated by iteratively moving a virtual particle in the direction in which diffusion is strongest, the principal diffusion direction (PDD). This direction corresponds to the major eigenvector of the diffusion tensor, which is the eigenvector corresponding to the largest eigenvalue. It is widely believed that for human brain

white matter, in areas where the diffusion tensors are highly anisotropic, the PDD is highly correlated with the orientation of the underlying fiber structure.

One way to visualize the fiber organization of white matter is to place a virtual particle inside a voxel in white matter and iteratively move it according to a velocity field defined by the principal diffusion direction. This trace will be aligned with the underlying fiber structures and visualizing it will give the impression of looking at actual fiber pathways.

This paper will in the following sections introduce a novel post processing method for visualization of fiber traces from DT-MRI. We will focus on enhancing the perception of organization and connectivity in the data. The method will not specifically address the shortcomings of fiber tracking, but assume that a set of fiber traces has already been obtained. Instead the main contribution of this paper will be to show how a spectral non-linear dimensionality reduction technique, such as Laplacian eigenmaps, can be applied to the problem of organizing fiber trace data. The main application will be visualization of large collections of fiber traces.

8.2 Previous Work

Visualization of DT-MRI still poses a challenge for the medical imaging community, since the data is high dimensional and contains a lot of interesting anatomical structure. A simple but effective way to visualize tensor data is to map the tensors to scalars or colors and then visualize the data using any method for volume or image visualization. Commonly used scalar mappings include Fractional Anisotropy Index, trace and the norm of the tensor. Color mapping has also been used to encode orientation of the PDD. While these mappings are good in some applications, they are unintuitive or insufficient in others.

To cope with the high dimensionality of tensor data, special tensor glyphs have been designed, see for instance (Westin et al., 1999). Commonly used glyphs are short line segments showing the orientation of the PDD and ellipsoids representing all six degrees of freedom of a tensor. Other interesting approaches to encode tensor shape and orientation are reaction diffusion patterns (Kindlmann et al., 2000) and line integral convolution (McGraw et al., 2002).

Fiber traces, as described in the introduction, have been successfully been used to reveal fiber pathways in the brain, see for instance (Basser et al., 2000). Often the traces have been represented by streamtubes (Zhang et al., 2003), sometimes in combination with coloring schemes and/or variation of the streamtube thickness according to some quality of the underlying tensor field.

In the area of post processing of fiber traces, prior to visualization, work on clustering of fiber traces have been reported recently. These approaches depend on a similarity measure between pairs of fiber traces, which is used in combination

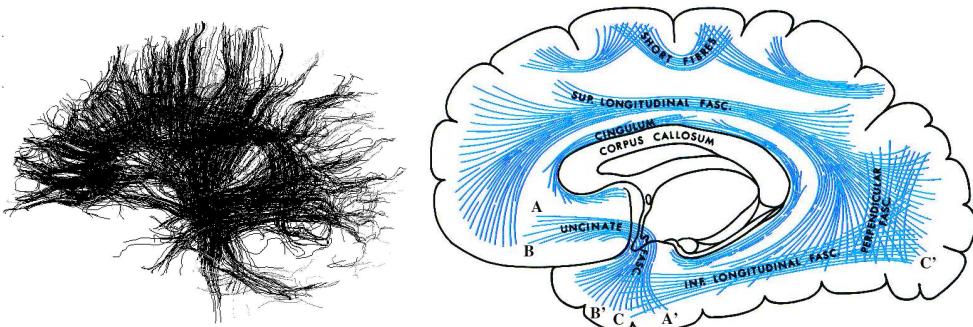


Figure 8.1: **Left:** Fiber traces from a human brain. Simple PDD fiber tracking have been initiated from and constrained to voxels with high anisotropy index. A sagittal view. The head facing left. **Right:** A schematic view of major fiber bundles in the brain. Adapted from Gray's *Anatomy of the Human Body* as displayed at Bartleby.com.

with a traditional clustering method (“fuzzy c-means clustering” (Shimony et al., 2002) and “K nearest neighbors” (Ding et al., 2003)). Outside the medical field, model based curve clustering has been studied in (Gaffney and Smyth, 2003). The method presented in this article will share many similarities with automatic clustering methods. It will however give a continuous coloring of the fiber traces, as opposed to the discrete set of labels assigned during clustering. It could also be considered as a preprocessing step to clustering. Similar to the clustering methods, our approach is automatic and involves no user intervention except parameter selection. This is in sharp contrast from manual approaches to organize traces into bundles, such as the virtual dissection proposed in (Catani et al., 2002). However, all the post processing methods for fiber traces share the same weakness: they rely on a good fiber tracking algorithm to perform well.

8.3 Embedding Fiber Traces – a Motivation

If fiber traces are initiated from seed points in the entire white matter, as in figure 8.1 left, a quick glance motivates the need for some kind of color mapping in order to enhance the perception of the fiber organization in the brain. We therefore propose a post processing step, prior to visualization, in which each fiber trace is assigned a color from a continuous RGB color space. The intuition is that similar traces should be assigned similar colors, while dissimilar traces are mapped to dissimilar colors. This will enhance the visualization of fiber bundles.

8.4 Spectral Clustering and Embedding

In order to map the fiber traces we use a spectral embedding technique called Laplacian eigenmaps which was recently proposed by Belkin and Niyogi in (Belkin and Niyogi, 2002). The core of the algorithm is the use of a local similarity measure, which is used to construct a graph in which each node correspond to a data point and where the edges represent connections to neighboring data points. It is the structure of this graph which represents the manifold to be discovered, which is accomplished through the solution of an eigenvalue problem which maps each data point to a low-dimensional Euclidean space. This mapping locally preserves the graph structure. In short, points close in the graph are mapped to nearby points in the new Euclidean space.

In our application, the data points are fiber traces. The effect we would like to obtain is that traces within a fiber bundle are mapped to similar points in the low-dimensional space. The manifolds we hope to reveal would correspond to a parameterization of a specific fiber bundle. Not a parameterization along the fibers – all points of a fiber trace should project to the same point in the new low-dimensional space – but in the direction perpendicular to the fibers. In the case of a thin bundle such as the cingulate fasciculus we would expect a clustering effect to dominate, all traces within this thin bundle should project to more or less a single point in a low dimensional space. On the other hand, a large bundle structure such as the corpus callosum can be parameterized along the anterior-posterior axis and we would expect it to be represented as a one-dimensional manifold.

While fiber traces naturally reside in a low dimensional 3-D space, a trace itself must be considered as a high-dimensional object, or at least an object which we have difficulties in representing as a point in a low dimensional vector space. Constructing an explicit global similarity measure for fiber traces is also somewhat difficult – to what extent are two traces similar? How can we come up with a similarity measure which corresponds to a mapping of traces into a low-dimensional space? Luckily Laplacian eigenmaps and other spectral methods only needs a local similarity measure, a measure which determine the similarity between a data point and its neighbors. This means that we only need to construct a similarity measure which is able to identify and measure similarity between two very similar traces. In the case of two very dissimilar traces, we may assume zero similarity.

Using this similarity measure, a graph is constructed in which nodes represent fiber traces and where edges connect neighboring traces.

8.5 Laplacian Eigenmaps

For an in depth explanation of Laplacian eigenmaps, as explained by Belkin and Niyogi, see (Belkin and Niyogi, 2002). In brief, the algorithm for Laplacian eigenmaps consists of three steps:

1. Construction of a graph where each node corresponds to a data point. Edges are created between nodes which are close to each other in the original space. A neighborhood of fixed size around each data point or the set of K nearest neighbors could for instance be used as criteria for creating the edges in the graph.
2. Weights are assigned to each edge in the graph. In general, larger weights are used for edges between points which are close to each other in the original space. In the simplest case, all weights are set to 1. A Gaussian kernel or similar could also be used.
3. Solution of the generalized eigenvalue problem:

$$D_{ij} = \begin{cases} \sum_{k=1}^N W_{ik} & \text{if } i = i \\ 0 & \text{if } i \neq j \end{cases} \quad (8.3)$$

$$L = D - W \quad (8.4)$$

$$Ly = \lambda Dy \quad (8.5)$$

where N is the number of nodes and L is called the Laplacian matrix of the graph. The eigenvectors derived from equation 8.3 are ordered according to their eigenvalues. Due to the structure of the L , the smallest eigenvalue will correspond to a constant eigenvector and is discarded, but the n eigenvectors corresponding to the next smallest eigenvalues are used as embedding coordinates for the data points in the new space.

We never performed the formation of the graph in step one explicitly, but performed a thresholding of the weights so that very small weights were set to zero, which corresponds to absence of an edge in the graph.

Laplacian eigenmaps share many similarities with other recent spectral algorithms for clustering and embedding of data, for instance Kernel PCA (Schölkopf et al., 1998) and spectral methods for image segmentation (Meila and Shi, 2001), and we expect a qualitatively similar behavior from all of them even if the interpretation of the results is somewhat different in the various methods. For a unifying view of the behavior of spectral embeddings and clustering algorithms, see (Brand and Huang, 2003). One of the most important aspects of spectral methods for clustering and embedding, including Laplacian eigenmaps, is the fact that they are all posed as eigenvalue problems, for which efficient algorithms are widely available.

8.6 Similarity Through Connectivity

There is no similarity measure given for fiber traces per se and therefore many ways of choosing the edge weights exist. In this initial effort to cluster and embed traces for visualization purposes, we will only try a simple but yet effective similarity measure.

The measure is based on the idea that two traces with similar end points should be considered similar. That is, we only look at the endpoints for a pair of fiber traces, and discard all other information. In figure 8.1 (right) we would for instance want a trace with endpoints $\{A, A'\}$ to have high similarity with a trace with endpoints $\{B, B'\}$. However, trace $\{C, C'\}$ should be considered dissimilar from both $\{A, A'\}$ and $\{B, B'\}$, even though they all share a common origin. This could also be interpreted as a measure of connectivity.

Here $f_{i,1}$ and $f_{i,\text{end}}$ corresponds to the first and last coordinates of the i th fiber trace and W_{ij} is the weight between nodes / fiber traces i and j :

$$f_i = (f_{i,1}, f_{i,\text{end}}), \quad (8.6)$$

$$\tilde{f}_i = (f_{i,\text{end}}, f_{i,1}), \quad (8.7)$$

$$W_{ij} = \begin{cases} 0 & \text{if } i = j \\ \exp\left(-\frac{\|f_i - f_j\|^2}{2\sigma^2}\right) + \exp\left(-\frac{\|\tilde{f}_i - \tilde{f}_j\|^2}{2\sigma^2}\right) & \text{if } i \neq j \end{cases} \quad (8.8)$$

We note that W_{ij} is symmetric with respect to i and j . This measure is also invariant to re-parameterization of the fiber trace, for instance reverse numbering the fiber trace coordinates. It will also give traces which connects similar points in space a large weight while dissimilar connectivity will result in a weight close to zero given that σ is chosen carefully.

This similarity measure will work fine in most cases where the fiber traces are not damaged and really connect different parts of the brain in an anatomically correct way. Other similarity measures used in clustering methods have been based on correlation measures between fiber traces (Shimony et al., 2002; Ding et al., 2003). Those correlation measures could be used as well to build up the graph needed by a spectral embedding method such as Laplacian eigenmaps. For the purpose of demonstration and under the assumption that the fiber traces are ok, the above described similarity should work fine and it is also faster to compute than correlation measures.

8.7 In Vivo DT-MRI Data

Real DT-MRI data from the brain of a healthy volunteer was obtained at the Brigham and Women's Hospital using LSDI technique on a GE Signa 1.5 Tesla Horizon Echospeed 5.6 system with standard 2.2 Gauss/cm field gradients. The time required for acquisition of the diffusion tensor data for one slice was 1 min; no averaging was performed. The voxel resolution was $0.85\text{mm} \times 0.85\text{mm} \times 5\text{mm}$.

A random sample of 4000 points inside white matter with diffusion tensors having high FA were selected as seed points for the fiber tracking. Traces were then created by tracking in both directions starting from these seed points, following the principal eigenvector of diffusion using a step length of 0.5mm and linear

interpolation of the tensors. The tracking was stopped when reaching a voxel with FA lower than certain threshold approximately corresponding to the boundary between white and gray matter. Fiber traces shorter than 10mm were removed. This resulted in a set of approximately 3000 fiber traces.

8.8 Experiments

The algorithm was implemented in MATLAB. While the number of fiber traces were at most 5000 the PDD tracking method, calculation of the graph Laplacian and the solution of the generalized eigenvalue problem could be performed without optimizations. MATLAB was used for visualization except in figure 8.5, where the in-house software 3-D Slicer (Gering, 1999; Gering et al., 1999) was used.

For the color mapping, the second, third and fourth eigenvector were scaled to fit into the interval [0, 1] and then used for the channels red, green and blue, to color the corresponding fiber traces.

The embedding of fiber traces into a RGB color space was tested first on synthetic data, then on real human brain DT-MRI data. The synthetic toy examples should be considered as illustrations of the method rather than near realistic or challenging experiments.

Figure 8.2 shows how the embedding into color space works for a set of fiber traces arranged as a Möbius strip. The traces on (left) are mapped into an RGB space, which determines the color of each trace. In the right plots, the image of this mapping in RGB space (first two embedding coordinates) is shown. Each dot in the right plots correspond to a single trace in the left plots. The circular structure of the Möbius strip can thus be seen in the geometry of the left image, in the coloring of the left image and in the shape of the fiber bundle after embedding it into RGB space to the right.

Figure 8.3 (left) shows how traces connecting opposite sides of a sphere are colored. Coloring according to the three first embedding coordinates. This set of traces has the topology of the “projective plane”, \mathbb{RP}^2 . Note that even though it is impossible to see the traces inside the sphere, we can deduce how traces connect by looking at the colors which match on opposite sides. However, the projective plane cannot be embedded in three dimensions without intersecting itself, which means that the color mapping of this set of traces is many-to-one in some sense.

Figure 8.3 (right) shows a synthetic example of four fiber bundles, two crossing each other and two having the same origin. Because of our similarity measure based on connectivity of the fiber trace endpoints, crossings and overlaps will not disturb the embedding. Laplacian eigenmaps will color each in a color close to its neighbors colors. In this case the clustering properties of Laplacian eigenmaps becomes obvious, which is welcomed as no obvious manifold structure exists in

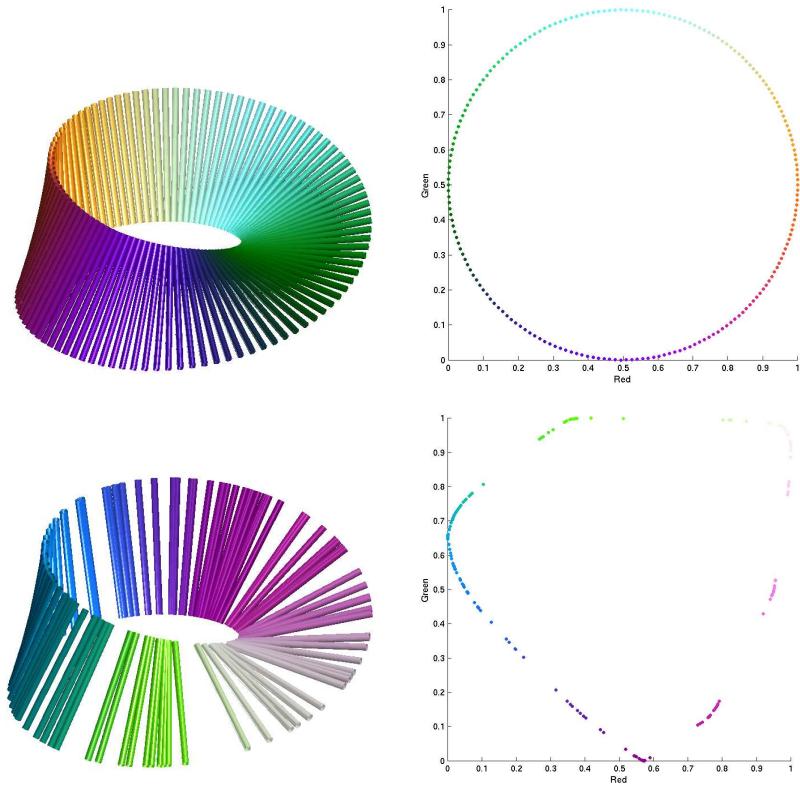


Figure 8.2: Synthetic toy examples of coloring “fiber traces” shaped as a Möbius strip.

Top: A very regular bundle (left) and its embedding (right). Note how the embedding finds a perfect circle. **Bottom:** A more random bundle (left) and its embedding using a little too small σ (right). Note how the embedding tends to enhance clusters in the data, but the topology is still somewhat a circle.

the data.

The experiments on real data in figures 8.4 and 8.5 show how the method works in practice. The value of the only parameter σ was chosen empirically. Starting with a large sigma is safe in general, while a too small sigma give unstable solutions of the eigenvalue problem. In figure 8.5 an example is shown where the fiber traces have been projected back to a T2 weighted coronal slice.

8.9 Discussion

All the figures show different aspects of the idea of using Laplacian eigenmaps, together with a custom made similarity measure, to enhance the visualization of fiber organization. Both the synthetic and real brain data show very promising results, and the colors reveal that the method has been able to organize and em-

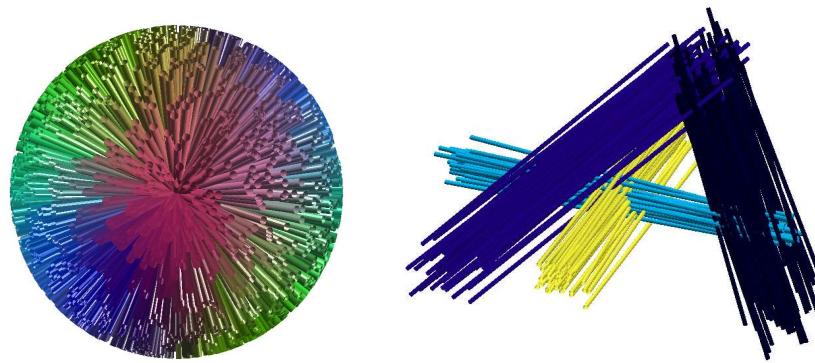


Figure 8.3: Synthetic examples with “fiber traces” connecting opposite points on a sphere (left) and with four fiber bundles (right), two crossing each other and two having the same origin. Coloring according to the three first embedding coordinates.

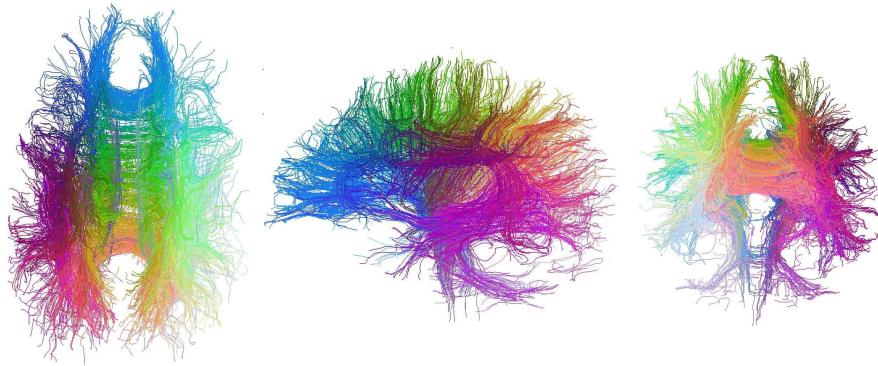


Figure 8.4: Fiber traces from a human brain, colored such that traces with similar endpoints have been assigned similar colors. Simple PDD fiber tracking have been initiated from and constrained to voxels with high anisotropy index. **Left:** Axial view. The head facing up. **Middle:** Sagittal view. The head facing left. **Right:** Coronal view. The head facing inwards.

bed the fiber traces into a space where different anatomical structures have been mapped to different positions. In the real brain data, it can for instance be noted that traces on the left hemisphere in general have a different color from those on the right. Small structures such as the cingulum, going from posterior to anterior above the corpus callosum, are also more visible thanks to the coloring.

The experiments presented in this paper have been chosen with great care. Finding the correct σ has not always been easy and what is a good embedding of fiber traces in RGB-space for visualization is subjective. Optimal choice of σ as well as an analysis of the stability for the embedding is certainly interesting topics for future research.

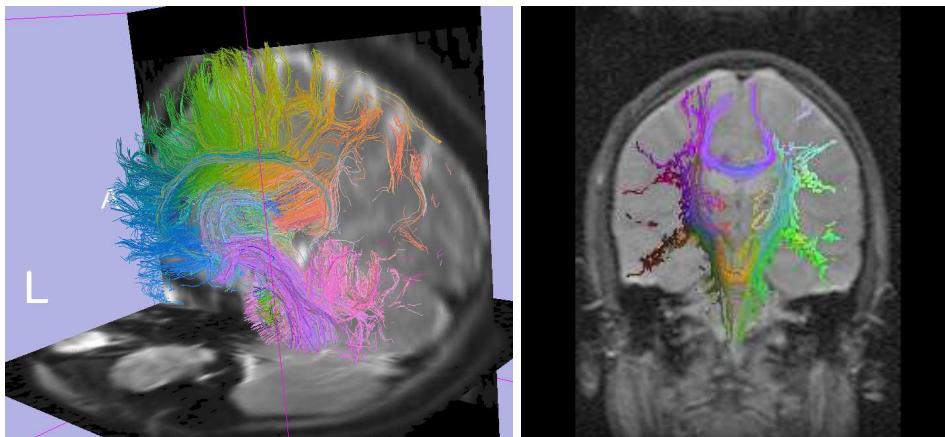


Figure 8.5: **Left:** Fiber traces from a human brain, colored such that traces with similar endpoints are assigned similar colors. A cutting plane is used to give a cross section view of corpus callosum and slices from a T2 weighted volume add additional understanding of the anatomy. Visualization done using the 3-D Slicer (Gering, 1999; Gering et al., 1999). **Right:** Fiber traces from a human brain, colored such that traces with similar endpoints are assigned similar colors. Only the intersection of the traces with a coronal T2 weighted slice is shown. This kind of voxel coloring could for instance assist when manually segmenting white matter in DT-MRI images.

We have so far not focused on optimizing the speed of this post processing method for fiber traces. After the coloring is only done once per dataset. However, for more than a maximum of 5000 fiber traces used in our experiments, we feel there is a need to take greater care in terms of memory management and speed. First of all the eigenvalue problem solved in Laplacian eigenmaps is sparse, given the right similarity measure. Also there exists methods to reduce the size of the eigenvector calculation by using sampling methods such as in the Nyström method (Fowlkes et al., 2001).

The similarity measure used so far is efficient, but simple. Correlation measures of fiber trace similarity have been used by other groups and this method for fiber trace visualization could definitely benefit from a better definition of local fiber trace similarity. Two issues raises. One is to define a better similarity measure which is able to “glue together” broken fiber traces, as fiber tracking is sensitive to noise. The other issue is speed, as the fiber trace similarity measure is evaluated for all pairs of traces. We have done experiments with highly efficient and more correlation-like similarity measures, but the results are still too preliminary to present here.

8.10 Conclusion

The goal of this project was to find a post processing method for DT-MRI fiber traces, to enhance the perception of fiber bundles and connectivity in the brain in general. We can conclude that despite the simplicity of the similarity function, this approach based on Laplacian eigenmaps has been able to generate anatomically interesting visualizations of the human brain white matter. Many interesting new topics arise in the light of this novel way of organizing DT-MRI data: clustering, segmentation and registration being prominent candidates for future research.

Acknowledgments

Thanks to Martha Shenton, Ron Kikinis, Stephan Maier, Steve Haker, Eric Pichon, Jonathan Sacks and Mats Björnemo for interesting discussions. This work was funded in part by NIH grants P41-RR13218 and R01 MH 50747, and CIMIT. It was also supported in part by the Post-doctoral Fellowship Program of Korea Science & Engineering Foundation (KOSEF).

Clustering Fiber Traces Using Normalized Cuts¹

Anders Brun^{a,d}, Hans Knutsson^a, Hae-Jeong Park^{b,c,d},
Martha E. Shenton^b and Carl-Fredrik Westin^d

^a Department of Biomedical Engineering, Linköping University, Sweden
 {andbr, knutte}@imt.liu.se

^b Clinical Neuroscience Division, Laboratory of Neuroscience, Boston VA Health Care System-Brockton Division, Department of Psychiatry, Harvard Medical School, Boston, MA, USA

martha_shenton@hms.harvard.edu

^c Department of Diagnostic Radiology, Yonsei University College of Medicine, Seoul, South Korea

parkhj@yumc.yonsei.ac.kr

^d Laboratory of Mathematics in Imaging, Harvard Medical School, Boston, MA, USA
 westin@bwh.harvard.edu,

Abstract: In this paper we present a framework for unsupervised segmentation of white matter fiber traces obtained from diffusion weighted MRI data. Fiber traces are compared pairwise to create a weighted undirected graph which is partitioned into coherent sets using the normalized cut (*Ncut*) criterion. A simple and yet effective method for pairwise comparison of fiber traces is presented which in combination with the *Ncut* criterion is shown to produce plausible segmentations of both synthetic and real fiber trace data. Segmentations are visualized as colored stream-tubes or transformed to a segmentation of voxel space, revealing structures in a way that looks promising for future explorative studies of diffusion weighted MRI data.

¹Published in the Proceedings of the 7th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'2004).

9.1 Introduction

Diffusion Weighted MRI (DWI) makes it possible to non-invasively measure water diffusion within tissue. In a volume acquired using DWI, each voxel contains a diffusion tensor or other higher order descriptor for the local water diffusion. In fibrous tissue such as muscles and human brain white matter, water tends to diffuse less in the directions perpendicular to the fiber structure. This makes it possible to study the local fiber orientations indirectly by interpreting the water diffusion within the voxel. From DWI data it is therefore possible to create so called fiber traces from virtual particles, traveling along the direction of maximum diffusion, starting from a set of seed points (Basser, 1995; Basser et al., 2000; Westin et al., 2002), a.k.a fiber tracking. Performing fiber tracking in DWI data from the human brain gives valuable insights about fiber tract connectivity, useful in for instance surgical planning and for the study of various diseases such as schizophrenia.

In our experiments we have exclusively used data from so called diffusion tensor MRI (DT-MRI) (Bihan et al., 1986), where the diffusion inside a voxel is described by a second order symmetric positive definite 3×3 tensor, which may be thought of as an ellipsoid. An elongated ellipsoid represents high diffusivity in a particular direction, which may be interpreted as the dominant orientation of the fibers going through that particular voxel. From this data, fiber traces were created within the white matter areas using a standard fiber tracking algorithm following the principal direction of diffusion based on a fourth-order Runge-Kutta integration scheme.

The contribution of this paper is a novel post processing method for clustering or segmentation of such fiber traces. Fiber traces are grouped according to a pairwise similarity function which takes into account the shape and connectivity of fiber traces. The clustering method we propose builds on so called normalized cuts, which have previously been introduced in the computer vision community by Shi and Malik (Shi and Malik, 2000) for automatic segmentation of digital images. This results in an unsupervised segmentation of human brain white matter, in which fiber traces are grouped into coherent bundles, applicable to any DWI technology able to produce fiber traces. For an overview of the method, see figure 9.1.

9.1.1 Previous Work

There are numerous examples where fiber traces from DWI have successfully revealed fiber tracts in the human brain, see for instance (Basser et al., 2000; Behrens et al., 2003a; Westin et al., 2002). Stream-tubes have often been used for visualization (Zhang et al., 2003), sometimes in combination with coloring schemes and variation of the stream-tube thickness according to some aspect of the underlying local diffusion descriptor. The idea of using fiber traces to obtain segmentations of white matter fiber tracts, as well as gray matter areas, have

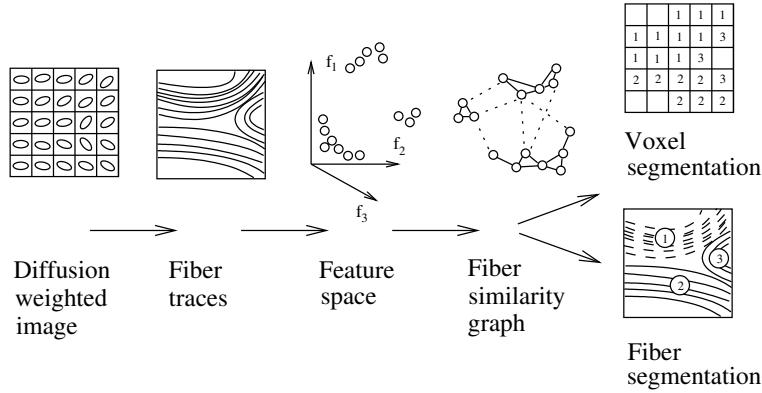


Figure 9.1: An overview of the proposed method. Whether the result should be in voxels or fiber traces depends highly on the application. Fiber traces are flexible, they are for instance able to represent multiple fiber directions going thru a point in space. Voxels are on the other hand more suitable for volume rendering.

been explored in a number of papers recently. In (Behrens et al., 2003a) a segmentation of deep gray matter structures is performed using probabilistic fiber tracking, which connects pre-segmented areas of the human cortex with the thalamus. There also exist manual approaches to organize fiber traces into fiber bundles, such as the virtual dissection proposed in (Catani et al., 2002). In (Brun et al., 2003) the idea of pseudo-coloring (soft clustering) fiber traces to enhance the perception of connectivity in visualizations of human brain white matter was presented. Some unsupervised approaches to clustering of fiber traces, similar to the one in this paper, have also been reported. For instance fuzzy c-means clustering (Shimony et al., 2002) and K nearest neighbors (Ding et al., 2003). Outside the area of medical image processing, clustering of curves (Gaffney and Smyth, 2003) has been reported.

9.2 Determining Fiber Similarity

Many clustering methods, including the *NCut* being used in this paper, operate on a graph with undirected weighted edges describing the pairwise similarity of the objects to be clustered. This graph may be described using a weight matrix W , which is symmetric and has values ranging from 0 (dissimilar) to 1 (similar).

A fiber trace, represented as an ordered set of points in space, is a fairly high-dimensional object. Therefore the pairwise comparison of all fiber traces could potentially be a time-demanding task if fiber trace similarity is cumbersome to calculate and the number of fiber traces is high. In this paper we propose to split the computation of similarity into two steps:

1. Mapping high-dimensional fiber traces to a relatively low-dimensional Eu-

clidean feature space, preserving some but not all information about fiber shape and fiber connectivity. This mapping is oblivious, acting on each fiber separately.

2. The use of a Gaussian kernel for comparison of points in the Euclidean feature space. This function acts on pairs of fiber traces.

It is important to point out early that even though the above mapping to a feature space may seem to be crude at a first glance, it works surprisingly well for fiber traces in practice. For a set of N fiber traces the first calculation above cost $O(N)$, while the second calculation cost $O(N^2)$ operations. This is also the reason for pre-processing the fiber data in the first step, making the second calculation more computationally efficient.

9.2.1 Mapping Fiber Traces to an Euclidean Feature Space

The position, shape and connectivity are important properties of a fiber trace to preserve in the mapping to a feature space. If we regard a fiber trace as just a set of points in space, we capture a sketch of these properties by calculating the mean vector \mathbf{m} and the covariance matrix \mathbf{C} of the points building up the fiber trace. In order to avoid non-linear scaling behavior, we take the square root of the covariance matrix, $\mathbf{G} = \sqrt{\mathbf{C}}$. Now the mapping of a fiber F may be described

$$\Phi(F) = (m_x, m_y, m_z, g_{xx}, g_{xy}, g_{xz}, g_{yy}, g_{yz}, g_{zz})^T, \quad (9.1)$$

which is a 9-dimensional vector. This mapping has the desirable property that it is rotation and translation invariant in the sense that the Euclidean distance between two fiber traces mapped to the 9-dimensional space is invariant to any rotations or translations in the original space. For applications where mean and covariance is not enough to discriminate between different clusters of fiber traces, higher order central moments could add more feature dimensions to the above mapping. Also, in cases when fiber connectivity is more important than shape, a higher weight could be given to the fiber trace end-points in the calculation of the mean vector and covariance matrix above. Different weights could also be given to the mean vector and the covariance components in the mapping in eq (9.1).

9.2.2 Using the Gaussian Kernel for Pairwise Comparison

When fiber traces have been mapped to points in a Euclidean feature space, they may be compared relatively easy for similarity. We choose Gaussian kernels (a.k.a. Radial Basis Functions in Neural Networks literature)

$$K(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right) \quad (9.2)$$

which are symmetric and contain a parameter σ which we may use to adjust the sensitivity of the similarity function. This function maps similar points in feature

space to unity and dissimilar points to zero.

9.2.3 Constructing W

By combining the mapping to a Euclidean feature space with the Gaussian kernel, we obtain the weights of an undirected graph describing the similarity between all pairs of fiber traces. The weights are stored in a matrix W defined as

$$w_{ab} = K(\Phi(F_a), \Phi(F_b)). \quad (9.3)$$

This matrix is expected to be sparse, having most of the values close to zero.

9.3 Normalized Cut and Clustering

Clustering, segmentation and perceptual grouping using normalized cuts was introduced to the computer vision community by Shi and Malik in (Shi and Malik, 2000). The points to be clustered are represented by a undirected graph $G = (V, E)$, where the nodes correspond to the points to be clustered and each edge has weight $w(i, j)$ which represent the similarity between point i and j . The cut is a graph theoretical concept which for a partition of the nodes into two disjunct sets A and B bipartitioning V is defined as

$$\text{cut}(A, B) = \sum_{u \in A, v \in B} w(u, v) \quad (9.4)$$

Using the cut, an optimal partitioning of the nodes may be defined as one that minimizes the cut. Intuitively this could be used for segmentation, while the minimum cut corresponds to a partitioning which keeps well connected components of the graph together. However, there is no bias in the minimum cut which says it should partition the graph in two parts of equal size. Shi and Malik therefore defined the normalized cut, which is defined as

$$N\text{cut}(A, B) = \frac{\text{cut}(A, B)}{\text{asso}(A, V)} + \frac{\text{cut}(A, B)}{\text{asso}(B, V)} \quad (9.5)$$

where $\text{asso}(A, V) = \sum_{u \in A, t \in V} w(u, t)$. This new measure $N\text{cut}$ tries to minimize the cut, while at the same time penalizing partitions in which one set of nodes is only loosely connected to the graph at large. If we define $x_i = 1$ when node $i \in A$, $x_i = -1$ when node $i \in B$, $d(i) = \sum_j w(i, j)$, $k = \frac{\sum_{x_i > 0} d_i}{\sum_i d_i}$, D is a matrix with $\mathbf{d} = d(i)$ in its diagonal and W is the matrix defined by $w(i, j)$ then it is shown in (Shi and Malik, 2000) that

$$N\text{cut} = \frac{(1+x)^T(D-W)(1+x)}{k\bar{1}^T D \bar{1}} + \frac{(1-x)^T(D-W)(1-x)}{(1-k)\bar{1}^T D \bar{1}}. \quad (9.6)$$

which can be shown to be equivalent to

$$Ncut = \frac{y^T(D - W)y}{y^T D y} \quad (9.7)$$

where $y_i \in \{1, -b\}$, $b = k/(1 - k)$, for $y = (1 + x) - b(1 - x)$ while $y^T \mathbf{d} = 0$. Relaxing the problem by allowing y to take any real values results in the minimization of the so called Rayleigh quotient, which can be minimized by solving

$$(D - W)y = \lambda Dy \quad (9.8)$$

It is shown in (Shi and Malik, 2000) that the second smallest eigenvector of eq. (9.8) minimizes the real valued version of the normalized cut.

In our implementation we used the second smallest eigenvector to obtain a 1-d ordering of the vertices of the graph. A random search was then performed to determine a good threshold for the bipartitioning of the graph. To test the goodness of a specific threshold the true discrete $Ncut$ was calculated. When the graph has been split into two, (Shi and Malik, 2000) recommends the partitioning continues recursively until the $Ncut$ raises above a certain value.

9.4 Results

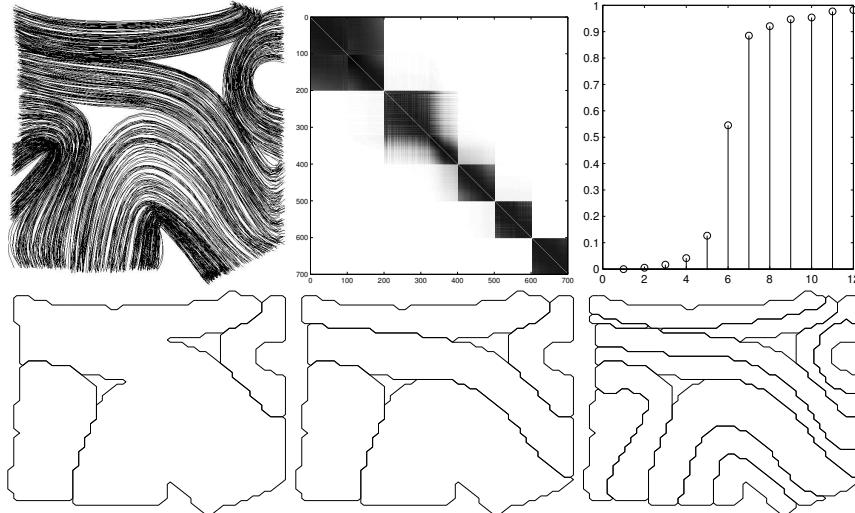


Figure 9.2: **Top left:** A set of synthetic fiber traces in 2-D. **Top middle:** The matrix W . Rows and columns sorted according to the second smallest eigenvector. The Gaussian kernel have been chosen so that five clusters present themselves naturally. **Top right:** The 15 smallest eigenvalues of $(D - W)/D$. **Bottom:** Segmentation obtained from recursive bipartitioning of the fiber traces. Maximum value of the $Ncut$ set to 0.2, 2.5 and 4.5 respectively.

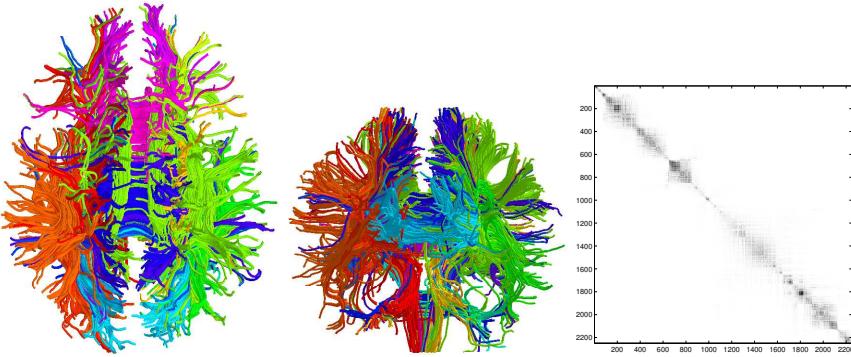


Figure 9.3: **Left:** Axial view of a segmentation obtained from recursive bipartitioning of the white matter fiber traces. Maximum value of the N_{cut} was set to 1.5 and Gaussian kernel $\sigma = 20$. . The colors of fiber traces indicate cluster membership. **Middle:** Coronal view. **Right:** The matrix W . Rows and columns sorted according to the second smallest eigenvector.

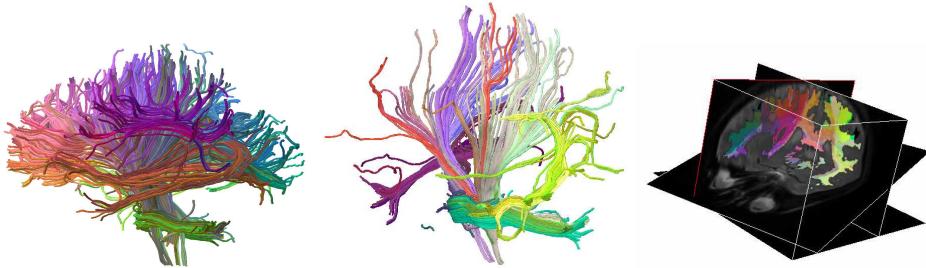


Figure 9.4: Pseudo-coloring of fiber traces. RGB colors are derived directly from scaled versions of the 2nd, 3rd and 4th eigenvector of $(D - W)/D$. Using a very large Gaussian kernel, $\sigma = 100$, results in a soft clustering effect. Note the enhanced perception of fiber connectivity and shape, despite the lack of discrete clusters. **Left:** Whole brain white matter visualized using pseudo-colored fiber traces. **Middle:** A subset of the white matter fiber traces visualized using pseudo-colored fiber traces. **Right:** Pseudo-coloring of voxels belonging to white matter enhance perception of connectivity in slices or volumes. To fill in holes in white matter when fiber traces were transformed to voxels, a nearest-neighbor approach combined with a white matter mask was used.

The method was tested on both synthetic datasets and fiber traces from real diffusion weighted MRI. Results are visualized using both stream-tubes and color coded voxel data. All algorithms were implemented in MATLAB.

- In figure 9.2 the method is tested on synthetically generated fiber traces in 2-D.
- In figure 9.3 real fiber traces obtained from DT-MRI was used as input.
- In figure 9.4 the method was tested with a very large value of σ and direct

mapping of the second, third and fourth eigenvector to colors similar to the approach described in (Brun et al., 2003).

9.5 Discussion

We have not yet investigated the effects of noise to the clustering. Also, the maximum number of fiber traces we have analyzed so far is only about 5000, due to the current implementation in MATLAB which does not fully exploit the sparsity of W . Never the less, the experiments show the potential of the $NCut$ criterion and the proposed measure of fiber similarity.

One insight from the experiments is that pseudo-coloring of fiber traces is very effective to reveal errors in fiber tracking algorithms. A collection of fiber traces may look ok at a first glance, but after pseudo-coloring, the anomalies are easily spotted. One example of this is in figure 9.4 (middle) where the red fiber traces may be identified instantly as outliers because they have a very different color than surrounding fiber traces.

In our experiments we have segmented fiber traces and then sometimes transformed the results back to voxel space. One may ask if it would be possible to segment voxels directly, and what features to use to discriminate voxels. A solution with obvious similarities to the approach presented in this paper would be to perform fiber tracking, possibly stochastic (Behrens et al., 2003a), from each voxel inside white matter and regard the fiber traces as a non-local features of the voxels – *macro features*.

The continuous coloring in figure 9.4 appears to be more visually pleasing than the discrete coloring according to the segmentation in figure 9.3. One may in fact ask if a structure such as the corpus callosum is meaningful to partition into several clusters or whether it is better described as one fiber bundle parameterized by a coordinate system going from anterior to posterior. One could think of the smoothly varying colors in figure 9.4 as coordinate systems, parameterizing all fiber traces in the cross bundle directions. It is in fact then also natural to add a parameterization of each bundle in the fiber direction.

In conclusion the proposed clustering method seems to be a promising new way to automatically reveal the global structure of white matter by segmentation of fiber traces obtained from DWI data. We believe this to be useful in for instance explorative studies of the brain and for visualization of DWI data in surgical planning applications. Important to note though is that all post processing methods for fiber traces share the same weakness: they all rely on a good fiber tracking algorithm to perform well.

9.6 Acknowledgments

Thanks to Marek Kubicki, Lauren O'Donnell, William Wells, Steven Haker, Stephan Maier, Robert McCarley and Ron Kikinis for interesting discussions and to Raúl San José Estépar for valuable comments on the manuscript.

This work was funded in part by NIH grants P41-RR13218 and R01 MH 50747, and CIMIT. It was also supported in part by the Post-doctoral Fellowship Program of Korea Science & Engineering Foundation (KOSEF).

A Tensor-Like Representation for Averaging, Filtering and Interpolation of 3-D Object Orientation Data¹

Anders Brun^{a,b}, Carl-Fredrik Westin^b, Steven Haker^b, Hans Knutsson^a

^a Department of Biomedical Engineering, Linköpings Universitet, Sweden

email: {andbr,knutte}@imt.liu.se

^b Laboratory of Mathematics in Imaging, Harvard Medical School, Boston, MA, USA

email: {westin,haker}@bwh.harvard.edu,

Abstract: Averaging, filtering and interpolation of 3-D object orientation data is important in both computer vision and computer graphics, for instance to smooth estimates of object orientation and interpolate between keyframes in computer animation. In this paper we present a novel framework in which the non-linear nature of these problems is avoided by embedding the manifold of 3-D orientations into a 16-dimensional Euclidean space. Linear operations performed in the new representation can be shown to be rotation invariant, and defining a projection back to the orientation manifold results in optimal estimates with respect to the Euclidean metric. In other words, standard linear filters, interpolators and estimators may be applied to orientation data, without the need for an additional machinery to handle the non-linear nature of the problems. This novel representation also provides a way to express uncertainty in 3-D orientation, analogous to the well known tensor representation for lines and hyperplanes.

10.1 Introduction

Averaging, filtering and interpolation of scalar and vector valued sets and signals using linear methods has been common practice in engineering science for a long time. However, when the data belongs to a non-linear manifold, theory is not as developed. This paper discuss an important special case of the latter, namely when the data belongs to the set of all 3-D orientations. Problems related to this

¹Published in the Proceedings of the IEEE International Conference on Image Processing (ICIP'05).

manifold has been studied in both computer vision and computer graphics. In computer animation, object and camera orientations are interpolated over time, to create smooth transitions between keyframes specified by the animator (Shoemake, 1985; Barr et al., 1992; Grassia, 1998; Lee and Shin, 2002; Johnson, 2003). In computer vision it is common to estimate the orientation of objects, such as airplanes and faces(Srivastava and Klassen, 2002), relative to the coordinate system of a camera. Other application areas include reduction of noise from raw data in sequence-dependent continuum modeling of DNA and the study of plate tectonics (Moakher, 2002).

Object orientation is often mixed up with rotations, because it is a well known fact that any 3-D orientation may be obtained by a single rotation of an object from a reference orientation. In short, orientation is a state, rotation is an action or a change of orientation. One early approach to deal with orientation data in computer graphics was to describe orientation using Euler angles, which specifies orientation as a sequence of rotations about three pre-chosen axes. In a classical paper by Shoemake(Shoemake, 1985), this approach was shown to be inferior to a representation using quaternions. The key observation was that unit quaternions could represent orientation and thus interpolation could be done on \mathbb{S}^3 . One drawback however is that antipodal quaternions, q and $-q$, represent the same rotation and thus quaternions has to be flipped to the same hemisphere prior to interpolation. This approach has been extended by several others, see for instance (Barr et al., 1992). Other approaches have used the intrinsic metric of the space of orientations and rotations, such as parameterization and interpolations using the logarithm and exponential map defined on the lie-group of rotations in 3-D, $SO(3)$ (Grassia, 1998; Lee and Shin, 2002; Moakher, 2002; Johnson, 2003). Other approaches have been based on e.g. intrinsic formulations of non-linear optimization problems (Lee and Shin, 2002), global linearization such as in (Johnstone and Williams, 1999) and extrinsic mean values for orientation data (Srivastava and Klassen, 2002). The approach presented in this paper is related to the notion of extrinsic mean. We will build from the quaternion representation of rotations, creating a novel one-to-one mapping of orientations into 16-dimensional Euclidean space. In this new representation, standard linear methods for interpolation and filtering can be applied, and the result is then projected back to the set of unit quaternions which are then interpreted as orientations.

10.2 Quaternions

Quaternions were invented by Sir William Rowan Hamilton in 1843, after a 15 year search for a successor of complex numbers. The history and theory of quaternions is too rich to cover in a short paper, and we will here focus on some main properties which make quaternions suitable for representing rotations in 3-D. A

quaternion q can be written on the forms

$$\begin{aligned} q &= q_1 + q_2i + q_3j + q_4k \\ &= (q_1, [q_2 \ q_3 \ q_4]^T) = (s, \mathbf{v}) \end{aligned} \quad (10.1)$$

where the coefficients q_1, q_2, q_3 and $q_4 \in \mathbb{R}$. The conjugate is then defined as

$$\bar{q} = q_1 - q_2i - q_3j - q_4k. \quad (10.2)$$

The purely imaginary parts of a quaternion satisfy Hamilton's rules

$$i^2 = j^2 = k^2 = -1 \quad (10.3)$$

$$ij = -ji = k \quad (10.4)$$

$$jk = -kj = i \quad (10.5)$$

$$ki = -ik = j. \quad (10.6)$$

Addition and multiplication can then be defined

$$\begin{aligned} a + b &= (a_1 + b_1) + (a_2 + b_2)i \\ &\quad + (a_3 + b_3)j + (a_4 + b_4)k \end{aligned} \quad (10.7)$$

$$\begin{aligned} ab &= (a_1b_1 - a_2b_2 - a_3b_3 - a_4b_4) \\ &\quad + (a_1b_2 + a_2b_1 + a_3b_4 - a_4b_3)i \\ &\quad + (a_1b_3 - a_2b_4 + a_3b_1 + a_4b_2)j \\ &\quad + (a_1b_4 + a_2b_3 - a_3b_2 + a_4b_1)k. \end{aligned} \quad (10.8)$$

The norm is defined as

$$n(a) = \sqrt{q\bar{q}} = \sqrt{\bar{q}q} = \sqrt{a_1^2 + a_2^2 + a_3^2 + a_4^2}, \quad (10.9)$$

and it is multiplicative

$$n(ab) = n(a)n(b). \quad (10.10)$$

A rotation in 3-D of a point \mathbf{p} about the unit vector $\hat{\mathbf{n}}$ by an angle θ may be calculated using quaternions using

$$p' = qpq^{-1} = qp\bar{q}, \quad (10.11)$$

where

$$q = (s, \mathbf{v}) = (\cos(\frac{\theta}{2}), \hat{\mathbf{n}} \sin(\frac{\theta}{2})) \quad (10.12)$$

and points are represented by

$$p = (0, \mathbf{p}). \quad (10.13)$$

From (10.11) and (10.12) it can be seen that every unit quaternion, $n(q) = 1$, may be interpreted as a rotation in 3-D, and every rotation in $SO(3)$ maps to antipodes on \mathbb{S}^3 with q and $-q$ representing the same rotation (Shoemake, 1985; Barr et al., 1992; Grassia, 1998; Lee and Shin, 2002).

10.3 Mapping Orientations to \mathbb{R}^{16}

The antipodal nature of the quaternion representation of rotations is analogous to that of representing line orientations using unit length vectors pointing along the line. These vectors might point in either of two directions. In \mathbb{R}^2 , line orientation may be represented using the double angle representation (Granlund, 1978). In n dimensions, $n \geq 2$, line orientations can be represented by taking the outer product of a unit vector $\hat{\mathbf{n}}$ pointing along the line, which results in a $n \times n$ symmetric positive semi-definite matrix (or tensor of order 2) (Knutsson, 1989).

$$T = \hat{\mathbf{n}}\hat{\mathbf{n}}^T = (-\hat{\mathbf{n}})(-\hat{\mathbf{n}}^T) \quad (10.14)$$

This tensor representation is invariant to flips of the unit vector. It identifies antipodal points on \mathbb{S}^{n-1} , which reveals that the topology of line orientation in n dimensions is equivalent to the projective plane $\mathbb{RP}^{(n-1)}$. Fully generalized, this representation may continuously represent not only line orientations, but also planes and hyper planes in higher dimensions, and it is called the structure tensor when used to analyze image and volume data local neighborhood.

Seen from the quaternion representation of rotations, $SO(3)$ maps to antipodal points on \mathbb{S}^3 . By identifying these antipodal points one realizes that $SO(3)$ has the topology of the projective space \mathbb{RP}^3 . Analogous to the tensor representation of line orientations, we now define the following map for unit quaternions:

$$Q = M(q) = M(-q) \quad (10.15)$$

$$= \mathbf{q}^T \mathbf{q} \quad (10.16)$$

$$= (q_1, q_2, q_3, q_4)^T (q_1, q_2, q_3, q_4) \quad (10.17)$$

$$= (\cos \frac{\theta}{2}, \sin \frac{\theta}{2} \hat{\mathbf{n}}^T)^T (\cos \frac{\theta}{2}, \sin \frac{\theta}{2} \hat{\mathbf{n}}^T) \quad (10.18)$$

$$= \begin{pmatrix} \cos^2 \frac{\theta}{2} & \cos \frac{\theta}{2} \sin \frac{\theta}{2} \hat{\mathbf{n}}^T \\ \cos \frac{\theta}{2} \sin \frac{\theta}{2} \hat{\mathbf{n}} & \sin^2 \frac{\theta}{2} \hat{\mathbf{n}} \hat{\mathbf{n}}^T \end{pmatrix} \quad (10.19)$$

$$= \frac{1}{2} \begin{pmatrix} (1 + \cos \theta) & \sin \theta \hat{\mathbf{n}}^T \\ \sin \theta \hat{\mathbf{n}} & (1 - \cos \theta) \hat{\mathbf{n}} \hat{\mathbf{n}}^T \end{pmatrix} \quad (10.20)$$

The new object Q is a 4×4 symmetric positive semi-definite matrix and thus the manifold of $SO(3)$ has been mapped to \mathbb{R}^{16} , or \mathbb{R}^{10} if symmetry is taken into account. This mapping from unit quaternions is two-to-one, $M(-q) = M(q)$. Combined with the mapping from $SO(3)$ to quaternions, which is one-to-two, a continuous one-to-one mapping from 3-D orientations into a Euclidean space has been obtained.

10.4 ... and Back Again

The basic idea of this paper is to apply linear methods on the manifold of orientations by inheriting the Euclidean metric of \mathbb{R}^{16} . In (Srivastava and Klassen, 2002)

the so called extrinsic mean is defined, which is the minimization of the sum of squared distance to all data points, using the extrinsic metric inherited from the space embedding the manifold

$$q_{mean} = \arg \min_{|q|=1} \sum_{i=1}^N d(q, q_i)^2 \quad (10.21)$$

While the most natural distance metric should be the intrinsic metric, the extrinsic metric may be a good approximation in many cases. It is shown in (Srivastava and Klassen, 2002) that finding the extrinsic mean on the manifold is equivalent to first finding the mean μ in the embedding space, and then find the point $q_{mean} = P(\mu)$ on the manifold which minimizes the distance to μ . Using the Frobenius norm on matrices $Q \in \mathbb{R}^{4^2}$ (equivalent to the Euclidean metric on \mathbb{R}^{16}) as extrinsic metric, we obtain the extrinsic mean of orientations as

$$q_{mean} = \arg \min_{|q|=1} \sum_{i=1}^N |(M(q), M(q_i))|_F^2 \quad (10.22)$$

$$= P\left(\frac{1}{N} \sum_{i=1}^N M(q_i)\right). \quad (10.23)$$

The orientation which maps to the point on the manifold closest to Q may be calculated using

$$P(Q) = \arg \min_{|q|=1} |M(q) - Q|_F^2 \quad (10.24)$$

$$= \arg \min_{|q|=1} \text{Tr}((qq^T - Q)(qq^T - Q)^T) \quad (10.25)$$

$$= \arg \max_{|q|=1} \text{Tr}(Q qq^T), \quad (10.26)$$

and it can be shown that choosing q to be the largest eigenvector of Q will minimize this distance.

The inverse mapping $P(Q)$ is not only useful for calculating the extrinsic mean. It may be used to map any point $Q \in \mathbb{R}^{4^2}$ back to the manifold of orientations, after performing any linear combination of data points mapped by M . In fact, any variational approach which ends up minimizing an energy function in \mathbb{R}^{4^2} which can be expressed as a monotone increasing function of $d(Q_{min}, Q)$, for some global minimum Q_{min} , should be possible to minimize on the manifold of orientations using $P(Q_{min})$. In particular, we may use $P(Q)$ to get back to the manifold after having performed interpolation and filtering in \mathbb{R}^{4^2} .

10.5 Rotation Invariance

Are linear operations on Q rotation invariant?

Lemma 1 (Rotation) *Applying a rotation q_r to q prior to mapping results in*

$$M(q_r q) = (R\mathbf{q})(R\mathbf{q})^T \quad (10.27)$$

$$= R(\mathbf{q}\mathbf{q}^T)R^T \quad (10.28)$$

for some 4×4 rotation matrix R .

Proof 1 *Quaternion multiplication is a linear operation on the coefficients. The norm is multiplicative and thus every $|q| = 1$ will be mapped into a new unit quaternion, so the mapping must be a rotation in \mathbb{R}^4 .*

Theorem 1 (Rotation invariance)

$$RM(q_a)R^T + RM(q_b)R^T = R(M(q_a) + M(q_b))R^T \quad (10.29)$$

$$\lambda RM(q_a)R^T = R(\lambda M(q_a))R^T \quad (10.30)$$

Proof 2 *Follows directly from matrix algebra.*

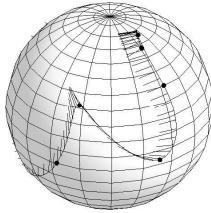
This is sufficient to make all linear operations rotation invariant. The interpretation should be that linear operations can be applied with the same result regardless of any prior rotation of the data set, just what is expected for a fair and well balanced representation.

10.6 Experiments

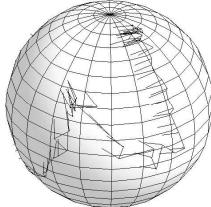
To test the new representation, two experiments were performed. First cubic splines were used to interpolate orientations in the new representation. Then directional noise was added and finally a FIR filter was applied on the sequence of orientation data to restore the signal. See figure 1. In order to visualize sequences of orientations, a curve is plotted on a sphere, which can be thought as representing the path of a camera pointing towards origo.

10.7 Discussion

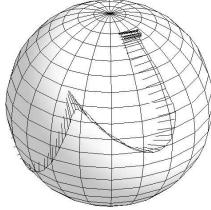
This paper describes how to obtain a continuous and rotation invariant representation of orientation, which can be used in a plug-and-play fashion with pretty much any method for averaging, filtering and interpolation of vector valued data. The calculation of extrinsic mean value is similar to the method described at page 115 in (Johnson, 2003) to calculate a mean orientation. However, what we propose here is a representation of orientation, in which interpolation and filtering can be done directly. Analogous to the tensor representation of line and hyper plane orientations (Knutsson, 1989), it may also be seen as a simultaneous representation of a mean value and uncertainty, when the rank of Q is larger than one, which could be used for instance to perform rotation invariant statistical tests on a set of orientations.



(a) Interpolated



(b) Noisy



(c) Smoothed

Figure 10.1: Interpolation and filtering of noisy orientation data. First a set of keyframe orientations (filled circles) are interpolated using cubic splines to form a continuous path in a). Then directional noise is added in b). Finally the noisy data in b) is filtered using a smoothing FIR filter of length 10. The path between the keyframes corresponds to the position of a virtual camera pointing towards origo. The line-glyphs tangent the sphere and represent the up-direction relative to the camera.

While the method is easy to implement, it is not yet fully understood what negative effects there might be of using a global embedding approach. Other methods based on the more natural intrinsic metric may also be both computationally faster and more direct to implement. Never the less, we believe the simplicity of the method will be attractive for at least some applications. For the image analysis community, the similarity of this representation with the tensor representation of line- and hyperplane orientations should be of particular interest. It would also be interesting to further investigate how low-level image features in 3-D volume data can be represented using this new representation. The matching of a small 3-D template, rotated in all possible orientations and translated all over the volume, is one example of a signal which can be described using the proposed representation. In the computer graphics community on the other hand, this representation may be

able to compete with the commonly used quaternions in some specific application areas. We guess the possibility to replace non-linear signal processing of 3-D object orientation data, with simple 1-D signal processing, might not always give the desired result but it will work well enough in many situations.

11

Fast Manifold Learning Based on Riemannian Normal Coordinates¹

Anders Brun^{a,c}, Carl-Fredrik Westin^c, Magnus Herberthson^b, Hans Knutsson^a

^a Department of Biomedical Engineering, Linköpings Universitet, Sweden
email: {andbr,knutte}@imt.liu.se

^b Department of Mathematics, Linköpings Universitet, Sweden
email: maher@mai.liu.se

^c Laboratory of Mathematics in Imaging, Harvard Medical School, Boston, MA, USA
email: westin@bwh.harvard.edu,

Abstract: We present a novel method for manifold learning, i.e. identification of the low-dimensional manifold-like structure present in a set of data points in a possibly high-dimensional space. The main idea is derived from the concept of Riemannian normal coordinates. This coordinate system is in a way a generalization of Cartesian coordinates in Euclidean space. We translate this idea to a cloud of data points in order to perform dimension reduction. Our implementation currently uses Dijkstra's algorithm for shortest paths in graphs and some basic concepts from differential geometry. We expect this approach to open up new possibilities for analysis of e.g. shape in medical imaging and signal processing of manifold-valued signals, where the coordinate system is "learned" from experimental high-dimensional data rather than defined analytically using e.g. models based on Lie-groups.

11.1 Introduction

A manifold can be seen as a generalization of a surface to higher dimensions. Locally a manifold looks like a Euclidean space, \mathbb{R}^N , but on a global scale it may be curved and/or compact, like a sphere or a torus. A manifold with a metric tensor defined at each point is called a Riemannian manifold.

¹Published in the Proceedings of the 14th Scandinavian Conference on Image Analysis (SCIA'05).

Recent developments in so called manifold learning has opened up new perspectives in non-linear data analysis. Classical methods such as Principal Components Analysis (PCA, a.k.a. the Karhunen-Loeve transform) and Multidimensional Scaling (MDS) efficiently finds important linear subspaces in a set of data points. Methods within the field of manifold learning are however able to identify non-linear relations as well. In this paper we present a new tool for data analysis of this kind, based on the concept of Riemannian normal coordinates.

Manifold learning has become an established field of research, Kohonen's Self Organizing Maps (SOM) (Kohonen, 1982) being an important early example. Characteristic for the newest generation of manifold learning techniques is efficiency and global convergence, in particular many of them are based on the solution of very large eigenvalue problems. This include for instance the recent Kernel PCA (Schölkopf et al., 1998), Locally Linear Embedding (Roweis and Saul, 2000), Isomap (Tenenbaum et al., 2000), Laplacian Eigenmaps (Belkin and Niyogi, 2002) and Hessian Eigenmaps (Donoho and Grimes, 2003).

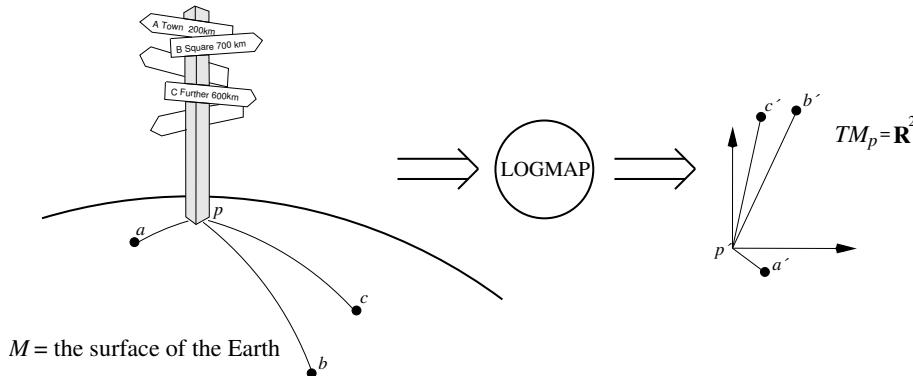


Figure 11.1: Traveling along a geodesic, starting at a specific location in a specific direction, will eventually take you to any place on the surface of the Earth. Riemannian normal coordinates captures this information, mapping points on the sphere to \mathbb{R}^2 in a way that direction and geodesic distance from the origin to any point is preserved. Riemannian normal coordinates are therefore quite natural to use for navigation on a manifold, at least in the vicinity of a point. Also note that geodesics on a manifold M (left) are mapped to lines in Riemannian normal coordinates (right).

Manifolds arise in data for instance when a set of high-dimensional data points can be modeled in a continuous way using only a few variables. A typical example is a set of images of a 3-D object. Each image may be represented as a very high-dimensional vector, which depends on the scene and a few parameters such as relative camera orientation, camera position and lighting conditions. Camera orientation itself is a good example of a non-linear manifold. The manifold of orientations, $SO(3)$, can be represented by the set of all rotation matrices. While the manifold-valued parameter space is equivariant to important features of the data, namely camera- and lighting information, it should also be invariant to un-

interesting things such as noise from the image sensors.

In the following sections we present a novel technique for manifold learning based on the concept of Riemannian normal coordinates. We have translated this technique from its original setting in differential geometry, to the task of mapping a set of experimental high-dimensional data points, with a manifold-like structure, to a low-dimensional space. An intuitive explanation of Riemannian normal coordinates is given in figure 11.1. They contain information about the direction and distance from a specific point on a manifold to other nearby points. The usefulness of such information for navigation is obvious, not only for navigating on the Earth, but also for creating user interfaces to navigate in manifold-valued data in general. The Riemannian normal coordinates are also closely related to geodesics and the exponential and logarithmic maps of Lie-groups, which have been used recently for the analysis of shape in medical images (Fletcher et al., 2003) and to perform time-dependent signal processing of orientation data (Lee and Shin, 2002).

11.2 Theory

In this section we briefly review some basic concepts of differential geometry necessary to understand the method we propose.

To each point p on a manifold M there is a associated tangent space, $T_p M$, consisting of a Euclidean space tangential to M at p . Derivatives at p of smooth curves passing through a point p belongs to $T_p M$.

A special class of curves defined on Riemannian manifolds are the geodesics, i.e. length minimizing curves on M . These define a metric $d(x, y)$ on a manifold derived from the length of a geodesic passing through x and y .

The Riemannian exponential map, $\exp(v) \in M$, $v \in T_p M$, is a function which maps points in the tangent space of p , to points on M . If $H(t)$ is the unique geodesic, starting at p with velocity v , then $\exp(v) = H(1)$. Intuitively this can be thought of as walking with constant velocity in particular direction on the manifold, from a point p , during one time unit. This mapping is one-to-one in a neighborhood of p and its inverse is the log map.

The set of points on M for which there exists more than one shortest path from p is called the *cut locus* of p . The cut locus of a point on a sphere is for instance its antipodal point. Some manifolds, such as \mathbb{R}^2 , lack a cut locus. Other manifolds, such as the torus, have a quite complex looking cut locus.

Given a point p and an orthonormal basis $\{\hat{e}_i\}$ for the tangent space $T_p M$, a Riemannian normal coordinate system is provided by the exponential mapping. A point $x \in M$ gets the coordinate (x^1, \dots, x^N) if $x = \exp(x^i \hat{e}_i)$.

The gradient of a scalar function f is a dual vector field which components are simply the partial derivatives (in the induced basis).

11.3 Method

Given a basis point p from a data set X and an orthonormal basis of the tangent space at p to a thought manifold M , we would like to, via the log map into $T_p M$, express all data points $x \in X$ using Riemannian normal coordinates. Due to the properties of Riemannian normal coordinates, this is equivalent to measuring the distance and direction from p to every other point in the data set. We choose to call this framework LOGMAP:

1. From a set of data points, X , sampled from a manifold M , choose a base point $p \in X$.
2. To determine the dimension of M , select a ball $B(p)$ of the K closest points around p . Then perform standard PCA in the ambient space for $B(p)$. This will give us $T_p M$, with $\dim T_p M = N$, where we choose any suitable ON-basis $\{\hat{e}_i\}$. All $y \in B(p)$ are mapped to $T_p M$ by projection on $\{\hat{e}_i\}$ in the ambient space. This is the Ψ -mapping in figure 11.2.
3. Approximate distances on M . In the current implementation we do this by defining a weighted undirected graph, with each node corresponding to a data point and with edges connecting each node to its L closest neighbors. Let the weights of these edges be defined by the Euclidean distance between data points in the ambient space. We then use Dijkstra's algorithm for finding shortest paths in this graph, to approximate the geodesic distances in M . This gives estimates of $d(x, y)$ for all $(x, y) \in X \times B(p)$.
4. To calculate the direction from p to every point $x \in X$, estimate $\mathbf{g} = \sum g^i \hat{e}_i = \nabla_y d^2(x, y)|_{y=p}$ numerically, using the values obtained in the previous step. While we only have values of $d^2(x, y)$ for $y \in B(p)$, we must interpolate this function in $T_p M$, e.g. using a second order polynomial, in order to calculate the partial derivatives at $\Psi(p)$.
5. Estimates of Riemannian normal coordinates for a point x are then obtained as $x^i = d(x, p) \frac{g^i}{|\mathbf{g}|}$.

In step 4) above, the numerical calculation of the gradient at p uses the squared distance function. The reason for not just taking the gradient at p of the plain distance function from x , which is known to point in the direction of the geodesic connecting p and x , is that it is not smooth for $p \approx x$. Using the square of the distance function, which is much easier to interpolate, solves this problem while giving a gradient in the same direction. However, when x is close to the cut locus of p , even the squared distance function becomes non-smooth. In the experiments shown in the next section, we have actually used a slightly more robust scheme

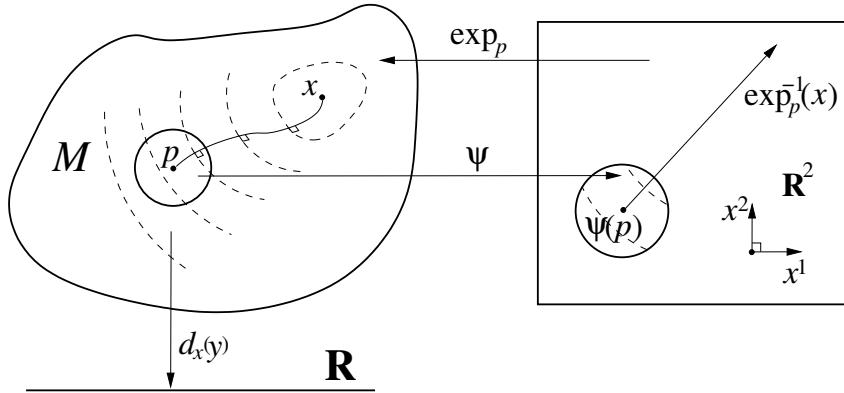


Figure 11.2: A schematic illustration of a geodesic from x to p in a manifold M . Dashed curves correspond to iso-levels of $d_x^2(y) = d^2(x, y)$. These iso-curves are perpendicular to every geodesic passing through x . The ball around p and the mapping Ψ defines a chart that maps a part of M to \mathbb{R}^2 . The domain of \exp is actually the tangent space of M at p , and it is natural to identify vectors in \mathbb{R}^2 with $T_p M$.

to estimate the gradient for points close to the cut locus. This was done by using the RANSAC algorithm (Fischler and Bolles, 1981), to select points close to p consistent with a second order polynomial model of the squared distance function.

11.4 Experiments

The LOGMAP method was evaluated using Matlab. The most critical part of the algorithm, the calculation of shortest paths, was borrowed from the Isomap implementation of Dijkstra's shortest paths (Tenenbaum et al., 2000). In the LOGMAP implementation, the selection of p was made interactively by the click on the mouse and the resulting log map was calculated almost in real time.

Three experiments on synthetic data are presented here to illustrate the behavior of the algorithm. In each of the experiments we have assumed knowledge of how to choose L , the number of neighbors for building the graph, and K , which determines the size of the neighborhood used for dimension estimation and later the estimation of gradients. It is important to point out that selection of these parameters is actually non-trivial for many data sets, e.g. when noise is present. We will not go further into the details of choosing these constants in this paper however.

11.4.1 The Swiss Roll

In the first experiment we use the “Swiss roll” data set, consisting of points sampled from a 2-D manifold, embedded in \mathbb{R}^3 , which looks like a roll of Swiss

cheese. It has been used before to illustrate methods for manifold learning, see e.g. (Tenenbaum et al., 2000; Roweis and Saul, 2000), and we include it mainly as a benchmark. A set of 2000 points from this data set were used in the experiment and the results are presented in figure 11.3. The experiment shows that the LOGMAP method correctly unfolds the roll and maps it to Riemannian normal coordinates in \mathbb{R}^2 .

It is important to note that the resulting mapping in the Swiss roll example is more or less isometric, which is expected for simple flat manifolds. This is similar to the behavior of Isomap. On the other hand, both Isomap and LOGMAP would fail to produce isometric embeddings if we would introduce a hole in the Swiss roll data set. This particular problem is solved by Hessian Eigenmaps for flat manifolds.

11.4.2 The Torus

In the second experiment we tested the method on a data set consisting of 2000 points from a torus embedded in 3-D. The results in figure 11.4 illustrate how the method cuts the coordinate chart at the cut locus of the point p . This particular behavior of “cutting up” the manifold allows us to save one dimension in this particular example. There is no embedding of the torus into \mathbb{R}^2 . Any standard method for dimension reduction, e.g. LLE, Laplacian Eigenmaps or Isomap, would embed this manifold into \mathbb{R}^3 at best. However, the automatic introduction of a cut by the LOGMAP method makes it possible to make a one-to-one mapping of this manifold to \mathbb{R}^2 .

11.4.3 The Klein Bottle

The third experiment finally, shown in figure 11.5, tests the method on truly high-dimensional data. The data set consists of 21×21 pixel image patches. Each of the 2-D image patches were rendered as a 1-D sine wave pattern with a specific phase and orientation. A small amount of normal distributed white noise was also added to the images. The resulting data set consisted of 900 data points, in a 441-dimensional space, representing image patches sampled uniformly from all possible values of phase and orientation. It is natural to assume that the intrinsic dimensionality of this data set is 2, since the variables phase and orientation adds one degree of freedom each.

We observed slightly different shapes of the cut locus, i.e. the border of the resulting map, depending on the choice of base point p . This was somewhat unexpected, but it has a logical explanation. Even though the data set seems to be highly symmetric in terms of orientation and phase, the square shape of the image patches themselves will break the otherwise expected rotation invariance and introduce variations in curvature on the manifold.

The mapping of image patches to \mathbb{R}^2 is visualized by simply using the image

patches as glyphs, placed at various locations in the plane. By carefully identifying the edges of the cut locus, we manually obtain an interpretation of the mapping shown in the top left of figure 11.5. This directed labeled graph reveals that the topology of this particular image manifold is actually the well known Klein bottle (Weisstein, 2005). Similar conclusions for the topology of local descriptions of phase and orientation has previously been described in (Tanaka, 1995; Swindale, 1996), where the topology of Gabor filters is derived from theoretical investigations. Our investigation is on the contrary experimental, and to the best of our knowledge it is a new example of how manifold learning can be used to experimentally infer the topology of a data set.

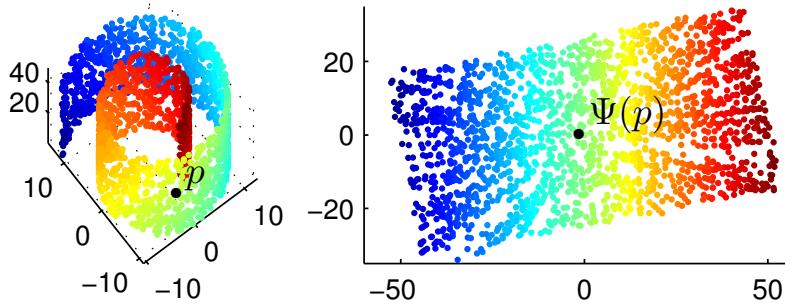


Figure 11.3: A set of 2000 points from the “Swiss roll” example (Tenenbaum et al., 2000). Colors correspond to the first Riemannian normal coordinate derived from the method. **Left:** The original point cloud embedded in 3-D. **Right:** Points mapped to 2-D Riemannian normal coordinates.

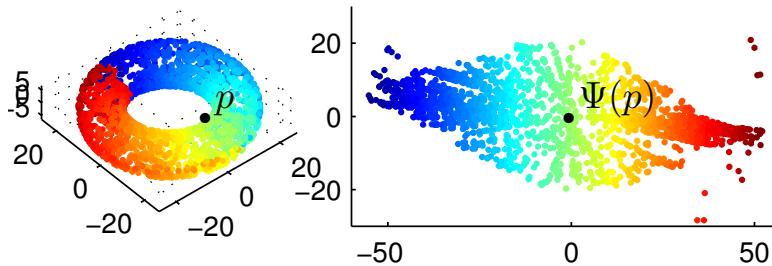


Figure 11.4: A set of 2000 points from a torus embedded in 3-D. Colors correspond to the first Riemannian normal coordinate derived from the method. **Left:** The original point cloud embedded in 3-D. Notice the discontinuity (red-blue) in the coordinate map, revealing a part of the “cut locus” of p . **Right:** Points mapped to 2-D Riemannian normal coordinates. Because the metric of a torus embedded in 3-D is not flat, the manifold is not mapped to a perfect rectangle. Some outliers are present, due to incorrect estimation of the gradient for points near the cut locus.

11.5 Discussion

The presented LOGMAP method is rather different from many other methods for manifold learning and dimension reduction, both in terms of the output and in terms of algorithmic building blocks.

The possibility of a cut, a discontinuity in the mapping at the so called cut locus, could be seen not only as a problem but also as a feature of the method. This allows for instance the torus and the Klein bottle to be visualized using a two-dimensional plot. Other methods, such as (Tenenbaum et al., 2000; Roweis and Saul, 2000; Schölkopf et al., 1998; Belkin and Niyogi, 2002; Donoho and Grimes, 2003), tries to find a continuous embedding of the manifold, and for that at least 4 dimensions are needed for the Klein bottle and 3 for the torus. (The top middle illustration in figure 11.5 is actually an example of an immersion and not an embedding of the Klein bottle in 3-D, meaning roughly that it intersects itself at some points.)

The use of other criteria for assigning a global coordinate system to a manifold could also be considered, for instance conformal mappings of 2-D manifolds. In almost every case when mapping a manifold to a low-dimensional space, some kind of distortion is introduced while some features of the original manifold will be preserved. For most manifolds, Riemannian normal coordinates create a very distorted mapping far away from the base point p , in some cases they even introduce a cut. However, they also preserve all geodesic distances and angles from p to other points on the manifold, which makes this mapping quite intuitive and particularly useful for the purpose of navigating inside a manifold. At least this is true in the vicinity of the base point p .

The LOGMAP method is built up by two major algorithmic building blocks:

1. Approximation of distances on a manifold given a set of sampled data points.
2. Calculation of gradients on manifolds, from a set of function values defined at the sampled data points.

For the first building block we have here used Dijkstra's method, mainly inspired by the Isomap implementation. This method has obvious problems to truthfully approximate distances, because distances are measured along zigzag trajectories in a graph. One way to make LOGMAP more accurate is therefore to switch to a more accurate method based on higher order approximations of the manifold.

The second building block, which is about calculating gradients, could also be improved a lot compared to the current implementation. Measuring gradients for smooth functions is not a problem, but for points close to the cut locus the distance function will introduce a discontinuity which makes the problem quite delicate. The difficulty of gradient estimation manifests itself by producing spurious points

in the mapping, most easily seen in the torus and the Klein bottle examples, close to the cut locus.

In this paper we have chosen examples of manifolds with low intrinsic dimensionality, mainly to illustrate the method, but in principle the method works for manifolds of higher dimensionality too. In the examples we have also used only little or no noise. While this can be seen as very optimistic assumptions about the data, we would like to stress the fact that the LOGMAP method does not try to explicitly deal with noise. In order to handle noise efficiently, it should either be removed prior to the use of LOGMAP or handled by more robust versions of the distance and gradient estimation steps within the LOGMAP framework. This is clearly an important area of future research.

Regarding the efficiency or speed properties of the LOGMAP method, it is important to mention that it, in contrast to many other methods for manifold learning, does not involve the solution of any large eigenvalue problem or any other large scale iterative minimization procedure. Instead it relies totally on the ability to fast approximate distances on the manifold and calculate gradients. A key observation is also that distances $d(x, y)$ are only calculated for pairs $(x, y) \in X \times B(p)$. This is far less demanding than calculating the distance for all pairs $(x, y) \in X \times X$, which is done in for instance Isomap.

In summary, we have introduced a novel method for manifold learning with interesting mathematical and computational properties. We have also provided an example of how manifold learning can assist in identifying a rather non-trivial manifold, in this case a Klein bottle, from a high-dimensional data set. We believe this to be of general interest to people within the fields of manifold learning and medical image analysis, to for instance develop better tools for shape analysis, and to inspire the future development of manifold learning and manifold-valued signal processing in general.

11.6 Acknowledgements

We gratefully acknowledge the support of the Manifold Valued Signal Processing project by the Swedish Research Council (Vetenskapsrådet, grant 2004-4721) and the SIMILAR network of excellence (<http://www.similar.cc>), the European research task force creating human-machine interfaces similar to human-human communication of the European Sixth Framework Programme (FP6-2002-IST1-507609).

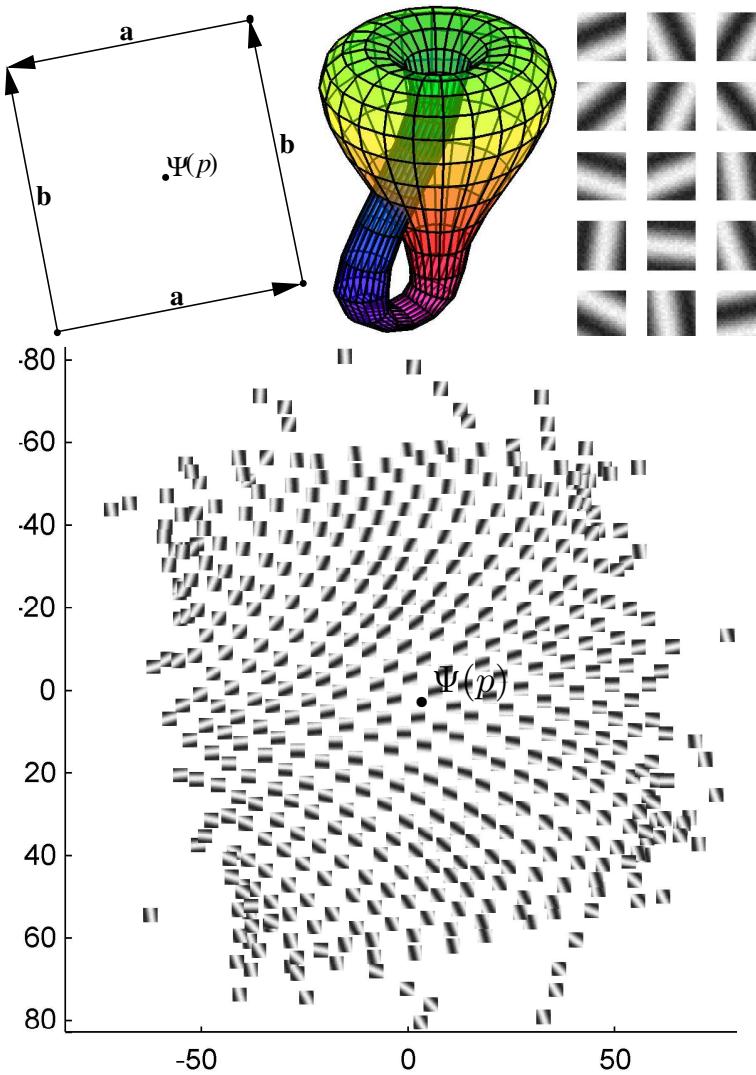


Figure 11.5: To test the proposed method on a high-dimensional data set, a set of 900 image patches, each being of 21×21 pixels with a characteristic orientation/phase, were generated and mapped to Riemannian normal coordinates. This experiment reveals the Klein bottle-structure of local orientation/phase in 2-D image patches! **Top left:** An idealized Klein bottle aligned to the mapping below. Edges correspond to the cut locus of p and should be identified according to the arrows. **Top middle:** An immersion of the Klein bottle in 3-D. **Top right:** 15 random examples of image patches used in the experiment. **Bottom:** The mapping of image patches to Riemannian normal coordinates using the proposed method.

Bibliography

- Barr, A. H., Currin, B., Gabriel, S., and Hughes, J. F. (1992). Smooth interpolation of orientations with angular velocity constraints using quaternions. *Computer Graphics*, 26(2):313–320.
- Basser, P. J. (1995). Inferring microstructural features and the physiological state of tissues from diffusion-weighted images. *NMR Biomed*, 8:333–344.
- Basser, P. J., Pajevic, S., Pierpaoli, C., Duda, J., and Aldroubi, A. (2000). In vivo fiber tractography using DT-MRI data. *Magnetic Resonance in Medicine*, 44:625–632.
- Behrens, T., Johansen-Berg, H., Woolrich, M., Smith, S., Wheeler-Kingshott, C., Boulby, P., Barker, G., Sillery, E., Sheehan, K., Ciccarelli, O., Thompson, A., Brady, J., and Matthews, P. (2003a). Non-invasive mapping of connections between human thalamus and cortex using diffusion imaging. *Nature Neuroscience*, 6(7):750–757.
- Behrens, T., Woolrich, M., Jenkinson, M., Johansen-Berg, H., Nunes, R., Clare, S., Matthews, P., Brady, J., and Smith, S. (2003b). Characterisation and propagation of uncertainty in diffusion weighted mr imaging. *Magn. Reson Med*, 50:1077–1088.
- Behrens, T. E. J. (2004). *MR Diffusion Tractography: Methods and Applications*. PhD thesis, University of Oxford.
- Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. In Diettrich, T. G., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems 14*, pages 585–591, Cambridge, MA. MIT Press.
- Bernstein, M., de Silva, V., Langford, J., and Tenenbaum, J. (2000). Graph approximations to geodesics on embedded manifolds. Technical report, Department of Psychology, Stanford University.
- Berry, M. W., Dumais, S. T., and O'Brien, G. W. (1995). Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):573–595.
- Bihan, D. L., Breton, E., Lallemand, D., Grenier, P., Cabanis, E., and Laval-Jeantet, M. (1986). Mr imaging of intravoxel incoherent motions: Application to diffusion and perfusion in neurologic disorders. *Radiology*, pages 401–407.

- Bishop, C. M., Svensen, M., and Williams, C. K. I. (1998). GTM: The generative topographic mapping. *Neural Computation*, 10(1):215–234.
- Björnemo, M., Brun, A., Kikinis, R., and Westin, C.-F. (2002). Regularized stochastic white matter tractography using diffusion tensor MRI. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'02)*, Tokyo, Japan.
- Blaas, J., Botha, C. P., Peters, B., Vos, F. M., and Post, F. H. (2005). Fast and reproducible fiber bundle selection in dti visualization. In *Proceedings of Visualization*. IEEE.
- Brand, M. and Huang, K. (2003). A unifying theorem for spectral embedding and clustering. In Bishop, C. M. and Frey, B. J., editors, *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*.
- Bregler, C. and Omohundro, S. M. (1994). Surface learning with applications to lipreading. In *Advances in Neural Information Processing Systems 6*, pages 43–50, San Francisco. Morgan Kaufmann.
- Brun, A., Björnemo, M., Kikinis, R., and Westin, C.-F. (2002). White matter tractography using sequential importance sampling. In *Proceedings of the ISMRM Annual Meeting (ISMRM'02)*, Honolulu, Hawaii.
- Brun, A., Knutsson, H., Park, H. J., Shenton, M. E., and Westin, C.-F. (2004). Clustering fiber tracts using normalized cuts. In *Seventh International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'04)*, pages 368 – 375, Rennes - Saint Malo, France. Springer, Berlin Heidelberg.
- Brun, A., Park, H.-J., Knutsson, H., and Westin, C.-F. (2003). Coloring of DT-MRI fiber traces using laplacian eigenmaps. In *Proceedings of the Ninth International Conference on Computer Aided Systems Theory (EUROCAST)*, volume 2809 of *Lecture Notes in Computer Science*.
- Brun, A., Westin, C.-F., Haker, S., and Knutsson, H. (2005a). A tensor-like representation for averaging, filtering and interpolation of 3-d object orientation data. In *IEEE International Conference on Image Processing (ICIP'05)*, Genoa, Italy.
- Brun, A., Westin, C.-F., Herberthson, M., and Knutsson, H. (2005b). Fast manifold learning based on riemannian normal coordinates. In *Proceedings of the 14th Scandinavian Conference on Image Analysis (SCIA'05)*, Joensuu, Finland.
- Catani, M., Howard, R. J., Pajevic, S., and Jones, D. K. (2002). Virtual in vivo interactive dissection of white matter fasciculi in the human brain. *NeuroImage*, 17:77–94.

- de Silva, V. and Tenenbaum, J. B. (2002). Global versus local methods in non-linear dimensionality reduction. In Becker, S., Thrun, S., and Obermayer, K., editors, *NIPS*, pages 705–712. MIT Press.
- Demartines, P. and Herault, J. (1997). Curvilinear component analysis: A self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks*, 8:1197–1206.
- Ding, Z., Gore, J. C., and Anderson, A. W. (2003). Classification and quantification of neuronal fiber pathways using diffusion tensor MRI. *Magnetic Resonance in Medicine*, 49:716–721.
- Donoho, D. L. and Grimes, C. (2003). Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *PNAS*, 100(10):5591–5596.
- Donoho, D. L. and Grimes, C. (2005). Image manifolds which are isometric to euclidean space. *J. Math. Imaging Vis.*, 23(1):5–24.
- Enders, F., Sauber, N., Merhof, D., Hastreiter, P., Nimsky, C., and Stamminger, M. S. (2005). Visualization of white matter tracts with wrapped streamlines. In *Proceedings of Visualization*. IEEE.
- Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Comm. of the ACM*, 24:381–395.
- Fletcher, P. T., Lu, C., Pizer, S. M., and Joshi, S. (2004). Principal geodesic analysis for the study of nonlinear statistics of shape. *IEEE Transactions on Medical Imaging*, 23(8):995–1005.
- Fletcher, P. T., S. Joshi, C. L., and Pizer, S. (2003). Gaussian distributions on lie groups and their application to statistical shape analysis. In *Proc. of Information Processing in Medical Imaging (IPMI)*, pages 450–462.
- Fowlkes, C., Belongie, S., and Malik, J. (2001). Efficient Spatiotemporal Grouping Using the Nyström Method. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2001*.
- Friedman, J. H. and Tukey, J. W. (1974). A projection pursuit algorithm for exploratory data analysis. *IEEE Trans. of Computers*, 23(9):881–890.
- Friman, O. and Westin, C.-F. (2005). Uncertainty in fiber tractography. In *Eighth International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'05)*, Lecture Notes in Computer Science 3749, pages 107–114, Palm Springs, CA, USA.
- Gaffney, S. and Smyth, P. (2003). Curve clustering with random effects regression mixtures. In Bishop, C. M. and Frey, B. J., editors, *Proc. Ninth Int. Workshop on AI and Stat.*, Florida.

- Gering, D. (1999). A System for Surgical Planning and Guidance using Image Fusion and Interventional MR. Master's thesis, Department of Electrical Engineering and Computer Science, MIT, Boston MA, USA.
- Gering, D. T., Nabavi, A., Kikinis, R., Grimson, W. E. L., Hata, N., Everett, P., Jolesz, F. A., and W. M. Wells, I. (1999). An integrated visualization system for surgical planning and guidance using image fusion and interventional imaging. In *MICCAI '99: Proceedings of the Second International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 809–819, London, UK. Springer-Verlag.
- Granlund, G. H. (1978). In search of a general picture processing operator. *Computer Graphics and Image Processing*, 8(2):155–178.
- Granlund, G. H. and Knutsson, H. (1995). *Signal Processing for Computer Vision*. Kluwer Academic Publishers. ISBN 0-7923-9530-1.
- Grassia, F. S. (1998). Practical parameterization of rotations using the exponential map. *J. Graph. Tools*, 3(3):29–48.
- Gudbjartsson, H. and Patz, S. (1995). The rician distribution of noisy mri data. *Magn Reson Med*, 34:910–914.
- Hagmann, P., Thiran, J.-P., Jonasson, L., Vandergheynst, P., Clarke, S., Maeder, P., and Meulib, R. (2003). DTI mapping of human brain connectivity: statistical fibre tracking and virtual dissection. *NeuroImage*, 19:545–554.
- Hastie, T. and Stuetzle, W. (1989). Principal curves. *J. Am. Stat. Assn.*, 84:872–876.
- Heiberg, E. B. (2001). *Automated Feature Detection in Multidimensional Images*. PhD thesis, Linköpings Universitet.
- Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:417–441, 498–520.
- Johnson, M. P. (2003). *Exploiting Quaternions to Support Expressive Interactive Character Motion*. PhD thesis, Massachusetts Institute of Technology.
- Johnstone, J. K. and Williams, J. P. (1999). A rational quaternion spline of arbitrary continuity. Technical report, Department of Computer and Information Sciences, University of Alabama at Birmingham. <http://www.cis.uab.edu/info/faculty/jj/cos.html>.
- Jonasson, L., Hagmann, P., Thiran, J.-P., and Wedeen, V. J. (2005). Fiber tracts of high angular resolution diffusion mri are easily segmented with spectral clustering. In *Proc. of the International Society for Magnetic Resonance Medicine (ISMRM)*, volume 13, Miami, Florida, USA.

- Jutten, C. and Herault, J. (1991). Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, (24):1–10.
- Karcher, H. (1977). Riemannian center of mass and millifier smoothing. *Commun. Pure Appl. Math*, 30(5):509–541.
- Karhunen, K. (1947). Über lineare methoden in der Wahrscheinlichkeitsrechnung. *Annales Academiae Scientiarum Fennicae, Series A1: Mathematica-Physica*, 37:3–79.
- Kayo, O. (2006). *Locally Linear Embedding Algorithm – Extensions and Applications*. PhD thesis, Department of Electrical and Information Engineering, University of Oulu. Draft version and personal communication.
- Kindlmann, G. (2004). Superquadric tensor glyphs. In *Proceedings of IEEE TVCG/EG Symposium on Visualization 2004*, pages 147–154.
- Kindlmann, G., Weinstein, D., and Hart, D. (2000). Strategies for direct volume rendering of diffusion tensor fields. *IEEE Transactions on Visualization and Computer Graphics*, 6(2):124–138.
- Knutsson, H. (1989). Representing local structure using tensors. In *The 6th Scandinavian Conference on Image Analysis*, pages 244–251, Oulu, Finland. Report LiTH-ISY-I-1019, Computer Vision Laboratory, Linköping University, Sweden, 1989.
- Knutsson, H., Andersson, M., Borga, M., and Wiklund, J. (2000). Automated generation of representations in vision. In *Proceedings of the 15th International Conference on Pattern Recognition*, volume 3, Barcelona, Spain. IAPR. Invited Paper.
- Knutsson, H. and Borga, M. (1999). Learning Visual Operators from Examples: A New Paradigm in Image Processing. In *Proceedings of the 10th International Conference on Image Analysis and Processing (ICIAP'99)*, Venice, Italy. IAPR. Invited Paper.
- Knutsson, H., Borga, M., and Landelius, T. (1998). Learning multidimensional signal processing. In *Proceedings of the 14th International Conference on Pattern Recognition*, volume II, pages 1416–1420, Brisbane, Australia. ICPR. (Also as report: LiTH-ISY-R-2039) Invited Paper.
- Kohonen, T. (1982). Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69.
- Kouby, V. E., Cointepas, Y., Poupon, C., Rivière, D., Golestani, N., Poline, J.-B., Bihan, D. L., and Mangin, J.-F. (2005). Mr diffusion-based inference of a fiber bundle model from a population of subjects. In Duncan, J. S. and Gerig, G., editors, *MICCAI*, volume 3749 of *Lecture Notes in Computer Science*, pages 196–204. Springer.

- Landelius, T. (1997). *Reinforcement Learning and Distributed Local Model Synthesis*. PhD thesis, Linköping University, Sweden, SE-581 83 Linköping, Sweden. Dissertation No 469, ISBN 91-7871-892-9.
- Landelius, T. and Knutsson, H. (1993). The learning tree, a new concept in learning. In *Proceedings of the 2:nd Int. Conf. on Adaptive and Learning Systems*. SPIE.
- Lee, J. and Shin, S. Y. (2002). General construction of time-domain filters for orientation data. *IEEE Transactions on Visualization and Computer Graphics*, 8(2):119–128.
- Lenz, R., Bui, T. H., and Hernandez-Andres, J. (2005). Group theoretical structure of spectral spaces. *Journal of Mathematical Imaging and Vision*, 23:297–313.
- Li, J. X. (2004). Visualization of high-dimensional data with relational perspective map. *Information Visualization*, 3(1):49–59.
- Lorentz, E. N. (1956). Empirical orthogonal function and statistical weather prediction, science report 1. Technical report, Department of meteorology, MIT. (NTIS AD 110268).
- Maddah, M., Mewes, A. U. J., and S. Haker, W. E. L. Grimson, S. K. W. (2005). White matter tract clustering and correspondence in populations. In *Proc. of MICCAI*, pages 140–147.
- McGraw, T., Vemuri, B. C., Wang, Z., Chen, Y., Rao, M., and Mareci, T. (2002). Line integral convolution for visualization of fiber tract maps from dti. In *MICCAI '02: Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part II*, pages 615–622, London, UK. Springer-Verlag.
- Meila, M. and Shi, J. (2001). A random walks view of spectral segmentation. In *Proceedings of the International Workshop on AI and Statistics(AISTATS)*, pages 177–182, Florida.
- Moakher, M. (2002). Means and averaging in the group of rotations. *SIAM Journal on Matrix Analysis and Applications*, 24(1):1–16.
- Moberts, B., Vilanova, A., and van Wijk, J. J. (2005). Evaluation of fiber clustering methods for diffusion tensor imaging. In *Proceedings of Visualization*. IEEE.
- Nadler, B., Lafon, S., Coifman, R., and Kevrekidis, I. (2006). Diffusion maps, spectral clustering and eigenfunctions of fokker-planck operators. In Weiss, Y., Schölkopf, B., and Platt, J., editors, *Advances in Neural Information Processing Systems 18*. MIT Press, Cambridge, MA.
- Nilsson, J. and Andersson, F. (2005). Nonlinear dimensionality reduction using

- circuit models. In *Proceedings of the 14th Scandinavian Conference on Image Analysis (SCIA'05)*, Joensuu, Finland.
- O'Donnell, L., Grimson, W. E. L., and Westin, C.-F. (2004). Interface detection in DTMRI. In *Seventh International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'04)*, Lecture Notes in Computer Science, pages 360–367, Rennes - Saint Malo, France.
- O'Donnell, L., Haker, S., and Westin, C.-F. (2002). New approaches to estimation of white matter connectivity in diffusion tensor MRI: Elliptic pdes and geodesics in a tensor-warped space. In *Fifth International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'02)*, pages 459–466, Tokyo, Japan.
- O'Donnell, L. and Westin, C.-F. (2005). White matter tract clustering and correspondence in populations. In *Proc. of MICCAI*, pages 140–147.
- O'Donnell, L. and Westin, C.-F. (2006). A high-dimensional fiber tract atlas. In *Proc. of the International Society for Magnetic Resonance Medicine (ISMRM)*, volume 14, Seattle, Washington, USA.
- Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:559–572.
- Pennec, X., Fillard, P., and Ayache, N. (2005). A riemannian framework for tensor computing. *International Journal of Computer Vision*, 65(1).
- Pierpaoli, C., Jezzard, P., Basser, P., Barnett, A., and Chiro, G. D. (1996). Diffusion tensor mr imaging of human brain. *Radiology*, 201.
- Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326.
- Schölkopf, B., Smola, A. J., and Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319.
- Seung, H. S. and Lee, D. D. (2000). The manifold ways of perception. *Science*, 290(5500):2268–2269.
- Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905.
- Shimony, J. S., Snyder, A. Z., Lori, N., and Conturo, T. E. (2002). Automated fuzzy clustering of neuronal pathways in diffusion tensor tracking. In *Proc. Intl. Soc. Mag. Reson. Med.*, volume 10, Honolulu, Hawaii.
- Shoemake, K. (1985). Animating rotation with quaternion curves. *SIGGRAPH Comput. Graph.*, 19(3):245–254.
- Spira, A., Kimmel, R., and Sochen, N. A. (2003). Efficient beltrami flow using a short time kernel. In *Scale-Space*, pages 511–522.

- Srivastava, A. and Klassen, E. (2002). Monte carlo extrinsic estimators for manifold-valued parameters. *Special issue of IEEE Transactions on Signal Processing*, 50(2):299–308.
- Strauss, W. A. (1992). *Partial differential equations: An introduction*. John Wiley & Sons Inc., New York.
- Swindale, N. V. (1996). Visual cortex: Looking into a klein bottle. *Current Biology*, 6(7):776–779.
- Tanaka, S. (1995). Topological analysis of point singularities in stimulus preference maps of the primary visual cortex. In *Proc. R. Soc. Lond. B*, volume 261, pages 81–88.
- Tenenbaum, J. B., de Silva, V., and Langford, J. C. (2000). A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2323.
- Torgerson, W. S. (1952). Multidimensional scaling. *Psychometrika*, 17:401–419.
- Turk, M. and Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86.
- Weisstein, E. W. (2005). Klein bottle. From MathWorld—A Wolfram Web Resource. <http://mathworld.wolfram.com/KleinBottle.html>.
- Westin, C.-F. (1991). Feature extraction based on a tensor image description. Thesis No. 288, ISBN 91-7870-815-X.
- Westin, C.-F., Maier, S. E., Khidhir, B., Everett, P., Jolesz, F. A., and Kikinis, R. (1999). Image processing for diffusion tensor magnetic resonance imaging. In *Medical Image Computing and Computer-Assisted Intervention*, Lecture Notes in Computer Science, pages 441–452.
- Westin, C.-F., Maier, S. E., Mamata, H., Nabavi, A., Jolesz, F. A., and Kikinis, R. (2002). Processing and visualization of diffusion tensor MRI. *Medical Image Analysis*, 6(2):93–108.
- Wikipedia (2005). Flag of the united nations — wikipedia, the free encyclopedia. [Online; accessed 1-February-2006].
- Young, G. and Householder, A. S. (1938). Discussion of a set of points in terms of their mutual distances. *Psychometrika*, 3:19–22.
- Zhang, S., Demiralp, Ç., and Laidlaw, D. H. (2003). Visualizing diffusion tensor mr images using streamtubes and streamsurfaces. *IEEE Transactions on Visualization and Computer Graphics*, 9(4):454–462.
- Zhang, Z. and Zha, H. (2002). Principal manifolds and nonlinear dimension reduction via local tangent space alignment.