Feature Extraction for Visual Analysis of DW-MRI Data

Dissertation zur Erlangung des Grades des
Doktors der Ingenieurwissenschaften
der Naturwissenschaftlich-Technischen Fakultäten
der Universität des Saarlandes

Vorgelegt durch

Thomas Schultz

MPI Informatik
Campus E1 4
66123 Saarbrücken
Germany

am 22. April 2009 in Saarbrücken

Betreuende Hochschullehrer – Supervisors

Prof. Dr. Hans-Peter Seidel
Max-Planck-Institut für Informatik, Germany

Prof. Dr. Holger Theisel
Otto-von-Guericke-Universität Magdeburg, Germany

Tag des Kolloquiums – Date of PhD defense

18. Juni 2009

Mitglieder des Prüfungsausschusses – Members of PhD committee

Vorsitz: Prof. Dr. Joachim Weickert
Universität des Saarlandes, Germany

1. Berichterstatter: Prof. Dr. Hans-Peter Seidel
Max-Planck-Institut für Informatik, Germany

2. Berichterstatter: Prof. Dr. Holger Theisel
Otto-von-Guericke-Universität Magdeburg, Germany

3. Berichterstatter: Prof. Dr. Gordon Kindlmann
University of Chicago, Illinois, USA

Vertreter der akademischen Mitarbeiter: Dr. Robert Strzodka
Max-Planck-Institut für Informatik, Germany

Dekan – Dean

Prof. Dr. Joachim Weickert
Universität des Saarlandes, Germany

Abstract

Diffusion Weighted Magnetic Resonance Imaging (DW-MRI) is a recent modality to investigate the major neuronal pathways of the human brain. However, the rich DW-MRI datasets cannot be interpreted without proper preprocessing. In order to achieve understandable visualizations, this dissertation reduces the complex data to relevant features.

The first part is inspired by topological features in flow data. Novel features reconstruct fuzzy fiber bundle geometry from probabilistic tractography results. The topological properties of existing features that extract the skeleton of white matter tracts are clarified, and the core of regions with planar diffusion is visualized.

The second part builds on methods from computer vision. Relevant boundaries in the data are identified via regularized eigenvalue derivatives, and boundary information is used to segment anisotropy isosurfaces into meaningful regions. A higher-order structure tensor is shown to be an accurate descriptor of local structure in diffusion data.

The third part is concerned with fiber tracking. Streamline visualizations are improved by adding features from structural MRI in a way that emphasizes the relation between the two types of data, and the accuracy of streamlines in high angular resolution data is increased by modeling the estimation of crossing fiber bundles as a low-rank tensor approximation problem.

Kurzzusammenfassung

Die Diffusions-Bildgebung erlaubt die Untersuchung von Nervenfaser-Verbindungen im menschlichen Gehirn. Sie erzeugt jedoch große Datenmengen, die ohne geeignete Vorverarbeitung kaum interpretiert werden können. Diese Dissertation erzeugt verständliche Visualisierungen der Daten, indem sie sie auf relevante Merkmale reduziert.

Der erste Teil orientiert sich an topologischen Methoden für Strömungsdaten. Neuartige Merkmale rekonstruieren mittels probabilistischer Traktographie unscharfe Faserbündelgrenzen. Die topologischen Eigenschaften bestehender Merkmale zur Extraktion von Skeletten der weißen Substanz werden aufgeklärt, und die Kernbereiche planarer Diffusion werden visualisiert.

Der zweite Teil ruht auf Methoden der Mustererkennung. Grenzstrukturen werden durch regularisierte Eigenwert-Ableitungen erkannt und Anisotropie-Isoflächen mittels Kantenerkennung in anatomisch bedeutsame Regionen segmentiert. Ein Strukturtensor höherer Stufe erweist sich als detaillierter Deskriptor der lokalen Struktur in Diffusion-Daten.

Der dritte Teil befasst sich mit Traktographie-Methoden. Die Stromlinien werden so durch Merkmale aus strukturellen MRT-Daten ergänzt, dass das räumliche Verhältnis zwischen beiden Datensätzen erkennbar wird. Die Genauigkeit von Stromlinien in Daten mit hoher Winkelauflösung wird erhöht, indem die Schätzung kreuzender Faserbündel als Approximation eines Tensors höherer Stufe modelliert wird.

Summary

During the past decade, Diffusion Weighted Magnetic Resonance Imaging (DW-MRI) has been used widely to investigate the white matter of the human brain. This dissertation presents techniques for automated extraction of features from DW-MRI data, which reduces the large datasets to relevant structural information, in order to produce comprehensible visualizations. These features span all scales of DW-MRI analysis, from edge detection and the analysis of orientation distribution functions (ODF) to larger-scale topological and extremal features.

Topological features have proven efficient for the visual analysis of flow fields. This dissertation demonstrates that an existing generalization of vector field topology to generic symmetric tensor fields does not produce useful results when applied to diffusion tensor (DT-MRI) data. Instead, novel topological features are proposed which are founded on the anatomical meaning of the data and reflect the uncertainty inherent in any connectivity estimate from diffusion imaging.

Anisotropy creases are a second type of features which capture the structural skeleton of diffusion data. This dissertation clarifies their properties and provides a reliable and efficient algorithm for their extraction. It shows that DT-MRI streamsurfaces, which were previously suggested to visually represent regions in which the diffusion tensor takes on planar shape, are not well-defined in general diffusion tensor data, and offers planarity ridges as an adequate replacement.

Several approaches to feature extraction in this dissertation are closely related to methods from image processing and computer vision. It is demonstrated how derivatives of eigenvalue functions can be used to distinguish several anatomically relevant types of edges in DT-MRI data. Filtered edge information is used to segment anisotropy isosurfaces into anatomically meaningful regions via an edge-based watershed transformation. To represent the local image structure of multivariate data such as DT-MRI fields more accurately, a higher-order structure tensor is proposed.

Often, diffusion images are not the only type of data acquired from a subject. This dissertation presents a method to visually integrate streamlines from DT-MRI fiber tracking with context from structural MRI. The method is modeled on an anatomical fiber preparation technique known as Klingler dissection. It demonstrates that opacity isosurfaces are expressive features in scalar fields and that they can be extracted and rendered efficiently.

Many types of features in DW-MRI data require to estimate the number, orientations, and volume fractions of individual nerve fiber tracts within a voxel. The final contribution of this dissertation is to improve the reliability of such estimates in cases where high angular resolution (HARDI) data is available. This is achieved by a low-rank approximation of higher order tensors, which are used to model the orientation distribution functions (ODFs) which stem from Q-Ball imaging and spherical deconvolution.

Zusammenfassung

Die Diffusions-Bildgebung hat sich innerhalb der letzten zehn Jahre als nützliches Instrument zur Untersuchung der weißen Substanz des menschlichen Gehirns erwiesen. Diese Dissertation entwickelt Verfahren zur automatischen Extraktion von Merkmalen, die geeignet sind, die umfangreichen Diffusions-Datensätze auf ihren strukturellen Gehalt zu reduzieren und verständliche bildliche Darstellungen zu erzeugen. Die vorgestellten Merkmale decken verschiedene Aspekte der Datenanalyse ab, von der Kantenerkennung und Analyse von Orientierungsdichtefunktionen bis hin zur Extraktion topologischer und struktureller Merkmale.

Topologische Merkmale werden häufig zur visuellen Analyse von Strömungsfeldern eingesetzt. Diese Dissertation zeigt, dass eine bestehende Generalisierung der Vektorfeld-Topologie auf generische symmetrische Tensorfelder nicht zur Analyse von Diffusions-Tensor (DT-MRI-)Daten geeignet ist. Statt dessen werden aus der anatomischen Bedeutung der Daten heraus neuartige topologische Merkmale entwickelt, die auch die Unsicherheit der aus Diffusions-Daten gewonnenen Verbindungsmaße berücksichtigen.

Extremalflächen von Anisotropie-Werten sind eine zweite Möglichkeit, ein strukturelles Skelett aus Diffusions-Daten zu extrahieren. Diese Dissertation klärt ihre topologischen Eigenschaften auf und entwickelt einen verlässlichen und effizienten Algorithmus zu ihrer Extraktion. Es wird gezeigt, dass DT-MRI-Stromflächen, die zuvor zur Visualisierung planarer Regionen vorgeschlagen wurden, mathematisch schlecht bestimmt sind, und dass Planaritäts-Maxima einen geeigneten Ersatz darstellen.

Einige der in dieser Dissertation behandelten Verfahren sind sowohl zur Visualisierung als auch zur Bildverarbeitung und Mustererkennung nützlich. So werden mittels der Ableitungen der Eigenwert-Funktionen anatomisch relevante Typen von Kanten in DT-MRI-Daten unterschieden. Mittels einer Wasserscheiden-Transformation und gefilterten Kanteninformationen werden anschließend Anisotropie-Isoflächen in anatomisch bedeutsame Bereiche segmentiert. Schließlich werden Strukturtensoren höherer Stufe eingeführt, die die lokale Struktur in multivariaten Daten (wie etwa DT-MRI) präziser repräsentieren.

Neben den Diffusions-Daten wird häufig auch ein struktureller MRT-Datensatz des gleichen Probanden angefertigt. Diese Dissertation stellt eine Möglichkeit vor, die Traktographie-Ergebnisse in der Form von Stromlinien im Kontext struktureller MRT-Daten darzustellen. Das Verfahren bildet die Klingler-Präparationsmethode nach und zeigt, dass Opazitäts-Isoflächen in Skalarfeldern ausdrucksstarke Merkmale sind, die effizient extrahiert und dargestellt werden können.

Viele Merkmale zur Interpretation von Diffusions-Daten erfordern eine Schätzung von Zahl, Ausrichtung und den jeweiligen Volumenanteilen einzelner Nervenfaserbündel. Der letzte Beitrag dieser Dissertation besteht darin, die Verlässlichkeit derartiger Schätzungen zu verbessern, wenn Daten hoher Winkelauflösung vorliegen. Hierzu wird die Orientierungsdichtefunktion eines Q-Ball oder Spherical Deconvolution-Modells als Tensor höherer Stufe dargestellt und durch einen Tensor niedrigen Rangs approximiert.

Danksagung

Niemand kann zwei Herren dienen, und so kann ich von Glück reden, dass meine beiden Betreuer Hans-Peter Seidel und Holger Theisel nie Gehorsam gefordert, sondern mich, auf sich wunderbar ergänzende Art, beständig ermutigt, angeregt und herausgefordert haben. Für Eure verlässliche Unterstützung danke ich Euch sehr herzlich.

Alfred Anwander hat diese Arbeit von Anfang an begleitet. Ich habe Dir nicht nur für die Bereitstellung von Datensätzen zu danken, sondern auch für Hilfe in hirnanatomischen Fragen, stimulierende Diskussionen und Literaturhinweise.

The work of Gordon Kindlmann has been a constant source of inspiration. Your research helped to spawn new ideas and your open software helped to put them into practice. I am glad that you agreed to review this thesis.

Several people have broadened my view by sharing their perspective on diffusion imaging or tensors. With particular fondness, I remember conversations with Joachim Weickert, Bernhard Burgeth, Lawrence Frank, and Michael Scheel.

Meine Zeit am MPI für Informatik ist ein Erlebnis, das ich nicht missen möchte, und ich danke allen, die auf die eine oder andere Weise zu dieser Tatsache beigetragen haben. Insbesondere haben mir Torsten Langer, Natascha Sauber, Carsten Stoll, Michael Wand und Rhaleb Zayer bei verschiedenen fachlichen Fragen zu größerer Klarheit verholfen. Ein positives Arbeitsklima ist kein Verdienst von Einzelnen. Dennoch danke ich stellvertretend Sabine Budde, die daran keinen kleinen Anteil hat.

Das Max Planck Center for Visual Computing and Communication hat diese Arbeit durch die Finanzierung meiner Stelle und meiner Reisen ermöglicht.

Ein gesunder wissenschaftlicher Rahmen allein reicht nicht aus, um die Arbeit an einer Dissertation zu bewältigen, und so möchte ich an dieser Stelle auch meinen Eltern danken, meinen Verwandten und allen Menschen, deren Freundschaft mir auch dann am Herzen liegt, wenn ich gerade einen Weltraumspaziergang in der Formelsphäre unternehme. Besonders Du, Katharina, schaffst es immer wieder, Anteil zu nehmen und zugleich dafür zu sorgen, dass die Bodenstation nie unbesetzt bleibt.

Contents

1. Introduction

Many methods of modern medical and neuroscientific research rely on imaging techniques whose results cannot be interpreted without adequate data processing, automated analysis, and visualization. One of these techniques is Diffusion Weighted Magnetic Resonance Imaging (DW-MRI), which provides a unique way to investigate the nerve fiber tracts ("white matter") within the human brain, by in vivo diffusion measurements [125, 164].

The microstructure of nerve fibers affects the diffusion behavior of water molecules, which can be measured by advanced MRI sequences. In particular, the cylinder-like geometry of nerve fibers constrains the molecular motion to a preferred direction parallel to the fiber bundle. Over the last 15 years, this has led to an increased interest in using non-invasive diffusion measurements for the diagnosis of stroke [205], for better understanding of nerve-related diseases [98, 128] and normal brain function [103, 126, 78, 45], as well as for improved planning of brain surgery [146].

Since the observed diffusion coefficients are directionally dependent, measurements have to be taken in a large number of directions to correctly capture the three-dimensional diffusion behavior. The diffusion tensor (DT-MRI) model [13] is used widely to summarize these measurements such that one can draw conclusions from them. It uses a symmetric 3×3 matrix to describe the diffusion process, leading to six degrees of freedom at each point in space. While this poses a challenge for the visualization already, it has recently become popular to use High-Angular Resolution Diffusion Imaging (HARDI), which acquires measurements in even more directions and employs more complex models [220, 76, 149, 219]. To convey an impression of the involved amount of data, Figure 1.1 presents all images that belong to a single two-dimensional slice in an example HARDI dataset.

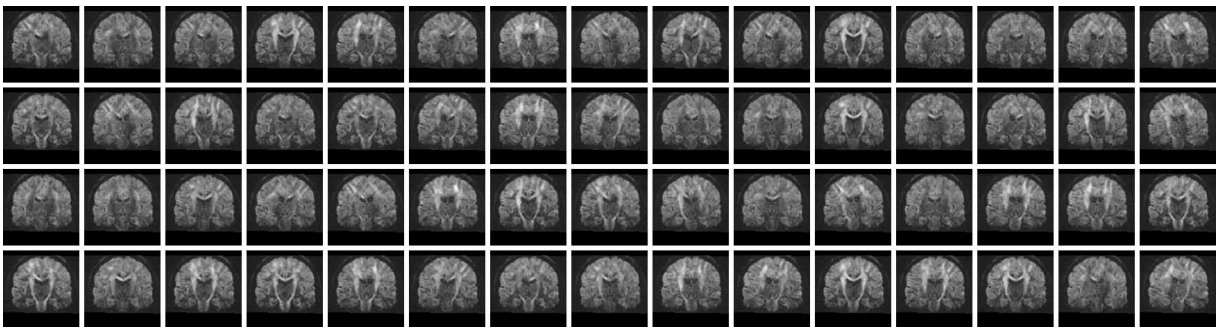


Figure 1.1.: These 60 images all show the same slice of the same brain, with different diffusion weightings. The total number of images in a single DW-MRI dataset is over 100 times greater than shown here.

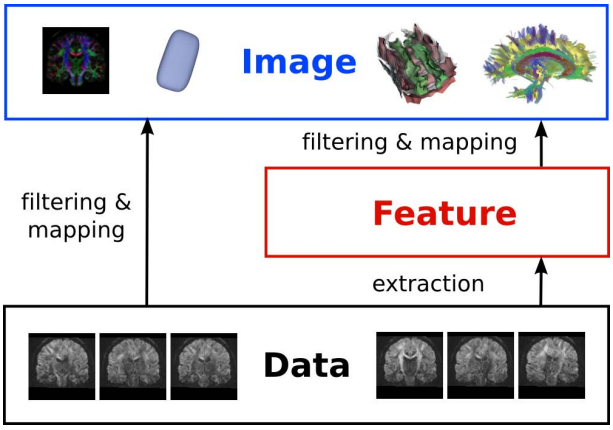


Figure 1.2.: Feature extraction reduces the given data to the relevant information and thus introduces a layer of abstraction between the data and the final visualization.

The goal of scientific visualization is to create digital images that represent scientific data in a way that it can be explored, understood, and communicated by human experts. In complex data, it is impossible to convey all available information at the same time, so visualization involves abstraction. Prior to our work, visualization of diffusion tensors was primarily achieved by mapping attributes of the tensor to shape, color, or texture of a visual representation (cf. Section 2.2.5). In such a framework, the abstraction is performed by mapping only some tensor attributes, or by mapping them only at certain locations.

In contrast, the general strategy pursued in this dissertation is to extract features as a separate layer of abstraction between the data itself and its visualization (Figure 1.2). In accordance with [166], we define features as "phenomena, structures or objects in a data set that are of interest for a certain research or engineering problem". As opposed to attributes, which are defined from individual tensors by simple formulas whose solutions are unique and easy to compute (cf. Section 2.2.4), feature extraction takes into account neighborhoods or even the complete dataset to detect non-local structures. Feature definitions often include model assumptions, and their extraction frequently involves problems like automated pattern recognition, segmentation, or clustering, which are computationally expensive and for which unique solutions may not exist.

Feature-based visualization shifts part of the burden of analyzing complex data to the computer. It offers the chance of reducing large datasets to the important information, and allows researchers to concentrate on the aspects relevant to their scientific problem. However, improper use of features carries the risk of hiding relevant details, or creating artifacts that stem from post-processing rather than from the data itself. In our work, we address these risks by careful evaluation: Results are compared to previous methods, inspected by experts, and, where possible, validated against ground truth.

Most of the methods presented in this dissertation concentrate on the diffusion tensor model. On one hand, DT-MRI data is complex enough so that the benefit from feature-based methods becomes evident. On the other hand, the basic processing and in-

terpretation of second-order tensors is understood well enough to allow for a well-founded treatment of higher level concepts like features. However, we expect that once the foundations for HARDI processing have been laid, the data reduction which is made possible by feature-based methods will play an even larger role for the visualization of HARDI data, since it contains up to one order of magnitude more information than a comparable DT-MRI dataset. Chapter 10 constitutes an initial step into this direction.

1.1. Outline

Individual feature definitions are specific to particular applications or problems. Therefore, it is natural to develop a variety of complementary methods. The ones proposed in this dissertation fall into three groups:

1. *Structural features* reduce the data to the cores and boundaries of regions that exhibit equivalent qualitative properties.

   Similar to the way in which a topological skeleton partitions a flow field into regions that connect the same source to the same sink (cf. Section 2.3.5), the topological features for diffusion MRI in Chapter 3 indicate regions that connect functionally distinct areas in the human brain.

   Anisotropy crease surfaces have been shown to represent the cores and boundaries of white matter structures [114]. In Chapter 4, the properties of these surfaces are clarified, and in Chapter 5, they are applied to the analysis of planar regions.

2. Often, methods for feature extraction in visualization are closely related to algorithms from *image processing* and *computer vision.*

   In Chapter 6, regularized eigenvalue derivatives are introduced as a way to detect different types of edges in DT-MRI data, and in Chapter 7, filtered boundary information is exploited to segment isosurfaces of diffusion anisotropy into anatomically meaningful regions.

   A descriptor of the local structure in DT-MRI data is proposed in Chapter 8 and visualized with a novel glyph for supersymmetric higher-order tensors. This feature not only allows for the visualization of local variability, but it also holds the potential to find its way back into image processing and computer vision.

3. *Fiber tracking* aims at reconstructing the trajectory of nerve fiber bundles, and is one of the most popular methods for the visualization of DW-MRI data.

   In Chapter 9, streamlines from fiber tracking are augmented with context from a coregistered structural MRI scan to indicate their relation to anatomical landmarks. The method is based on a novel type of features in scalar data and combines two three-dimensional datasets in a single intuitive rendering.

   To support fiber tracking in HARDI data, an improved method for the estimation of fiber directions from orientation distribution functions (ODFs) is presented in

Chapter 10. The resulting directions can be considered as features of the underlying ODFs. At the same time, they lay the foundation for future work on the feature-based visualization of HARDI data.

Most of the material presented in this dissertation has successfully undergone peer review [191, 192, 188, 190, 189, 194, 193]. Beside the summary of background material in Chapter 2, as well as an improved exposition and an updated discussion of related work throughout the thesis, details that were previously unpublished are the three-dimensional glyphs, the formal proof, and the results on diffusion tensor data in Chapter 8.

1.2. Data and Software

Most experiments in this dissertation have been performed on a dataset which was provided by the Max Planck Institute for Human Cognitive and Brain Science (Leipzig). It consists of diffusion-weighted images (DWIs) acquired on a Siemens 3T Trio scanner at $b = 1000\,\mathrm{s/mm^2}$ in 60 isotropically distributed gradient directions (3 averages each), plus one non-diffusion weighted T_2 image (7 averages), voxel size 1.72 mm (isotropic). We received the images pre-registered to compensate motion and imaging artifacts, but we have performed the diffusion modeling ourselves.

While it is not ideal to rely on a single dataset for testing, it is common in the context of visualization research. This is only partly due to the fact that visualization as a discipline concentrates more on developing novel concepts than on fine-tuning implementations to a point at which they become polished products for domain scientists. Without access to a scanner, it is difficult to obtain a larger number of datasets for comparison. Therefore, we gained additional insight by creating synthetic data with varying model assumptions, and it is our hope that as part of our future work, we will be able to validate at least some of the presented methods on a more representative set of real measurements.

Chapter 4 presents results on a CT dataset[1] which is courtesy of Bernd Tomandl (Division of Neuroradiology, University of Erlangen). Figure 9.1 (d) is reproduced from [133] with the kind permission of S. Karger AG, Basel.

In our implementation, we have made frequent use of the teem library[2]. Some of our experiments have also employed camino[3], BioTensor[4], and CImg[5]. Inkscape[6] and the GIMP[7] were used to create most illustrations.

[1] Available from `http://www.volvis.org/`
[2] Available from `http://teem.sf.net/`
[3] Available from `http://www.cs.ucl.ac.uk/research/medic/camino/`
[4] Available from `http://software.sci.utah.edu/`
[5] Available from `http://cimg.sf.net/`
[6] Available from `http://www.inkspace.org/`
[7] Available from `http://www.gimp.org/`

2. Background

The background material in this chapter is taken from three fields: *Tensors* (Section 2.1) are used as a mathematical tool for modeling our data. A good understanding of *diffusion imaging* (Section 2.2) is essential to design features that make sense in the context of this field of application. *Topological methods* (Section 2.3) are a popular example of feature-based visualization, and will serve as the starting point for some of our own methods.

Occasionally, we will also make use of some basic notions from *differential geometry:* Chapter 5 mentions the second fundamental form of a surface, and Section 8.3.2 uses the parametric form of a curve and the definition of curvature. However, this is not within our primary focus, so the interested reader is referred to [169] for details.

2.1. Tensors

This section gives an introduction to tensors and some important concepts related to them, starting from vectors (Section 2.1.1) and ranging from tensor definition (Sections 2.1.2 and 2.1.3), (multi)linear and homogeneous forms (Sections 2.1.4 and 2.1.5) to results on tensor decomposition (Section 2.1.6) and approximation (Section 2.1.7). A clear focus is on the background needed to understand the main part of this dissertation, but occasionally, definitions are given in slightly greater generality than absolutely required, to make it easier for the reader to get into the specified literature.

We assume some concepts which can be found in introductory texts on linear algebra [44]. This includes standard matrix operations like matrix multiplication $\mathbf{MN}$, transpose $\mathbf{M}^{\mathrm{T}}$, inverse $\mathbf{M}^{-1}$, trace $\mathrm{tr}(\mathbf{M})$, determinant $\det(\mathbf{M})$, and Frobenius norm $\|\mathbf{M}\|$. Italics a denote scalars, boldface $\mathbf{a}$ indicates vectors, upper-case boldface $\mathbf{A}$ denotes second-order tensors, and upper-case calligraphic letters $\mathcal{A}$ are used for general tensors. A function f from spaces V and W to Y will be written $f : V \times W \to Y$. Further notational conventions are introduced along with the respective definitions.

2.1.1. Vectors and Coordinates

Even though a vector $\mathbf{v}$ is defined in an abstract way, as an element of some vector space V, computations on vectors usually rely on a component representation which follows from the introduction of a maximal set of linearly independent basis vectors $\{\mathbf{b}_i \,|\, i = 1, \ldots, n\}$, where n is the dimension of V. Each vector $\mathbf{v}$ has two kinds of components. Its *contravariant* components v^i are the unique coefficients required to represent $\mathbf{v}$ as a linear combination of the basis vectors:

$$\mathbf{v} = v^i \mathbf{b}_i \tag{2.1}$$

Equation (2.1) and the remainder of this section employ the *Einstein summation convention,* according to which using an index exactly twice in the same expression implies summation over all its possible values. When co- and contravariant components are involved, summation has to happen over one co- and one contravariant index.

By convention, contravariant components v^i are used to describe elements from V and are arranged as *column vectors*. We write this representation of $\mathbf{v}$ as $[\mathbf{v}]_{\mathbf{B}}$, where $\mathbf{B}$ denotes the basis with respect to which the components are defined. In case of the canonical basis of $\mathbb{R}^n$, we simplify the notation and let $\mathbf{v}$ denote both the (abstract) vector itself and the column vector of its components. The distinction will always be clear from the context.

In the remainder of this dissertation, we assume that all vector spaces are equipped with an inner product, which defines a set of *covariant* components v_i by projecting $\mathbf{v}$ onto $\mathbf{b}_i$:

$$v_i = \mathbf{v} \cdot \mathbf{b}_i \tag{2.2}$$

Covariant components are used to describe linear functionals from V to the underlying field F, and are written as *row vectors*. These linear mappings $V \to F$ form the *dual* vector space V^*.

When changing from one basis to another, co- and contravariant components follow different transformation rules. However, it is possible to convert contravariant components to covariant ones and vice versa via the metric tensor G_{ij} and its inverse G^{ij}, respectively:

$$v_i = G_{ij} v^j \qquad \text{with} \qquad G_{ij} = \mathbf{b}_i \cdot \mathbf{b}_j \tag{2.3}$$

When the basis is orthonormal, the metric tensor equals the unit tensor, so co- and contravariant components coincide. Since all applications in this dissertation assume orthonormal bases, we will ignore the distinction between co- and contravariant components from now on, and write all indices as subscripts.

A matrix $\mathbf{R}$ that transforms the components v_i of a vector $\mathbf{v}$ with respect to some basis $\{\mathbf{b}_i\}$ to components $\widetilde{v}_i$ with respect to a new basis $\{\widetilde{\mathbf{b}}_i\}$ via $[\mathbf{v}]_{\widetilde{\mathbf{B}}} = \mathbf{R}[\mathbf{v}]_{\mathbf{B}}$ is formed by the column vectors that contain the components of the old basis vectors in terms of the new ones. In this dissertation, both $\{\mathbf{b}_i\}$ and $\{\widetilde{\mathbf{b}}_i\}$ are orthonormal and share the same orientation (i.e., right-handed in $\mathbb{R}^3$), so the resulting transformation is a rotation, and the corresponding matrix $\mathbf{R}$ is orthogonal with $\det(\mathbf{R}) = 1$.

2.1.2. Second-Order Tensors

In full generality, the *tensor product* of two vector spaces V and W over a common field F is defined as a new vector space $V \otimes W$ over F and a bilinear map $\bigotimes : V \times W \to V \otimes W$ such that for each bilinear function $f : V \times W \to Y$ to any vector space Y, there exists a unique linear map $\hat{f} : V \otimes W \to Y$ such that $f = \hat{f} \circ \bigotimes$. Using $\mathbf{v} \otimes \mathbf{w}$ as shorthand notation for $\bigotimes(\mathbf{v}, \mathbf{w})$, this condition can be written as $f(\mathbf{v}, \mathbf{w}) = \hat{f}(\mathbf{v} \otimes \mathbf{w})$. For clarity, a schematic illustration is given in Figure 2.1.

All tensor products of two given vector spaces are isomorphic to each other in a natural way (i.e., without making arbitrary choices) [61]. This makes it legitimate to let the term

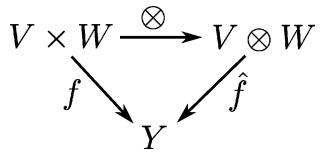


Figure 2.1.: The tensor product allows one to replace a bilinear map $f : V \times W \to Y$ by an equivalent linear map $\hat{f} : V \otimes W \to Y$ from a product space $V \otimes W$.

"tensor product" also denote the bilinear map $\bigotimes$, and to introduce it in a slightly less general, but more intuitive way [33].

Let $\mathbf{v}$ and $\mathbf{w}$ be elements from inner product spaces V and W, respectively, both over a common field F. Their tensor product is the *dyad* $\mathbf{v} \otimes \mathbf{w}$, and is defined by the fact that inner multiplication with any vector $\mathbf{x}$ from W produces a vector from V, according to

$$(\mathbf{v} \otimes \mathbf{w}) \cdot \mathbf{x} = \mathbf{v}\,(\mathbf{w} \cdot \mathbf{x}) \tag{2.4}$$

Based on this, a *dyadic* is defined as any sum of dyads. Since the dyadics form a vector space over F and the mapping defined by Equation (2.4) fulfills the requirements of the bilinear map $\bigotimes$ in the general definition of the tensor product, dyadics are nothing but *second-order tensors* from $V \otimes W$. Since the vectors that form the individual dyads can be expressed in bases $\{\mathbf{b}_i^V \mid i = 1, \ldots, n\}$ and $\{\mathbf{b}_j^W \mid j = 1, \ldots, m\}$ of V and W, respectively, a second-order tensor $\mathbf{T}$ can be written in terms of $n \times m$ components T_{ij} such that

$$\mathbf{T} = T_{ij}\mathbf{b}_i^V \otimes \mathbf{b}_j^W \tag{2.5}$$

By convention, the components T_{ij} are arranged in a matrix. Assuming column vectors and the standard rules of matrix-vector multiplication, $\mathbf{v} \otimes \mathbf{w}$ can be written as $\mathbf{v}\mathbf{w}^{\mathrm{T}}$, and multiplication of tensor $\mathbf{T}$ with vector $\mathbf{v}$ as $\mathbf{T}\mathbf{v}$. Finally, the coordinate transformation expressed by matrix $\mathbf{R}$ in the previous section is applied to $\mathbf{T}$ by $[\mathbf{T}]_{\widetilde{\mathbf{B}}} = \mathbf{R}[\mathbf{T}]_{\mathbf{B}}\mathbf{R}^{\mathrm{T}}$.

2.1.3. Higher-Order Tensors

Higher-order tensors arise when at least one of the vector spaces of which a tensor product is taken is the result of a tensor product already. The order l of the resulting tensors equals the sum of the orders of the two spaces involved in its definition, where vectors count as first-order tensors. As an example, consider a *triad* $\mathbf{v} \otimes \mathbf{w} \otimes \mathbf{y}$ of vectors from V, W and Y, which results from taking the tensor product of $V \otimes W$ with Y. It is defined by the fact that inner multiplication with a vector $\mathbf{x}$ from Y produces a second-order tensor:

$$(\mathbf{v} \otimes \mathbf{w} \otimes \mathbf{y}) \cdot \mathbf{x} = \mathbf{v} \otimes \mathbf{w}\,(\mathbf{y} \cdot \mathbf{x}) \tag{2.6}$$

Again, sums of triads are called *triadics,* and they are the same as *third-order tensors.* This definition generalizes to orders higher than three in the obvious way and results in *polyads,* whose sums are called *polyadics,* or *higher-order tensors.* Since the components of an order-l tensors are addressed by l indices, higher-order tensors and their operations

can no longer be expressed in standard matrix notation. However, the Frobenius norm generalizes naturally to order-l tensors $\mathcal{T}$:

$$\|\mathcal{T}\| = \sqrt{T_{i_1 i_2 \ldots i_l} T_{i_1 i_2 \ldots i_l}} \tag{2.7}$$

2.1.4. Linear Forms and Tensor Rank

From our definition of tensors in Equations (2.4) and (2.6), it is clear that order-l tensors can be interpreted as linear forms that map vectors to tensors of order $(l-1)$. In fact, a tensor corresponds to l different linear mappings, since we may choose to sum over any of its l indices. This is called the inner product along the mth *mode* of the tensor, and is written $\mathbf{T} \cdot_m \mathbf{x}$. Let $\mathbf{x}^V$, $\mathbf{x}^W$ and $\mathbf{x}^Y$ be vectors from V, W, and Y, respectively. Then, for the triad from the previous section, the possible tensor-vector products are:

$$\begin{aligned}
(\mathbf{v} \otimes \mathbf{w} \otimes \mathbf{y}) \cdot_1 \mathbf{x}^V &= \mathbf{w} \otimes \mathbf{y} \left(\mathbf{v} \cdot \mathbf{x}^V\right) \\
(\mathbf{v} \otimes \mathbf{w} \otimes \mathbf{y}) \cdot_2 \mathbf{x}^W &= \mathbf{v} \otimes \mathbf{y} \left(\mathbf{w} \cdot \mathbf{x}^W\right) \\
(\mathbf{v} \otimes \mathbf{w} \otimes \mathbf{y}) \cdot_3 \mathbf{x}^Y &= \mathbf{v} \otimes \mathbf{w} \left(\mathbf{y} \cdot \mathbf{x}^Y\right)
\end{aligned} \tag{2.8}$$

In linear algebra, the *rank* r of a linear mapping is defined as the dimension of its image. This notion of rank is transferred to tensors via their corresponding linear mappings. In the second-order case, the ranks along the first and second mode are equal, which corresponds to the fact that row and column rank of a matrix coincide. In the general higher-order case, this is no longer true [94, 53], so tensor rank in the linear algebra sense becomes a tuple $(r_1, r_2, \ldots, r_l)$.

When all involved vectors are non-zero, polyads have rank $(1, 1, \ldots, 1)$, so they are referred to as rank-1 tensors. Based on them, a unique tensor rank r is defined as the minimum number of rank-1 terms such that the given tensor can be written as a sum of them [94]. As opposed to the "linear algebra rank", we will call this number r "decomposition rank" or simply "rank" of a tensor. In the second-order case, both notions coincide.

In part of the literature (including [149]), "tensor rank" is used to denote the number of indices required to address the tensor components. In this dissertation, that number is always called "tensor order", and "rank" is strictly reserved for the two definitions given above.

2.1.5. Multilinear and Homogeneous Forms

Taking the inner product of an order-l tensor with a vector produces a tensor of order $(l-1)$, which can, again, be interpreted as a linear form, now mapping vectors to tensors of order $(l-2)$. This way, the tensor-vector product can be repeated l times, until a scalar (a "zeroth-order tensor") is left. Since each step is linear, the overall mapping from l vectors to a scalar is a *multilinear form*.

In the remainder of this dissertation, all vector spaces which are involved in tensor products coincide. Thus, one may identify all arguments of such multilinear forms. In the

second-order case, the scalar-valued form is bilinear, and identification of its arguments leads to a *quadratic form.* Writing out its component notation in $\mathbb{R}^2$ yields:

$$\mathbf{v}^{\mathrm{T}}\mathbf{M}\mathbf{v} = M_{11}v_1^2 + (M_{12} + M_{21})\, v_1 v_2 + M_{22}v_2^2 \tag{2.9}$$

From Equation (2.9), it is clear that the components of a second-order tensor that represents a quadratic form can be chosen such that they are *symmetric* ($M_{12} = M_{21}$). We let Sym_n denote the space of symmetric $n \times n$ matrices. In the general case, identifying the arguments of the scalar-valued form of a tensor $\mathcal{T}$ produces a *homogeneous form* $T(\mathbf{v})$:

$$T(\mathbf{v}) = T_{i_1 i_2 \ldots i_l} v_{i_1} v_{i_2} \cdots v_{i_l} \tag{2.10}$$

The components of a tensor that represents a homogeneous form can be chosen such that they are invariant under any index permutation. This has been called *total symmetry* [141] or *supersymmetry* [37, 121, 170]. Other authors argue that this concept is a straightforward generalization of matrix symmetry and should not be given a new name [46]. However, unlike matrices, higher-order tensors may exhibit different types of symmetries [141]. In particular, the fourth-order covariance tensors that arise in the context of DT-MRI [14, 113] possess a certain set of symmetries, but are not invariant under all index permutations.

In the case of supersymmetric tensors, the mode along which an inner product is taken does not play a role, so it does not have to be specified as a subscript. Instead, the number of times the product is repeated can be written as a superscript. In this notation, Equation (2.10) reads $T(\mathbf{v}) = \mathcal{T} \cdot^l \mathbf{v}$.

Tensor order l determines the number of vector components that enter the homogeneous form, which leads to *antipodal symmetry* $T(-\mathbf{v}) = T(\mathbf{v})$ for even l, and to *antisymmetry* $T(-\mathbf{v}) = -T(\mathbf{v})$ for odd l. When the underlying vector space is $\mathbb{R}^2$, the homogeneous polynomial can be restricted to the unit circle, resulting in a periodic function on the real line. It can be shown that for tensors with even order l, the resulting function space is the same as the one of Fourier series which are truncated after order l, and in which all odd coefficients are set to zero. In $\mathbb{R}^3$, the corresponding result is an equivalence of homogeneous polynomials on the unit sphere and truncated spherical harmonics. Details on this conversion are given in Appendix A.

When the homogeneous form of a tensor $\mathcal{T}$ is positive (non-negative) for all vectors $\mathbf{v}$, $\mathcal{T}$ is called *positive definite (positive semidefinite).*

2.1.6. Eigenvectors and Tensor Decomposition

A *unit eigenvector* of a second-order tensor $\mathbf{T}$ is a unit vector $\mathbf{e}$ whose direction is invariant under tensor-vector multiplication, $\mathbf{T}\mathbf{e} = \lambda\mathbf{e}$. The corresponding scalar λ is called an *eigenvalue* of $\mathbf{T}$ and found as a root of the *characteristic equation*

$$\det(\lambda\mathbf{I} - \mathbf{T}) = 0 \tag{2.11}$$

The same scalar may be counted multiple times as an eigenvalue, according to the number of times it appears as a root of Equation (2.11). Eigenvectors to an eigenvalue of

multiplicity m form an m-dimensional vector space. In this dissertation, eigenvalues are sorted such that $\lambda_1 \geq \lambda_2 \geq \lambda_3$.

Symmetric tensors $\mathbf{T}$ over $\mathbb{R}^n$ can be written in terms of n real eigenvalues λ_i and n pairwise orthogonal eigenvectors $\mathbf{e}_i$:

$$\mathbf{T} = \sum_{i=1}^{n} \lambda_i \mathbf{e}_i \otimes \mathbf{e}_i \tag{2.12}$$

Equation (2.12) is known as the *eigendecomposition* or the *spectral decomposition* of the tensor. The orthonormal eigenvectors can be used as the rows of a matrix that rotates the tensor into its *eigenframe,* in which it takes on diagonal form.

When the quadratic form of a second-order tensor is restricted to the unit sphere, the eigenvectors indicate its *stationary points,* in which the gradient magnitude of the spherical function vanishes. The corresponding eigenvalues indicate the function values at these maxima, minima, and saddle points. Consequently, the tensor is positive (semi)definite as defined in Section 2.1.5 if and only if all eigenvalues are positive (non-negative).

For a tensor of order l, the partial derivative of the homogeneous form T with respect to axis x_j is given by

$$\frac{\partial T(\mathbf{v})}{\partial x_j} = l \times T_{j i_1 i_2 \ldots i_{l-1}} v_{i_1} v_{i_2} \cdots v_{i_l - 1} \tag{2.13}$$

The restriction of the homogeneous form T to the unit sphere has a stationary point when $\nabla T(\mathbf{v})$ aligns with $\mathbf{v}$. Thus, these stationary points are given by the system

$$\begin{cases} \mathcal{T} \cdot^{l-1} \mathbf{e} & = \lambda \mathbf{e} \\ \|\mathbf{e}\| & = 1 \end{cases} \tag{2.14}$$

Due to the analogy of Equation (2.14) with the eigenvector definition in the second-order case, vectors $\mathbf{e}$ that fulfill it have been called Z-eigenvectors of $\mathcal{T}$ [170]. The corresponding Z-eigenvalues λ equal the value of the homogeneous form at the stationary point and thus reveal its definiteness. However, Z-eigenvectors are generally not pairwise orthogonal, and they do not determine the higher-order tensor uniquely.

A decomposition of higher-order tensors into a minimum number of r rank-1 terms has first been considered by Hitchcock [94], and was later rediscovered in the context of psychometrics [222], where it became known as *canonical decomposition* [39] or *parallel factor analysis* [88]. For an order-l tensor $\mathcal{T}$, it reads:

$$\mathcal{T} = \sum_{i=1}^{r} \lambda_i \mathbf{e}_{i_1} \otimes \mathbf{e}_{i_2} \otimes \cdots \otimes \mathbf{e}_{i_l} \tag{2.15}$$

If $\mathcal{T}$ is symmetric, one may identify $\mathbf{e}_{i_1} = \mathbf{e}_{i_2} = \cdots = \mathbf{e}_{i_l}$ to obtain a *supersymmetric canonical decomposition* [46] which is analogous to the spectral decomposition in Equation (2.12), except that the involved vectors are no longer pairwise orthogonal. Unfortunately, algorithms for practical computation of the canonical decomposition are rare. Comon and Mourrain [47] present an implementation for tensors over $\mathbb{R}^2$, but the problem in $\mathbb{R}^3$ is still open.

2.1.7. Low-Rank Tensor Approximation

Given an order-l tensor $\mathcal{T}$ of decomposition rank r, the task of rank-k tensor approximation is to find a tensor $\widetilde{\mathcal{T}}$ of rank $k < r$ which minimizes the distance to $\mathcal{T}$ in terms of the Frobenius norm:

$$\arg\min_{\widetilde{\mathcal{T}}} \|\mathcal{T} - \widetilde{\mathcal{T}}\| \quad \text{such that} \quad \widetilde{\mathcal{T}} = \sum_{i=1}^{k} \lambda_i \mathbf{v}_{i_1} \otimes \mathbf{v}_{i_2} \otimes \cdots \otimes \mathbf{v}_{i_l} \tag{2.16}$$

In a supersymmetric approximation, vectors $\mathbf{v}_{i_1} = \mathbf{v}_{i_2} = \cdots = \mathbf{v}_{i_l}$ are identified. For second-order tensors, a best rank-k approximation can be found via the *singular value decomposition* [231], which represents any real $n \times m$ matrix $\mathbf{M}$ in terms of an orthogonal $n \times n$ matrix $\mathbf{U}$, an orthogonal $m \times m$ matrix $\mathbf{V}$, and a diagonal $n \times m$ matrix $\mathbf{\Sigma}$ that holds non-negative *singular values* σ_i:

$$\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathrm{T}} \tag{2.17}$$

For symmetric, positive semidefinite matrices, the singular value decomposition coincides with the eigendecomposition. An optimal rank-k approximation is found by setting all but the k largest σ_i to zero and multiplying out Equation (2.17). Since Eckart and Young first formulated this theorem in modern matrix notation [65], it is commonly referred to as "Eckart-Young's theorem". However, Stewart [207] suggests that the result should rather be attributed to Schmidt [186], who had previously formulated an equivalent theorem in terms of integral equations.

Unfortunately, this elegant result does not carry over to higher-order tensors: Even when a canonical decomposition can be obtained, truncating it is not guaranteed to give a best rank-k approximation. De Silva and Lim [54] even pointed out that in the topology generated by the Frobenius norm, the set of tensors of rank $r \leq k$ is not closed when order $l > 2$ and rank $k > 1$. Thus, sequences of rank-k tensors may converge to a tensor of rank $r > k$, which consequently does not have an optimal rank-k approximation.

2.2. Diffusion Imaging

This section introduces our field of application, diffusion-weighted imaging of the human brain. Our presentation of diffusion and MRI physics (Sections 2.2.1 and 2.2.2) is limited to the basics needed to understand the modeling issues detailed in Sections 2.2.3, 2.2.6, 2.2.7, and 2.2.8. In particular, we describe the behavior of protons in the presence of strong magnetic fields from the perspective of classical physics, which is sufficient to explain the observations of larger ensembles, as those found in the volume elements (voxels) of magnetic resonance imaging. For the quantum mechanical description required to understand the process on a per-atom basis, the interested reader is referred to [232].

For the diffusion tensor model, some scalar measures (Section 2.2.4) and visualization techniques (Section 2.2.5) are introduced which are important tools for the interpretation of the data. More general introductions on the acquisition, processing, and visualization of diffusion tensor data can be found in references [167, 224, 252].

2.2.1. Diffusion in the Human Brain

Diffusion is a process in which random molecular heat motion causes a net transport of molecules from a region of higher concentration to one of lower concentration. The concentration gradient does not, however, apply a force to the individual molecules, and neither does it impose a preferred direction of motion on them. During a given period of time, the same fraction of molecules from the lower concentration compartment is transported to the one with higher concentration as vice versa. It is only due to the larger number of molecules in the high concentration compartment that the absolute number of molecules that are transported from high to low concentration prevails.

This net mass transport is described by Fick's laws [70, 49]. In the case of free isotropic diffusion, Fick's first law relates the net chemical flux $\mathbf{f}$ (amount of substance per area and time) to the concentration gradient ∇c via the diffusion coefficient d:

$$\mathbf{f} = -d\nabla c \tag{2.18}$$

When no concentration gradient is present, the net flow vanishes, but the diffusion process itself does not stop: Individual molecules still perform their heat motion. This process is called *self-diffusion* and is what is measured in the context of diffusion imaging.

Within the brain, self-diffusion is not free, but affected by tissue microstructure. This introduces a dependence of the measured diffusion coefficient on *diffusion time:* When observing the movement of a molecule for such a short time that it is unlikely to meet an obstacle, it appears free and the resulting diffusion coefficient is high. As the observation time becomes longer, interactions with obstacles happen more frequently and the resulting diffusion coefficient appears to drop. Because of this, the quantity measured in diffusion imaging is often referred to as an *apparent diffusion coefficient* (ADC).

Neurons consist of a cell body *(soma)* which integrates incoming activations and a long thin *axon* which conducts action potentials away from the cell. In the human brain, the cell bodies are concentrated in the *gray matter,* while the axons form nerve fiber bundles that constitute the *white matter* and connect various centers of gray matter.

When investigating the brain, diffusion time is chosen such that the effects of barriers on the scale of axons can be observed. Individual axons are only a few micrometers in diameter and much smaller than the achievable voxel size (around one millimeter). However, they are organized in *fiber bundles* of larger scale and in many regions of the white matter, this organization is coherent enough such that a directional dependence *(anisotropy)* of the apparent diffusivities can be observed on a voxel level.

Empirically, it has been found that water diffuses more freely in direction of the fiber bundles than across them, but the exact causes of this have yet to be elucidated. Existing studies indicate that anisotropy increases with the proliferation of insulating myelin sheaths around the axons during brain maturation [181], but some diffusion anisotropy is already observed in unmyelinated fibers [244] and the dependence of the apparent diffusion coefficient on diffusion time does not support models which are purely based on impenetrable barriers [127]. Instead, the widely used diffusion tensor model [13] (Section 2.2.3) assumes that diffusion is *hindered,* but not *restricted* by structures like cell membranes, organelles, and macromolecules.

2.2.2. Diffusion-Weighted MRI

Magnetic resonance imaging exploits the high amount of water in human soft tissue to produce tomographic images non-invasively. Protons, the nuclei of hydrogen, possess a magnetic moment and align in an external magnetic field, either parallel (in a low energy state) or antiparallel (in a high energy state). Moreover, the proton dipole moments *precess* (rotate) at the *Larmour frequency* $\omega_L = \gamma B_0$, where γ is the gyromagnetic ratio, a constant specific to the nucleus, and B_0 is the strength of the external field.

The direction of the external field is taken as the $\mathbf{z}$ axis of a local coordinate system. Orthogonal to it, fixed $\mathbf{x}$ and $\mathbf{y}$ axes are defined relative to the measurement equipment. Additionally, $\mathbf{x}'$ and $\mathbf{y}'$ axes form a frame of reference that rotates at ω_L around $\mathbf{z}$. In the equilibrium, a higher fraction of nuclei aligns parallel, resulting in a net magnetization vector $\mathbf{m}$ parallel to the external field ($\|\mathbf{m}\| = m_0$).

A radio-frequency (RF) electromagnetic pulse at ω_L excites the rotating protons to their high energy state. This reduces the number of parallel protons and, therefore, the longitudinal net magnetization m_z. Moreover, the excited nuclei precess in phase, which creates a rotating net magnetization $m_{x'}$ and $m_{y'}$ in the transverse plane. According to the *Bloch equations* [25], the components of the net vector $\mathbf{m}$ return to their equilibrium states exponentially when the RF pulse is turned off:

$$\frac{\mathrm{d}m_z}{\mathrm{d}t} = -\frac{m_z - m_0}{T_1} \qquad \frac{\mathrm{d}m_{x'}}{\mathrm{d}t} = -\frac{m_{x'}}{T_2^*} \quad \frac{\mathrm{d}m_{y'}}{\mathrm{d}t} = -\frac{m_{y'}}{T_2^*} \tag{2.19}$$

The parameters T_1 and T_2^* are the time constants of the exponential restoration of the longitudinal and transverse magnetization, respectively. They are specific to certain types of tissue, so grayscale images that pronounce differences in them display tissue boundaries. T_2^* is both influenced by spin-spin interactions within the tissue, which destroy phase coherence irreversibly, and slight inhomogeneities in the external field, which lead to additional phase dispersal through variations in ω_L. Since the latter effect is not specific to the tissue, it is common to reduce its influence by a so-called *spin-echo sequence* [86].

In a spin-echo sequence, a 90° pulse is applied to flip the net magnetization vector into the $\mathbf{xy}$-plane. After some time τ_δ, a 180° pulse inverts the direction of the precession. Since static protons keep their individual Larmour frequencies, the spins rephase after $2\tau_\delta$ and produce a spin-echo (SE, cf. Figure 2.2). A common metaphor for this effect is to imagine a group of runners who start from a common line and are told to turn around and return to that line after τ_δ has passed. If each runner keeps his individual, constant speed, they will cross the starting line together. The maximal amplitude of subsequent echoes decays exponentially with time constant T_2. It is mainly influenced by spin-spin interactions.

The Bloch equations neglect the fact that protons are not static during the measurement, but perform a constant heat motion. In particular, moving spins in an inhomogeneous external field do not have a constant Larmour frequency. Modeling this effect allows for an estimation of the self-diffusion coefficient by introducing known inhomogeneities (gradients) into the external field [38].

The first pulse sequence which was widely used to measure self-diffusion has been de-

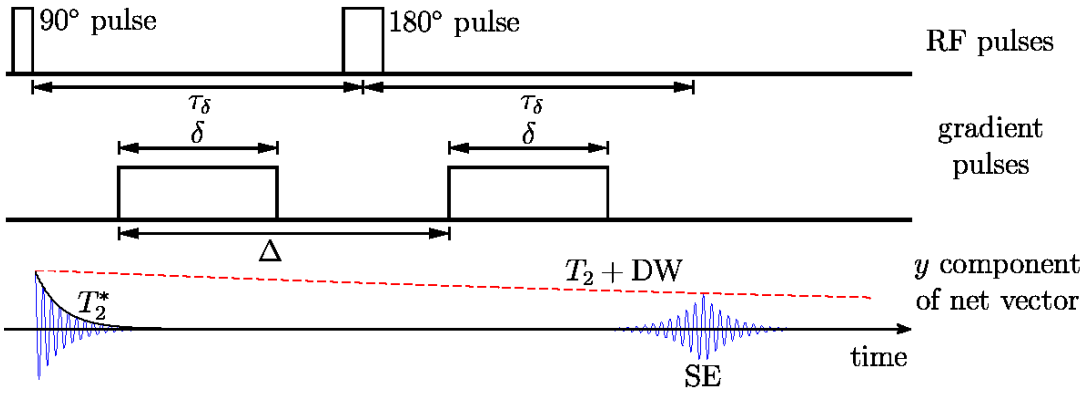


Figure 2.2.: In the Stejskal-Tanner sequence, the decay of spin-echoes (SE) depends on T_2 and on the diffusion weighting (DW) caused by the gradient pulses.

veloped by Stejskal and Tanner [206]. It places a *gradient pulse* of length δ between the 90° and the 180° pulse of the traditional spin-echo sequence (Figure 2.2). A second gradient after the 180° pulse reverses this effect for nuclei that kept their position within the diffusion time Δ. If the spin has moved, it will experience different phase shifts by the two gradient fields, leading to a weaker spin-echo (SE). This sequence was integrated into tomographic imaging and used to investigate the human brain by LeBihan et al. [125]. The resulting slice images are called diffusion-weighted images (DWIs) and compared to a non-diffusion weighted T_2 image to estimate the apparent diffusion coefficient.

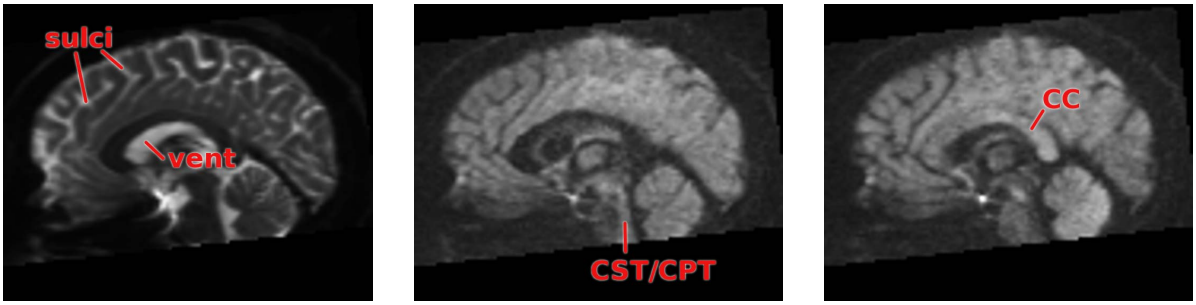
The image intensity $S(\mathbf{g})$ of a voxel measured by a Stejskal-Tanner sequence is related to the intensity S_0 in an unweighted spin-echo sequence and the apparent diffusivity $d(\mathbf{g})$ by the following equation [206]:

$$S(\mathbf{g}) = S_0 \mathrm{e}^{-bd(\mathbf{g})} \tag{2.20}$$

The b-value collects the measurement parameters ($b = \gamma^2\delta^2(\Delta - \delta/3)\|\mathbf{g}\|^2$, where γ, δ and Δ are defined as above, and $\|\mathbf{g}\|$ is the magnitude of the diffusion gradient). Even though image intensity in MRI is not calibrated to a fixed scale, apparent diffusion coefficients d estimated via Equation (2.20) have the units of diffusivity. ADCs in tissue are usually lower than the diffusivity of free water, which is $\approx 2.3 \times 10^{-3}\,\mathrm{mm}^2/s$ at 25°C [206].

Since the diffusion gradient $\mathbf{g}$ is applied along a single direction, only the projection of the molecular movement onto that direction attenuates the signal. By taking multiple diffusion-weighted measurements with different gradient directions, it becomes possible to model the directional dependence of the apparent diffusivity $d(\mathbf{g})$, which is the key to inferring fiber directions from diffusion data of the human brain.

The sagittal slices in Figure 2.3 illustrate this. Free water has a high T_2, so the fluid-filled *sulci* and *ventricles* appear hyperintense (bright) in the unweighted image (a). However, under diffusion weighting (b/c), the unhindered heat motion of free water causes signal attenuation. In homogeneous fiber tracts, the attenuation depends on the gradient direction: In (b), the gradient is aligned with the *corpus callosum* (CC), so this structure appears hypointense (dark). In contrast, the gradient in (c) is more closely aligned with



(a) Baseline ($b = 0\,\mathrm{mm}^2/s$) (b) Gradient $[0.98\ 0.06\ 0.17]^{\mathrm{T}}$ (c) Gradient $[-0.06\ 0.31\ 0.95]^{\mathrm{T}}$

Figure 2.3.: In the baseline image (a), cerebrospinal fluid appears bright. The appearance of anisotropic structures like CST/CPT or CC in diffusion-weighted images (b) and (c) depends on the gradient direction.

the *corticospinal / corticopontine tract* (CST/CPT). Due to overall variations in MR values, Subfigure (a) maps the range $[0, 1200]$, while the dynamic range of Subfigures (b) and (c) is $[0, 180]$.

2.2.3. The Diffusion Tensor Model

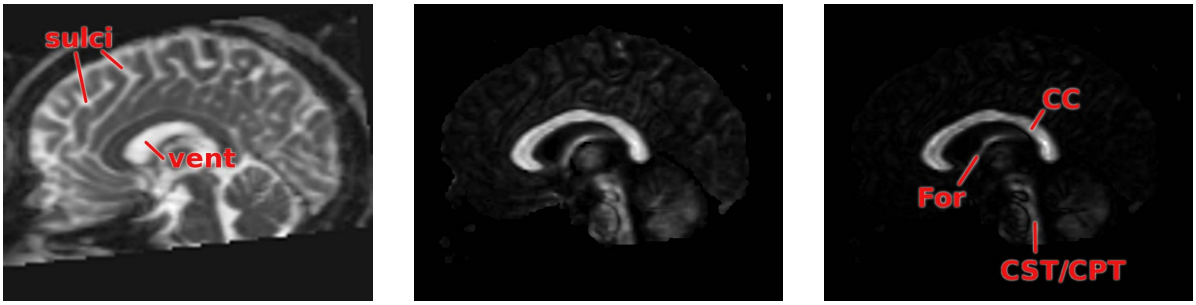
To model the diffusion anisotropy in tissue, the *diffusion tensor* (DT-MRI) model by Basser et al. [13] approximates the diffusivity function $d(\mathbf{g})$ with a quadratic form, written as a symmetric second-order diffusion tensor $\mathbf{D}$:

$$d(\mathbf{g}) = \mathbf{g}^{\mathrm{T}}\mathbf{D}\mathbf{g} \tag{2.21}$$

Since the symmetric tensor $\mathbf{D}$ has six degrees of freedom, at least six diffusion-weighted images are required to estimate it, plus a baseline image S_0. When exactly that number of images is available, tensor components can be estimated in closed form [240]. However, for robust estimation of anisotropy and principal direction, it is advisable to use more than the minimum number of directions [106]. In this case, diffusion tensor estimation can be formulated as a least squares problem on the logarithms $\ln(S/S_0)$ [13].

When a diffusion-weighted image exhibits higher intensity than the baseline image S_0, Equation (2.20) yields a negative apparent diffusion coefficient, which does not make sense physically and can be attributed to measurement errors or noise. Consequently, some methods for processing diffusion tensors assume that $\mathbf{D}$ is positive (semi)definite. When required, this constraint can be enforced after tensor estimation by clamping negative eigenvalues, which corresponds to a projection to the cone of positive definite tensors. Variational methods [217] integrate positive definiteness and additional spatial regularity constraints into the estimation process itself. The Cholesky factorization [230] and the Log-Euclidean framework [71] offer additional alternatives to enforce positive definiteness.

When using the DT-MRI model, typical b-values for the measurements are on the order of $b = 700\,\mathrm{s/mm}^2$ to $b = 1000\,\mathrm{s/mm}^2$ [240, 220, 106].



(a) Mean Diffusivity (b) Fractional Anisotropy (c) Linearity c_l

Figure 2.4.: Scalar measures of the amount and directional dependence of water diffusion provide contrast between anatomical structures and can be used for diagnosis.

2.2.4. Scalar Measures in DT-MRI

Tensor-valued data is more difficult to visualize and to interpret than the grayscale images produced by traditional MRI. Fortunately, several clinical and scientific problems can already be addressed by considering scalar measures that capture certain anatomically relevant attributes of the diffusion tensor. These measures are chosen such that they are invariant under rotation of the coordinate frame. They can be visualized as traditional slice images or by volume rendering [116], and they serve as quantitative measures in scientific studies [98, 128].

The most fundamental scalar measure is *mean diffusivity* (MD), the average diffusivity over the full unit sphere. It equals one third of the diffusion tensor trace:

$$\mathrm{MD}(\mathbf{D}) = \frac{1}{3}\,\mathrm{tr}(\mathbf{D}) = \frac{1}{3}\,(D_{xx} + D_{yy} + D_{zz}) \tag{2.22}$$

In healthy subjects, mean diffusivity is approximately constant over gray and white matter, and much larger in the cerebrospinal fluid [164] (cf. Figure 2.4 (a)). An important clinical application of diffusion imaging is due to the fact that decreased mean diffusivity serves as an early indicator of stroke. In the subacute and chronic phases, renormalization and increase of MD is observed [205].

Let $\mathbf{I}$ denote the unit matrix. The trace-free tensor

$$\overline{\mathbf{D}} = \mathbf{D} - \frac{1}{3}\,\mathrm{tr}(\mathbf{D})\,\mathbf{I} \tag{2.23}$$

is the *deviatoric* (anisotropic part) of $\mathbf{D}$. Based on its magnitude, Basser and Pierpaoli [16] proposed two measures of overall directional dependence, *fractional anisotropy* (FA) and *relative anisotropy* (RA):

$$\mathrm{FA}(\mathbf{D}) = \sqrt{\frac{3}{2}}\frac{\|\overline{\mathbf{D}}\|}{\|\mathbf{D}\|} \qquad \mathrm{RA}(\mathbf{D}) = \sqrt{\frac{1}{3}}\frac{\|\overline{\mathbf{D}}\|}{\mathrm{MD}(\mathbf{D})} \tag{2.24}$$

Both FA and RA are dimensionless quantities which equal zero for perfectly isotropic tensors ($\mathbf{D} = s\mathbf{I}$). For perfectly linear (rank-one) tensors, FA = 1, RA = $\sqrt{2}$. Higher values of FA and RA are only attained by indefinite tensors.

Anisotropy is high in white matter, but low in gray matter and cerebrospinal fluid. This is shown in Figures 2.4 (b) and (c), where For denotes the *fornix*, while the labels CC and CST/CPT are used as above. Decreased diffusion anisotropy is typically interpreted as a result of damaged nerve tissue structure. However, anisotropy in healthy subjects is highly variable over different regions of the white matter, so the effects of registration misalignment and the choice of filtering kernel cannot be neglected when studying the relation between diseases and changes in anisotropy (cf. [203] and references therein). Moreover, in the case of crossing fiber bundles, degeneration of one bundle can lead to a paradoxical increase in anisotropy, as the fibers of the remaining bundle now appear more coherently organized [163].

FA and RA quantify overall directional dependence, without differentiating between linear ($\lambda_1 \gg \lambda_2 \approx \lambda_3$) and planar ($\lambda_1 \approx \lambda_2 \gg \lambda_3$) anisotropy. This distinction is called *mode* or *skewness*. The skewness measure s employed by Bahn [11] and the parameter "mode" used by Criscione [50] and introduced to DT-MRI by Ennis and Kindlmann [67] are defined as

$$s = \arctan\left(\frac{\sqrt{3}\,(\lambda_2 - \lambda_1)}{2\lambda_3 - \lambda_2 - \lambda_1}\right) \qquad \text{mode} = 3\sqrt{6}\,\det\left(\frac{\overline{\mathbf{D}}}{\|\overline{\mathbf{D}}\|}\right) \tag{2.25}$$

For perfectly linear tensors, $s = \pi/3$, mode $= 1$. For planar tensors, $s = 0$, mode $= -1$. To avoid confusion with the notion of tensor mode which was introduced in Section 2.1.4, we will only use the term "skewness" to refer to the distinction between linear and planar diffusion from now on. Skewness is affected by noise more strongly than mean diffusivity or anisotropy [12]. When eigenvalues are taken as the axes of a space of tensor shapes, measures of anisotropy and skewness can be derived by changing to cylindrical or spherical coordinates [11]. It is noteworthy that even though MD and FA are frequently analyzed together, they are not strictly orthogonal within that framework. Instead, a measure of overall diffusivity which is orthogonal to FA is given by the diffusion tensor norm $\|\mathbf{D}\|$ [11, 67].

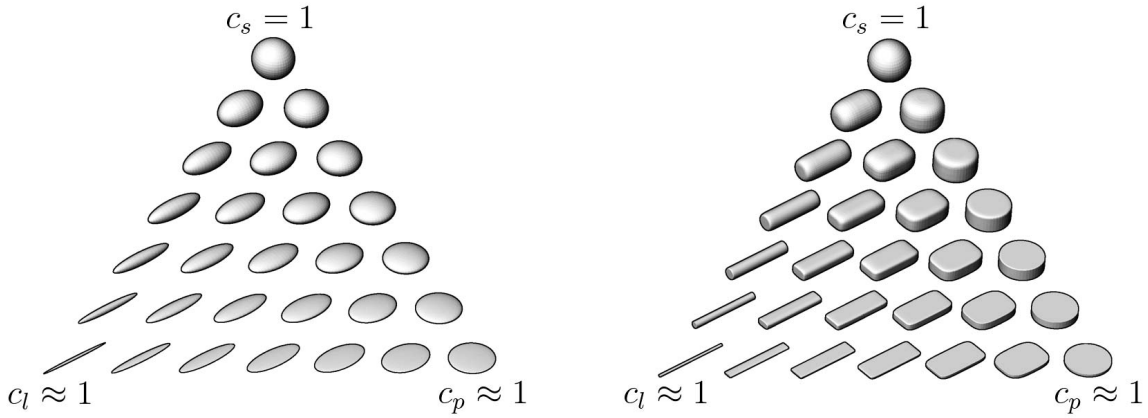
The anisotropy measures by Westin et al. [241] integrate skewness by providing coordinates with respect to perfectly linear (c_l), planar (c_p) and spherical tensors (c_s):

$$c_l = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2 + \lambda_3} \qquad c_p = \frac{2\,(\lambda_2 - \lambda_3)}{\lambda_1 + \lambda_2 + \lambda_3} \qquad c_s = \frac{3\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} \tag{2.26}$$

For positive definite tensors, c_l, c_p, and c_s lie in the range $[0, 1]$ and fulfill the barycentric coordinate constraint $c_l + c_p + c_s = 1$. Figure 2.5 presents glyph visualizations of this barycentric shape space. More details on tensor glyphs are given in the following section.

Both skewness measures in Equation (2.25) classify a tensor as equally linear and planar ($s = \pi/6$, mode $= 0$) if $\lambda_1 - \lambda_2 = \lambda_2 - \lambda_3$. However, the Westin measures from Equation (2.26) would consider the same tensor twice as planar as linear ($c_p = 2 \times c_l$). This discrepancy is fixed by an alternative normalization of the Westin measures [241]:

$$\hat{c}_l = \frac{\lambda_1 - \lambda_2}{\lambda_1} \qquad \hat{c}_p = \frac{\lambda_2 - \lambda_3}{\lambda_1} \qquad \hat{c}_s = \frac{\lambda_3}{\lambda_1} \tag{2.27}$$



(a) Visualization with ellipsoid glyphs.

(b) Visualization with superquadric glyphs.

Figure 2.5.: The Westin measures place tensors into a barycentric shape space with linear, planar, and spherical tensors at the corners.

Derivatives of scalar measures indicate structural boundaries in the data. Since mean diffusivity is a linear function of tensor components, its derivative can be computed by evaluating the measure on the original grid points and taking finite differences. For the nonlinear fractional anisotropy, Kindlmann et al. [114] take derivatives via the chain rule. Since sorted eigenvalues are not everywhere differentiable, edge maps with respect to the Westin measures are less straightforward to compute. It is one contribution of this dissertation to define regularized eigenvalue derivatives which are everywhere well-defined and allow for such edge maps (Chapter 6).

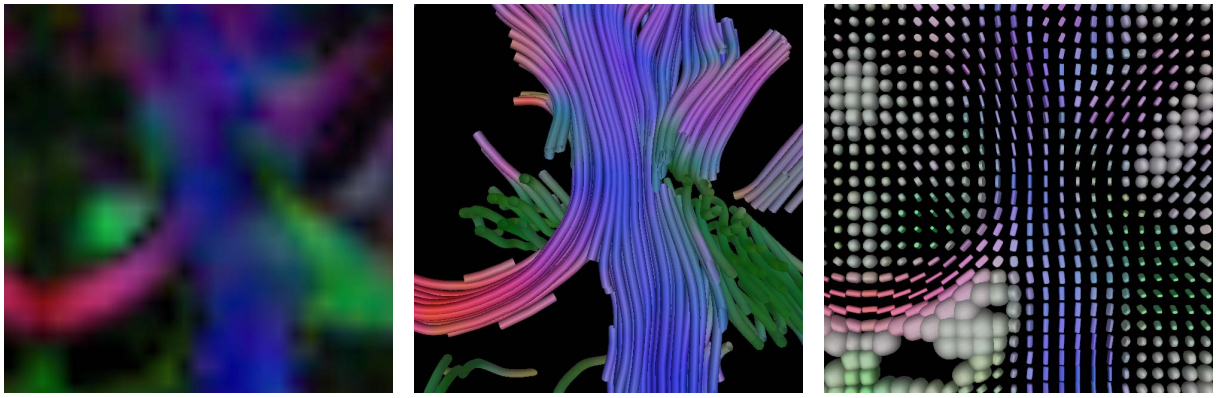
A conceptually different way to measure boundaries in DT-MRI data is to consider the gradient of the tensor field itself, a third-order tensor [154]. Ennis and Kindlmann [67] demonstrate how this tensor field gradient can be projected onto the gradients of scalar measures, taken over the space of second-order tensors, to differentiate between various types of boundaries. A more detailed discussion of this approach is given in Section 6.4.

2.2.5. Standard Visualization Techniques for DT-MRI

In brain data, the orientation of linear diffusion tensors allows one to draw conclusions about the orientation of fiber structures within the respective voxel. The easiest way to convey this direction visually is to *color code* the principal eigenvector in slice images (Figure 2.6 (a)) or volume renderings.

The most common color scheme, XYZ-RGB, maps the coordinates of the eigenvector $\mathbf{e}$ with respect to an orthonormal basis $\{\mathbf{b}_i\}$ of $\mathbb{R}^3$ to the red, green, and blue color channels, respectively. $\mathbf{b}_i$ are chosen with respect to the patient, such that $\mathbf{b}_2$ and $\mathbf{b}_3$ span the sagittal plane, and $\mathbf{b}_2$ is aligned with the anterior-posterior intercommissural line [155].

The alpha value A (or, alternatively, the saturation) of the resulting color is modulated by a measure of linearity to avoid visualization of meaningless eigenvector directions in



(a) Color coded slice image (b) Streamlines (tractography) (c) Superquadric glyphs

Figure 2.6.: Color maps, streamlines, and glyphs are standard techniques for DT-MRI visualization. Here, they display a part of a coronal slice of brain data.

isotropic regions. In the simplest case, an $[R, G, B, A]$ tuple is given as

$$R = |\mathbf{e} \cdot \mathbf{b}_1| \qquad G = |\mathbf{e} \cdot \mathbf{b}_2| \qquad B = |\mathbf{e} \cdot \mathbf{b}_3| \qquad A = c_l \tag{2.28}$$

This mapping creates undesired ambiguities: Mirroring $\mathbf{e}$ at any of the coordinate planes preserves the resulting color. However, since eigenvectors lack orientation, DT-MRI color schemes have to map vectors $\mathbf{v}$ and $-\mathbf{v}$ to the same color. Moreover, to avoid artifactual edges, continuous changes in direction should lead to continuous changes in color. Pajevic and Pierpaoli [155] have shown that any scheme that fulfills both of these constraints suffers from undesired ambiguities.

When the goal is to visualize both orientation and anisotropy at the same time, Equation (2.28) should not be used in its unmodified form, since perceived brightness is not the same for pure red, green, and blue on typical displays, and it is also affected by the blending of colors. In practice, different heuristic variations of Equation (2.28) are in use (e.g., [155, 197]). By convention, they share the anatomical interpretation that red indicates the mediolateral direction, green anterior–posterior, and blue superior–inferior.

Techniques for vector field visualization can be applied to the principal eigenvector field of tensor data, provided that one takes into account the facts that eigenvectors lack orientation and that they become ill-defined at degeneracies ($\lambda_1 = \lambda_2$). An important vector field-based method is the integration of streamlines which are tangential to the principal eigenvector field. In the context of DT-MRI, this is called *fiber tracking* [143, 15, 237, 144]. Beside the definition of reasonable termination criteria, a successful streamline-based visualization requires an appropriate strategy for seeding and culling [251, 223, 138].

Streamlines like the ones in Figure 2.6 (b) convey the inferred trajectory of nerve fiber bundles in an intuitive manner, but one should keep in mind that the individual lines do not have clear anatomical correlates. Single axons are far below image resolution. On the other hand, fiber bundles usually comprise more than a single streamline. Consequently, approaches have been developed to automatically cluster those streamlines that belong to a common tract [201, 60, 32, 250], and to construct geometric hulls around them [66, 43].

Another common tool for vector field visualization is *line integral convolution* (LIC) [34]. It produces a dense image of the field, in which the intensity of pixel values along streamlines is correlated. When modulating transparency with an anisotropy measure, LIC can also be applied to eigenvector fields [246]. However, in three-dimensional data, clear and effective rendering of the dense volume produced by LIC is a challenge. Existing strategies to reveal interior structures include volumetric clipping and interactive dye injection [200], sparse noise textures and depth-revealing halos [99], transfer functions that emulate sparse noise, and clipping by dynamic geometry [174].

Both streamline integration and LIC have been modified to use the full tensor information: The ellipsoidal cross sections of hyperstreamlines [55] indicate the directions of the smaller two eigenvectors and the relative magnitude of the corresponding eigenvalues. HyperLIC [253] uses a convolution kernel whose spatial extent depends on the full tensor. Both methods have been applied to DT-MRI data [251, 253].

Tensor glyphs are geometric objects whose shape, orientation, and size convey the full information at discrete points of a tensor field. The most basic type of tensor glyphs are ellipsoids whose semi-axes are aligned with the eigenvectors and scaled with the eigenvalues of the tensor. Example tensor ellipsoids are shown in Figure 2.5 (a). A formal definition and a generalization to higher-order tensors is presented in Section 8.3.

The effectiveness of ellipsoidal glyphs is limited by the fact that the projection of quite different ellipsoids can result in the same two-dimensional ellipse in image space. Often, shading does not provide sufficient cues of the three-dimensional shape. Other types of glyphs avoid this ambiguity [239]. In particular, superquadrics can be parametrized such that sharp edges convey the direction of clearly defined eigenvectors [110]. Examples of the resulting tensor glyphs are given in Figures 2.5 (b) and 2.6 (c).

Superquadric tensor glyphs have been used for visualization of both brain [110] and myocardial DT-MRI data [68], and have been extended to applications outside diffusion MRI [100]. Beside choosing an appropriate type of glyph, careful seeding can further improve glyph-based visualizations [117].

2.2.6. Higher-Order Diffusion Models

Validation studies [129, 52] have indicated that in voxels with a single predominant fiber orientation, the direction of the highest apparent diffusion coefficient, as given by the principal eigenvector of a second-order diffusion tensor, is well-aligned with this fiber direction. Unfortunately, many voxels contain more than a single predominant direction, due to fiber bundles that cross, fan out, bend sharply, or pass each other.

To better capture such complex voxels, it has been proposed to model the apparent diffusivity function $d(\mathbf{g})$ in a more flexible way, using spherical harmonics [76, 3] or, equivalently, higher-order tensors [149]. Such *high angular resolution diffusion imaging* (HARDI) requires diffusion-weighted measurements in more directions. Since diffusion imaging projects the molecular motion onto the (unoriented) line defined by $\mathbf{g}$, all models of apparent diffusivity exhibit antipodal symmetry $d(-\mathbf{g}) = d(\mathbf{g})$.

Unfortunately, the maxima of the diffusivity profile no longer align with any fiber direction when a voxel contains more than a single tract [226, 220]. Commonly, n crossing

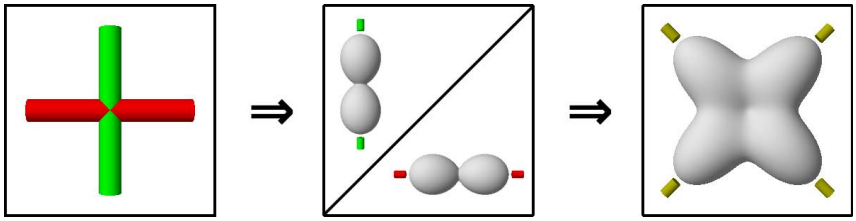


Figure 2.7.: Crossing tracts act as separate compartments, whose apparent diffusivities add non-linearly. Consequently, the maxima of the combined diffusivity function are not aligned with any fiber direction.

tracts are modeled by independent compartments with volume fractions c_i ($\sum_{i=1}^{n} c_i = 1$) and quadratic diffusivities d_i in each. The apparent overall diffusivity d is a non-linear mixture of the d_i, and is no longer quadratic. The modified version of Equation (2.20) and the resulting expression for d are:

$$S(\mathbf{g}) = S_0 \sum_{i=1}^{n} c_i \, \mathrm{e}^{-b d_i(\mathbf{g})} \quad \Rightarrow \quad d(\mathbf{g}) = -\frac{1}{b} \ln \sum_{i=1}^{n} c_i \, \mathrm{e}^{-b d_i(\mathbf{g})} \tag{2.29}$$

Figure 2.7 illustrates the case of a 90° crossing, in which maxima are offset by 45°. The exact shape of d depends on the b-value. HARDI measurements commonly use a larger b than DT-MRI ($b = 1000\,\mathrm{s/mm^2}$ to $b = 3000\,\mathrm{s/mm^2}$), to achieve sharper peaks [220, 213].

2.2.7. Q-Space Imaging and the Q-Ball Method

Compared to higher-order models of diffusion, *q-space imaging* [35] allows for more direct conclusions about possible fiber directions. They describe the spin displacement within the experimental diffusion time by a displacement vector $\mathbf{r}$ and reconstruct the voxel-averaged probability density function (PDF) $P(\mathbf{r})$ of these vectors. Q-space imaging exploits the fact that $P(\mathbf{r})$ is related to the attenuated signal $S(\mathbf{q})$ via the Fourier transform $\mathcal{F}$:

$$P(\mathbf{r}) = \mathcal{F}[S(\mathbf{q})] \tag{2.30}$$

$S(\mathbf{q})$ is given as a function of the diffusion wavevector $\mathbf{q} = (2\pi)^{-1}\gamma\mathbf{g}\delta$, where γ is a nucleus-specific constant, $\mathbf{g}$ is the diffusion gradient, and δ is its duration, as in Section 2.2.2. A Cartesian sampling of q-space, followed by a discrete Fourier transform, has been performed to obtain P [233], but this procedure is infeasible for routine investigation, since it is time consuming and requires extremely high values of $\|\mathbf{g}\|$. Moreover, for practical analysis, P is integrated in radial direction to obtain an orientation distribution function (ODF) $\psi(\mathbf{u})$ of directions $\mathbf{u}$ on the unit sphere ($\|\mathbf{u}\| = 1$), which discards much of the acquired information.

Thus, measurements are generally taken on a ball in q-space, with fixed gradient magnitude $\|\mathbf{g}\|$ and duration δ. Assuming that the signal decays exponentially with $\|\mathbf{g}\|$ (Equation (2.20)), the expected intensity values on a Cartesian grid in q-space can be

computed numerically [151]. It is even possible to evaluate the radial part of the integral analytically, which leads to the *diffusion orientation transformation* [150].

An even simpler method is *q-ball imaging* [219]: It uses the Funk-Radon transform, a mapping from the sphere to the sphere, which treats each point as a pole and assigns the integral over the associated equator to it. Q-ball imaging exploits the fact that the Funk-Radon transform of the measurements $S(\mathbf{q})$, which are again taken on a ball in q-space, provides a good approximation to $\psi(\mathbf{u})$. Q-ball reconstruction is greatly simplified by working in a spherical harmonics basis, which allows for an analytic solution of the Funk-Radon transform [5, 92, 57]. For our own implementation, we have followed the detailed description in [57].

Q-space methods are *model-free* in the sense that they characterize the diffusion process without assuming free, hindered, or restricted diffusion. Even though ODF peaks have been found to align with crossing fiber bundles in some regions of the white matter, one should keep in mind that q-ball ODFs describe diffusion, not a fiber distribution [221]. To draw conclusions about white matter structure, q-space measurements have been combined with models of the diffusion process within and around fiber bundles [10].

In case of restricted diffusion, $P(\mathbf{r})$ need not be symmetric. In theory, this could help to distinguish crossing (X-shaped PDF) from branching tracts (Y-shaped PDF). In practice, one considers the magnitude of the complex signal $S(\mathbf{q})$, since its phase is corrupted by unavoidable bulk motion. This introduces antipodal symmetry $P(\mathbf{r}) = P(-\mathbf{r})$ [131].

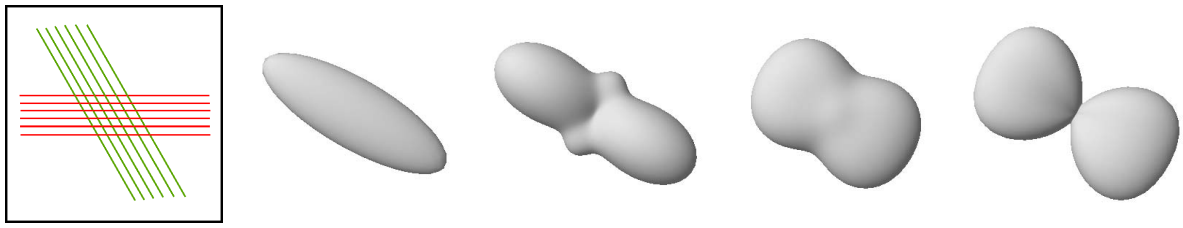
2.2.8. The Spherical Deconvolution Model

A direct way of inferring fiber orientations from high angular resolution measurements is *spherical deconvolution* [213]: It is based on the assumption that the signal from a single, well-organized fiber population can be described by an axially symmetric response function R which is assumed constant over the whole brain and for all types of fibers. Then, the measured signal S is expressed as the convolution of a fiber orientation density function F with R, taken over the unit sphere. In the notation of [58], both $\mathbf{u}$ and $\mathbf{w}$ denote unit vectors, and R is defined as a function on $[-1, 1]$. Then, the convolution integral reads

$$S(\mathbf{u}) = \int_{\|\mathbf{w}\|=1} R(\mathbf{u} \cdot \mathbf{w}) F(\mathbf{w}) d\mathbf{w} \tag{2.31}$$

After S has been measured, the response function R is estimated from voxels which are thought to contain a single fiber tract. The fiber distribution F is then obtained from Equation (2.31) by spherical deconvolution, which is simplified by expressing S using spherical harmonics and R in a rotational harmonic representation [89].

Both q-ball imaging and spherical deconvolution result in functions on the sphere, but their semantics differ: While a q-ball orientation *distribution* functions defines the probability of a spin displacement, an orientation *density* function from spherical deconvolution yields a fiber volume fraction. However, in practice, the spherical deconvolution ODF can be considered a sharpened version of the q-ball ODF [58]. For simplicity, we use the term "orientation distribution function" (ODF) for the results of both methods in the remainder of this dissertation.



(a) (b) (c) (d) (e)

Figure 2.8.: A 60° fiber crossing (a) is not captured by the DT-MRI model (b). Unlike in a higher-order diffusion model (c), individual fiber peaks add linearly in q-ball (d) and spherical deconvolution ODFs (e).

As a final overview, Figure 2.8 presents a visual comparison of some common models for diffusion MRI, based on synthetic data from a 60° fiber crossing (a). Modeling the diffusivity profile using a second-order DT-MRI model (b) or a higher-order tensor (c) fails to indicate the tract directions. In q-ball imaging (d) and the spherical deconvolution (e), individual fiber bundles cause peaks which are aligned with the bundle directions, and add linearly in the final ODF. However, Figure 2.8 also demonstrates that, since the original peaks have finite width, they interfere and cannot simply be separated by locating maxima in the combined ODF. This problem will be the topic of Chapter 10.

2.3. Topology in Visualization

Topological methods have been introduced to scientific visualization by Helman and Hesselink [90], who proposed to visualize flow fields by extracting topological skeletons. Since then, topological visualization has been considered widely to reduce complex data to structural features. This section introduces basic concepts of point set topology (Section 2.3.1) and vector field topology (Section 2.3.2), and clarifies the notions of genericity and structural stability (Section 2.3.3). The focus is on the aspects of the topology of steady vector and tensor fields that are crucial to understand the first part of this dissertation.

In order to emphasize the close relation between vector and eigenvector fields, Sections 2.3.4 and 2.3.5 treat both topics in parallel. Even though we investigate 3D topology in the main part of the dissertation, some concepts are first introduced in 2D, where they are easier to understand. Crease surfaces and surface topology are at the center of interest in Chapter 4, and are introduced in Sections 2.3.6 and 2.3.7, respectively.

A general overview on topology can be found in [28]. Vector field topology is closely related to the study of dynamical systems [243]. A broad introduction to the topological visualization of vector and tensor fields is given in [236, 216, 184, 214].

2.3.1. Brief Introduction to Topology

Let $f : X \to Y$ be a function from set X to set Y. According to the $\epsilon - \delta$ definition of continuity, f is continuous with respect to metrics $d_x : X \times X \to \mathbb{R}$ and $d_y : Y \times Y \to \mathbb{R}$

if, for any $a \in X$ and any $\epsilon > 0$, there exists a $\delta > 0$ such that

$$d_x(a, b) < \delta \Rightarrow d_y(f(a), f(b)) < \epsilon \tag{2.32}$$

Despite the formal dependence on a particular choice of d_x and d_y, $\epsilon - \delta$ continuity is preserved under whole classes of different metrics, indicating that a more general definition is possible. Therefore, the topological definition of continuity is in terms of open sets: A function $f : X \to Y$ is continuous if

$$U \subseteq Y \text{ is open} \Rightarrow f^{-1}(U) \subseteq X \text{ is open} \tag{2.33}$$

Define the ϵ-ball $B_\epsilon(a)$ around $a \in X$ with respect to metric d as

$$B_\epsilon(a) = \{b \in X \mid d(a, b) < \epsilon\} \tag{2.34}$$

and call a subset $U \subseteq X$ "open" if, for each $u \in U$, there is an ϵ such that $B_\epsilon(u) \subseteq U$. Then, the definitions in Equations (2.32) and (2.33) are equivalent, and two metrics generate the same topology if their families of open sets coincide [28].

This definition can be generalized even further: A topological space (X, T) is a set X with any collection T of subsets of X, which are called "open" and which fulfill the following three axioms:

1. The intersection of two open sets is open.
2. The union of any collection of open sets is open.
3. The empty set $\emptyset$ and the whole space X are open.

The open sets that are induced by a metric fulfill this definition, so all metric spaces are topological spaces. However, the converse is not true: There exist "non-metrizable" topological spaces which cannot be generated by a metric. In our applications, we will only deal with metrizable spaces.

We will now provide some topological definitions which are used in the remainder of this section. A *homeomorphism* is a function $f : X \to Y$ between topological spaces X and Y that is bijective, continuous, and has a continuous inverse f^{-1}.

A set $N \subseteq X$ is a *neighborhood* of some point $x \in N$ if there exists an open set $U \subseteq N$ that contains x. A set $K \subseteq X$ is *closed* if its complement $X \setminus K$ is open. Sets may be both open and closed *(clopen)* in the same topology, and X is *connected* if and only if its only clopen subsets are $\emptyset$ and X.

For each set $A \subseteq X$, the *interior* $\mathrm{int}(A)$ is its largest open subset $U \subseteq A$. Similarly, the *closure* $\mathrm{cl}(A)$ is the smallest closed set F that contains A ($A \subseteq F \subseteq X$). The boundary of $A \subseteq X$ is $\mathrm{bdy}(A) = \mathrm{cl}(A) \cap \mathrm{cl}(X \setminus A)$. $A \subseteq X$ is *dense* in X if $\mathrm{cl}(A) = X$.

An *open covering* of a set $A \subseteq X$ is a collection $\{U_i\}$ of open sets such that $A \subseteq \bigcup_{i \in I} U_i$. A is *compact* if every open covering has a finite subcover

$$A \subseteq U_{i_1} \cup U_{i_2} \cup \ldots \cup U_{i_n} \quad \text{with} \quad i_1, i_2, \ldots i_n \in I \tag{2.35}$$

If X is equipped with a metric d, compactness can be defined in a more intuitive way. A *Cauchy sequence* is a sequence (x_n) such that for each $\epsilon > 0$, there is some number $N > 0$ such that for all $i, j > N$, $d(x_i, x_j) < \epsilon$. A is *complete* if every Cauchy sequence in A converges in A. Moreover, A is *bounded* if for any $\epsilon > 0$, it can be covered by finitely many ϵ-balls. Then, A is compact in the topology generated by the metric if and only if it is complete and bounded [28].

2.3.2. Vector Field Topology

The vector fields in this dissertation are given as C^k continuous ($k \geq 1$) functions $\mathbf{v} : D \to \mathbb{R}^d$ ($d \in \{2, 3\}$) on a compact domain $D \subset \mathbb{R}^d$. On the set of such functions, the C^r topology ($0 \leq r \leq k$) is generated by a metric that takes into account function values and derivatives up to order r [1].

A tangent curve $\mathbf{x}_\mathbf{v}$ of is the trajectory of a massless particle that is advected by $\mathbf{v}$. For initial position $\mathbf{x}_0$, it depends on time t, according to the following initial value problem:

$$\frac{d\mathbf{x}_\mathbf{v}(t;\mathbf{x}_0)}{dt} = \mathbf{v}\big(\mathbf{x}_\mathbf{v}(t;\mathbf{x}_0)\big) \quad \text{with} \quad \mathbf{x}_\mathbf{v}(0;\mathbf{x}_0) = \mathbf{x}_0 \tag{2.36}$$

Since $\mathbf{v}$ does not depend on any parameters, it is called "steady", and tangent curves coincide with *streamlines.* In the related context of dynamical systems, steady vector fields are called "autonomous", and their streamlines are called "orbits".

Two vector fields $\mathbf{v}$ and $\mathbf{w}$ are *structurally equivalent* if there exists a homeomorphism $h : D \to D$ that maps the tangent curves of $\mathbf{v}$ to those of $\mathbf{w}$, preserving orientation, but ignoring parametrization by time. Formally,

$$h\big(\mathbf{x}_\mathbf{v}(t;\mathbf{x}_0)\big) = \mathbf{x}_\mathbf{w}\big(\gamma(t;\mathbf{x}_0); h(\mathbf{x}_0)\big) \tag{2.37}$$

where $\gamma(t;\mathbf{x}_0)$ is a reparametrization of the tangent curve starting at $\mathbf{x}_0$ that increases with t and thus preserves orientation [243]. If γ is the identity, $\mathbf{v}$ and $\mathbf{w}$ are *conjugate.*

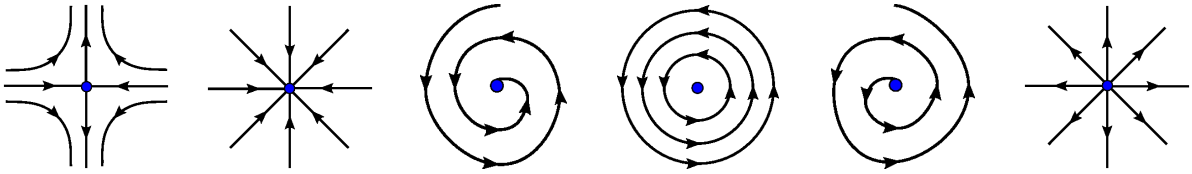
A topological analysis of vector fields examines properties that are invariant over all structurally equivalent fields. In particular, reparametrization by an increasing function γ does not affect the *asymptotic behavior* of tangent curves as $t \to \pm\infty$.

A point $\mathbf{x}_0^\omega$ is an ω limit point of $\mathbf{x}_0$ if there exists a sequence (t_n) with $t_n \to \infty$ such that $\mathbf{x}_\mathbf{v}(t_n;\mathbf{x}_0) \to \mathbf{x}_0^\omega$. Analogously, α limit points are defined by letting $t_n \to -\infty$ [243]. This definition considers a sequence (t_n) instead of taking $\lim_{t\to\infty} \mathbf{x}_\mathbf{v}(t;\mathbf{x}_0)$ directly to cover the case of closed streamlines, which will be introduced in Section 2.3.5. Points $\mathbf{x}_0$ whose tangent curve converges to a closed streamline have more than a single $\mathbf{x}_0^\omega$ or $\mathbf{x}_0^\alpha$.

The ω and α *limit sets* of a field are the sets of all ω and α limit points of all points in D. Since they are invariant under structural equivalence, they are at the center of interest of vector field topology.

2.3.3. Genericity and Structural Stability

The analysis of steady fields concentrates on *generic* properties and *structurally stable* states. In practice, generic properties are those which are met under general conditions and remain stable under small perturbations.



(a) $R_1 > 0$, $R_2 < 0$, $I_i = 0$ (b) $R_i < 0$, $I_i = 0$ (c) $R_1 = R_2 < 0$, $I_1 = -I_2 \neq 0$ (d) $R_i = 0$, $I_1 = -I_2 \neq 0$ (e) $R_1 = R_2 > 0$, $I_1 = -I_2 \neq 0$ (f) $R_i > 0$, $I_i = 0$

Figure 2.9.: In $\mathbb{R}^2$, first order critical points are classified as saddles (a), attracting / repelling nodes (b/f), attracting / repelling foci (c/e), and centers (d).

Generic properties are true "almost everywhere". Formally, a property is generic in some topological space X if and only if there exists a set $U \subseteq X$ which is a countable intersection of open dense sets in which all elements have the desired property. Since vector fields in the C^r topology form a *Baire space* [243], U itself is open and dense. Openness ensures that a generic property is preserved by sufficiently small perturbations. Density guarantees that any element that does not have the generic property can be approximated arbitrarily closely by an element that has it.

In vector field topology, a field $\mathbf{v}$ is called structurally stable if, in the C^1 topology, there exists a neighborhood N around $\mathbf{v}$ such that any $\mathbf{w} \in N$ is structurally equivalent to $\mathbf{v}$. In general, genericity is stronger than structural stability: Structural stability of $\mathbf{v}$ guarantees that some open set around $\mathbf{v}$ is structurally equivalent. Thus, any generic property that is preserved under structural equivalence holds for all structurally stable fields. On the other hand, properties that hold for all structurally stable cases need not be generic, since the set of stable fields may not be dense in the underlying space. Peixoto's theorem [158] states that stable vector fields on two-dimensional compact and orientable manifolds are dense, but this is no longer true in higher dimensions [202].

2.3.4. Critical Points and Degeneracies

In steady two-dimensional flows $\mathbf{v} : D \to \mathbb{R}^2$, *critical points* are an important type of limit points. Critical points can be detected locally as points $\mathbf{p}$ at which the vector field magnitude vanishes ($\|\mathbf{v}(\mathbf{p})\| = 0$). If we ignore walls along which the vector magnitude may be constrained to be zero, this generically happens at isolated points. Critical points are the only places at which streamlines may intersect asymptotically.

Generically, the Jacobian matrix $\mathbf{J}_\mathbf{v}(\mathbf{p})$ at critical points has full rank (*linear* or *first order critical point*). In this case, the real $R_i = \Re(\lambda_i)$ and imaginary $I_i = \Im(\lambda_i)$ parts of its eigenvalues can be used to classify the point as a saddle, node, focus, or center, as illustrated in Figure 2.9. Centers (d) are only structurally stable when the field is constrained to zero divergence ($\nabla \cdot \mathbf{v} \equiv 0$).

The vicinity of a general (nonlinear) critical point $\mathbf{p}$ can be segmented into regions of different asymptotic flow behavior:

1. In a *hyperbolic sector* (Figure 2.10 (a)), streamlines diverge from $\mathbf{p}$ in both directions.

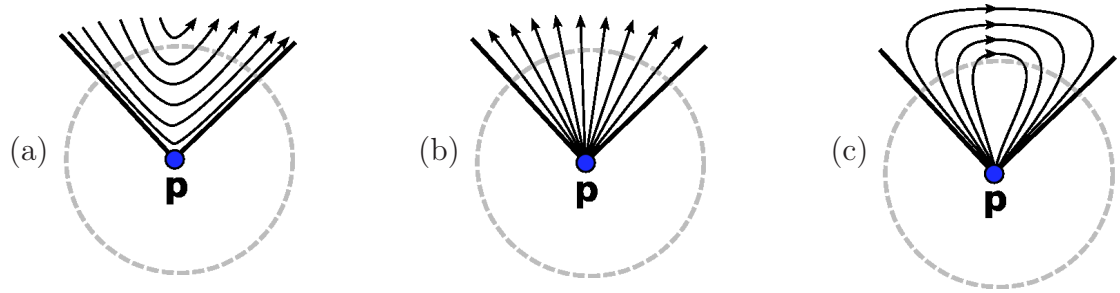


Figure 2.10.: General critical points $\mathbf{p}$ can have hyperbolic (a), parabolic (b), or elliptic (c) sectors. Only (a) and (b) are structurally stable.

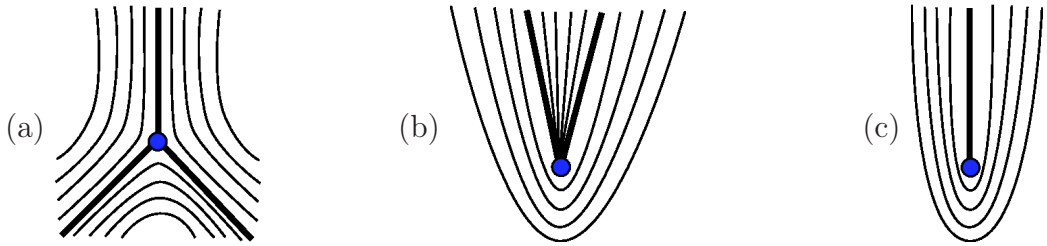


Figure 2.11.: Linear degenerate points in tensor fields can be classified as trisectors (a), double wedges (b), or single wedges (c).

2. In a *parabolic sector* (Figure 2.10 (b)), streamlines reach $\mathbf{p}$ in one direction.

3. In an *elliptic sector* (Figure 2.10 (c)), streamlines reach $\mathbf{p}$ in both directions.

Critical points are characterized by their *Poincaré index*, which is defined as the number of counterclockwise revolutions of $\mathbf{v}(\mathbf{x})$ when $\mathbf{x}$ moves in counterclockwise direction along the boundary of a ball $B_\epsilon(\mathbf{p})$ that does not contain any other critical point. Let n_e and n_h be the number of elliptic and hyperbolic sectors, respectively. Then, the index of a critical point is given by [72]

$$\text{index} = 1 + \frac{n_e - n_h}{2} \tag{2.38}$$

The index of an arbitrary curve in the vector field equals the sum of indices of the enclosed critical points. It follows from the continuity of the field that indices in vector fields are always integers. Among the generic critical points, saddles have index -1, all other first order critical point have index 1.

Tensor topology as defined by Delmarcelle and Hesselink [56] is the topology of *tensor lines,* the tangent curves in eigenvector fields. In two dimensions, the analogs of critical points are *degenerate* or *semiumbilic points* in which both eigenvalues are equal. At these loci, the eigenvectors become ill-defined and tensor lines intersect.

Similar to first order critical points, linear degenerate points are classified based on first derivatives of the tensor field [56]. Figure 2.11 presents the three structurally stable classes of degenerate points in $\mathbb{R}^2$, trisectors (a) and two types of wedges: Double wedges have

a full parabolic sector (b), while single wedges only have one individual tensor line that enters the degeneracy (c). The Poincaré index is transferred to tensor fields by counting the number of counterclockwise revolutions of the (unoriented) eigenvector. Due to the lack of orientation, indices in tensor fields can be half-integers. Trisectors have index $-\frac{1}{2}$, wedges have index $\frac{1}{2}$.

In three dimensions, there are different types of tensor degeneracies: At type P (planar) degeneracies, the larger two eigenvalues are equal (major and medium tensor lines intersect), while type L (linear) features involve the smaller eigenvalue pair. In a triple degeneracy, all eigenvalues coincide. In generic 3D tensor data, triple degeneracies are structurally unstable, but type L and type P features form stable lines [51, 254]. Stable tensor degeneracies in 3D are restricted to a plane, and can be classified like 2D degenerate points within that plane [255].

2.3.5. Separatrices, Basins, and Faces

Critical points are not the only type of limit points in planar vector field topology: *Closed streamlines* are integral curves that are traversed by a massless particle periodically. They can only be detected globally, and act as sources or sinks. Moreover, streamlines end at *inflow* or *outflow parts* at the boundary of the domain. These parts are separated from each other by *boundary switch points,* at which the flow is parallel to the boundary. If the flow is constrained to zero magnitude along walls in the domain, streamlines may also end in *attachment nodes* and *detachment nodes.*

A full topological characterization of a planar vector field connects the limit points by *separatrices,* streamlines which form the boundary of hyperbolic sectors. In the first order case, integration of separatrices starts at saddles, boundary switch points, and attachment / detachment nodes. Additionally, closed streamlines act as separatrices.

Limit points and separatrices form the nodes and edges of the *topological skeleton.* For presentation, these nodes and edges are placed in the locations from which they were extracted, so even though the features themselves are based on topology alone, their visualization also takes into account the employed metric, in the sense that distances between limit points, as well as lengths and crossing angles of separatrices are preserved.

The topological skeleton partitions the domain into regions of uniform asymptotic flow behavior: In each region, all streamlines start in the same source and end in the same sink. This is formalized by the notions of basins and faces. The α *basin* of a source $\mathbf{p}$ is the set of all points that have $\mathbf{p}$ as an α limit point. Similarly, if $\mathbf{p}$ is a sink, its ω basin consists of all points that have $\mathbf{p}$ as their ω limit point. The *faces* of the topological skeleton result from intersecting all α and ω basins [183].

Due to the lack of orientation, the distinction between α and ω basins does not carry over to tensor fields. Otherwise, their topological skeleton is comparable: Separatrices bound hyperbolic sectors and closed tensor lines can be detected using similar methods as in vector fields [245]. In 3D, Zheng et al. [255] have rendered separating surfaces emanating from degenerate lines as arrays of hyperstreamlines, but a complete topological partitioning of the domain has not been attempted.

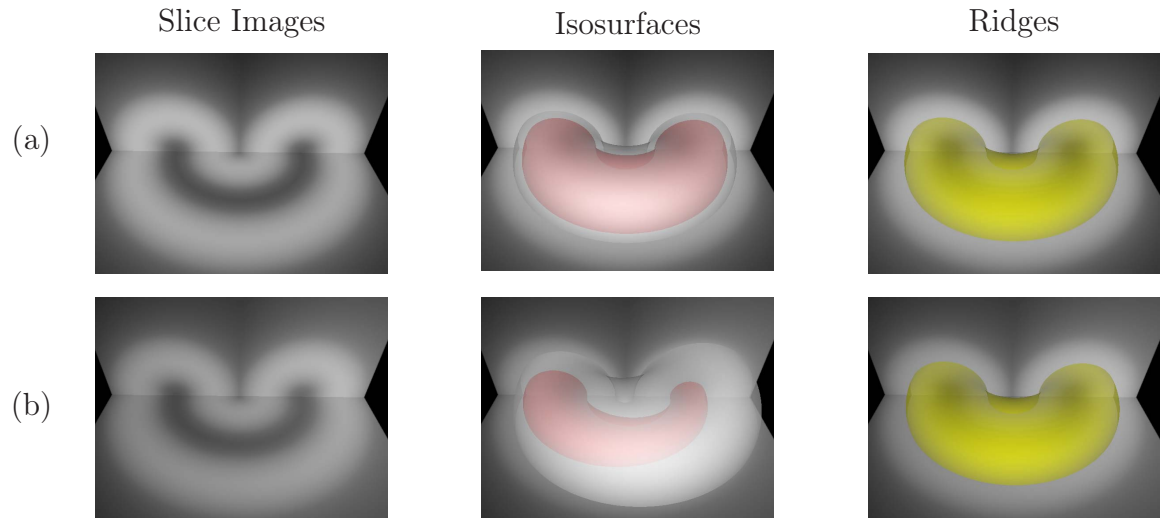


Figure 2.12.: Ridges capture extremal structures. In (a), an isosurface with an extremal isovalue is extracted as a ridge surface. After adding a slight gradient in (b), the ridge is almost unaffected, but no longer corresponds to an isosurface.

2.3.6. Height Creases

Local extrema are characteristic structures of scalar fields. Generically, they occur in isolated points. For cases in which higher-dimensional extremal features are more appropriate, there exist crease definitions, which generalize local extrema to ridges (constrained local maxima) and valleys (constrained local minima).

As an example, consider the datasets presented in Figure 2.12, which attain large values on the surface of a torus. Even though the values in (a) are constant along the torus, traditional algorithms for isosurface extraction [132] produce a slightly larger (gray) and a slightly smaller (red) surface, since they require a non-extremal isovalue. In contrast, algorithms for crease surface extraction successfully reconstruct the shape. In Figure 2.12 (b), a slight gradient in horizontal direction is overlaid on the original data. In this case, isosurfaces do not even provide a usable approximation of the torus, while the crease geometry is more or less unchanged.

The crease definition considered in this dissertation has been introduced to visual computing by Haralick [87], who suggested to use creases to capture highlight and shadow lines in natural images. Haralick defines creases as lines in a 2D image along which the first directional derivative, taken in a direction which extremizes the second directional derivative, changes sign. This is known as the "height crease definition", since it is motivated by treating the intensity profile of an image as a height field and it is closely related to the notion of ridges and valleys in surface topography.

Different crease definitions have been proposed and there has been some dispute over the "correct" one [120]. After a theoretical analysis and visual comparison of results, Eberly et al. [64] conclude that height creases are most suitable for digital image analysis. Our definition of d-dimensional creases in n-dimensional space follows their idea, but adopts

the simplified notation used in [115].

Assume a C^2 scalar field $f : \mathbb{R}^n \to \mathbb{R}$. Let $\mathbf{g} = \nabla f$ be its gradient and $\mathbf{H}$ be its Hessian with eigenvectors $\mathbf{e}_i$ and eigenvalues λ_i, $i \in \{1, 2, \ldots, n\}$, sorted such that $\lambda_1 \geq \ldots \geq \lambda_n$. Then, a d-dimensional height ridge is given by the conditions

$$\forall_{d<i\leq n} \quad \mathbf{g} \cdot \mathbf{e}_i = 0 \quad \wedge \quad \lambda_i < 0 \tag{2.39}$$

Intuitively, this means that f attains a local maximum in the $n-d$ directions of strongest convexity. For ridge surfaces in $\mathbb{R}^3$, this definition simplifies to

$$\mathbf{g} \cdot \mathbf{e}_3 = 0 \quad \wedge \quad \lambda_3 < 0 \tag{2.40}$$

The valleys of f are exactly the ridges of $-f$, so they need not be discussed separately. The *crease surfaces* defined by Equation (2.40) are two-dimensional manifolds and should not be confused with *surface creases*, which are lines of extremal curvature on general surfaces [19].

Even though height creases are not preserved by homeomorphisms, they are comparable to topological features in that they extract a structural skeleton of the data. For example, creases are used to extract medial cores from grayscale images [165]. A more formal relation to topological visualization lies in the fact that critical points in vector fields are subsets of the local minima in vector field magnitude. Similarly, degenerate lines in 3D tensor fields are subsets of crease lines in eigenvalue skewness [215].

2.3.7. Surface Topology

The topological analysis of surfaces is based on the notion of manifolds. An *n-manifold* M^n is a topological space in which each point $\mathbf{p}$ has a neighborhood which is homeomorphic to Euclidean n-space, without any boundaries $\mathbf{b}$ or ramifications $\mathbf{r}$ (cf. Figure 2.13). A *surface* is a two-manifold. A *surface with boundary* includes points with a neighborhood homeomorphic to the Euclidean half-space.

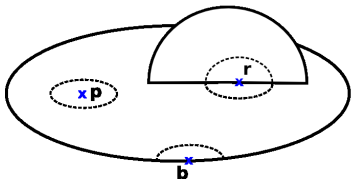


Figure 2.13.: A non-manifold surface patch, with an inner point $\mathbf{p}$, a boundary point $\mathbf{b}$, and a ramification $\mathbf{r}$

In each Euclidean neighborhood, we can introduce a coordinate system. On a *differentiable manifold,* the change between such local coordinate systems is smooth. If there exist local coordinate systems such that any change between them preserves orientation, the manifold is *orientable,* and its orientation allows one to distinguish two sides.

For the formal definition, we need additional notions. A topological space X is *Hausdorff* if for any two $x, y \in X$, $x \neq y$, there are disjoint open sets U and V with $x \in U$ and $y \in V$. In Hausdorff spaces, limits of sequences are unique. All metric spaces are Hausdorff.

A *basis* B of a topological space X is a collection of subsets of X such that the open sets are exactly the unions of members of B. The space X is *second countable* if it has a countable basis. Euclidean spaces are second countable, since the ϵ-balls with rational ϵ around rational coordinates are countable and form a basis.

A local coordinate system on a manifold M^n is given by a *chart* (U, ϕ), which is a pair of an open set $U \subseteq M^n$ and a homeomorphism $\phi : U \to U' \subseteq \mathbb{R}^n$, where U' is open. A *smooth atlas* is a collection of charts such that:

1. Each point $x \in M^n$ is in the domain of some chart.
2. Between two charts $\phi : U \to U'$ and $\psi : V \to V'$ with intersecting domains, the change of coordinates $\phi\psi^{-1} : \psi(U \cap V) \to \phi(U \cap V)$ is C^∞.

The definition of a differentiable manifold requires an atlas that contains a maximal set of charts that fulfill the above conditions. In practice, one works with a smaller atlas that may contain a few charts, or even just a single one.

Finally, a differentiable manifold is defined as a second countable Hausdorff space M^n with a maximal smooth atlas. It is orientable if there exists an atlas such that the Jacobian of $\phi\psi^{-1}$ has a positive determinant for any two ϕ, ψ and for all points in the domain. Topological properties of manifolds are the ones which are invariant under homeomorphisms. This includes orientability and the number of boundary components and handles.

Part I.

Structural Features in DW-MRI Data

3. Topological Features in Diffusion MRI Data

Topological methods have proven effective for the visualization of flow fields (cf. Section 2.3). They are based on the connectivity in the vector field: The domain is partitioned into regions in which all streamlines connect the same source to the same sink, or in other words, into regions in which the flow exhibits the same asymptotic behavior.

Connectivity in the human brain is investigated via diffusion MRI and algorithms for tractography (cf. Section 2.2). The fact that estimating connectivity is one of the main motivations for acquiring diffusion MRI data suggests that a topological analysis should be beneficial. It is the goal of our work [192] to find an adequate method for this.

Given the popularity of the second-order diffusion tensor model (cf. Section 2.2.3), a natural starting point is the work by Delmarcelle and Hesselink [56], which generalizes the concepts of vector field topology to second-order tensor fields. Based on their definitions, further research has been conducted on 3D tensor topology [93, 254, 255]. More specifically, Zheng et al. [256] have argued that applying tensor topology to DT-MRI is likely to prove beneficial. While they expect noise artifacts to dominate a naïve topological visualization of DT-MRI data sets, they suggest that an additional selection of the most important features would produce a "simple yet powerful representation" [255]. However, no results from applying tensor topology to DT-MRI have been published so far.

In Section 3.2, we discuss the interpretation of the features from tensor topology and present both experimental results and theoretical arguments which suggest that degenerate lines are, unfortunately, not useful in the context of DT-MRI. As a replacement, we propose a new paradigm for topological visualization of diffusion MRI data (Section 3.3). In contrast to the existing "tensor topology", we refer to it as "diffusion MRI topology" to reflect the fact that we neither restrict ourselves to the second-order diffusion tensor model, nor do we expect our approach to be useful for tensor fields that describe different phenomena (e.g., stress tensors [254]).

The further structure of this chapter is as follows: In Section 3.4, we propose an algorithm to extract the new type of features. In Section 3.5, we demonstrate the robustness of our method under noise and present experimental results to illustrate that the novel features allow a meaningful interpretation of the data. Finally, in Section 3.6, we conclude the chapter and discuss possible directions for future work.

3.1. Related Work

Based on the fundamental ideas of topological visualization, our work defines anatomically relevant features in diffusion MRI data. Tricoche et al. [215] proposed an alternative generalization of tensor topology: Starting from the observation that degenerate lines coincide with crease lines of skewness, they propose to extract creases in fractional anisotropy in-

stead, and they obtain meaningful results in DT-MRI data. Their work appeared later than our own [192], but it connects prior work on anisotropy crease surfaces [114] with tensor topology. In contrast to our work, anisotropy creases rely on scalar anisotropy rather than connectivity, and they extract structural skeletons rather than partitioning the domain. In this sense, both methods are complementary. An in-depth discussion and extension of anisotropy creases is given in Chapters 4 and 5 of this dissertation.

The works by Enders et al. [66] and by Chen et al. [43] are similar to ours in that they depict fiber pathways as a whole. However, their approach is to cluster and to wrap streamlines from a deterministic fiber tracking method. In contrast, our pipeline does not involve any streamlines. Rather, we partition grey matter voxels based on the results of a probabilistic method and infer the pathways that connect them in a second step.

Jonasson et al. [104] also depict fiber tracts as a whole, but aim at an interactive segmentation, rather than at a visualization of the dataset. Their approach relies on the placement of an initial seed for a surface growing algorithm, which is driven by the similarity of diffusion tensors in adjacent voxels, and it does not determine connectivity explicitly.

Our paradigm for topological diffusion MRI visualization draws on methods for cortex parcellation, which were used in the neuroscience community [103, 6, 118, 160]. They show that changes in connectivity profiles allow for the partitioning of grey matter into functionally distinct regions. Our focus is to construct a novel visualization method based on this insight. Moreover, existing approaches do not consider the asymptotic behavior of the employed tractography methods, so they do not constitute a topological analysis.

To reflect the uncertainty inherent in the inferred connectivity, our topological features are fuzzy. In flow topology, uncertainty has not yet played a major role. Salzbrunn and Scheuermann [182] have introduced "fuzzy" streamline predicates as a means to define characteristic sets of predicates which are difficult to compute directly. However, they do not use them to visualize uncertainty. To the best of our knowledge, we present the first approach in which fuzzy topological features convey the level of confidence to the user.

3.2. Degenerate Lines in DT-MRI Fields

The expressive power of topological flow visualization is largely owed to the physical meaning of the critical points from Section 2.3.4: Sinks, sources, and saddles clearly are important locations in flow fields. Despite the formal analogy, the interpretation of degenerate features in DT-MRI data differs fundamentally. Unlike in vector fields, connectivity in DT-MRI can only be inferred in a probabilistic sense. The major hyperstreamlines of a diffusion tensor field can be interpreted as maximum likelihood pathways [17], but type P features are merely locations in which no single direction has maximum likelihood, not places in which the pathway "ends" in the sense in which a streamline ends in a sink.

Even if this means that degenerate lines have limited relevance for the topology of neuronal fiber pathways, they may still constitute an interesting tool for the analysis of DT-MRI data if they provide stable features in practice. Unfortunately, our experiments indicate that this is not the case.

3.2.1. Experimental Setup

In order to explore the potential of degenerate features in DT-MRI data, we implemented the prediction-correction scheme from [254], which is based on discriminant constraint functions and Hessian factorization. To obtain the best possible results even under difficult conditions, we allow for a large number of Newton-Raphson iterations in the correction phase and repeat failed steps with an extremely small stepsize.

The experiments were performed on the data described in Section 1.2. To avoid visual clutter, the analysis is limited to a region of interest which spans $21 \times 29 \times 14$ voxels at the center of the corpus callosum. As suggested in [256], we only consider regions of sufficient anisotropy (FA ≥ 0.2). The tractography in Figure 3.1 (a) is obtained by major eigenvector integration and is shown from superior. It includes the corpus callosum (in red), the cingulum bundles (in green) and a small part of the pyramidal tract (in dark blue, at the right and left image boundaries).

While type L features are not part of the major hyperstreamline topology, Zheng et al. [256] suggested that they may be of particular interest for DT-MRI, so we included them in our analysis. For comparison, we also extracted both streamlines and degenerate lines after adding Gaussian noise to the DWI and T_2 images and re-estimating the diffusion tensors. The standard deviation was chosen as $\sigma = A/\mathrm{SNR}$ with $\mathrm{SNR} \in \{12, 8\}$, where A was the average of signal intensities within the white matter mask.

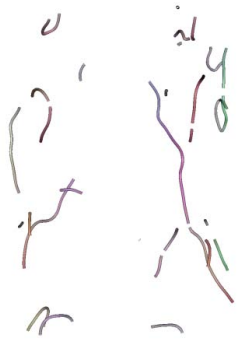
3.2.2. Results

The degenerate lines in Figure 3.1 are colored using the same XYZ-RGB scheme as the tractography. For type P (L), the color indicates the minor (major) eigenvector direction of the tensor field. The degenerate lines themselves are generally not aligned with any eigenvector direction, so the color coding does not indicate the direction of the features. Rather, red type L features are located within the corpus callosum, green ones are in the cingulum bundle, and blue ones are in the pyramidal tract. Unfortunately, the degenerate features do not have any evident correlation with known anatomical structures.

Since our dataset has ten times the minimum number of DWIs required to estimate the tensors, the noise can be considered low and moderate, respectively, which is reflected by the fact that the major features of the tractography remain discernible at all noise levels. Still, the degenerate features of type L change significantly. Even though type P features appear more stable, this result depends on the choice of interpolation. In Figure 3.1, we used a smooth (C^2) B-spline approximation of the tensor data [154], which stabilizes feature extraction. Figure 3.2 illustrates the effect of using trilinear interpolation instead. On exactly the same data, results differ noticeably.

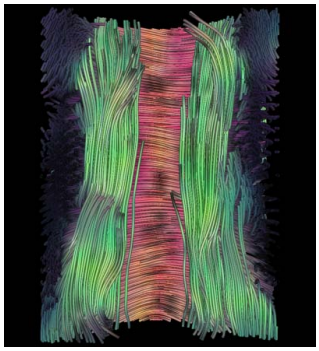
Finally, we demonstrate that the encountered instabilities neither indicate a general flaw in the concept of tensor topology, nor an error in our implementation. To this end, we present results on a randomly generated dataset, similar to the one used by Zheng et al. [254]. Figure 3.3 shows both type L (cool colors) and type P features (warm colors). In this case, changing the interpolation scheme alters the exact shape of the features slightly, but generally leaves them well-recognizable.

(a) Fiber tracts, no added noise

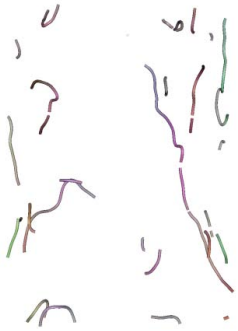


(b) Type P, no added noise

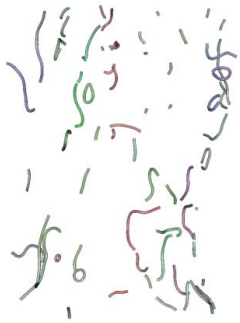
(c) Type L, no added noise



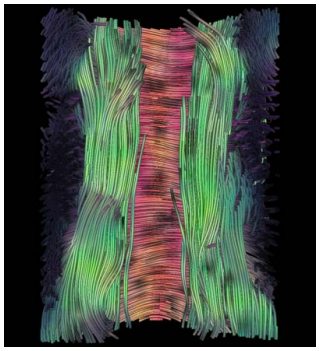
(d) Fiber tracts, SNR=12



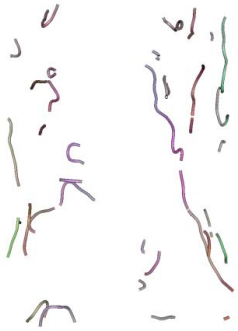
(e) Type P, SNR=12



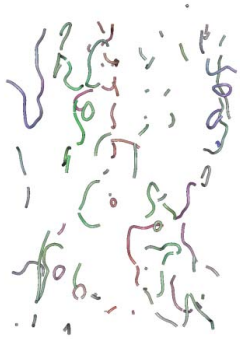
(f) Type L, SNR=12



(g) Fiber tracts, SNR=8

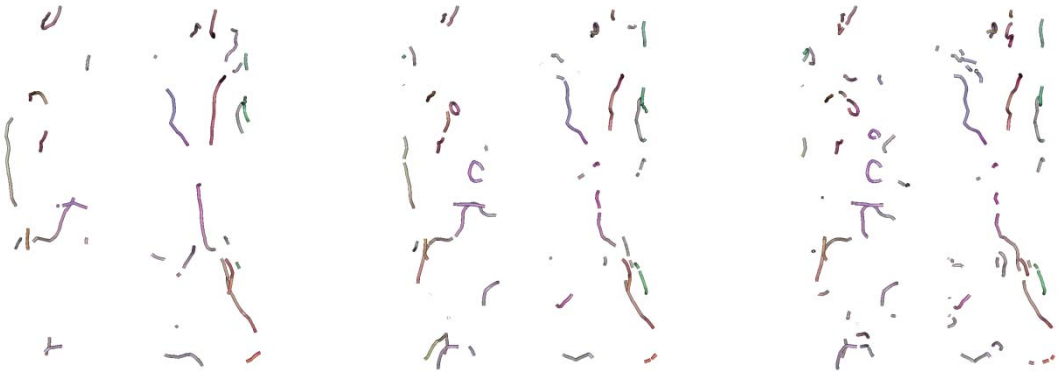


(h) Type P, SNR=8



(i) Type L, SNR=8

Figure 3.1.: A comparison of type P and type L features under Gaussian noise shows significant changes, even for moderate levels of noise.



(a) No added noise (b) SNR=12 (b) SNR=8

Figure 3.2.: Type P features with linear interpolation instead of B-spline approximation as in Figure 3.1. Feature lines depend significantly on the choice of interpolation.

(a) B-spline approximation (b) Trilinear interpolation

Figure 3.3.: In a generic dataset, the degenerate lines are far less affected by the choice of interpolation.

3.2.3. Interpretation

In our experiments, we have made two discouraging observations: First, degenerate lines in DT-MRI data do not have a clear relation to anatomical structures. Second, they are very sensitive to noise and to the choice of interpolation.

One reason for this lies in specific properties of DT-MRI data. Diffusion tensor fields from the human brain are not fully unconstrained second-order tensor fields. Rather, investigating streamlines in DT-MRI data assumes a single dominant fiber bundle per voxel, and typical single fiber models result in perfectly linear tensors [17]. In other words, we can expect regions in which the DT-MRI model is appropriate to be densely filled with type L features, whose exact location will depend on factors outside the model, including artifacts from noise and interpolation. This explains the large number of unstable type L features in Figure 3.1.

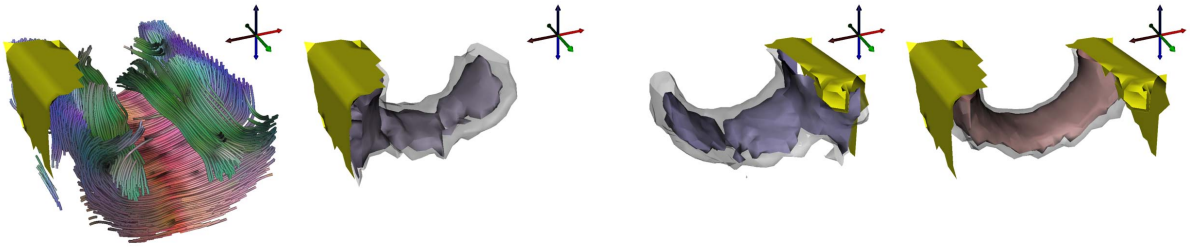
A second reason for the observed instability of degenerate lines applies to tensor fields in general: Tricoche et al. [215] have recently pointed out that degenerate lines can be expressed as crease lines of skewness. Since skewness depends on higher eigenvalue moments, it is sensitive to perturbations [12]. In addition, height creases depend on higher derivatives (cf. Section 2.3.6). Thus, noise can be expected to affect degenerate lines noticeably in general.

3.3. Topological Features in Diffusion MRI Data

Previous research on tensor topology has started from mathematical analogies to flow topology [56], which is appropriate to define stable features in generic tensor data. In our work, we are concerned with finding features that have a meaningful interpretation in the context of our particular application domain, so we choose brain anatomy as the starting point of our reasoning.

The white matter pathways whose connectivity we would like to investigate, are formed by axons. Even though axons have an orientation – they start at a cell soma and end in a synapse – diffusion imaging does not reveal this polarity, so we cannot distinguish if a connection endpoint is a source or a sink. So far, this agrees with tensor topology, which does not make this distinction either.

Critical points in flow topology are an instance of the more general notion of limit sets: They are locations in which a streamline integration starts or ends. In general, such limit sets do not necessarily form points. For example, the degenerate locations in 3D tensor topology form lines. At the scope of diffusion images of the brain, neuronal pathways end at surfaces, namely, at the interfaces between grey and white matter, or between white matter and the boundary of the domain. Recently, so-called cortex parcellation studies have shown that, to a certain extent, functionally distinct regions within grey matter can be found by considering changes in their connectivity profile [103, 6]. We will call connected regions of uniform connectivity, which are likely to represent anatomically meaningful units, *critical regions,* and identify them as the suitable limit sets for our diffusion MRI topology.



(a) A critical region (yellow surface) (b) Its 0.4-/0.25-basins (purple/transparent) (c) The counterpart of (b) on the right side (d) Their common 0.4- and 0.25-faces

Figure 3.4.: The basin of a critical region (a) consists of the voxels from which a probabilistic tractography reaches the region (b), (c). A face of two regions consists of the voxels that connect them (d).

As discussed in Section 3.2, the endpoints of streamlines that result from deterministic fiber tracking methods do not necessarily coincide with endpoints of the underlying neuronal pathways, so we do not consider them appropriate for defining a diffusion MRI topology. Instead, we base our analysis on the asymptotic behavior of a probabilistic fiber tracking approach [119] that employs the widely used diffusion tensor model and will be described in Section 3.4.1. Alternative methods, which may or may not depend on diffusion tensors (e.g., [17]), could be plugged into our framework, making its use independent of the preferred choice of diffusion and fiber models.

3.3.1. Critical Regions and Basins

The fact that the selected fiber tracking method provides a probabilistic connectivity measure has to be reflected in the definition of topological features from its results. In topological flow visualization, the α-basin of a source is the union of all streamlines that emerge from it. Accordingly, the ω-basin of a sink is the union of streamlines that end in it [183]. In analogy to these notions, we define the p-basin of a critical region as the set of points from which a probabilistic tractography reaches that region with probability $P \geq p$. For a point that connects two regions, we expect that around half of the particles end in each region, so we typically consider p-basins with $p < 0.5$.

To clarify these basic notions visually, we present some examples obtained with the method from Section 3.4, on the same region of interest as in Figure 3.1. Since it is taken from the center of the brain, the critical regions segment the domain boundaries rather than the cortex. Figure 3.4 (a) shows the deterministic tractography from a posterior/left viewpoint and a sample critical region as a yellow surface. It corresponds to the left endpoints of the fibers that pass through the central part of the corpus callosum and extends to a portion of the internal capsule. This is understandable, since fibers from both structures intermingle in this region and are not clearly separated anatomically.

Figure 3.4 (b) presents the same critical region with its 0.4-basin instead of the tractography. To provide a confidence interval, the 0.25-basin is rendered transparently. As

expected, the basin extends over the central part of the corpus callosum and down towards the internal capsule. For the XYZ-RGB color coding of basins and faces, a weighted average is computed from the tensors within the corresponding structure, with the local probabilities as weights. Thus, the purple color of the basin indicates the mixture of fibers that run through the corpus callosum (red) and the internal capsule (blue).

3.3.2. Faces

In flow visualization, one is typically interested in the *faces* which result from all intersections of α- and ω-basins. These are regions of uniform asymptotic flow behavior, i.e., regions in which all streamlines emerge from the same source and end in the same sink. Taken together, they form the topological skeleton of a flow field.

In diffusion MRI topology, the corresponding notion is the p-face of a pair of critical regions, consisting of the set of points which connect both regions with probability $P \geq p$. The computation of this probability is left to Section 3.4.5. As an illustration, Figure 3.4 (c) presents the counterpart of the basin in (b) on the right side of the corpus callosum. Figure 3.4 (d) shows the common 0.4- and 0.25-faces of the two critical regions, which clearly depict the part of the corpus callosum that connects both sides.

3.4. Extraction of Topological Features

Figure 3.5 gives an overview of the processing pipeline that will be described in this section. It comprises a preprocessing step in which the fiber tracking is performed (Section 3.4.1), a clustering step which forms the critical regions (Sections 3.4.2 and 3.4.3), as well as algorithms for the extraction and ranking of faces for examination by the user (Sections 3.4.4 and 3.4.5). We expect that a topological visualization of diffusion MRI data will be of specific interest to researchers in neuroscience, so the proposed method aims to provide a sensible initial visualization, with the option of subsequent user interaction for formation and testing of specific hypotheses.

3.4.1. Preprocessing

As a first step in finding our topological features, we perform a probabilistic fiber tracking, using a 3D variant of the algorithm proposed by Koch et al. [119]. First, it classifies voxels as white matter, grey matter, and cerebrospinal fluid (CSF). The volume outside the brain is already masked during tensor estimation, based on low signal values. Voxels with a tensor trace $\mathrm{tr}(\mathbf{D})/3 > 10^{-9}\,\mathrm{m}^2/\mathrm{s}$ are marked as cerebrospinal fluid and grey matter is distinguished from white matter based on an anisotropy threshold (white matter: FA > 0.2). Isolated white or grey matter voxels are caused by fluctuations around the FA threshold, and removed in a post-processing step. Consequently, we call non-white matter voxels adjacent to white matter *interface voxels* and we add dummy voxels when white matter reaches the boundary of the domain.

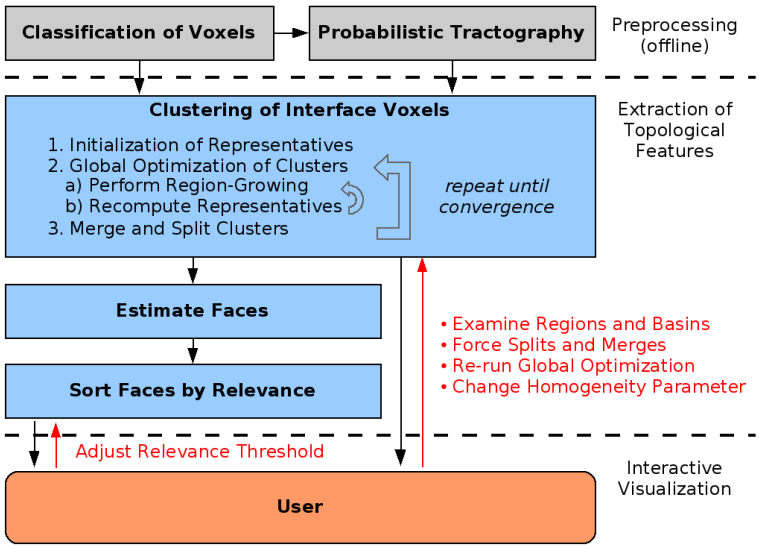


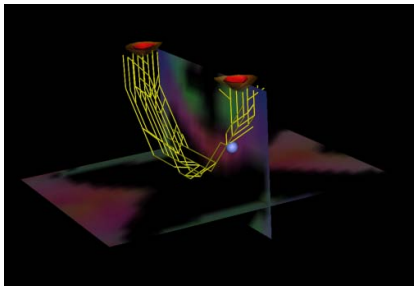
Figure 3.5.: An overview of the processing pipeline. The user can interact with it in a number of ways to test specific hypotheses.

The tractography itself is based on a random walk of particles at voxel resolution. Let $\mathbf{r}_{mn}$ be the unit vector pointing from voxel m to a voxel n in its 26-neighborhood $\mathcal{N}$ and $d_m(\mathbf{r}_{mn}) = \mathbf{r}_{mn}^T \mathbf{D}_m \mathbf{r}_{mn}$ be the apparent diffusivity in that direction, derived from the diffusion tensor $\mathbf{D}_m$. Then, Koch et al. define the transition probability $p(m \to n)$ from voxel m to n as

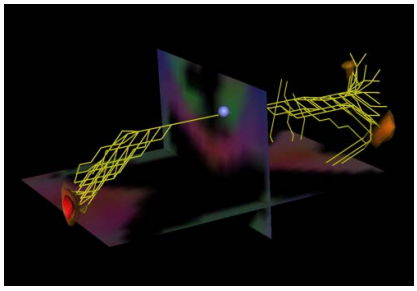
$$p(m \to n) = \frac{[d_m(\mathbf{r}_{mn}) + d_n(\mathbf{r}_{mn})]^a}{\sum_{n' \in \mathcal{N}} [d_m(\mathbf{r}_{mn'}) + d_{n'}(\mathbf{r}_{mn'})]^a} \tag{3.1}$$

where the exponent a is determined empirically and fixed at $a = 7$. Taking the exponent focuses the diffusivity profile to its major direction, which is likely to align with an actual fiber direction, while allowing for a certain surrounding spread. Some authors have used the product of diffusivities instead of the sum to adapt this method. In this modified form, Equation (3.1) has produced plausible cortex parcellations [6] and results that agreed with findings from fMRI [78].

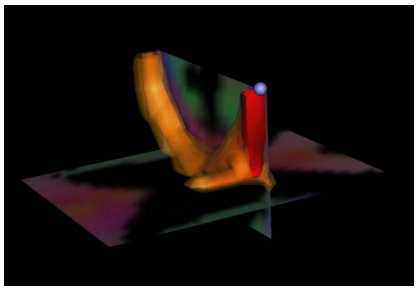
After the first step, Koch et al. restrict the probability distribution to directions that deviate less than 90° from the previous step. We make two small improvements to this: First, we additionally set the transition probabilities to CSF voxels to zero, because it is anatomically impossible that fiber tracts end in the CSF-filled ventricles. Second, we do not simply truncate the distribution at 90°, but rather weight the probabilities in forward direction with $\cos\phi$, where ϕ is the angle between $\mathbf{r}_{mn}$ and the current tracking direction $\mathbf{t}$, calculated from the direction $\mathbf{r}'_{mn}$ of the previous step as $\mathbf{t} = \mathbf{D}_m \mathbf{r}'_{mn}$. This definition of $\mathbf{t}$ accounts for the fact that the fiber direction changes from voxel to voxel and is analog to the "outgoing" direction in the tensorline propagation by Weinstein et al. [237]. Section 3.5.2 presents an example where these modifications are necessary to obtain correct results.



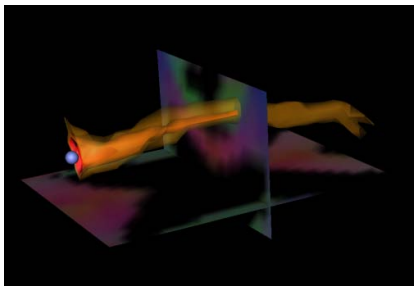
(a) Random walks in the *corpus callosum*



(b) Random walks in the *cingulum*



(c) Footprint in the *corpus callosum*



(d) Footprint in the *cingulum*

Figure 3.6.: Random walks from white matter voxels to the interface are generated. Consequently, footprints are computed that show from which parts of the white matter an interface voxel was reached.

The random walk is terminated when the particle reaches an interface voxel. For each white matter voxel, we trace 10 000 particles and record the percentage that goes to the individual interface voxels. Figures 3.6 (a) and (b) illustrate this: Starting from the seeding voxels (blue balls), a few random walks in the *corpus callosum* and *cingulum,* respectively, are visualized as yellow tubes. Red isosurfaces indicate interface voxels that were reached by a relevant percentage of the particles.

Similar to previous methods that pre-compute a deterministic tractography [24], the preprocessing is performed offline. For the region of interest in Figure 3.7 (4 948 white matter voxels), it takes more than six minutes on a 2 GHz Athlon 64 processor. Our modifications to the original algorithm account for 25% of this time.

3.4.2. Clustering Criteria for Critical Regions

Cortex parcellation studies have computed and clustered a correlation matrix for the interface voxels in the region of interest, either manually [103] or with k-means [6]. However, forming critical regions within a topological visualization method requires that the number of clusters is chosen automatically, based on the data. Moreover, we cannot ensure connectivity of the critical regions when considering only the correlation matrix, since it does not contain any information about voxel adjacency. Consequently, a novel approach is required for the clustering of critical regions. This subsection introduces some notation and formalizes suitable clustering criteria. A custom algorithm which fulfills these

requirements will then be presented in the following subsection.

Let $\mathcal{W}$ be the set of white matter voxels w, $W = |\mathcal{W}|$. Similarly, $\mathcal{I}$ is the set of interface voxels i, $I = |\mathcal{I}|$. Then, the tractography result for voxel w can be written as a vector $\mathbf{t}(w)$ of dimension I, where $t_i(w)$ is the percentage of particles originating from w that reached i. From this, we define the footprint $\mathbf{f}(i)$ of an interface voxel as a vector of dimension W:

$$f_w(i) = \mathrm{FA}(\mathbf{D}_w) \cdot t_i(w) \tag{3.2}$$

Weighting particles with fractional anisotropy has not been done by previous authors and is not strictly necessary to get sensible results. However, it helps to stabilize the clustering in the presence of noise, since the principal direction in regions of low FA is unreliable. For the interface voxels marked by blue balls in Figures 3.6 (c) and (d), isosurfaces indicate which white matter voxels have a super-threshold contribution to the footprint.

A clustering Γ of the interface voxels is a partition of $\mathcal{I}$ into C clusters $\Gamma_1, \ldots, \Gamma_C$, where we require that each Γ_c is connected. The number of clusters C is not known a priori and changes as part of the clustering process. For each cluster c, the footprint $\mathbf{F}(c)$ is defined as the accumulated footprint of its members:

$$\mathbf{F}(c) = \sum_{i \in \Gamma_c} \mathbf{f}(i) \tag{3.3}$$

The similarity $\psi_c(i)$ between a cluster c and an interface voxel i is defined as

$$\psi_c(i) = \frac{\mathbf{f}(i) \cdot \mathbf{F}(c)}{\|\mathbf{f}(i)\| \cdot \|\mathbf{F}(c)\|} \tag{3.4}$$

Since all involved vectors have non-negative components, $\psi_c(i) \in [0, 1]$. From this, the homogeneity Ψ_c of a cluster c is defined as

$$\Psi_c = \frac{\sum_{i \in \Gamma_c} \|\mathbf{f}(i)\| \psi_c(i)}{\sum_{i \in \Gamma_c} \|\mathbf{f}(i)\|} \tag{3.5}$$

Since the total number of particles that arrive at an interface voxel depends on the number of white matter voxels in its neighborhood, it is appropriate to normalize ψ_c by the product of footprint magnitudes in Equation (3.4). In contrast, the weighting in Equation (3.5) reflects the fact that interface voxels with a small number of particles should contribute less to the overall homogeneity of a cluster.

Let γ be a function that maps each interface voxel i to its cluster c ($\gamma(i) = c \Leftrightarrow i \in \Gamma_c$). Then, a clustering agrees with the data if the total homogeneity $\overline{\Psi}$ is high:

$$\overline{\Psi} = \frac{\sum_{i \in \mathcal{I}} \|\mathbf{f}(i)\| \psi_{\gamma(i)}(i)}{\sum_{i \in \mathcal{I}} \|\mathbf{f}(i)\|} \tag{3.6}$$

If we leave the problem unconstrained, $\overline{\Psi}$ reaches its optimum for the trivial clustering, in which each interface voxel has its own cluster. Thus, we require that the homogeneity of each individual cluster c should approximately equal a parameter h ($\Psi_c \approx h$).

In our experiments, values around $h \approx 0.2$ generally gave useful results. However, part of the insight in [6] has been gained by trying various values of k for the k-means clustering, so leaving h as a user-defined parameter is useful for allowing an interactive exploration of the data. Also, the authors of [6] try to discover whether the data supports further subdivision of specific clusters, so we allow interactive splitting and merging of user-selected clusters. A subsequent global optimization of $\overline{\Psi}$ indicates if a split resulted in valid sub-clusters: In that case, surrounding clusters should not change much.

3.4.3. Clustering Algorithm

To find a clustering according to the criteria of the previous section, our method proceeds in two steps: The first step follows a greedy local strategy to create an initial clustering Γ. The second step globally optimizes both the clustering and the number C of clusters with respect to $\overline{\Psi}$, preserving the conditions of connectivity and cluster homogeneity $\Psi_c \approx h$. Similar two-step methods have previously been used in computer vision to reduce the complexity of segmenting images into an unknown number of regions [30].

A common building block of both steps is a variant of the k-means algorithm that uses a fast-marching region-growing scheme to ensure connectivity of the resulting clusters. Like k-means, it iteratively computes new cluster footprints $\mathbf{F}^{n+1}$ from a given clustering Γ^n and subsequently uses them to re-assign all interface voxels to new clusters Γ^{n+1}. Convergence is assumed when only a small percentage (2%) of the voxels is re-assigned to a different cluster.

The footprints $\mathbf{F}^{n+1}$ are determined by evaluating Equation (3.3). Consequently, for each cluster c, the voxel $i \in \Gamma_c^n$ with the highest similarity $\psi_c^{n+1}(i)$ is selected as a seed point. Starting from these seeds, voxels which have not yet been assigned to Γ^{n+1} are added to an adjacent cluster c. In order to optimize $\overline{\Psi}$, voxels are added in descending order of their similarity $\psi_c^{n+1}(i)$. Thus, good-matching voxels are assigned early on, while dissimilar voxels are initially left free, which gives more suitable clusters the chance to become adjacent to them. This scheme is efficiently implemented using a priority queue.

To make the algorithm more stable, we replace the parameter h with two parameters, h^+ and h^-, where h^+ is slightly larger than h^-. If the average similarity Ψ_c of a cluster is smaller than h^-, the cluster is split, to allow for a more precise adaptation to the data. On the other hand, if merging two adjacent clusters would lead to a cluster homogeneity which is still larger than h^+, the merge is performed.

For initialization, each connected component on the interface is treated as a cluster and subdivided until h^- is reached. At this stage, the region-growing only acts locally on the voxels of the two newly formed sub-clusters. When a cluster is split, one half of its members are assigned arbitrarily to each of the two new clusters. After the first iteration of the region-growing algorithm, the results are again connected and converge to an optimum. In rare cases, this "careless" initialization causes very small sub-clusters to split off. This is acceptable, since such clusters will be re-merged later.

The initial clustering ignores interface voxels i with $\|\mathbf{f}(i)\| < 0.2$. This exploits the fact that many of the final clusters are separated by regions of small footprint magnitude. Ignoring these voxels is an extremely simple and cheap way to identify connected compo-

nents that are likely to coincide with relevant clusters. Experiments have indicated that using this heuristic does not affect the final result noticeably, but nearly doubles the speed of the clustering process.

When the initial clusters have been found, the region-growing is used to extend them to all interface voxels with $\|\mathbf{f}(i)\| > 0$, and to refine them until global convergence. After merging and splitting clusters as appropriate, this procedure is iterated until no more merges or splits are necessary. Since the initial clustering is usually quite good, convergence is reached quickly.

In our implementation, we exploit the fact that most interface voxels only connect to a small fraction of the white matter, i.e., the footprint vectors $\mathbf{f}$ are sparse. Thus, we store them as lists of <voxel index, value> pairs rather than as full-length arrays, which reduces the cost of evaluating Equations (3.3) and (3.4). On the region of interest shown in Section 3.3 ($I = 3\,408$, $W = 4\,948$), the full clustering took 1.3 s on a 2 GHz Athlon 64 processor. Afterwards, small modifications to h, or user-specified splits and merges, followed by a global optimization, take around half a second, making these operations appropriate for interactive exploration of the data.

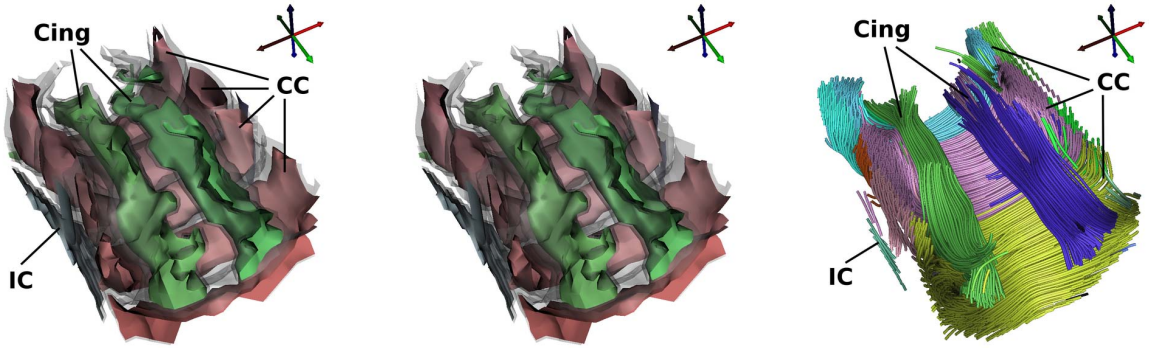
3.4.4. Definition of Faces

According to the definition in Section 3.3.2, finding the faces in diffusion MRI topology requires to determine the probability that a given voxel connects any two critical regions. This information can be collected in the tractography step by using particle pairs that leave the starting voxel in opposite directions. Pairs of interface voxels that are reached this way are connected through the starting voxel.

Even though the target space of such pairs is of order I^2, only a few pairs are actually connected, so for small enough regions of interest, a sparse representation makes this approach feasible. For example, in the region discussed above, probabilistic tractography from a single white matter voxel reaches 285 individual interface voxels on average, but only 1 135 voxel pairs. Still, this exact solution may become prohibitively expensive on larger regions of interest. Already in the case of Figure 3.9, the probabilities of more than $4 \cdot 10^7$ voxel pairs have to be stored. Thus, we present a simple heuristic that estimates the face probabilities from the cluster footprints $\mathbf{F}$ alone. Its fundamental idea is to divide the particles reaching a given region according to the ratio of particles that went to the remaining regions and to let them vote for a connection to these regions.

Let $T_c(w)$ be the percentage of particles from a white-matter voxel w that reach cluster c. If a single region collects more than 50% of the particles ($T_c(w) > 0.5$), we assume that the voxel w connects that region to itself with probability $P_{cc}(w) = 2 \cdot T_c(w) - 1$. To compute the connectivity between different regions, we remove these probabilities from T_c: Let $T'_c(w) = T_c(w) - P_{cc}(w)$ be the reduced percentages (such that $T'_c(w) \leq 0.5$) and let $\widetilde{P}(w) = \sum_c P_{cc}(w)$ be the probability that w connects any critical region to itself. Then, the estimated probability $P_{ab}(w)$ that w connects clusters a and b ($a \neq b$) is given by

$$P_{ab}(w) = T'_a(w) \cdot \frac{T'_b(w)}{1 - T'_a(w) - \widetilde{P}(w)} + T'_b(w) \cdot \frac{T'_a(w)}{1 - T'_b(w) - \widetilde{P}(w)} \tag{3.7}$$



(a) Exact faces (b) Heuristic faces (c) Clustered streamlines

Figure 3.7.: The nine most relevant faces in the region of interest. Exact faces (a) are well-approximated by our heuristic (b). A comparison with clustered streamlines (c) confirms their anatomical relevance.

Equation (3.7) is an ad hoc definition, designed to satisfy the requirements that the resulting probabilities are non-negative and form a partition of unity. Its symmetry $P_{ab}(w) = P_{ba}(w)$ reflects the fact that we cannot distinguish between sources and sinks.

Figure 3.7 provides a comparison of faces obtained from tracking particle pairs (in (a)) and faces computed with our heuristic (in (b)). Visually, the approximation appears adequate. For an objective comparison, we averaged the absolute difference between exact and approximated $P_{ab}(w)$ over $\mathcal{W}$ for the nine displayed faces. The absolute deviation ranged between 0.0007 and 0.0095, with an overall average of 0.0057. Over the voxels relevant for display ($P_{ab}(w) > 0.33$, corresponding to the confidence bounds in Figure 3.7), the average relative deviation of the heuristic from the exact method was between 0.5% and 10%, the overall average being 5%.

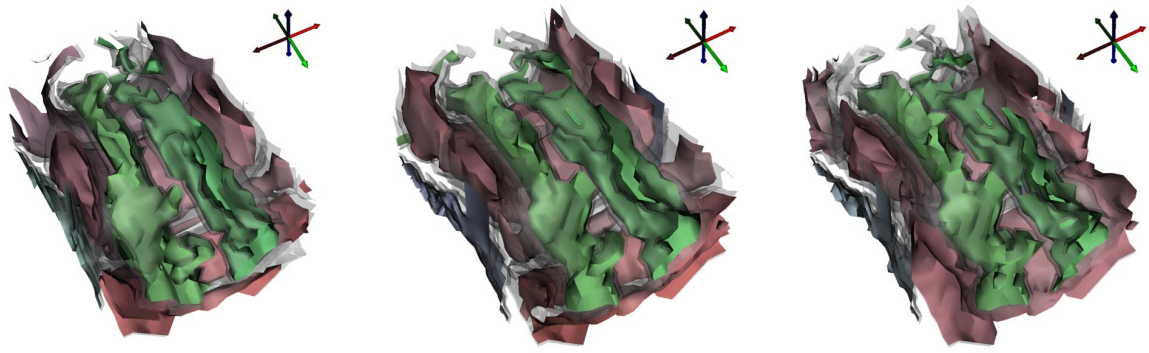
3.4.5. Selection of Relevant Faces

Like 3D flow topology, diffusion MRI topology suffers from the fact that three-dimensional faces may occlude each other. To alleviate this problem, we define a metric of face relevance, which helps the user to select only the important faces for display.

A face is relevant if the voxels it contains belong to it with a high probability. Thus, we formalize the relevance ρ_{ab} of a face between clusters a and b as its summed probability P_{ab} over $\mathcal{W}$, normalized by the magnitude of the joint footprint:

$$\rho_{ab} = \frac{\sum_{w \in \mathcal{W}} P_{ab}(w)}{\|\mathbf{F}_a + \mathbf{F}_b\|} \tag{3.8}$$

Equation (3.8) is evaluated for all possible pairs of clusters and the resulting faces are sorted according to their value of ρ. Then, the user can add faces until cluttering occurs or less important faces start to appear. In Figure 3.7, the nine most relevant p-faces



(a) Faces in less ideal data (b) Faces with SNR=12 (c) Faces with SNR=8

Figure 3.8.: The major faces ($\rho > 5$) remain recognizable at increased levels of noise.

have been selected this way and rendered with $p = 0.66$ (confidence bounds at $p = 0.33$). Computing and sorting took 0.27 s for exact faces, and around 10 ms with the heuristic.

Figure 3.7 (c) compares our results to a deterministic tractography, pseudo-colored according to a clustering with normalized cuts [32]. Both methods agree on the main features: The *corpus callosum* (CC) is subdivided into multiple regions and separated from the *cingulum bundles* (Cing). Only a few streamlines were seeded inside the *internal capsule* (IC, truncated by the region of interest), making it more recognizable in Figure 3.7 (a) and (b).

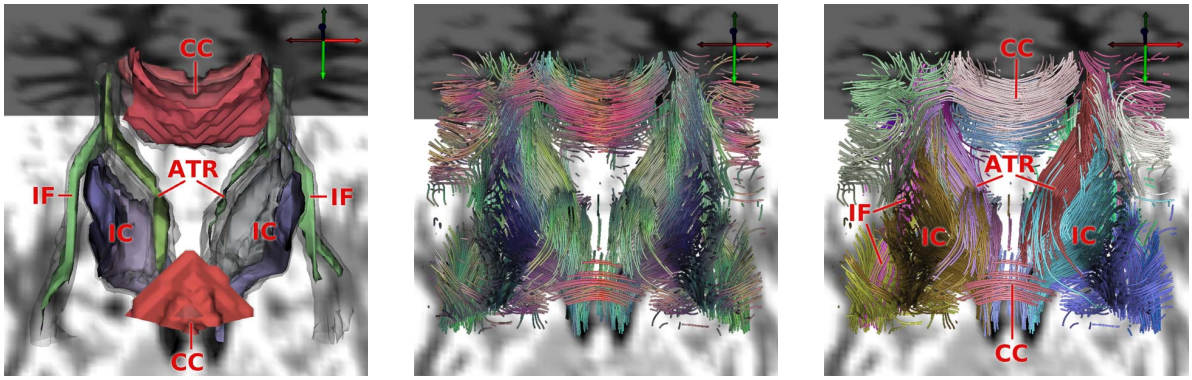
Note that in Figure 3.7 (c), some streamlines of the *corpus callosum* are clustered as part of the *cingulum bundle* and vice versa. This is due to the fact that Brun et al. [32] project the fibers to a low-dimensional feature space in which these streamlines are not well-separated. In contrast, our clustering works directly on the high-dimensional voxel footprints. Also, clustering the 2 290 displayed streamlines took 90 s, which made fine-tuning of the parameters more time intensive than with our method.

3.5. Experimental Results

3.5.1. Robustness under Noise

We tested the robustness of our features under both real and synthetic measurement noise. Figure 3.8 (a) presents results on a second dataset from the same subject as in Figure 3.7. It uses the same setup described in Section 1.2, but includes only one, rather than three, measurements per direction. This reduces the measurement time for a full-brain scan to 15 minutes, at the cost of stronger physical noise.

For direct comparison with the results in Section 3.2.2, Figures 3.8 (b) and (c) additionally show faces in the datasets which were corrupted with additive Gaussian noise. Since the noise causes higher variability in the voxel footprints, we have selected a lower homogeneity target h than in Figure 3.7 to obtain a comparable number of critical regions ($h = 0.24$ without noise, $h = 0.22$ with artificial noise, $h = 0.21$ with physical noise). In all examples, we show the 0.66- and 0.33-faces with a relevance value $\rho > 5$.



(a) 12 out of the 23 highest-ranking faces

(b) Standard tractography in XYZ-RGB coloring

(c) Standard tractography, pseudo-colored by cluster

Figure 3.9.: The relevant structures are also found in this larger region (a). Results from a deterministic tractography (b) and streamline clustering (c) confirm the result.

Even though the exact clustering changed slightly, all major structures remain well recognizable at all noise levels. Timings were around 2 s in all cases.

3.5.2. Results on a Larger Region of Interest

Figure 3.9 presents results on a second region of interest, which spans $49 \times 39 \times 25$ voxels in the center of the brain, right below the *corpus callosum*. It is shown from posterior/superior, with additional context from FA slices. As indicated by the streamlines in Figure 3.9 (b), this region contains more white matter than could be shown occlusion-free in a single rendering. Therefore, we sorted the 1 176 faces using the ρ-criterion and manually selected twelve out of the 23 highest-ranking ones for display. They depict the *inferior fronto-occipital fasciculus* (IF), the *internal capsule* (IC), the *anterior thalamic radiation* (ATR) and parts of the *corpus callosum* (CC, truncated by region of interest), which are also partly reflected in the streamline clustering in Figure 3.9 (c).

In this experiment, it becomes apparent that the probabilistic tractography method used for pre-processing is not well-suited for tracing thin fibers over long distances, since few random paths traverse them fully. The notable deviation between the core ($p = 0.4$) and the confidence bounds ($p = 0.2$) of IF and ATR in the rendering reflects the high amount of uncertainty that results from this.

Other authors have faced the same problem. In their analysis of long-range connectivity, Hagmann et al. [83] rely on a deterministic model. Perrin et al. [160] stick to a probabilistic method, but modify it heuristically by giving their particles inertia and letting them create child particles as they traverse long fascicles. To some degree, we profit from our own heuristic modifications, described in Section 3.4.1: With the original method [119], we were not able to reproduce the IF and ATR at all. However, finding a reliable and well-founded algorithm for the probabilistic estimation of long-range connectivity was outside the scope of our work.

	I	W	time	time/I	WM/I
Figure 3.7	3 408	4 948	1.3 s	0.4 ms	413
1/2 of Figure 3.9	7 803	12 765	10.8 s	1.4 ms	771
Figure 3.9	14 633	24 787	22.2 s	1.5 ms	852

Table 3.1.: Time per interface voxel depends on the average number of white matter voxels connected to them (WM/I), but is approximately constant otherwise.

To show that our clustering itself remains feasible for larger regions of interest, Table 3.1 summarizes the timings from our experiments. For comparison, we have included a region of interest that covers only the left half of Figure 3.9. While the input size doubles from the second to the third row, the time spent per interface voxel remains almost constant. However, the clustering for Figure 3.7 is much cheaper. This is partially explained by the sparse representation of voxel footprints, which exploits lower number of white matter voxels connected to an average interface voxel (WM/I).

3.6. Conclusion and Future Work

The motivation for this work was a lack of connectivity-based features in diffusion MRI data that would resemble the expressiveness of the features in flow topology. We have closed this gap by defining suitable, anatomically meaningful topological features in brain diffusion MRI and proposing a method for their extraction.

At the same time, we have contributed a method for the visual analysis of results from probabilistic tractography. While visualization research focused on deterministic streamline techniques, researchers interested in quantitative connectivity studies have deemed the confidence measures provided by probabilistic methods indispensable for their work and are lacking appropriate methods for visualizing their results. In recent papers, slice projections [18] or volume renderings [6] of scalar connectivity values derived from the tractography constitute the state of the art.

A second contribution is to explain why degenerate lines are not useful in DT-MRI data. With this result, we do not question the fact that tensor topology holds the potential to extract interesting features from other types of tensor fields. In fact, Chapter 4 of this dissertation points out the role that degenerate lines in Hessian fields play for crease surfaces, and it even makes use of the algorithm by Zheng et al. [254].

While our work solves some open issues, it also leaves a number of questions to future research. To reduce the complexity of our approach, we have neglected the uncertainty in the critical regions themselves. It could be worthwhile to investigate if a probabilistic clustering further improves visualization. Also, more work could be done on the rendering of features: Currently, we assign uniform colors to the basins and faces. Textures could add information relevant for interpreting the probabilistic tractography, like local fiber orientations and their variance, or the local density of particles.

Out of the need to reflect the uncertainty inherent in our data, we have defined prob-

abilistic versions of some basic topological features, leading to expressive visualizations. While it is outside the scope of the present work, it would be interesting to derive a rigorous mathematical framework for fuzzy topological visualization that might rest on existing fuzzy set theory [97], and to apply it to other cases in which uncertainty visualization may play a role.

Finally, our approach integrates two topics of active research in the neuroscience community, namely, finding probabilistic fiber tracking methods that reliably reproduce fiber tracts known from anatomy (e.g., cf. [18]), and clustering grey matter voxels in a way that reflects functional units (e.g., cf. [6]). Our work has both benefited from this research and leads to a method that could help neuroscientists to better explore their data. It is our hope that having these tasks as part of our visualization pipeline will continue to create synergies between the two exciting fields of visualization and neuroscience.

4. Topology and Extraction of Crease Surfaces

Crease surfaces are two-dimensional manifolds along which a scalar field assumes a local maximum (ridge) or a local minimum (valley) in a constrained space. As demonstrated in Section 2.3.6, they are complementary to isosurfaces in that they are able to capture extremal structures in the data.

Despite the fact that applications of crease surfaces in visualization and computer graphics range from medical imaging [114] to vector field visualization [178, 157] and surface reconstruction from noisy point clouds [209], their mathematical properties have not been investigated thoroughly. In particular, all previous algorithms for crease surface extraction ignore degeneracies in the involved eigenvectors.

Our work [193] points out that such degeneracies are practically relevant, since they are structurally stable and affect the topology of the creases: Unlike isosurfaces, crease surfaces have boundaries and are not necessarily orientable. Because of these differences, the marching cubes case table is inappropriate for crease surface extraction, even though it was widely used for this task. Based on this insight, we propose an efficient algorithm which produces more accurate representations of crease surfaces.

This chapter is organized as follows: After discussing related work (Section 4.1), we describe our theoretical results on crease surface topology (Section 4.2) and present our novel algorithm for crease surface extraction (Section 4.3). We then demonstrate that our method is a clear improvement over the state of the art (Section 4.4), before we conclude the chapter (Section 4.5).

The results presented in this chapter are application-independent: They improve crease surface extraction in general. In the context of our efforts to visualize data from DW-MRI, the motivation to address this problem was provided by a previous work [114], which showed that anisotropy crease surfaces are meaningful features in DW-MRI. A more detailed discussion and extension of this particular application is presented in the following chapter.

4.1. Related Work

Crease lines have been studied extensively by Pizer et al. in the context of medial core extraction [165], which generalizes the Blum medial axis analysis of binary objects [27] to finding the core of objects in grayscale images. Pizer et al. employ a medial function, which yields high values at the center of an object, and they extract its core as ridges in medial function values. Lindeberg [130], whose formalism differs slightly from the presentation in Section 2.3.6, and Damon [51] carefully investigated the behavior of height creases in Gaussian scale space.

Crease lines have also been used for finding vortex cores in vector field visualization, for example by Miura and Kida [140] and by Sahner et al. [179], and for the extraction of characteristic lines in symmetric tensor fields by Tricoche et al. [215]. In this context, the parallel vector approach by Peikert and Roth [156] and the feature flow fields by Theisel and Seidel [211] provide popular algorithms for crease line extraction.

For crease surfaces, Furst et al. have proposed the "marching cores" algorithm [80], which addresses the problem of finding 2D creases in a 4D (3D+scale) space. In their "marching ridges" method [79], Furst and Pizer even extend this to the extraction of one- and two-dimensional creases from spaces of arbitrary dimension. To deal with such high complexity, they make simplifying assumptions, such as that the boundary of each face is only intersected twice by a ridge. Intersections are found as changes of sign in $\mathbf{g} \cdot \mathbf{e}_i$, after imposing a local orientation on $\mathbf{e}_i$ via a principal component analysis.

So far, the application of ridge surfaces in visualization has been restricted to single-scale analysis, so it has been sufficient to find crease surfaces in 3D space. To find skeletal structures in data from diffusion tensor MRI, Kindlmann et al. [114] have extracted ridge surfaces as isosurfaces of $\mathbf{g} \cdot \mathbf{e}_i = 0$, using the marching cubes algorithm [132] after imposing a per-cell orientation on $\mathbf{e}_i$ by tracking eigenvectors along subsampled cell edges.

Sadlo and Peikert [178] have used marching cubes on an adaptive grid to extract ridge surfaces which separate regions of different flow behavior in unsteady vector fields, using the original rule from [79] to orient eigenvectors. Another recent work on vector field visualization by Sahner et al. [180] has used crease surfaces, but employed a different, watershed-based definition. In computer graphics, Süßmuth and Greiner [209] used marching cubes to reconstruct surface meshes from noisy point clouds by extracting ridges of point cloud density. They use the height crease definition, but do not provide details on their way of orienting eigenvectors.

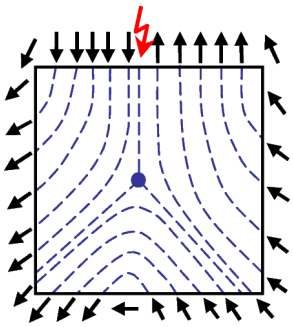
4.2. On the Topology of Crease Surfaces

In this section, we will discuss differences between the topology of isosurfaces and crease surfaces, which show that the marching cubes case table is inappropriate for crease extraction. This will lead us to a novel algorithm which is specialized for the extraction of 2D creases from 3D fields.

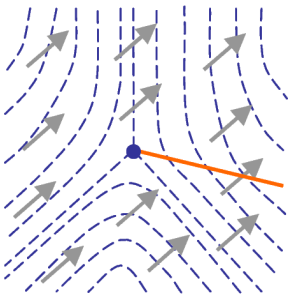
4.2.1. Degenerate Lines as Boundaries

Unit eigenvectors are only defined up to their sign. However, previous algorithms for the extraction of crease surfaces rely on a locally consistent sign of the involved eigenvector, so prior work has suggested different heuristics to impose a local orientation on it [145, 79, 114].

Unfortunately, orienting the eigenvector along the boundary of a cell face is impossible when the Hessian has a degeneracy in the interior of the face. This is due to the half-integral Poincaré indices of the three generic types of degenerate points in tensor fields (cf. Section 2.3.4). For example, Figure 4.1 (a) depicts a degenerate point (blue) and



(a) Orienting eigenvectors can lead to a contradiction.



(b) Lines along which $\mathbf{g} \perp \mathbf{e}_i$ end in degeneracies.

Figure 4.1.: Non-orientability of eigenvectors (a) leads to the fact that creases end at degenerate points (b).

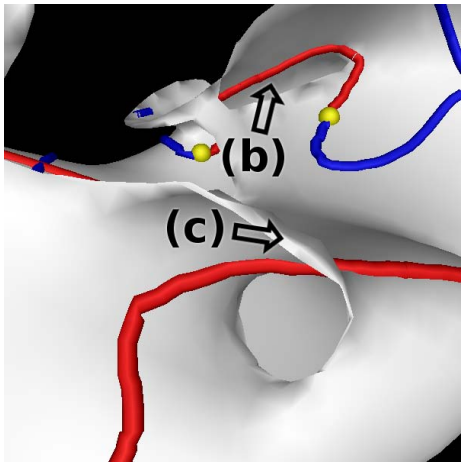
indicates eigenvector directions in the plane around it by dashed blue lines. As indicated by the arrows, trying to impose a consistent sign along the cell boundary around the degeneracy (black) leads to a contradiction. This case is not rare in practice: Degenerate loci in symmetric 3D tensor fields generally form stable lines (cf. Section 2.3.4).

In the context of crease surfaces, degenerate locations have traditionally been discussed as a source of numerical difficulty when imposing a local orientation on eigenvectors (e.g., [145]). Only recently, it has been pointed out that their presence implies that eigenvectors are not orientable in principle [157]. To the best of our knowledge, it has so far not been discussed that degenerate lines actually constitute one type of crease surface boundaries: Beside the obvious type of boundaries, which are caused by the side constraint on the eigenvalue ($\lambda_3 < 0$ or $\lambda_1 > 0$, respectively), ridge surfaces are bounded by type L degenerate lines ($\lambda_2 = \lambda_3$), and valley surfaces are bounded by type P lines ($\lambda_1 = \lambda_2$).

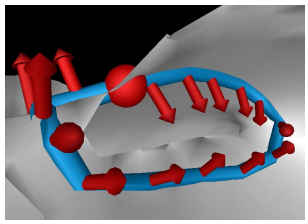
For crease lines in 2D, this insight follows directly from the Poincaré index of the degenerate point: Along the boundary of an ϵ-environment around it, the eigenvector turns $\pm\frac{1}{2}$ times (cf. Figure 2.11), while changes in the gradient can be neglected for sufficiently small ϵ. Thus, both vectors are orthogonal (i.e., the crease intersects the boundary) exactly once – the crease ends inside of it. Figure 4.1 (b) illustrates this: Along the crease (orange), gradient vectors (gray) are orthogonal to the eigenvectors (blue). Behind the degenerate point, both vectors are parallel, so the crease ends.

This argument carries over to crease surfaces in 3D by projecting the gradient vector to the eigenplane of the repeated eigenvalue (the part outside the eigenplane is orthogonal to the relevant eigenvector anyway) and observing that generic 3D degenerate points behave just like 2D degeneracies within that plane (cf. Section 2.3.4). This also clarifies that in general, crease surfaces do not branch, since this would require degeneracies with index $\pm(\frac{3}{2} + n)$, $n \in \mathbb{N}_0$, which are not structurally stable in 3D.

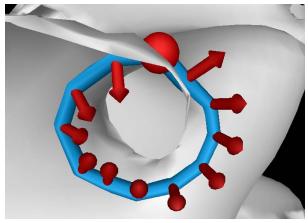
Extracting the skeleton of a bifurcating structure as a crease surface typically does *not* result in a non-manifold sheet. Rather, one part of the surface ends shortly before meeting the other one. In our experience, it is exactly this case in which degenerate lines occur as crease surface boundaries most frequently in practice.



(a) Cutting along the red and blue lines yields orientable pieces.



(b) Example path 1



(c) Example path 2

Figure 4.2.: A non-orientable ridge surface, extracted from an MRI volume dataset (a). Non-orientable paths are shown in detail in (b) and (c).

Note that corresponding results for crease lines in Gaussian scale space have been obtained in the context of medial cores by Damon [51], in a work which has not found adequate attention in the visualization community: Among other things, Damon proved that degenerate loci of symmetric 3×3 matrices form stable lines in 3D (cf. Propositions 8.1 and 9.1 in his work), which was later rediscovered by Zheng et al. [254].

4.2.2. Non-orientability of Creases

The fact that $\mathbf{g} \cdot \mathbf{e}_i = 0$ defines a surface with boundary even before considering any further constraints introduces the possibility that crease surfaces may not be orientable, i.e., it may not in general be possible to assign a normal vector field with consistent sign to a crease surface. This problem has been encountered by previous authors [115, 178], but so far, it has not been discussed whether it is a true property of creases or merely a numerical artifact of existing extraction techniques. Also, examples of non-orientable creases have not been published so far.

Peikert and Sadlo [157] propose to extract crease surfaces as subsets of the zero isosurface of a scalar measure

$$d = \det(\mathbf{g}|\mathbf{Hg}|\mathbf{HHg}) \tag{4.1}$$

which first appears in a work by Süßmuth and Greiner [209]. The fact that creases can be expressed as a filtered isosurface suggests that they are orientable. However, the scalar field d changes sign in an ϵ-band around the crease not only in normal direction, but also along the surface. More precisely, $d = 0$ not only when $\mathbf{g}$ is orthogonal to the selected eigenvector ($\mathbf{e}_3$ for ridges, $\mathbf{e}_1$ for valleys), but to an arbitrary eigenvector. Along parallel vector lines $\mathbf{g} \parallel \mathbf{e}_i$, $\mathbf{g}$ is orthogonal to *both* remaining eigenvectors $\mathbf{e}_j$ ($j \neq i$), so in these places, the zero isosurface of d self-intersects – the sign of d changes along the crease.

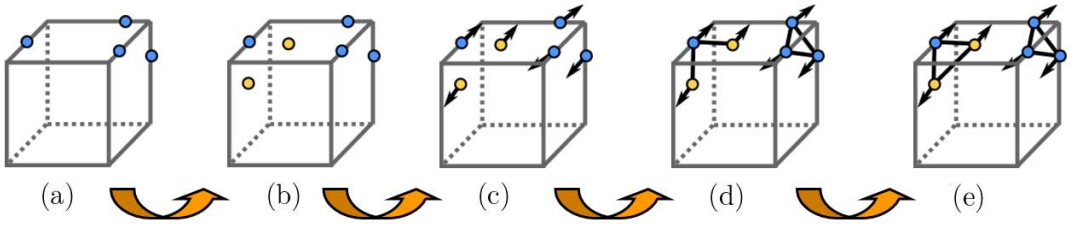


Figure 4.3.: Our algorithm creates the mesh per-cell, by finding intersection points (a)/(b), estimating normals (c), and connecting the points based on them (d)/(e).

Figure 4.2 presents a ridge surface from a real-world MRI dataset. In two places, marked by arrows, it shows surface pieces which are homeomorphic to the Möbius strip. This establishes the fact that non-orientability is, in fact, a true property of creases. To give a better visual impression of the non-orientability, Subfigures (b) and (c) show closed paths (light blue) along which the normal (red) cannot be oriented – at the points marked by a red ball, a contradiction occurs. In (a), the parallel vector lines $\mathbf{g} \parallel \mathbf{e}_1$ are shown in red, $\mathbf{g} \parallel \mathbf{e}_2$ in blue, type P degeneracies are shown as yellow spheres. This illustrates that cutting the crease along these lines would result in pieces that could be oriented using the scalar d from Equation (4.1).

4.3. Extraction of Crease Surfaces

4.3.1. Basic Idea

From the observations made in the previous section, it follows that marching cubes is not suitable for the extraction of crease surfaces. Since isosurfaces are closed, it only considers cases in which the boundary of each cell face is intersected an even number of times. If there really is an odd number of intersections (because the crease ends inside the cell), applying the marching cubes case table will either add spurious triangles or create a hole. In existing algorithms as used in [114] and [178], either of these options happens at random.

Peikert and Sadlo [157] have proposed to solve this problem by using marching cubes to extract a superset of creases and filtering out irrelevant parts afterwards. This is theoretically appealing, but unfortunately, it is infeasible in practice, since the marching cubes algorithm cannot handle the self-intersections which occur in the zero isosurface of their scalar field d. In fact, it follows from the non-orientability of crease surfaces that it is generally not possible to close them in $\mathbb{R}^3$ without introducing self-intersections.

The algorithm we propose instead is also based on cell marching, but does not rely on the marching cubes case table to determine topology. Figure 4.3 gives an overview of our pipeline: We first extract individual intersections of cell edges with the crease surface (a) and of cell faces with the degenerate lines which bound the crease (b). We then estimate surface normals at these points (c) and use them to select the most likely topology, both

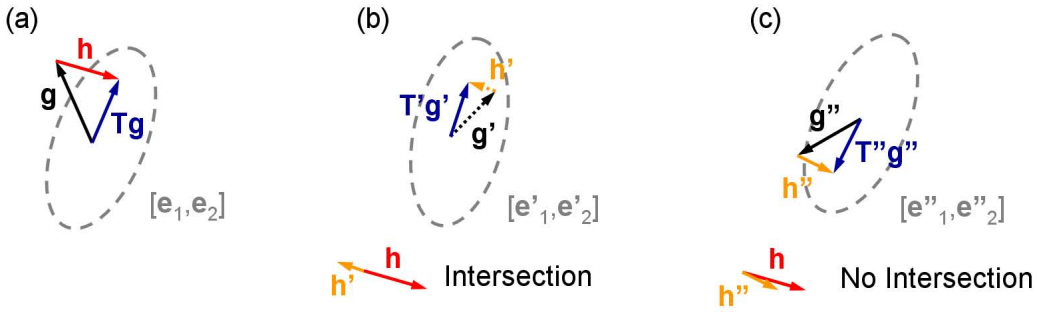


Figure 4.4.: An intersection of $\mathbf{g}$ with the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$ is detected by considering the difference vector $\mathbf{h}$ that results from projecting $\mathbf{g}$ to the plane.

on the faces (d) and within the cell (e). Taken together, this leads to closed polygons over the boundary of each cell, which can be triangulated to form the final mesh.

Without showing results, Eberly [63] has proposed a similar strategy to extract 2D creases from 3D fields. However, he assumes that the intersections of crease surfaces with cell faces can be described as the zero contour of a bilinear function. Like marching cubes, this does not allow creases to terminate within a cell.

To keep the notation simple, we will restrict our discussion to ridge surfaces. Valleys are obtained by straightforward analogies or by extracting ridges of $-f$.

4.3.2. Finding Edge Intersections

Our algorithm makes extensive use of a differentiable symmetric tensor field $\mathbf{T}(\mathbf{x})$ which is derived from the Hessian field $\mathbf{H}(\mathbf{x})$ and has $\mathbf{g}(\mathbf{x})$ as an eigenvector to eigenvalue 1 if and only if $\mathbf{x}$ is a point on the ridge. Its eigenvectors $\mathbf{e}'_i$ and eigenvalues λ'_i are defined from those of the Hessian matrix ($\mathbf{e}_i$ and λ_i) as:

$$\mathbf{e}'_i := \mathbf{e}_i \tag{4.2}$$

$$\lambda'_1 := 1 \quad \lambda'_2 := 1 \quad \lambda'_3 := \begin{cases} 0 & \text{if } \lambda_2 - \lambda_3 > \theta \\ \left(1 - \frac{\lambda_2 - \lambda_3}{\theta}\right)^2 & \text{else} \end{cases} \tag{4.3}$$

The definition of λ'_3 makes sure that $\mathbf{T}$ is a differentiable function of $\mathbf{H}$ and that it remains well-defined as $(\lambda_2 - \lambda_3) \to 0$ and $\mathbf{e}_3$ becomes ill-conditioned. We assume that this starts to play a role when $(\lambda_2 - \lambda_3)$ drops below a threshold θ, which we fixed empirically at 0.5% of the dynamic range in our data. As a result of this formulation, degenerate loci are counted as being on the ridge regardless of $\mathbf{g}$. This is reasonable, since we have shown in Section 4.2.1 that crease surfaces are bounded by these lines.

To detect intersections of the ridge with cell edges, we consider the vector

$$\mathbf{h}(\mathbf{x}) := \mathbf{T}(\mathbf{x})\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{x}) \tag{4.4}$$

which is zero if and only if $\mathbf{x}$ is a point on the ridge. Otherwise, $\mathbf{h}$ indicates the direction in which the gradient $\mathbf{g}$ moves when being projected onto the eigenplane of $\mathbf{T}$ (cf. Figure 4.4).

Let $\mathbf{h}_1$ and $\mathbf{h}_2$ be the respective vectors at the endpoints of an edge. We assume that the ridge intersects the edge if $\mathbf{h}_1 \cdot \mathbf{h}_2 < 0$, i.e., if the gradient has changed from one side of the plane to the other. The relative magnitudes of $\mathbf{h}_1$ and $\mathbf{h}_2$ provide an estimate of the point of intersection.

It may appear even easier to locate edge intersections by bracketing zero crossings in the scalar d from Equation (4.1). Unfortunately, d often has very close pairs of zero crossings, of which only one indicates a ridge and which are difficult to find in practice. This happens near parallel vector lines $\mathbf{g} \parallel \mathbf{e}_i$, for the reasons mentioned in Section 4.2.2.

4.3.3. Extracting the Boundary

To find the endpoints of the ridge on the cell faces, we localize type L degeneracies via the gradient descent by Zheng et al. [254]. We have augmented it with an Armijo stepsize selection [7] to improve its convergence properties and we repeat it from different starting positions on the face in cases where it runs into local minima. To save computations, this is only done on faces whose boundary is intersected an odd number of times.

Each ridge that enters a face should either leave it again or end in a type L degeneracy. It is important to ensure this algorithmically to achieve a consistent final triangulation. If no degeneracy is found, this typically means that we have missed an edge intersection. In fact, edges along which $\mathbf{T}$ varies strongly may be intersected multiple times. To handle this, we bisect an edge if the values $\mathbf{T}_1$ and $\mathbf{T}_2$ at its endpoints differ too much. The exact condition used in our current implementation is

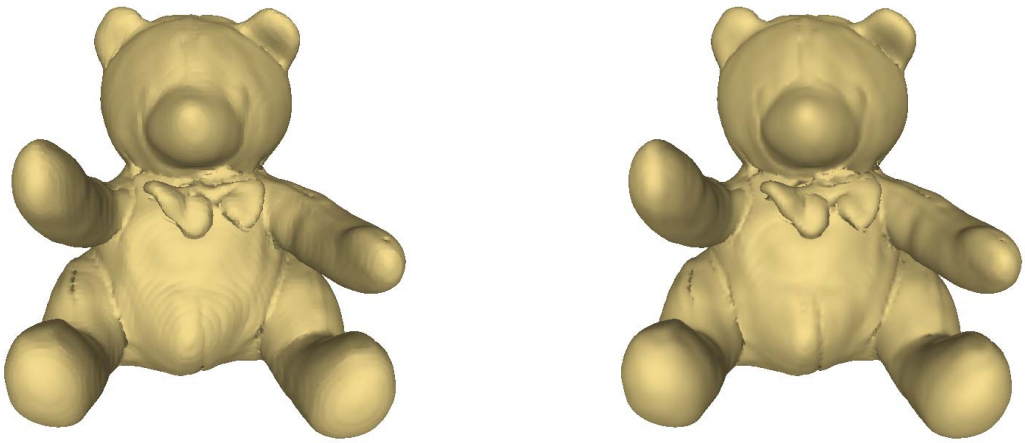
$$\frac{\operatorname{tr}(\mathbf{T}_1^T \mathbf{T}_2)}{\sqrt{\operatorname{tr}(\mathbf{T}_1^T \mathbf{T}_1)} \cdot \sqrt{\operatorname{tr}(\mathbf{T}_2^T \mathbf{T}_2)}} < \Theta$$

The threshold Θ is increased iteratively while the total number of intersections is odd and no degeneracy has been found.

4.3.4. Estimating Normals

The fact that $\mathbf{h}(\mathbf{x}) = \mathbf{0}$ for all points $\mathbf{x}$ on the surface allows us to compute the surface normal at $\mathbf{x}$. Directional derivatives of $\mathbf{h}$ tangential to the surface are $\mathbf{0}$, so the Jacobian $\nabla\mathbf{h}$ has rank one, with the only non-zero eigenvector in normal direction. The fact that the normal computed this way is only defined up to sign is not a limitation, since the ridge surface is non-orientable anyway.

Since $\nabla\mathbf{h} = \nabla\mathbf{T}\mathbf{g} + \mathbf{T}\nabla\mathbf{g} - \nabla\mathbf{g}$ involves the gradient of $\mathbf{T}$, which is in turn defined in terms of the Hessian, normal estimation assumes that f is at least C^3 continuous. Despite dealing with third derivatives and applying some computational simplifications (detailed in Section 4.3.6), we found that the normals obtained this way are of reasonable quality and can be used both for estimating local topology and for rendering (cf. Figure 4.5).



(a) Normals estimated from discrete mesh [210]

(b) Normals computed from scalar field f

Figure 4.5.: Despite the use of third derivatives, our analytic normals appear smooth (b). For comparison, Subfigure (a) shows normals estimated from the mesh.

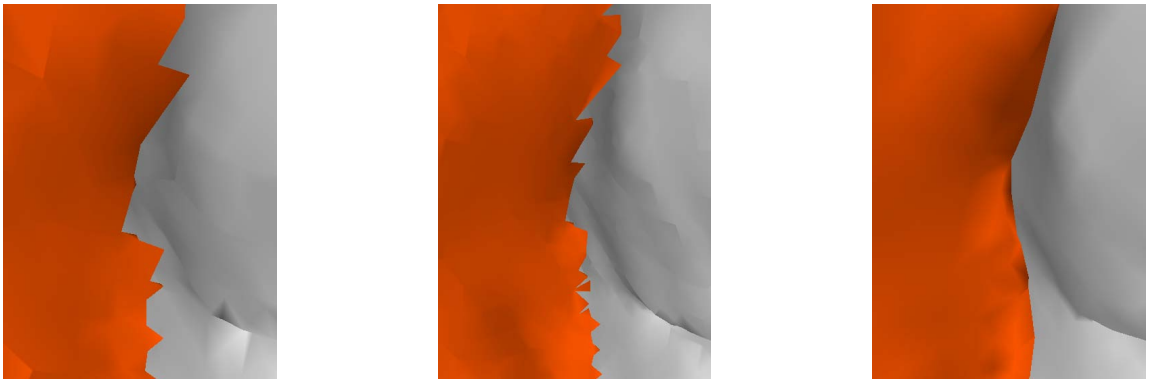
4.3.5. Generating the Mesh

Our theoretical analysis in Section 4.2 revealed two facts: First, crease surfaces end at degenerate lines; second, they may not be orientable. To reflect these insights, per-cell processing has to concentrate on finding the lines that bound the surface. Orientability, on the other hand, is a global property, and will result automatically if we make the correct per-cell decisions.

Consistency of the final mesh is guaranteed by estimating the connectivity per-face, and sharing the results between adjacent cells. Moreover, the boundary points are connected pairwise per-cell and thus form continuous lines in the final mesh. As an example, Figure 4.6 illustrates the case discussed in Section 4.2.1, in which one part of a crease (red) terminates just before it would meet another one (gray) in a non-manifold configuration. Since marching cubes does not allow the surface to terminate within a cell, it produces a zig-zag edge (a), even when using a finer resolution (b). On the other hand, our method extracts a smooth degenerate line which bounds the surface.

To estimate per-face connectivity, we connect the extracted intersection points pairwise. Since the total number of points per face is low, we simply enumerate all possible pairings and exclude the ones that would lead to a self-intersection (2D line-line intersection test). Among the permissible options, we choose the one which agrees best with the computed normals, i.e., the one which minimizes the sum of absolute dot products of connection lines and normals at their endpoints.

On each face, we extracted a degeneracy if and only if its boundary was intersected an odd number of times. Since edges are shared between adjacent faces, this leads to an even number of degenerate points per cell, which are connected in a similar manner. In rare cases, a cell has more than two degenerate points. In that case, we use the quads defined



(a) MC with PCA (b) as (a), 2× res. (c) Proposed method

Figure 4.6.: Unlike marching cubes (a) and (b), our method creates a smooth representation of crease surfaces that terminate at degenerate lines (c).

by any pair of degenerate points and their respective neighbors on the face to check for self-intersections (3D triangle-triangle intersection tests after arbitrary subdivision).

After these steps, each cell contains a set of closed polygons (cf. Figure 4.3 (e)). Triangles are used as is, quads are subdivided arbitrarily. We triangulate larger polygons via a triangle fan with an additional vertex at the barycenter.

4.3.6. Implementation

Trilinear interpolation is widely used for its computational efficiency. Since creases require C^2 continuity, more advanced interpolation becomes obligatory. Like Kindlmann et al. [115], we convolve the given sample points with a C^2 cubic B-spline kernel. However, we store the resulting values, gradients, and Hessians at each grid point and interpolate them trilinearly in between. A very similar approximation is made when using the Phong shading model [162], which interpolates surface position and normal independently.

We found that this approximation greatly speeds up the bisection of edges and iterative search for degenerate points on faces, while the resulting changes to the crease are on the order of a small additive Gaussian perturbation of f. Even approximating third derivatives by taking finite differences in the trilinearly interpolated Hessian field did not introduce any notable artifacts in the resulting normals. Note that this choice is an implementation detail which could be changed without having to alter any part of the algorithm.

Our extraction algorithm only produces exact boundaries where the ridge ends at a degenerate line. The side constraint ($\lambda_3 < 0$) is taken into account by excluding cells for which no vertex meets the constraint. This causes zig-zag boundaries, which are straightened by triangle trimming in a postprocess. This choice was motivated by the fact that crease surfaces are typically filtered using application-specific rules anyway, so both tasks are easily combined. Moreover, it avoids complex special cases that would otherwise occur in the extraction when the two types of boundaries meet.

4.3.7. Rendering

On modern graphics hardware, it is straightforward to render non-oriented surfaces, simply by discarding the sign of the normal in the lighting computation. As an example, let $\mathbf{n}$ denote the surface normal, $\mathbf{l}$ the vector towards the light source. The diffuse term in the Phong shading model [162] usually involves $\max\{\mathbf{n}\cdot\mathbf{l}, 0\}$. To render non-oriented surfaces, we simply replace this expression with $|\mathbf{n}\cdot\mathbf{l}|$ in a vertex shader program [176].

4.4. Results

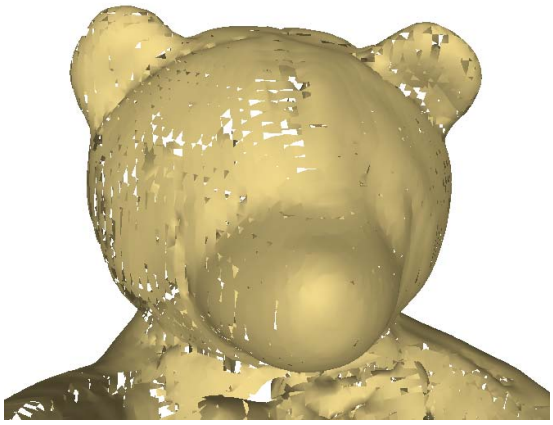
4.4.1. Setup and Qualitative Results

To validate our method, we extracted boundary ridges in a volume dataset from a CT scan of a teddy bear. We chose this dataset because the bear is composed from different materials, which makes it difficult to extract using simple isosurfacing. We resampled the dataset to $118 \times 118 \times 105$ cells with isotropic edge length $l = 3\,\text{mm}$. To detect the boundaries, we computed the gradient magnitude by convolution with directional-derivative-of-Gaussian kernels at $\sigma = 3.3\,\text{mm}$. From the resulting dataset, we then extracted height ridges at the data grid resolution using the proposed method, and compared them to results of marching cubes, using both eigenvector tracking (as in [115]) and principal component analysis (PCA, as in [178]) as a preprocess.

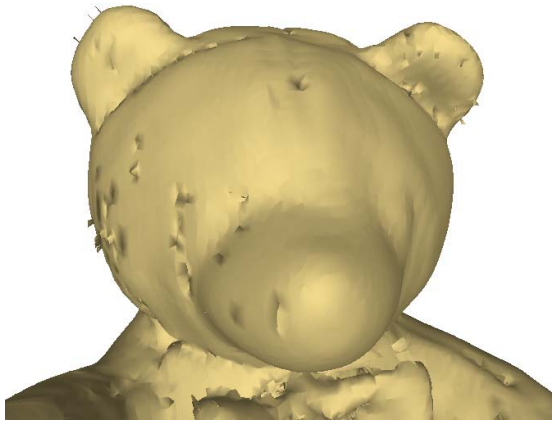
All previous authors have found it necessary to filter out noise-related parts of creases. Like Haralick [87], we used a threshold on the ratio $\|\mathbf{g}\|/\lambda_3$ to restrict the ridge to its most salient part. Moreover, we put a threshold on absolute value and performed connected component analysis to remove a background object present in the dataset. As shown in Figure 4.7, the visual impression of our result (c) is clearly better than the ones from marching cubes at the same resolution.

Eigenvector tracking cannot process cells in which any edge is near a type L Hessian degeneracy. These skipped cells lead to the large number of small holes in (a). PCA processes all cells, but fails to find a consistent orientation in the presence of Hessian degeneracies or large eigenvector variations. This leads to the spikes and holes in (b). The degeneracies which cause these problems run through the same cells as the affected parts of the bear, but belong to surfaces that end in the vicinity of the bear and are filtered out during post-processing. Since the effect of Hessian degeneracies is not controlled in previous methods, they can affect any surface that intersects the cell. This problem can be mitigated by refining the extraction grid, which makes it less likely that a degeneracy of a "noise" ridge runs through the same cell as a legitimate ridge we would like to keep.

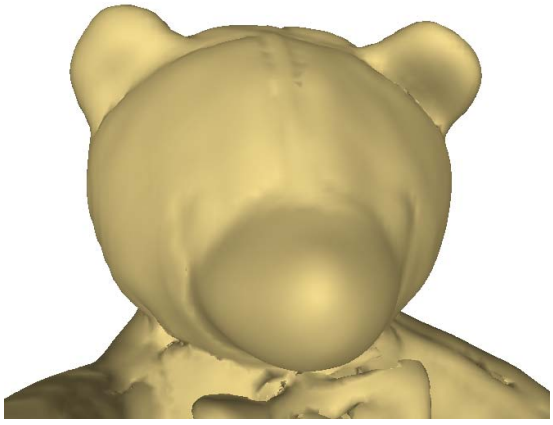
Previous authors have exploited this: Sadlo and Peikert [178] propose an adaptive refinement around the crease, and Kindlmann et al. [115] globally use a grid which is by factor 5 finer than the data grid. Indeed, at twice the original resolution, marching cubes with eigenvector tracking produces a result which looks comparable to ours (d). However, adaptive refinement cannot avoid ragged crease boundaries (cf. Figure 4.6 and Table 4.3), leads to overtesselation and comes at considerable computational expense (cf. Table 4.1).



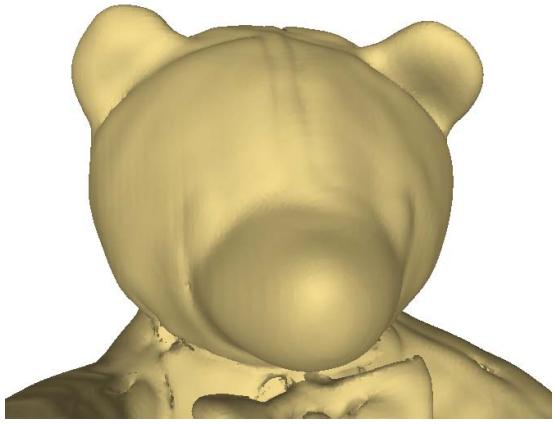
(a) Marching cubes with eigenvector tracking



(b) Marching cubes with PCA



(c) Proposed method



(d) Eigenvector tracking at double resolution

Figure 4.7.: At the original data resolution, the proposed algorithm for crease extraction (c) provides much better results than marching cubes, (a) and (b).

Method	Time (s)	# triangles
evec tracking	24+1	66,662
evec tracking (2× res.)	180+4	578,489
PCA	18+2	201,068
proposed method	38+3	246,040

Table 4.1.: Timings and triangle counts, including filtering.

Method	area	error (mm)
evec tracking	$0.21\,\mathrm{m}^2$	0.02 *(0.14)*
PCA	$0.49\,\mathrm{m}^2$	0.07 *(0.31)*
proposed method	$0.47\,\mathrm{m}^2$	0.02 *(0.13)*

Table 4.2.: Mean absolute and RMS (in italics) error in face position, as measured by a gradient descent.

4.4.2. Quantitative Results

Table 4.1 presents algorithm performance, in terms of consumed wall time (on a 2 GHz laptop) and generated geometry. It shows that the improved accuracy of our algorithm comes at moderate additional computational expense. In particular, it is more than four times faster than marching cubes on the refined grid, the only alternative that provides acceptable quality.

Moreover, we conducted two quantitative experiments which support the observations from the previous subsection. First, we evaluated the accuracy of the extracted surfaces by taking a large number of samples from the mesh ($1.5\,\mathrm{mm}^{-2}$, uniformly at random), and measuring the distance to the nearest point on the crease, as found by a gradient descent in the direction which minimizes the squared norm $\|\mathbf{h}\|^2$ of $\mathbf{h}$ from Equation (4.4). This gradient descent is only used for evaluation, not during crease extraction. Table 4.2 lists the resulting average absolute and root mean square (RMS, in italics) distances. It clearly shows the increased error of the PCA result, which is due to erroneous triangles. The table also lists the total area of the bear, illustrating the fact that eigenvector tracking only reconstructs part of the surface.

In a second experiment, we considered the boundary components of the meshes and created a histogram of their length in terms of the number of individual edges. The results in Table 4.3 confirm that marching cubes produces small holes in the surface. In particular, eigenvector tracking at both resolutions misses a large number of single triangles, due to skipped cells. Vertices in which more than two boundary edges meet are an indicator of spurious holes. In eigenvector tracking at the original resolution, more than 7% of all boundary vertices are affected. In marching cubes with PCA, it is slightly less than 1%; in our proposed method, such configurations do not occur by design. Note that many of the longer boundary components are a consequence of the fact that the crease also represents the stuffing of the bear.

Method	Bdy components of length				
	3	**4**	**5**	**6**	**> 6**
evec tracking	2158	59	408	69	353
evec tracking (2× res.)	2277	97	469	80	533
PCA	44	31	20	67	923
proposed method	0	10	0	0	136

Table 4.3.: Marching cubes produces a large number of spurious short boundary components on crease surfaces.

For further validation, it would have been ideal to implicitly represent a known surface as a crease and to compare the mesh extracted from the resulting scalar field to the initial ground truth. Unfortunately, it is not obvious how to transform a surface to a well-defined height crease, and to the best of our our knowledge, this topic has not been addressed in the literature.

4.5. Conclusion and Future Work

Crease lines have a long tradition in image processing and computer vision. Crease surfaces, their two-dimensional generalization, can offer a versatile tool for visualization, both to capture boundaries which cannot be characterized as isosurfaces, and to extract object cores or skeletal structures. However, using crease surfaces widely requires a full understanding of their properties and reliable numerical methods for their extraction.

This chapter has promoted this goal by clarifying the topological properties of crease surfaces and proposing a novel algorithm for their extraction. The transformed Hessian approach in Section 4.3.2 provides a unified framework for the detection of crease surface intersections, estimation of surface normals, and for a gradient descent to the crease surface, without the need to orient eigenvectors. The results of our algorithm have been shown to be more accurate than the ones of existing methods.

To fully harness the potential of creases, future work should investigate their scale space behavior. This will require the extraction of surfaces from a four-dimensional space. However, understanding crease surfaces in 3D is a necessary step towards that more complex goal. Moreover, creases at a single scale have proven sufficient to gain insights both in prior work [114, 178], and in the following chapter of this dissertation.

5. Planarity Ridges for DT-MRI Visualization

A standard way to visualize DT-MRI data is to integrate streamlines which are everywhere tangential to the principal eigenvector of the tensor field, and are interpreted as estimated fiber pathways (cf. Section 2.2.5). However, this method is inappropriate for regions where fiber tracts cross or fan out, since the diffusion tensor becomes planar, i.e., its larger two eigenvalues are similar in magnitude, and there is no single preferred direction.

To transfer the idea of streamlines to such areas, Zhang et al. [251] proposed to integrate streamsurfaces, which are everywhere tangential to the plane spanned by the major and medium eigenvectors. In our work [193], we show that the surfaces generated by their algorithm are ill-defined and do not have a clear interpretation: The method makes arbitrary choices which have a strong influence on the final result. This is explained by the fact that in typical DT-MRI data, surfaces that are everywhere tangential to the major and medium eigenvectors do not exist.

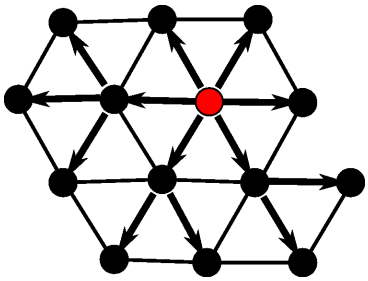
In a second step, we combine the basic idea of anisotropy creases [114] with results on crease surface extraction and eigenvalue derivatives from Chapters 4 and 6 of this dissertation, to provide a viable alternative to streamsurfaces. Crease surfaces in planarity are well-defined, and they fulfill the goal for which streamsurfaces were developed: To illustrate the major regions of planar diffusion.

This chapter is organized as follows: After a more detailed review of related work (Section 5.1), we demonstrate that DT-MRI streamsurfaces are ill-defined (Section 5.2). Then, we introduce planarity ridges (Section 5.3) and evaluate them (Section 5.4), before we conclude the chapter (Section 5.5).

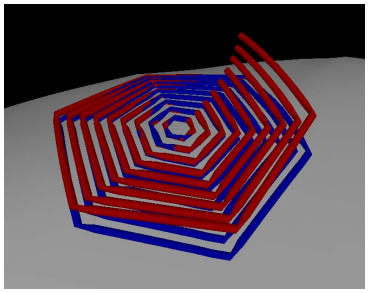
5.1. Related Work

In their work on DT-MRI streamsurfaces, Zhang et al. [251] included the caveat that their definition relies on the assumption that the Lie bracket of the involved eigenvector fields lies within their common plane. They considered it overly complex to verify that assumption, but stated that it would likely be fulfilled, since they did not experience problems in practice. Despite the fact that this integrability condition has never been checked, streamsurfaces are frequently mentioned as a standard tool for DT-MRI visualization [252, 224, 167].

In vector field visualization, streamsurfaces are defined as surfaces which are traced out by advecting a given seed line along the field. This only involves integration along a single vector field, and does not entail the problems that arise from trying to be everywhere tangential to two vector fields at the same time. Streamsurfaces in the well-defined sense of flow visualization have been extracted from tensor fields by Jeremić et al. [102]. Sondershaus and Gumhold [204] call the surfaces from [251] "diffusion surfaces", to avoid



(a) Vertices are added radially from the seed, breadth first.



(b) Integration along n-rings leaves the surface.

Figure 5.1.: Streamsurface extraction relies on the order in which vertices are added (a). If the surface were well-defined, the red and blue lines in (b) would coincide.

confusion of these different definitions. In this chapter, we stick to the more widespread name "streamsurface", but prefix it with "DT-MRI" to emphasize the difference.

Sondershaus and Gumhold [204] argue that since the minor eigenvector is well-defined in the regions in which DT-MRI streamsurfaces are integrated, the surfaces should be manifolds with boundary. In fact, DT-MRI streamsurfaces try to be everywhere perpendicular to the minor eigenvector field. However, vector fields only define an everywhere orthogonal surface if their derivatives obey a specific symmetry, which corresponds to the symmetry of the second fundamental form [77], and is not fulfilled by general eigenvector fields.

The scheme for DT-MRI streamsurface extraction presented in [204] differs from the one in [251] in that it is based on the operations of face-based surface compression [82], which makes it suitable to extract two-manifolds with arbitrary topology. Vilanova et al. [223] present a different method that allows DT-MRI streamsurfaces to have boundary components, and they combine streamlines with streamsurfaces by creating additional seed points when entering or leaving a region of planarity. However, both works employ the original integration rule of [251], which we will show to produce non-consistent results, depending on the order in which vertices are added.

5.2. DT-MRI Streamsurfaces are Ill-Defined

The original algorithm for DT-MRI streamsurface extraction grows a mesh of equilateral triangles from a given seed point. Zhang et al. [251] perform the integration along the edges which are marked by arrows in Figure 5.1 (a), but this choice is arbitrary. Their integrability condition in terms of the Lie bracket has an alternative formulation which is much easier to check in practice: If the surface resulting from their algorithm is well-defined, finding the position of a vertex by integrating along any other path in the mesh should produce the same result. In particular, integration along cycles should return to the initial position.

To test this, we integrated cycles along the n-rings $n \in \{1, \ldots, 10\}$ around the seed

point. Integration started at a vertex of the previously extracted DT-MRI streamsurface and the first step was made in direction of its neighbor on the n-ring in counter-clockwise direction. Further integration was carried out in the plane spanned by the minor eigenvector and the "incoming" vector of the previous step, using the exact rule from [251]. After each n-th step, we turned the incoming vector 60° to the left within the current tangent plane.

If the rule for surface integration were well-defined, the resulting trajectory should coincide with the corresponding n-ring on the surface. Figure 5.1 (b) shows that this condition is violated: On the presented streamsurface (gray), the n-rings are shown in blue. Our trajectories, which clearly depart from the surface, are red. Since we made conservative choices for stepsize (one fifth of a cell edge length) and numerical integration scheme (fifth-order Runge-Kutta at 64-bit floating point precision), such strong differences in such a small neighborhood cannot be explained by numerical errors. Also, integration was limited to a domain where the second eigenvalue was much larger than the third, so degeneracies have not played a role.

The algorithm in [251] expands the surface breadth-first, so adjacent vertices are integrated along similar paths. In our experience, the algorithm becomes unstable when this order is changed to depth-first, which should not be the case if the surface were well-defined. Moreover, when allowing for holes in the surface (as in [223]), highly deformed triangles occur when a boundary component is closed and vertices whose integration paths had departed for some time become adjacent again.

It appears possible to address these algorithmic problems by adopting methods from computer vision: In shape from shading, an estimated normal field is used to infer surface geometry. In this context, there exist various strategies to deal with "nonintegrable" vector fields (cf. [41] and references therein). However, this does not change the fact that surfaces which are everywhere aligned with the two principal directions of diffusion generally do not exist. Because of this, we feel that, in the context of our application, the meaning of surfaces that are "as well-aligned as possible" with these directions would be unclear.

5.3. Planarity Ridges for DT-MRI Visualization

Our proposed substitute for DT-MRI streamsurfaces extends the idea of anisotropy creases: Kindlmann et al. [114] have extracted crease surfaces of fractional anisotropy (FA), a scalar measure which quantifies the overall directional dependence of diffusion (cf. Equation (2.24) and Section 2.3.6). Anisotropy creases have been shown to represent the skeleton of white matter structures, and they have proven to produce repeatable results over a range of subjects [115]. To obtain surfaces that represent regions of planarity, we replace FA with c_p, a specific measure of planarity (cf. Equation (2.26)).

5.3.1. Partial Derivatives of c_p

Extracting creases of c_p requires formulas for the first and second partial derivatives of c_p with respect to the tensor field. Without loss of generality, we only consider $c_{p,x}$ and $c_{p,xy}$.

According to the quotient rule, they are given as

$$c_{p,x} = \frac{\partial c_p}{\partial x} = \frac{A}{B}$$
$$c_{p,xy} = \frac{\partial^2 c_p}{\partial x \partial y} = \frac{A_y B - A B_y}{B^2}$$

with

$$\begin{aligned}
A &= 2\lambda_1 (\lambda_{2,x} - \lambda_{3,x}) + 2\lambda_2 (-\lambda_{1,x} - 2\lambda_{3,x}) + 2\lambda_3 (\lambda_{1,x} + 2\lambda_{2,x}) \\
A_y &= 2\lambda_1 (\lambda_{2,xy} - \lambda_{3,xy}) + 2\lambda_2 (-\lambda_{1,xy} - 2\lambda_{3,xy}) + 2\lambda_3 (\lambda_{1,xy} + 2\lambda_{2,xy}) + \\
&\quad 2\lambda_{1,y} (\lambda_{2,x} - \lambda_{3,x}) + 2\lambda_{2,y} (-\lambda_{1,x} - 2\lambda_{3,x}) + 2\lambda_{3,y} (\lambda_{1,x} + 2\lambda_{2,x}) \\
B &= (\lambda_1 + \lambda_2 + \lambda_3)^2 \\
B_y &= 2 (\lambda_1 + \lambda_2 + \lambda_3) (\lambda_{1,y} + \lambda_{2,y} + \lambda_{3,y})
\end{aligned}$$

First partial eigenvalue derivatives $\lambda_{i,x}$ are found by rotating the corresponding tensor derivative $\mathbf{D}_x$ to the eigenframe of the original tensor $\mathbf{D}$. Second partial eigenvalue derivatives $\lambda_{i,xy}$ are given by rotating the second partial tensor derivative $\mathbf{D}_{xy}$ to the same frame, but additionally require a correction based on first *eigenvector* derivatives $\mathbf{e}_{i,x}$. Let $\mathbf{I}$ denote the identity matrix and $\mathbf{T}^+$ the Moore-Penrose inverse of $\mathbf{T}$. Then [134]:

$$\begin{aligned}
\lambda_{i,x} &= \mathbf{e}_i^T \mathbf{D}_x \mathbf{e}_i \\
\mathbf{e}_{i,x} &= (\lambda_i \mathbf{I} - \mathbf{D})^+ \mathbf{D}_x \mathbf{e}_i \\
\lambda_{i,xy} &= \mathbf{e}_i^T \mathbf{D}_{xy} \mathbf{e}_i + \mathbf{e}_i^T \mathbf{D}_x \mathbf{e}_{i,y} + \mathbf{e}_i^T \mathbf{D}_y \mathbf{e}_{i,x}
\end{aligned}$$

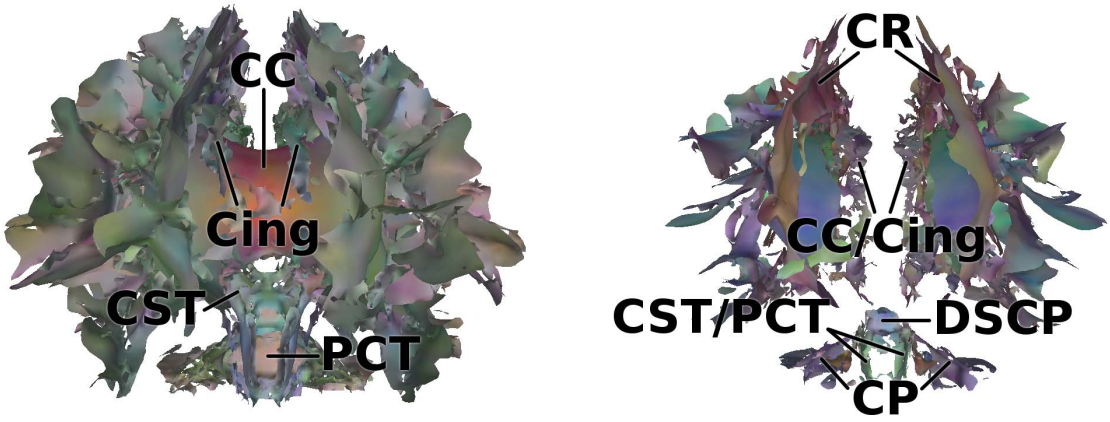
Planarity c_p is defined in terms of sorted eigenvalues, which are not necessarily differentiable at points where two eigenvalues coincide. We address this problem by using the regularized eigenvalue derivatives from Chapter 6.

5.3.2. Extracting Planarity Ridges

We extracted both ridges in fractional anisotropy and in planarity c_p on the dataset from Section 1.2. Gaussian pre-filtering with $\sigma = 1.72\,\text{mm}$ was used and the ridges were filtered to areas with FA > 0.2 / $c_p > 0.2$.

Our algorithm for crease surface extraction from Chapter 4 facilitated the processing of full-brain DT-MRI scans at the original resolution: The anisotropy crease in Figure 5.2 (a) was extracted on the original data grid ($93 \times 116 \times 93$) within 26 s, while Kindlmann et al. [115] report six minutes even after subsampling their data by a factor of two (i.e., to $48 \times 48 \times 28$ voxels), due to the extremely fine extraction grid that they had to use.

Despite the fact that as a non-linear function, FA has higher spatial frequency than the underlying tensor field itself [115], direct visual comparison between creases extracted from the approximation in Section 4.3.6 and ones from exact derivatives did not reveal any notable differences.



(a) Ridge surface in FA, colored by major eigenvector.

(b) Ridge surface in c_p, colored by minor eigenvector.

Figure 5.2.: Unlike ridges in FA (a), ridges in c_p (b) specifically illustrate the cores of planar regions. Therefore, they can replace the ill-defined streamsurfaces.

Figure 5.2 compares the ridges in FA and c_p in a frontal view. Note that different color schemes are used: (a) employs standard RGB-XYZ coloring of the major eigenvector, while (b) color codes the minor eigenvector (e.g., red denotes fanning perpendicular to the x axis), since no principal direction may be defined in planar regions.

As expected, ridges in c_p show the cores of planar regions: They capture the fanning in the *corona radiata* (CR) and the *cerebellar peduncles* (CP), the crossing at the *decussation of the superior cerebellar peduncle* (DSCP) and, due to partial voluming, interfaces between *corpus callosum* and *cingulum* (CC/Cing), as well as between *corticospinal tract* and *pontine crossing tract* (CST/PCT). In comparison, the FA ridge (a) also includes structures with linear diffusion, like the CC, Cing, CST and PCT. Unlike streamsurfaces, planarity ridges cannot be integrated from arbitrary positions, which alleviates issues of seeding and culling. Their parameters are scale (i.e., amount of pre-smoothing) and a threshold for post-filtering.

5.4. Evaluation of Planarity Ridges

For evaluation, we presented our planarity ridges to a neuroscientist. In this process, we found it helpful to add further anatomical context. We added streamlines from fiber tracking [143], which we rendered semi-transparently to mitigate problems with occlusion. Moreover, we seeded superquadric glyphs [110] on the surface.

Figure 5.3 shows an overview of the left hemisphere, seen from a medial cutting plane. The fused rendering with the streamlines allowed our collaborator to confirm our annotations from Figure 5.2 (b) and, in addition, to identify planar regions corresponding to the interfaces between *anterior thalamic projections* and *corticospinal tract* (ATP/CST), as well as between *internal capsule* and *putamen* (IC/Put). Within the *precuneus*, the

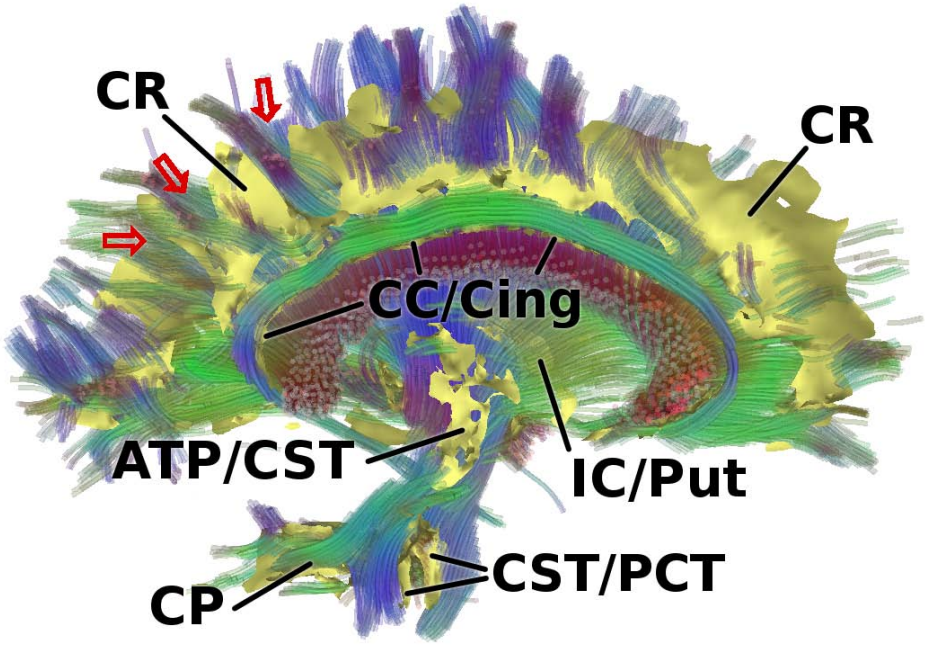


Figure 5.3.: Merging the planarity ridge with semi-transparent streamlines made it easier to recognize the anatomical relevance of its components in this medial view.

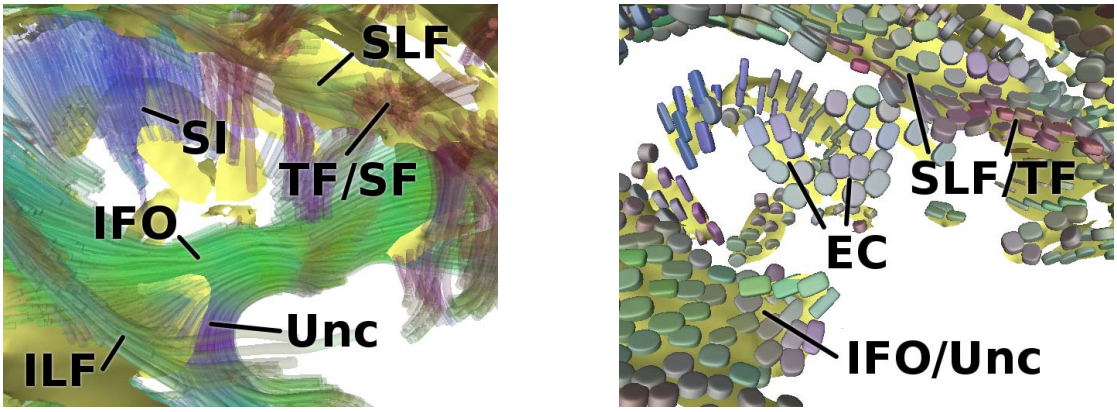
planarity ridge exhibits some characteristic dents (red arrows). In this region, the planarity is due to the fanning of the *corona radiata* (CR) and to its intersection with the SLF III, a component of the *superior longitudinal fasciculus*. It is weaker in places where fiber bundles run, in a more coherent manner, into one of the cortical gyri.

To get a more detailed view on a part of the planarity ridge, Figure 5.4 presents a closeup of the right hemisphere, near the *lateral sulcus*. The annotated tracts in Figure 5.4 (a) are the *superior longitudinal fasciculus* (SLF) which intersects with the *transcallosal fibers* (TF) and the *short association fibers* (SF), the *subinsular white matter* (SI), the *inferior fronto-occipital fasciculus* (IFO) which intermingles with the *uncinate fasciculus* (Unc), as well as the *inferior longitudinal fasciculus* (ILF).

The superquadric glyphs in Figure 5.4 (b) confirm that the planarity ridges in this region capture the intersection of SLF with TF and SF, and the bifurcation of IFO and Unc. Moreover, a planar region exists in the *external/extreme capsule* (EC), where the tracts of the *subinsular white matter* (SI) originate. For our evaluation needs, we found it sufficient to place the glyphs via a simple stratified surface sampling. If desired, a more even distribution could be achieved by implementing glyph packing [117] on surfaces.

5.5. Conclusion

In this chapter, we have demonstrated that DT-MRI streamsurfaces, which have been considered a standard tool for DT-MRI visualization, are mathematically ill-defined, depend



(a) Planarity ridges with annotated fiber tracts

(b) Annotated planarity ridges with glyphs

Figure 5.4.: Beside using streamlines (a), seeding superquadric glyphs on the surface (b) helped to identify planarity ridges in this lateral closeup.

on arbitrary choices in their extraction, and thus should not be used.

As a viable alternative, we have proposed to extract planarity ridges as surfaces that illustrate the regions of planar diffusion. We have derived the required formulas, found appropriate filtering criteria, and combined the results with streamlines and glyphs to allow for evaluation by an expert.

In our experiments, the algorithm from Chapter 4 proved crucial for the extraction of anisotropy creases from full-brain datasets at original resolution.

Part II.

Computer Vision for DW-MRI Visualization

6. Eigenvalue Derivatives for Edge Detection in DT-MRI

In grayscale images, edges are lines across which image intensity changes rapidly, and the magnitude of the image gradient is a common measure of edge strength. In tensor-valued images, edges have a more complex structure: Tensors have several degrees of freedom, which can be classified as invariant under rotation *(shape)* and rotationally variant *(orientation)*. Consequently, there are different *types* of edges, corresponding to changes in the different degrees of freedom.

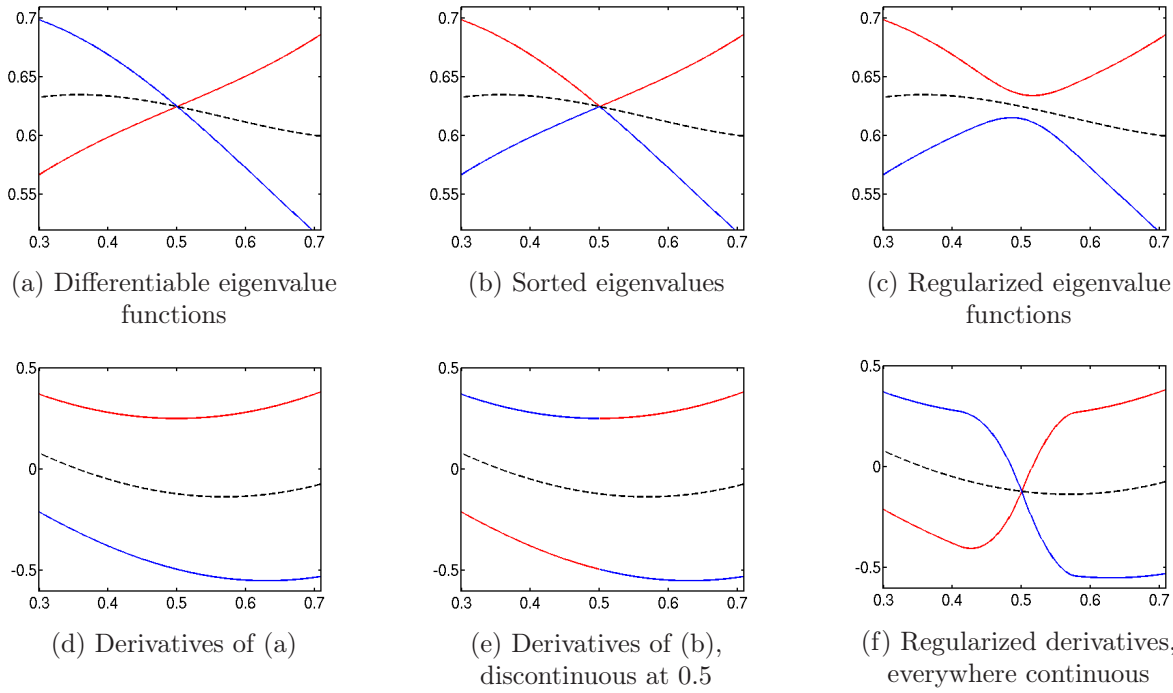
In our work, we are most interested in tensor-valued images from diffusion tensor magnetic resonance imaging (DT-MRI) [13], which is introduced in Section 2.2.3. Edge maps of DT-MRI data have first been created by Pajevic et al. [154]. They distinguish two types of edges by either considering the full tensor information or only its deviatoric (trace-free) part. More recently, Kindlmann et al. [113] have presented a framework based on invariant gradients, which separates six different types of edges, corresponding to all six degrees of freedom present in a symmetric 3×3 tensor. Based on a preliminary description of this approach [111], we demonstrated the practical relevance of differentiating various types of edges in matrix data for segmentation and smoothing in an earlier work [187].

The contribution of our current work [190] is to suggest eigenvalue derivatives as a fundamental tool to discern various types of edges in tensor-valued images. To this end, we summarize some results from perturbation theory [109], which show how to find the derivatives of eigenvalues from tensor derivatives. Since all shape measures in DT-MRI can be defined in terms of eigenvalues (cf. Section 2.2.4), this allows one to map edges with respect to arbitrary shape measures. Moreover, the existing framework based on invariant gradients [113] can be formulated in terms of eigenvalue derivatives, which allows us to simplify and to extend it.

Arsigny et al. [9] have proposed to process the matrix logarithm of diffusion tensors to ensure that the results remain positive definite. Consequently, they use the gradient of the transformed tensor field for edge detection. Section 6.5 shows that eigenvalue derivatives can also be used to analyze various types of edges in this setting, which has not been attempted before.

6.1. Related Work

In the context of DT-MRI, perturbation theory has previously been used by Anderson [4] to study the impact of noise on anisotropy measures and fiber tracking. This work differs from ours not only in scope, but also in methods, since it considers finite deviations from a noise-free tensor and differentiability of eigenvalues is not relevant to his task.



(a) Differentiable eigenvalue functions

(b) Sorted eigenvalues

(c) Regularized eigenvalue functions

(d) Derivatives of (a)

(e) Derivatives of (b), discontinuous at 0.5

(f) Regularized derivatives, everywhere continuous

Figure 6.1.: Sorting eigenvalues may lead to non-differentiable cusps at degeneracies (b). Regularized eigenvalues are sorted, but everywhere differentiable.

O'Donnell et al. [147] have distinguished two different types of edges in DT-MRI data by manipulating the certainties in a normalized convolution approach. Their results resemble the ones obtained from deviatoric tensor fields [154].

Kindlmann et al. [115] extract crease geometry from edges with respect to one specific shape measure (fractional anisotropy), and demonstrate the anatomical relevance of the resulting surfaces. Our work lays the foundation for a refinement of this approach, which is demonstrated in Chapter 5 of this dissertation.

6.2. Regularized Eigenvalue Derivatives

This section will assume a tensor field $\mathbf{D}(t)$ which is differentiable in a single scalar $t \in \mathbb{R}$ and denote its derivative with respect to t by $\mathbf{D}'(t)$. From perturbation theory, it is known that for a differentiable, symmetric tensor field, there exist differentiable eigenvalue functions (cf. Chapter two in [109]). They are given by the diagonal elements of $[\mathbf{D}'(t)]_{\mathbf{E}}$, which is obtained by applying the rotation matrix $\mathbf{E}$ whose rows are the orthonormal eigenvectors of $\mathbf{D}(t)$ to the derivative $\mathbf{D}'(t)$:

$$[\mathbf{D}'(t)]_{\mathbf{E}} = \mathbf{E}\mathbf{D}'(t)\mathbf{E}^{\mathrm{T}} \tag{6.1}$$

However, these differentiable eigenvalue functions do not produce sorted eigenvalues, and ordering them generally introduces non-differentiable cusps at points t at which $\mathbf{D}(t)$

is degenerate. This is illustrated in Figure 6.1: Even though eigenvalues can be described by differentiable functions (red and blue lines in Figures 6.1 (a) and (d)), sorted eigenvalues are non-differentiable where the original functions cross (Figure 6.1 (b) and (e)). We propose to solve this problem by regularizing the derivatives to make them everywhere continuous. This is achieved by smoothly blending the derivatives near degeneracies (Figure 6.1 (f)).

Within a small neighborhood of a degeneracy, the repeated eigenvalue typically splits into different eigenvalues, which constitute its λ-group [109]. Even if a repeated eigenvalue itself is not differentiable, the mean value of its λ-group is (dashed black lines in Figure 6.1). From $[\mathbf{D}']_{\mathbf{E}}$, this mean derivative can be extracted as the average of the diagonal entries that belong to the duplicated eigenvalue. Our regularized derivatives $\tilde{\lambda}'_i$ are defined to preserve this mean derivative ($\tilde{\lambda}'_1 + \tilde{\lambda}'_2 = (\lambda_1 + \lambda_2)'$).

In Figure 6.1 (c), the regularized derivatives $\tilde{\lambda}'_i$ have been integrated numerically to obtain regularized eigenvalue functions $\tilde{\lambda}_i$, which behave mostly like the sorted eigenvalues in Figure 6.1 (b), but are modified locally around degeneracies to be everywhere differentiable. These functions are shown purely for illustration. Our implementation blends the derivatives directly and does not make use of the resulting regularized eigenvalue functions.

In order to decide at which point we should start to blend the derivatives of two eigenvalues λ_i and λ_j, we introduce the measure ρ_{ij} of their relative distance:

$$\rho_{ij} = \frac{|\lambda_i - \lambda_j|}{\lambda_i + \lambda_j} \tag{6.2}$$

Since we aim at diffusion tensors, this definition assumes that $\lambda_i > 0$. Blending starts when ρ drops below a threshold ϵ, which was set to $\epsilon = 0.05$ in our experiments.

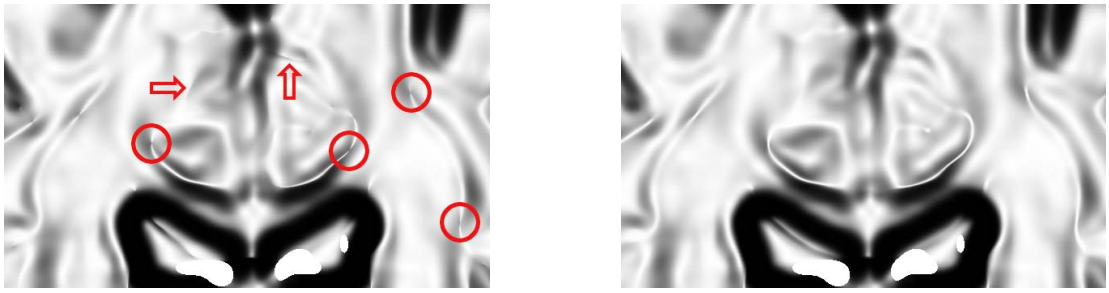
Let $w_{ij} = (\rho_{ij}/\epsilon - 1)^2$. With this, the exact formula we used to define $\tilde{\lambda}'_1$ in terms of the discontinuous derivatives λ'_i is:

$$\tilde{\lambda}'_1 = \begin{cases} \lambda'_1 & \text{if } \rho_{12} \geq \epsilon \\ \left(1 - \frac{1}{2}w_{12}\right)\lambda'_1 + \frac{1}{2}w_{12}\lambda'_2 & \text{if } \rho_{12} < \epsilon \wedge \rho_{23} \geq \epsilon \\ \left(1 - \frac{1}{2}w_{12} - \frac{1}{6}w_{12}w_{23}\right)\lambda'_1 + & \text{if } \rho_{12} < \epsilon \wedge \rho_{23} < \epsilon \\ \quad \left(\frac{1}{2}w_{12} - \frac{1}{6}w_{12}w_{23}\right)\lambda'_2 + \frac{1}{3}w_{12}w_{23}\lambda'_3 & \end{cases} \tag{6.3}$$

$\tilde{\lambda}'_3$ is defined analogously, $\tilde{\lambda}'_2$ is simply obtained as $\tilde{\lambda}'_2 = \lambda'_1 + \lambda'_2 + \lambda'_3 - \tilde{\lambda}'_1 - \tilde{\lambda}'_3$.

Figure 6.2 compares magnitude maps of $\nabla\lambda_1$ and $\nabla\tilde{\lambda}_1$ in a slice of DT-MRI data. The artifacts in Figure 6.2 (a) become more pronounced when considering edge maps of non-linear functions of sorted eigenvalues, like the Westin measures.

An alternative way to blend eigenvalue derivatives is to find a rotation $\bar{\mathbf{E}}$ such that those diagonal entries of $[\mathbf{D}']_{\bar{\mathbf{E}}}$ which correspond to a repeated eigenvalue equal its λ-group mean derivative. This is more complex to implement, but has the advantage of cleanly separating changes in shape (diagonal elements) from changes in orientation (off-diagonal) while preserving the total derivative magnitude, as measured by the rotationally invariant Frobenius norm. The key to this method is to observe that in case of a degeneracy, we are



(a) $\|\nabla\lambda_1\|$ (without regularization) (b) $\|\nabla\tilde{\lambda}_1\|$ (with regularization)

Figure 6.2.: In edge maps, ignoring degeneracies leads to small, but noticeable artifacts (a), which are fixed by the proposed regularization (b).

free to choose any set of mutually orthogonal eigenvectors $\mathbf{e}_i$ which span the eigenspace of the repeated eigenvalue.

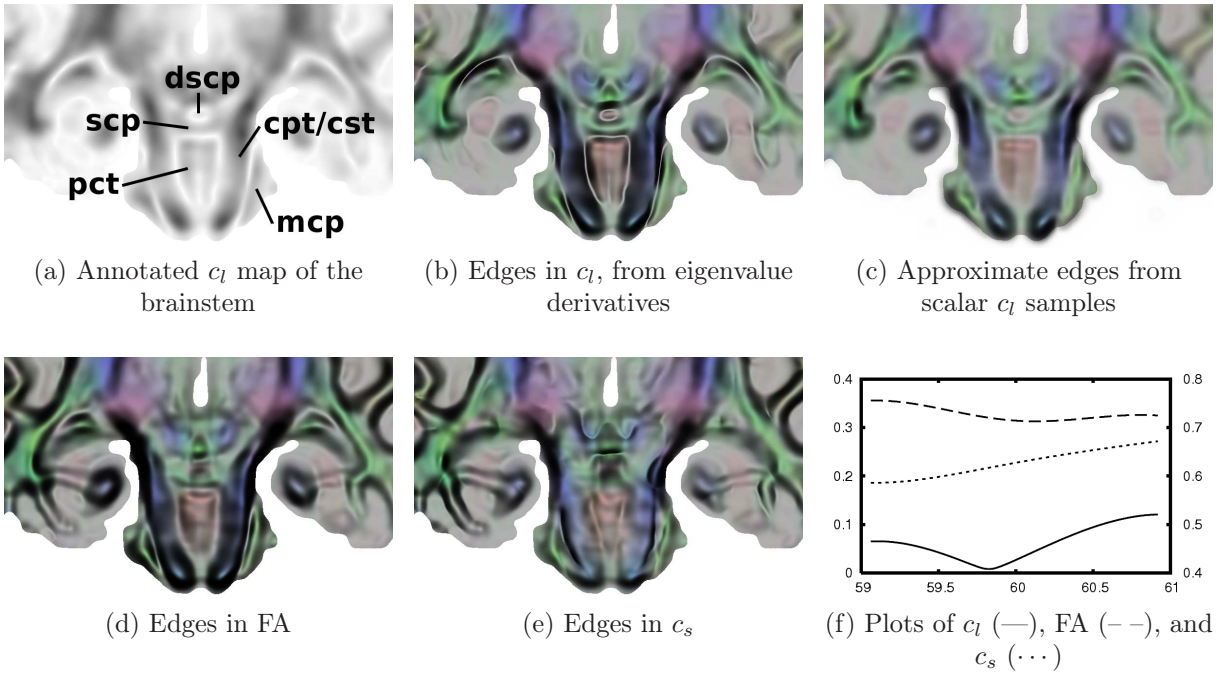
Let $\mathbf{b}_i$ be the vectors of the assumed basis, and let $D'^{(1)}_{j,k}$ be entry (j,k) of the matrix representation $\mathbf{D}'^{(1)} = [\mathbf{D}']_{\mathbf{E}}$. Then, our algorithm works as follows:

1. If $\rho_{1,2} < \epsilon$, create $\mathbf{D}'^{(2)}$ by rotating $\mathbf{D}'^{(1)}$ around $\mathbf{b}_3$ such that $D'^{(2)}_{1,1} = D'^{(2)}_{2,2}$.
2. If $\rho_{2,3} < \epsilon$, create $\mathbf{D}'^{(3)}$ by rotating $\mathbf{D}'^{(2)}$ around $\mathbf{b}_1$ such that $D'^{(3)}_{2,2} = D'^{(3)}_{3,3}$.
3. If $\rho_{1,3} < \epsilon$, identify i such that $D'^{(3)}_{i,i}$ is in between the remaining two diagonal entries, $D'^{(3)}_{j,j} \leq D'^{(3)}_{i,i} \leq D'^{(3)}_{k,k}$. If $D'^{(3)}_{i,i}$ is larger (smaller) than $\mu = 0.5 \cdot (D'^{(3)}_{j,j} + D'^{(3)}_{k,k})$, rotate around $\mathbf{b}_k$ ($\mathbf{b}_j$) such that $D'^{(4)}_{i,i} = \mu$. Afterwards, rotate around $\mathbf{b}_i$ such that $D'^{(5)}_{j,j} = D'^{(5)}_{k,k}$.

The final matrix $\mathbf{D}'^{(5)}$ equals $[\mathbf{D}']_{\tilde{\mathbf{E}}}$. The correct angles ϕ for the rotations are found by writing the desired elements of $\mathbf{D}'^{(n+1)}$ as trigonometric functions of elements from $\mathbf{D}'^{(n)}$ and ϕ and solving the specified equalities for ϕ. To avoid visible boundaries that would result from a fixed threshold ϵ, we perform a gradual transition between the non-degenerate and the degenerate case ($\rho = 0$) by scaling rotation angles ϕ by $(1 - \rho/\epsilon)$. Due to periodicity, there are infinitely many values of ϕ which solve the given trigonometric equalities. However, scaling ϕ for interpolation only produces the expected result when selecting the smallest possible value of ϕ.

6.3. Experimental Results

We used component-wise convolution with a cubic B-spline kernel [154] to obtain a differentiable tensor field from the discrete sample values. This method preserves positive definiteness and implies slight smoothing. Figure 6.3 (a) presents a c_l map of a coronal section of the brainstem. It reveals several tracts, which have been annotated by an expert: The *pontine crossing tract* (pct), the *superior cerebellar peduncle* (scp), the *decussation*



(a) Annotated c_l map of the brainstem

(b) Edges in c_l, from eigenvalue derivatives

(c) Approximate edges from scalar c_l samples

(d) Edges in FA

(e) Edges in c_s

(f) Plots of c_l (—), FA (– –), and c_s ($\cdots$)

Figure 6.3.: Edges in c_l, based on eigenvalue derivatives, separate adjacent fiber tracts in DT-MRI data (b). Neither evaluating c_l at grid points (c) nor mapping edges in other shape measures (d+e) produces results of comparable quality.

of the superior cerebellar peduncle (dscp), the *corticopontine/corticospinal tract* (cpt/cst), and the *middle cerebellar peduncle* (mcp). Figure 6.3 (b) is produced by computing the total derivative of c_l,

$$c_l' = \frac{\lambda_1(-2\lambda_2' - \lambda_3') + \lambda_2(2\lambda_1' + \lambda_3') + \lambda_3(\lambda_1' - \lambda_2')}{(\lambda_1 + \lambda_2 + \lambda_3)^2} \tag{6.4}$$

and using regularized eigenvalue derivatives to evaluate it. Minima in edge strength nicely separate adjacent fiber bundles, which was confirmed by overlaying the edges onto a color coded direction map. Figure 6.3 (c) illustrates that due to the non-linearity of c_l, it is not sufficient to evaluate this measure at grid points and to construct an edge map by computing gradients from the resulting scalar samples.

Similar results have previously been obtained by Kindlmann et al. [115], who employ valley surfaces of fractional anisotropy (FA) to reconstruct interfaces between adjacent tracts of different orientation. Figure 6.3 (d) presents an FA edge map of the same region. Unlike c_l, FA can be formulated directly in terms of tensor components, so exact edge maps do not require eigenvalue derivatives. A comparison to Figure 6.3 (b) suggests that c_l produces more pronounced fiber path boundaries. In particular, the dscp is hardly separated in the FA edge map.

The observation that a *shape* measure like FA can be used to find boundaries in *orientation* has been explained by the fact that partial voluming and component-wise interpolation lead to more planar shapes in between differently oriented tensors [115]. Since c_l is

more sensitive to changes between linearity and planarity than FA is, it is better suited to identify such boundaries.

Further evidence for this reasoning is given in Figure 6.3 (e), which presents edges in c_s, an isotropy measure that completely ignores the difference between linearity and planarity, and which consequently is less effective at separating tracts than FA. The visual impression is confirmed by Figure 6.3 (f), which plots c_l (solid line), FA (dashed line) and c_s (dotted line, uses the axis on the right) against vertical voxel position along a straight line that connects the centers of dscp and scp. It exhibits a sharp minimum in c_l, a shallow minimum in FA, and no extremum in c_s.

6.4. Invariant Gradients in Terms of Eigenvalue Derivatives

The currently most sophisticated method for detecting different types of edges in DT-MRI data has been suggested by Kindlmann et al. [113]. It is based on considering shape invariants J_i as scalar functions over Sym_3, the vector space of symmetric, real-valued 3×3 matrices, and computing their gradient $\nabla_{\mathbf{D}} J_i$, which is an element from Sym_3 for each tensor $\mathbf{D}$. Then, a set $\{\widehat{\nabla_{\mathbf{D}} J_i}\}$ of normalized orthogonal gradients is used as part of a local basis and the coordinates of a tensor derivative $\mathbf{D}'$ with respect to that basis indicate the magnitude of change which is aligned with changes in the corresponding invariant J_i.

Bahn [11] treats the eigenvalues as a fundamental parameterization of the three-dimensional space of tensor shape. Within this eigenvalue space $\mathcal{S} \cong \mathbb{R}^3$, he proposes a cylindrical and a spherical coordinate system, where both the axis of the cylinder and the pole of the sphere are aligned with the line of triple eigenvalue identity ($\lambda_1 = \lambda_2 = \lambda_3$). The resulting coordinates are closely related to standard DT-MRI measures like mean diffusivity and FA (cf. Section 2.2.4). Ennis and Kindlmann [67] point out that the two alternative sets of invariants in their own work, K_i and R_i, are analogous to the cylindrical (K_i) and spherical (R_i) eigenvalue coordinate systems, respectively. However, they state that the latter cannot be easily applied for edge detection, because they are not formulated in terms of tensor components.

A connection between both approaches can be made via eigenvalue derivatives: Restricting tensor $[\mathbf{D}']_{\bar{\mathbf{E}}}$ from Section 6.2 to its diagonal yields a vector in $\mathbb{R}^3$, which describes the shape derivative in eigenvalue space. Moreover, the analogous definitions of the tensor scalar product $\langle \mathbf{A}, \mathbf{B} \rangle = \mathrm{tr}(\mathbf{A}^{\mathrm{T}}\mathbf{B})$ and the standard dot product on $\mathbb{R}^3$ preserve magnitudes and angles when converting between both representations. This means that once $\mathbf{D}'$ has been rotated such that eigenvalue derivatives are on its diagonal, we can alternatively analyze shape changes in eigenvalue space $\mathcal{S}$ or in Sym_3, and obtain equivalent results.

This insight simplifies the derivation of invariant gradients: Instead of having to isolate them from the Taylor expansion (as in the appendix of [67]), invariants J_i can now be considered as functions over eigenvalue space $\mathcal{S}$, and their gradients $\nabla_{\mathcal{S}} J_i$ in $\mathcal{S}$ are simply found via the basic rules of differentiation. This makes it possible to extend the invariant

(a) Annotated c_l map (coronal section)

(b) AO measure from [113]

(c) Sharper AO measure involving $\nabla\widehat{c_p}$

Figure 6.4.: Extending the invariant gradients framework towards the Westin measures allows for a sharper version of the adjacent orthogonality (AO) measure.

gradients framework towards the Westin measures. The corresponding gradients are

$$\nabla_{\mathcal{S}}\, c_l \sim \begin{pmatrix} 2\lambda_1 + \lambda_3 \\ -2\lambda_1 - \lambda_3 \\ -\lambda_1 + \lambda_2 \end{pmatrix} \quad \nabla_{\mathcal{S}}\, c_p \sim \begin{pmatrix} -\lambda_2 + \lambda_3 \\ \lambda_1 + 2\lambda_3 \\ -\lambda_1 - 2\lambda_2 \end{pmatrix} \quad \nabla_{\mathcal{S}}\, c_s \sim \begin{pmatrix} -\lambda_3 \\ -\lambda_3 \\ \lambda_1 + \lambda_2 \end{pmatrix} \tag{6.5}$$

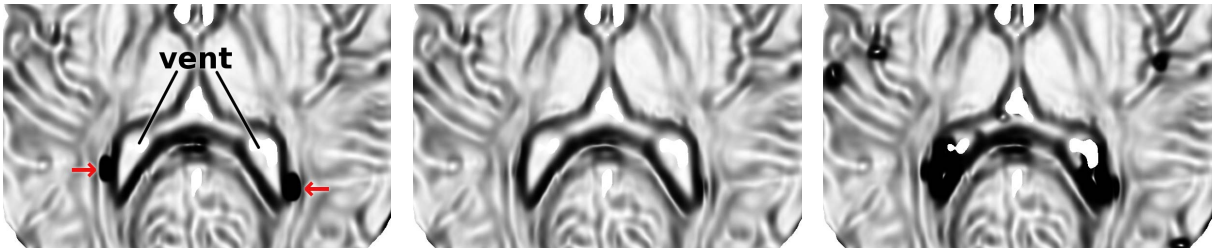
where scalar prefactors have been omitted for brevity, because the gradients will be normalized before use. All three are orthogonal to $\nabla_{\mathcal{S}}\,\|\mathbf{D}\| \sim (\lambda_1, \lambda_2, \lambda_3)^{\mathrm{T}}$. It has been pointed out [67] that the gradients of the Westin measures cannot be used as part of a basis of tensor shape space, which follows immediately from the fact that they provide three coordinates for a two-dimensional space. However, one may still select an arbitrary measure (c_l, c_p, or c_s) as part of an orthonormal basis of $\mathcal{S}$. The basis is then constructed from $\widehat{\nabla_{\mathcal{S}}}\|\mathbf{D}\|$, the normalized selected gradient from Equation (6.5), and a third vector which is the cross product of the first two and captures any remaining changes in shape.

The fiber tracts in Figure 6.4 (a) are *superior longitudinal fasciculus* (slf), *internal capsule* (ic), *corpus callosum* (cc), *cingulum* (cing), and *fornix* (fx). Figure 6.4 (b) presents a map of the adjacent orthogonality (AO) measure, defined in [113] from the coordinates of the tensor field derivative in the invariant gradients framework as

$$\mathrm{AO} = \sqrt{\|\nabla\widehat{R_3}\|^2 + \|\nabla\widehat{\phi_3}\|^2} \tag{6.6}$$

It separates differently oriented tracts, based on shape changes towards planarity, measured by $\nabla\widehat{R_3}$, and rotations around $\mathbf{e}_3$, measured by $\nabla\widehat{\phi_3}$. For detailed information on rotation tangents $\widehat{\mathbf{\Phi}_\mathbf{i}}$, which are used to analyze changes in orientation, the reader is referred to [113].

For the task of separating differently oriented tracts, we select a basis of $\mathcal{S}$ that has $\nabla_{\mathcal{S}}\, c_p$ as one of its axes. Since c_p reacts more specifically to changes in planarity than R_3 does, a sharper version of AO is then obtained by replacing $\nabla\widehat{R_3}$ with $\nabla\widehat{c_p}$. In particular, this produces clear borders in some locations where the original formulation of AO indicates no or only very unsharp boundaries (marked by arrows in Figure 6.4 (c)). Overlaying them on a color coded direction map confirms that they correspond precisely to tract interfaces.



(a) $\|\nabla \log \mathbf{D}\|$ via finite differences

(b) $\|\nabla \log \mathbf{D}\|$ via chain rule

(c) same as (a), after tensor estimation as in [71]

Figure 6.5.: Approximating $\|\nabla \log \mathbf{D}\|$ via finite differences of logarithms leads to artifacts near steep edges (a/c). They are avoided by using the chain rule (b).

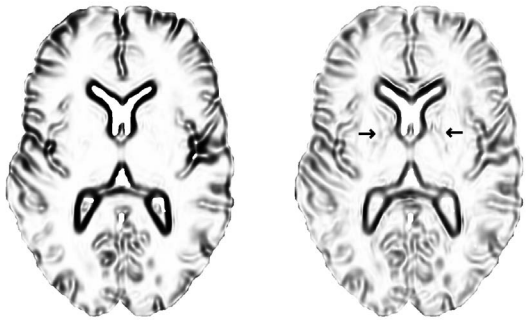
6.5. Edge Detection in the Log-Euclidean Framework

The fact that negative diffusivities do not have any physical meaning restricts diffusion tensors to the cone of positive definite matrices, which is closed under addition and multiplication by positive scalars. Arsigny et al. [9] point out that after taking the matrix logarithm, one may process diffusion tensors with arbitrary (even non-convex) operations without leaving the positive definite cone in the original space, because the inverse map, the matrix exponential, maps all real numbers back to positive values. The matrix logarithm $\log \mathbf{D}$ of a diffusion tensor $\mathbf{D}$ is computed by performing its spectral decomposition (Equation (2.12)) and taking the logarithm of the eigenvalues.

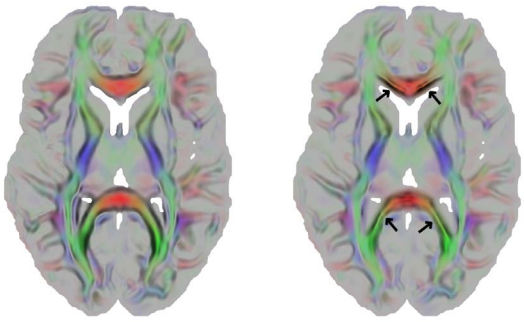
Within this framework, edge strength is measured as $\|\nabla \log \mathbf{D}\|$. We call this the Log-Euclidean edge detector, in contrast to the standard Euclidean edge detector $\|\nabla \mathbf{D}\|$. To simplify computations, it has been suggested to evaluate $\log \mathbf{D}$ at sample positions and to approximate $\|\nabla \log \mathbf{D}\|$ by taking finite differences between the resulting matrices [9]. However, as we have seen in Figure 6.3 (c), approximating the derivative of a nonlinear function via finite differences may not produce sufficiently exact results. In fact, near steep edges like those between the *ventricles* (vent) and brain tissue, we observed artifacts in approximated Log-Euclidean edge maps, marked by two arrows in Figure 6.5 (a).

This problem can be avoided by applying the multivariate chain rule of differentiation, which is again simplified by considering eigenvalue derivatives. In the eigenframe of $\mathbf{D}$, the Jacobian of the matrix logarithm $\log \mathbf{D}$ takes on diagonal form: Let $D_{i,j}$ be entry (i,j) of $[\mathbf{D}]_{\bar{\mathbf{E}}}$, $L_{i,j}$ be the corresponding entry of $[\log \mathbf{D}]_{\bar{\mathbf{E}}}$. Then, $\partial L_{i,i}/\partial D_{i,i} = D_{i,i}^{-1} = \lambda_i^{-1}$ and for $i \neq j$, $\partial L_{i,j}/\partial D_{i,j} = (\log D_{i,i} - \log D_{j,j})/(D_{i,i} - D_{j,j})$. All other partial derivatives vanish. Thus, we simply obtain $[(\log \mathbf{D})']_{\bar{\mathbf{E}}}$ from $[\mathbf{D}']_{\bar{\mathbf{E}}}$ by multiplication of entries (i,i) by λ_i^{-1} and multiplication of entries (i,j), $i \neq j$, by $(\log \lambda_i - \log \lambda_j)/(\lambda_i - \lambda_j)$. The resulting corrected map is shown in Figure 6.5 (b).

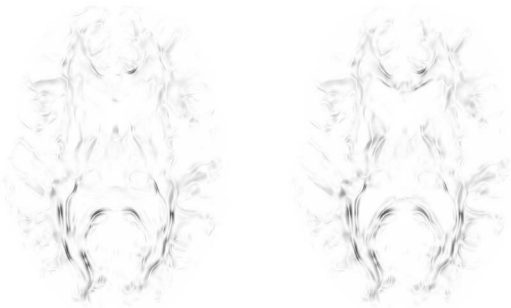
We confirmed that the artifacts in Figure 6.5 (a) are not caused by estimating the tensors via the standard least squares method [13] and clamping the rare negative eigenvalues to a small positive epsilon afterwards. They are still present when using the gradient descent approach from [71], which integrates the positive definiteness constraint into the estimation process itself (Figure 6.5 (c)).



(a) Changes in overall diffusivity



(b) Changes in anisotropy



(c) Changes in skewness

Figure 6.6.: In (a) and (b), different types of edges are distinguished more cleanly by a Euclidean (left) than by a Log-Euclidean edge detector (right).

The reformulation of the invariant gradients framework in terms of eigenvalue space allows us to apply it to Log-Euclidean edge detection, simply by considering the natural logarithms of eigenvalues as the fundamental axes of tensor shape space. Similar to the Euclidean case, the cylindrical coordinate system from the work of Bahn [11] can be used to separate meaningful types of edges. Figure 6.6 (a) shows edges in overall diffusivity, Figure 6.6 (b) edges in anisotropy. In both cases, the Euclidean result is on the left, the Log-Euclidean one on the right.

The Log-Euclidean approach measures overall diffusivity by the matrix determinant instead of the trace. In addition to eigenvalue magnitude, the determinant also reflects eigenvalue dispersion, which explains why the contours of some fiber tracts appear in the right image of Figure 6.6 (a). In the Euclidean case, they are isolated more cleanly in the anisotropy channel. Consequently, anisotropy contours appear more blurred in the Log-Euclidean case. Overlaying them on a principal eigenvector color map indicates that they are offset towards the inside of fiber tracts in some places (arrows in Figure 6.6 (b)). Maps of the third shape axis, which captures transitions between linearity and planarity, were extremely similar (Figure 6.6 (c)). The dynamic range of values in Subfigure (a) is much greater than in (b) and (c), so the mapping of gradient magnitude to grayscale has been adjusted per subfigure in order to improve overall contrast.

Note that a spherical coordinate system does not produce meaningful results for a Log-Euclidean edge detector, since the origin is now at $\lambda_1 = \lambda_2 = \lambda_3 = e^0 = 1$, and the distance to this point is no longer a measure of overall diffusivity.

6.6. Conclusion

Given the ubiquity of the spectral decomposition in DT-MRI processing, eigenvalue derivatives are a natural candidate for the analysis of local changes in this kind of data. In this chapter, we have used them to generate edge maps with respect to the widely used Westin shape measures [241], which we have shown to identify anatomical interfaces in real DT-MRI data and to allow for a more specific analysis of changes in tensor shape than it has been possible with previously suggested edge detectors.

The existing edge detection framework based on invariant gradients [113] is both simplified and easily extended by considering it in terms of eigenvalue derivatives. Finally, we have applied our results to analyze the Log-Euclidean edge detector [9]. We have both corrected a source of artifacts in its previously proposed form and demonstrated that it, too, allows for separation of different types of edges, yet with a slightly lower anatomical specificity than the more traditional Euclidean detector.

7. Segmentation of Anisotropy Isosurfaces

Diffusion anisotropy is the degree to which the apparent diffusivity in a voxel is directionally dependent. In the human brain, high anisotropy indicates coherently organized nerve fibers. Consequently, measures of anisotropy play an important role both in clinical practice and in medical studies (cf. Section 2.2.4).

The isosurfaces of scalar measures like fractional anisotropy (FA) or linearity (c_l) outline the contours of major white matter tracts and thus provide a large-scale overview of diffusion datasets [257]. However, these surfaces are complex and convoluted, so structures deep within the brain are occluded by structures further outside from most viewpoints. User-defined clipping planes reduce this problem, but do not offer a natural and convenient tool to eliminate curved parts of the surface.

In the previous section, we discussed several types of edge information in diffusion tensor data. In our work [191], we show how such indicators of anatomical boundaries can be used to segment anisotropy isosurfaces into meaningful regions. Subsequently, this segmentation can be used to restrict the surface to the parts a user is interested in, and a coloring based on the segmented regions helps to visually identify anatomically relevant parts at first glance.

This chapter is organized as follows: In Section 7.1, we discuss existing methods for mesh segmentation and relate them to our own work. Section 7.2 formalizes our segmentation criterion, which is then used by the algorithm described in Section 7.3. Results are discussed in Section 7.4, before Section 7.5 concludes the chapter.

7.1. Related Work

Our work uses the magnitude of directional derivatives in the tensor field as a criterion to segment anisotropy isosurfaces. While previous work has confirmed that the magnitude of the tensor field gradient indicates anatomically meaningful boundaries in the data [154, 147, 112] and can be used to steer geodesic active contours [69, 187], such differential information has so far not been used to segment isosurface meshes. Moreover, we consider directional derivatives rather than the full gradient magnitude, which is critical for our results.

Watershed-type segmentation methods offer themselves as a natural candidate for the task at hand, since they are steered by a scalar "height" field and the derivative magnitude can be used to define such a scalar field on the isosurface. The watershed principle has previously been used for surface segmentation by Mangan and Whitaker [135] and several others (cf. [42] and references therein). However, these works have a completely different background: They aim at segmenting general surface meshes according to their geometric

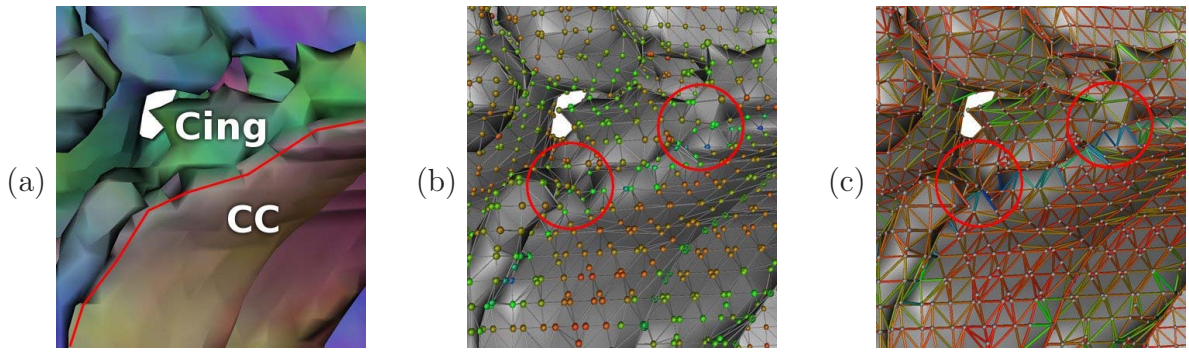


Figure 7.1.: For separating the narrow *cingulum bundle* from the *corpus callosum* (a), edge weights (c) are more appropriate than vertex heights (b).

properties, while our method depends on the underlying tensor field and finds regions which are not defined by surface geometry.

Both Rettmann et al. [173] and Vivodtzev et al. [225] have segmented isosurfaces from cranial MRI. Their approaches are based on geodesic depth and curvature, respectively, which are suitable to segment the gyri and sulci of the cortical surface, but do not allow segmentation of the major white matter structures depicted by anisotropy isosurfaces. Our method draws on information about fiber orientation, which is specific to DT-MRI and not present in the conventional MRI data these two works deal with.

Our work contributes to the field of general mesh segmentation in that it uses edge weights as a segmentation criterion, while all previous watershed-based mesh segmentation methods define a height field on the vertices. In this respect, the work by Page et al. [153] comes closest to our approach, since it uses a *directional* height field. However, their method only considers directional information in the final stage of the segmentation, when all regions have already been found and 70–90% of the vertices have been labeled based on vertex-specific information.

As an alternative to anisotropy isosurfaces, direct volume rendering of DT-MRI data has been suggested [116]. While that approach is not limited to depicting isosurfaces of anisotropy metrics, it has been used to produce results that are comparable to renderings of isosurfaces [224] and share their problems regarding occlusions. In this work, we concentrate on isosurfaces because our segmentation method requires explicit geometry.

7.2. Definition of Edge Weights

High local contrast in DT-MRI data, as measured by the Frobenius norm of the tensor field derivative, indicates an anatomical boundary [154]. Standard watershed-based segmentation algorithms like [135] define a height field on the mesh vertices. However, we found that this strategy only works well for sufficiently broad regions which are separated by a clear layer of boundary vertices, and leads to unsatisfactory results on narrow structures.

In Figure 7.1 (a), the goal is to separate the *cingulum bundle* (Cing, green) from the *corpus callosum* (CC, red). Figure 7.1 (b) color-codes the derivative magnitude in the tangent plane at mesh vertices (red indicates low magnitude, blue shows high magnitude). In the places marked by red circles, the cingulum is only few vertices wide, and using vertex heights would prevent different parts of it from being merged before a merge with the corpus callosum occurs.

Therefore, we propose to define an edge weight w, based on the magnitude of the directional derivative along the edge,

$$w = \int_{\text{edge}} \left\| \frac{\partial \mathbf{D}(\mathbf{r}(t))}{\partial \mathbf{e}} \right\| ds \tag{7.1}$$

where $\mathbf{e}$ is the direction of the edge, and the edge is parametrized as $\mathbf{r}(t)$. This rule allows us to express that vertices should be included in the same region as *specific* neighbors. Figure 7.1 (c) shows that with this rule, edges that separate Cing from CC receive a high weight, while edges within the Cing are much lighter.

In Equation (7.1), we intentionally take the integral of the derivative magnitude along the edge rather than its average. The resulting bias towards clustering geometrically close vertices together stabilizes the segmentation in cases where marching cubes generated extremely short edges. In the implementation, we approximate the integral by evaluating the magnitude of the directional derivative at the center of the edge, and multiplying the result with the edge length. As in [154], derivatives are obtained by convolution with the derivative of a C^2 piecewise-cubic reconstruction kernel.

In cerebrospinal fluid (CSF), the apparent diffusion coefficients are much higher than within tissue, leading to large diffusion tensor traces. At ventricle boundaries, this leads to spurious derivative magnitudes that dominate those between different white matter regions. Since tensor trace has been reported to be approximately constant over functional tissue [164], it is safe to filter out this disturbance, without losing relevant information elsewhere. Like previous authors [154], we achieve this by considering the derivative of the trace-free deviatoric from Equation (2.23).

Taking directional derivatives along FA or c_l isosurfaces implicitly excludes variations in these measures, so our segmentation exploits the part of the data which has been ignored by the anisotropy measure. Since we explicitly eliminate the influence of tensor trace, the only remaining degrees of freedom are those anisotropy changes which are not captured by our metric (e.g., skewness in case of FA), and changes in orientation. Thus, our segmentation criterion is conceptually similar to the adjacent orthogonality measure (cf. Equation (6.6) and Figure 6.4), which makes it plausible that it will yield anatomically meaningful results.

We compute the directional derivatives from the gradient, via the chain rule. The gradient $\nabla \mathbf{D}$ of a tensor field is a third-order tensor, which can be thought of as a three-vector of its partial derivatives $\mathbf{D}^{(x)}$, $\mathbf{D}^{(y)}$, and $\mathbf{D}^{(z)}$. A directional derivative is obtained by taking the inner tensor-vector product with a unit-length vector $\mathbf{e} = (e_x, e_y, e_z)^{\mathrm{T}}$ that defines the desired direction. Combined with the filtering of tensor trace, the formula for

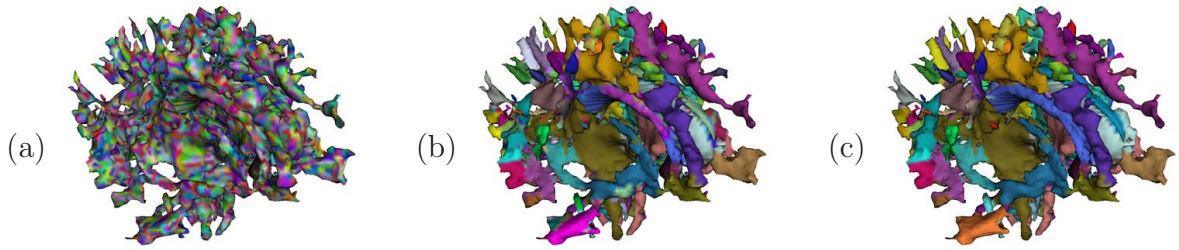


Figure 7.2.: Our watershed-based algorithm reduces an initial oversegmentation (a) by region merging (b) and a final cleaning step (c).

the traceless directional derivative magnitude M reads

$$M = \left\| \sum_{i \in \{x,y,z\}} e_i \left[\mathbf{D^{(i)}} - \frac{1}{3} \mathrm{tr}\left(\mathbf{D^{(i)}}\right) \mathbf{I} \right] \right\| \tag{7.2}$$

7.3. Performing the Segmentation

Like most watershed-based algorithms, our segmentation method finds initial regions based on local minima, followed by a merging process to reduce the oversegmentation which typically occurs in the first stage. Additionally, our algorithm requires a third stage in which it removes undesired "noise" regions that have not been matched by the merging criterion. Figure 7.2 gives an overview of these three phases.

7.3.1. Finding Initial Regions

The idea behind the initial step is that each vertex most likely lies in the same region as the vertex to which it is connected by the edge of least weight w. From that neighbor, we recursively traverse the mesh until we either meet a vertex that already has a label, which is then copied to all vertices on our path, or until we enter a vertex through its minimal edge, and form a new region.

This step of our algorithm is very similar to the original method by Mangan and Whitaker [135], except that it uses edge weights to decide on the transitions. On a typical isosurface mesh with 44k vertices, it produces almost 13k regions, which is a strong oversegmentation (Figure 7.2 (a)).

7.3.2. Merging Close Regions

Watershed methods frequently use closely related criteria for the initial (over-)segmentation and the merging. For example, previous authors [135, 173] have simply merged

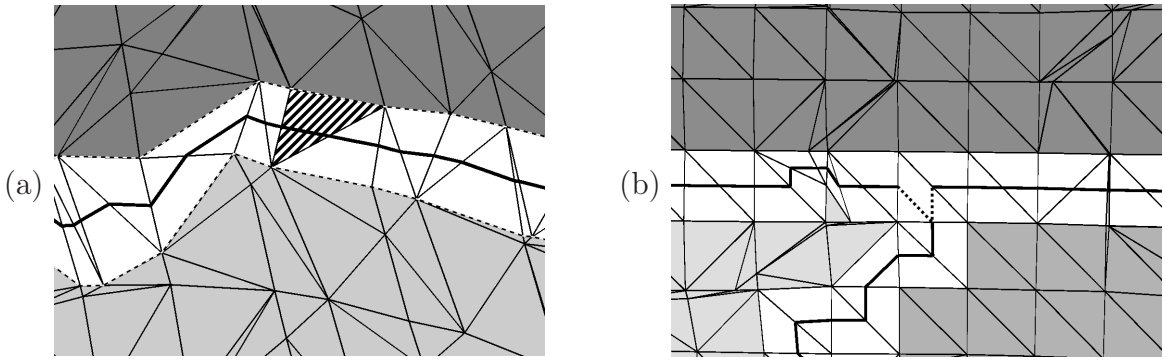


Figure 7.3.: (a) The area between two adjacent regions is their boundary (white). Boundary lines are shown dashed, the mid-boundary line is strong black. A single boundary triangle has been hatched.
(b) At junctions, mid-boundary lines can be connected in different ways. For the line coming from below, the two alternatives are shown dotted.

regions based on the lowest point on their common boundary. In our context, high edge weights along a large part of a boundary are a good indicator that the adjacent regions are distinct, even if they share one or two lighter edges. Consequently, we decide to integrate the directional derivative magnitudes over the full boundary.

Figure 7.3 (a) clarifies the terminology used to formulate our merging rule: Triangles whose vertices belong to more than one region are called *boundary triangles* and form the *boundary* between adjacent regions. Boundary vertices are connected by *boundary lines.* The midpoints of edges within the boundary are connected by the *mid-boundary line.*

The integrated weight W along a boundary is given by

$$W = \iint_{\text{boundary}} \sqrt{\sum_{i=1}^{2} \left\| \frac{\partial \mathbf{D}(\mathbf{b}(s,t))}{\partial \mathbf{v}_i} \right\|^2} \, dA \tag{7.3}$$

where $\mathbf{b}(s,t)$ parameterizes the surface between the boundary lines and the $\mathbf{v}_i$ are any two unit vectors that span the surface element dA. In practice, we again filter the derivatives, as described by Equation (7.2).

The integral in (7.3) is approximated for each boundary triangle by evaluating the derivative magnitudes at the midpoint of each of its edges, averaging the results and multiplying them with the area of the triangle. We found this scheme both reliable and efficient to implement, since it allows us to re-use the tensor field gradients that were used to define the edge weights.

The length of boundary lines is not an indicator of region saliency: In particular, distinct regions can meet along short boundary lines. Consequently, we define the effective weight $W' = W/l$, where l is the length of the mid-boundary line.

Junctions are triangles at which three boundaries meet. We distribute the information from junction triangles to the three adjacent boundaries by adding one third of the triangle's weight, area and mid-boundary length to each of them. While the integral in

(7.3) can be evaluated as before, the length of the mid-boundary line is now estimated by averaging over both possible configurations (cf. Figure 7.3 (b)).

Since the boundaries and the resulting weights are changed by merges, the order of merge operations plays an important role for the final result. We keep all boundaries in a priority queue and iteratively merge the lightest one, until a user-defined threshold on W' is reached.

The merging step is the core of our method, reducing the 13k initial regions in the above example to 623 more meaningful ones (Figure 7.2 (b)). In terms of mesh segmentation in general, it is also the main novelty of our approach. We expect that it can be transferred to other segmentation goals, as long as a reasonable definition of edge weights can be given.

7.3.3. Cleaning up

Many watershed approaches that are based on vertex heights merge based on basin depth, defined as the difference between boundary height and minimal height within a region [135, 173]. In contrast, our algorithm only considers absolute boundary weights. Consequently, regions that are fully contained in areas of high gradient magnitude are left as noise, since none of their boundaries are ever selected in the merging step.

The resulting noise regions are mostly either very small, or thin and elongated regions along the boundary of two larger ones. In a final step, we force a merge of all regions for which the number of internal edges is lower than the number of edges that form the boundary. This captures both types of noise regions, without introducing any new parameters. In all cases, large and legitimate regions were preserved.

The merging partner for the noise regions was again selected using the lowest effective weight W'. In our example, the final step removed 536 out of 623 regions left after merging (Figure 7.2 (c)).

7.3.4. Notes on the Implementation

In an efficient implementation of the described segmentation, the mesh is traversed only once, for the initial labeling. The subsequent merging and cleaning stages are performed via simple data structures for regions and boundaries. For each region, we store the number of internal and boundary edges, a list of all boundary indices, and a list of all regions that have been merged into this region. For each boundary, we store the associated region and edge indices, as well as the accumulated weights W, and mid-boundary line length l.

Merging itself is performed using a priority queue that holds all boundary indices and their effective weights W'. We maintain an array that keeps track of the final region to which each region of the initial oversegmentation has been merged. If a third region is adjacent to both of the regions we are about to merge, we also need to merge the corresponding boundaries, and update the priority queue. In addition, some care has to be taken as a merge may turn a former junction into a boundary triangle and the boundary information has to be updated accordingly.

(a) (b)

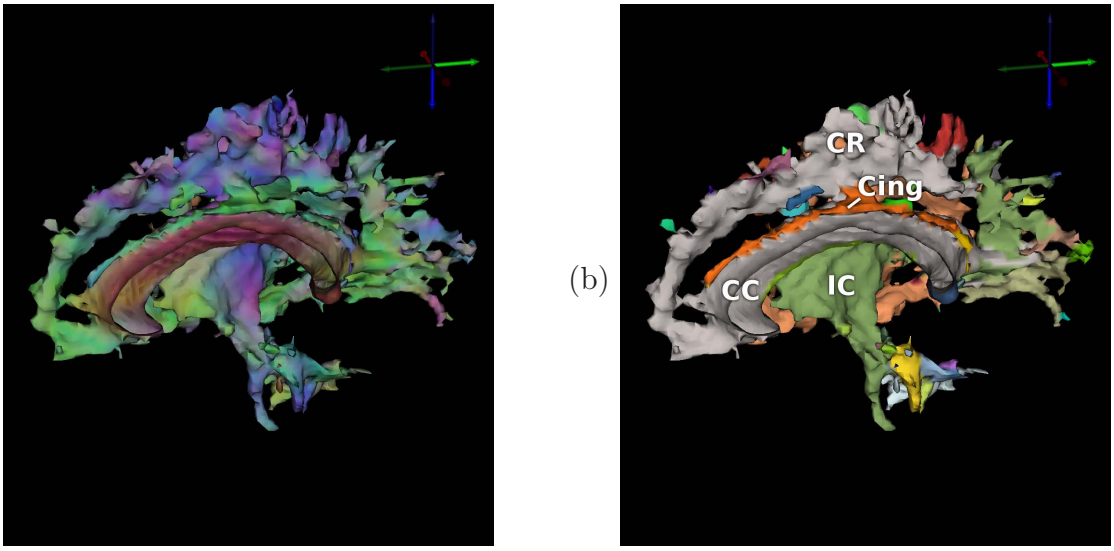


Figure 7.4.: A part of an FA=0.5 isosurface, viewed from the midsagittal plane. (a) shows the standard XYZ-RGB color scheme, (b) presents an annotated segmentation result in random pseudocolors.

For typical datasets, our prototype implementation performs the whole segmentation in less than three seconds on a 2 GHz Athlon 64 processor. This is true both for the example discussed above (44k vertices), and for all results presented in the following section.

7.4. Results

Anisotropy isosurfaces typically consist of one large, central surface and several smaller ones around it. We have performed a connected component analysis and only retained the main component in all our examples. Thus, the presented regions have all been found by our segmentation method. Like previous authors [257, 224], we have tried different values to find an isolevel which clearly depicts the structures of interest.

7.4.1. Segmentation Results and Region-Based Clipping

Figure 7.4 presents a sample result from running our algorithm on an isosurface at FA=0.5. In Subfigure (b), we assign a random pseudocolor to each region in order to emphasize the boundaries found by our method. A comparison to images in a brain atlas [195] suggests that our segmentation correctly captures a number of anatomic structures. For example, the *cingulum bundle* (Cing) is clearly separated from the *corpus callosum* (CC), and the *internal capsule* (IC) is recognized as a region of its own.

Note that the *corpus callosum* has not been separated from the *corona radiata* (CR), reflecting the fact that there is a smooth transition between both structures: In fact, the fibers from the *internal capsule* which fan out in the *corona radiata* are intermingled

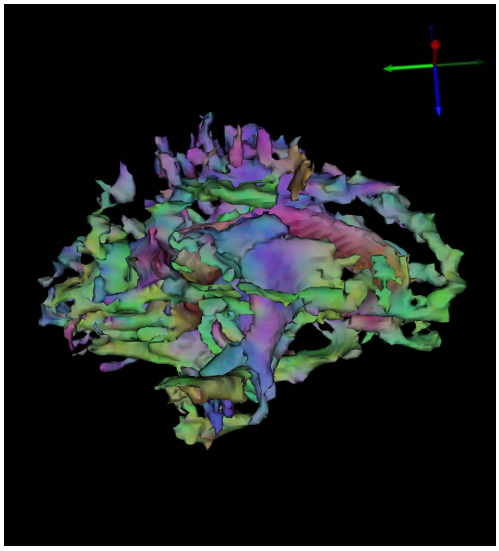
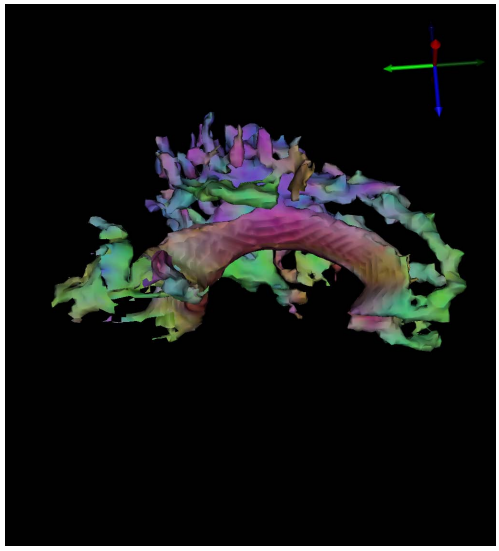
(a)  (b) 

Figure 7.5.: Segmentation results can be used to reveal parts of the surface which are otherwise occluded. In (b), parts of the surface in (a) have been clipped to give a better view on the *corpus callosum*.

with fibers that pass the *corpus callosum*. Thus, a purely boundary-based segmentation method cannot distinguish them.

In Figure 7.5, we demonstrate how the segmentation results can be used to clip parts of the surface that may occlude parts we are more interested in. Figure 7.5 (a) displays the largest connected component of an FA isosurface, while in Figure 7.5 (b), the user has selected the region that corresponds to the *corpus callosum* and the *corona radiata* with the mouse and clipped all other regions to get a better view on its ventral part.

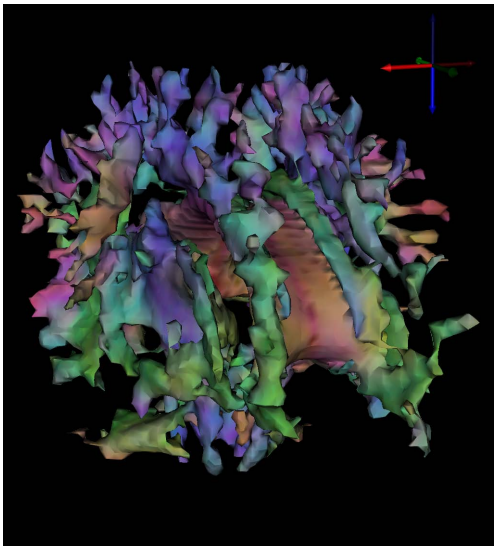
7.4.2. Coloring Regions by Representative

Once we have obtained a segmentation, it is possible to compute one tensor per region that represents its average diffusion behavior. We determine this representative as a weighted average of the data within the region, where the tensor at each vertex is weighted by the sum of areas of all adjacent triangles that belong to the same region. Consequently, a region color can be chosen based on the representative tensor and the standard XYZ-RGB color scheme (cf. Section 2.2.5). To avoid visualizing ill-defined principal eigenvector directions when $\lambda_1 \approx \lambda_2$, color saturation is modulated with $\sqrt{c_l}$.

In order to ensure expressive region representatives, we modify the termination criterion for the segmentation: In addition to providing a threshold on the effective boundary weight W', we now specify a threshold for the linear anisotropy c_l of the region representative that would result from a merge. If it is too low, we are no longer able to assign a clear color to the resulting region. Consequently, the selected boundary is removed from the queue without causing a merge.

Figure 7.6 (b) shows that this modification allows one to abstract from variations within

(a)



(b)

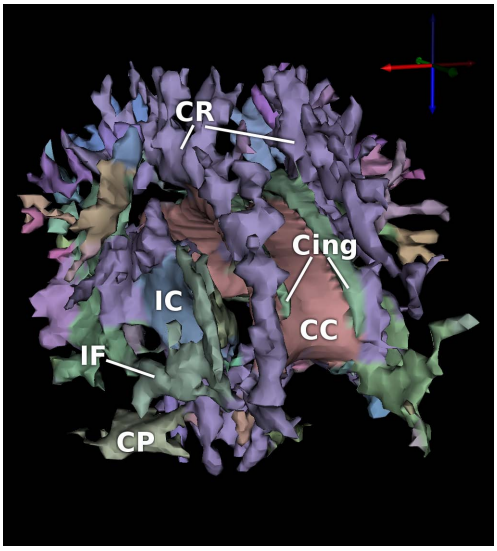


Figure 7.6.: An isosurface at $c_l = 0.26$, seen from the front/top/right. In Subfigure (b), the segmentated regions are colored by their representative tensor. The annotations illustrate that our method has identified anatomically relevant regions.

anatomic regions, and to tell apart several anatomical units at first sight, while still indicating their overall orientation: Again, a region has been identified that corresponds to the *internal capsule* (IC). The *cerebellar peduncle* (CP) and the *inferior fronto-occipital fasciculus* (IF) had also been segmented, but were mostly occluded in Figure 7.4. The *corpus callosum* (CC) is separated from the *cingulum bundle* (Cing), and the region-based c_l criterion even allowed to distinguish it from the *corona radiata* (CR).

7.5. Conclusion and Future Work

In this work, we have suggested that the part of diffusion tensor data which is ignored by anisotropy measures can be used to segment anisotropy isosurfaces in an anatomically meaningful manner, effectively adding information to the visualization and allowing the user to concentrate on parts of the surface that may be of particular interest.

We have demonstrated practical segmentation results using an efficient edge-based watershed approach, which extends methods that have been employed in the context of geometry-based mesh segmentation. The segmentated regions have been shown to reflect a number of anatomically distinct structures. We have used the results to provide an abstracted view on the data that only shows relevant structures and their overall properties.

The isosurfaces described in this paper have been obtained by evaluating FA or c_l for each tensor in the dataset and running the standard Marching Cubes algorithm [132] on the resulting scalar grid. This reflects current practice [257, 197], but introduces an error: In high-curvature regions, we have observed artifacts that stem from the fact that both FA and c_l are nonlinear in the tensor values. Upsampling the tensor field helps, but

makes isosurface extraction too slow to be practicable. In the future, this problem may be addressed by the development of an adaptive method for anisotropy isosurface extraction.

It may also be interesting to see where the results of our method differ when applied to high angular resolution diffusion (HARD) MRI measurements, which do not employ the standard second-order diffusion tensor model. Different anisotropy metrics for such data have been proposed [75, 152]. We expect that it would not be difficult to extract isosurfaces from them and to segment those by appropriately modifying our definition of edge weights.

8. A Higher-Order Structure Tensor

The second-order structure tensor is a common descriptor of local image structure. It is formed by taking the outer product of the image gradient with itself, and averaging the result over a local neighborhood. The major eigenvalue and -vector of the tensor indicate the amount and principal direction of contrast in the neighborhood, and the eigenvalue differences measure the degree to which the neighborhood has a clear principal direction.

The structure tensor was introduced for edge and corner detection [74] and orientation estimation [23]. Since then, it has been applied to a wide variety of problems in image processing and computer vision, including optic flow estimation [22], image diffusion [234], texture segmentation [177], image inpainting [218], and image compression [81].

A structure tensor for vector- and tensor-valued images can be defined by adding the structure tensors of the individual components. This is equivalent to an early idea by Di Zenzo [59] for finding the dominant edge direction in a color image. In the context of processing diffusion tensor data, structure tensors were used for edge detection [147], active contour segmentation, and diffusion-based smoothing [69, 187].

The goal of our work [194] is to overcome the limitation of the traditional structure tensor to a single dominant orientation. To this end, we generalize the second-order structure tensor to a higher-order tensor model. This allows us to investigate the neighborhoods around corners and junctions in greater detail, and to identify cases in which the direction of contrast varies notably over the channels of a multivalued image.

This chapter is organized as follows: After discussing related work in Section 8.1, we introduce our new higher-order structure tensor (HOST) along with some useful mathematical tools in Section 8.2. A novel, maxima-enhancing glyph for higher-order tensors is presented in Section 8.3, and used to visualize experimental results on 2D grayscale and color images, as well as on 3D diffusion tensor data (Section 8.4). Finally, Section 8.5 concludes this chapter and points out directions of future research.

8.1. Related Work

Arseneau and Cooperstock [8] use structure tensors to represent more than a single orientation by placing second-order tensors in discrete directional bins and deriving parameters of a multimodal directional distribution function from them. However, they use the structure tensors only as an intermediate representation to produce a final model which is no longer tensor-based. In contrast, our approach is an extension of the structure tensor itself. Moreover, their work concentrates on lifting the constraint of antipodal symmetry, a property which our approach preserves. Finally, they only present results in 2D, while our implementation covers both the two- and three-dimensional case.

Herberthson et al. [91] have used outer products to handle pairs of orientations. However, their approach neither generalizes to more than two directions, nor does it indicate cases in which representing a single orientation is sufficient.

Generalizations of the structure tensor which do not aim at representing more than a single dominant orientation include a modification to detect spiraling, cross-like, and parabolic shapes [21], and the introduction of nonlinear local averaging [235], which led to nonlinear structure tensors [31].

8.2. A Higher-Order Structure Tensor

8.2.1. Definition of the HOST

The standard second-order structure tensor $\mathbf{J}_\rho$ is given by taking the outer product of the image gradient ∇f with itself, and convolving the result with a Gaussian kernel G_ρ with standard deviation ρ:

$$\mathbf{J}_\rho := G_\rho * \left(\nabla f \, \nabla f^T\right) \tag{8.1}$$

The structure tensor representation is independent of the sign of ∇f. Thus, gradients that have the same direction, but opposite orientation do not cancel in the convolution. The eigenvectors and -values of $\mathbf{J}_\rho$ describe local image structure. For example, the principal eigenvector indicates the direction of largest contrast. However, the matrix representation is insufficient if there is more than one dominant direction in a neighborhood: A structure tensor $\mathbf{J}_\rho$ which describes two orthogonal, equally strong directions will have two equal eigenvalues and no longer indicate a principal direction.

This effect is avoided by a higher-order structure tensor $\mathcal{J}_\rho$, formed by repeating the outer product. Taking the outer product of a vector $\mathbf{v}$ with itself l times will be written $\mathbf{v}^{\otimes l}$. It yields an order-l tensor, indexed by $i_1, i_2, \ldots, i_l$:

$$\left(\mathbf{v}^{\otimes l}\right)_{i_1 i_2 \ldots i_l} := v_{i_1} \cdot v_{i_2} \cdots v_{i_l} \tag{8.2}$$

A structure tensor $\mathcal{J}_\rho$ can be interpreted through its induced homogeneous form $J(\mathbf{u})$, which is defined by repeating the inner tensor-vector product of $\mathcal{J}_\rho$ and $\mathbf{u}$ until a scalar is left, as described by Equation (2.10). $J(\mathbf{u})$ specifies the local contrast in a given direction $\mathbf{u}$ and will thus be referred to as a *contrast function*.

To ensure antipodal symmetry of the homogeneous form, tensor order l is chosen to be even. For a second-order structure tensor, J is unimodal, which reflects the fact that it is suitable to model only one dominant direction. For higher orders, J can become multimodal, which allows for a more accurate representation of corners, junctions, and multivalued images.

We consider it a sensible requirement that the values of the contrast function should remain comparable, independent of the tensor order that we use. When evaluated in direction of the gradient, the contrast function yields the squared gradient magnitude in the second-order case. However, taking the outer product l times would raise the gradient magnitude to the lth power. We compensate this by scaling the gradient vector

beforehand. Thus, an order-l structure tensor $\mathcal{J}_\rho$ that reduces to the well-known second-order tensor $\mathbf{J}_\rho$ for $l = 2$ is given by

$$\mathcal{J}_\rho := G_\rho * \left(\frac{\nabla f}{|\nabla f|^{\frac{l-2}{l}}} \right)^{\otimes l} \tag{8.3}$$

In some applications, it is beneficial to have a contrast function that gives the non-squared gradient magnitude [29]. This can be achieved by replacing the exponent $\frac{l-2}{l}$ by $\frac{l-1}{l}$ in Equation (8.3).

8.2.2. Generalized Tensor Trace

The second-order structure tensor trace has been used as a substitute of the squared gradient magnitude [69]. For the higher-order case, Özarslan et al. [152] have proposed a generalized trace operation "gentr" in 3D, which is based on integrating J over the unit hemisphere Ω and reduces to the standard matrix trace for $l = 2$:

$$\mathrm{gentr}(\mathcal{J}) := \frac{3}{2\pi} \int_\Omega J(\mathbf{u})\, d\mathbf{u} \tag{8.4}$$

In 2D, Ω is one half of the unit circle, and the normalization factor $\frac{3}{2\pi}$ is to be replaced with $\frac{2}{\pi}$. Since the generalized trace of an order-l tensor over $\mathbb{R}^2$ equals its Fourier coefficient a_0^l, we can use the results from Section A.2 to verify that

$$\mathrm{gentr}(\mathcal{J}) = a_0^l = 2 \sum_{i=0}^{l/2} [\mathcal{J}]_{2i} \frac{(l-1)!!}{(l-2i)!! \cdot (2i)!!} \tag{8.5}$$

where $l!!$ is the *double factorial,* i.e., the product of integers in steps of two.

In Equation (8.3), we scaled the gradient magnitude such that the maximum value of J is invariant to the tensor order. However, maxima become narrower with increasing l, so the generalized trace decreases with $\frac{(l-1)!!}{l!!}$.

8.2.3. The Canonical Decomposition

Many applications of the second-order structure tensor depend on its spectral decomposition into eigenvectors and eigenvalues (e.g., [234, 122, 218, 69, 81]). As detailed in Section 2.1.6, the supersymmetric canonical decomposition (sCand) comes closest to the spectral decomposition in the sense that it offers a complete decomposition of the tensor into unit vectors $\mathbf{e}_i$ and associated scalars λ_i.

Our experiments with the sCand are based on a re-implementation of the algorithm in [47]. However, that algorithm returns unnormalized vectors $\mathbf{v}_i$ and becomes numerically unstable when $\mathbf{v}_i$ aligns with the y-axis: In such cases, $\|\mathbf{v}_i\|$ tends to infinity, while the corresponding scalar λ_i tends to zero. We work around this problem by reconstructing a tensor $\mathcal{J}'$ only from those $\mathbf{v}_i$ which have a reasonable magnitude. Then, the residual

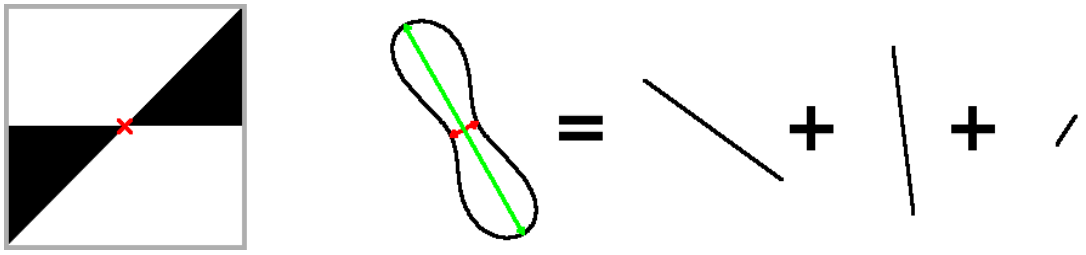


Figure 8.1.: The canonical decomposition can be used to recover individual directions from a higher-order structure tensor.

$\widetilde{\mathcal{J}} := \mathcal{J} - \mathcal{J}'$ can be rotated by 90° to obtain the remaining $\mathbf{v}_i$. In the array representation from Section A.1, the rotation is performed by reversing the array and multiplying all entries $[\mathcal{J}]_i$ with an odd index i by -1.

Figure 8.1 visualizes a sample result of the sCand. Even though the gradient directions in the neighborhood of the considered pixel are too close to be resolved as individual maxima in the contrast profile of an order-four tensor, they are well approximated by the two largest generalized eigenvectors.

Our prototype implementation is in C, and uses routines from Lapack[1] and the Numerical Recipes [168]. It found the sCand of 160 000 order six structure tensors over $\mathbb{R}^2$ in around 2.5 s on a 2 GHz Athlon 64. Unfortunately, no algorithms are known which compute the sCand for tensors of order $l > 3$ over $\mathbb{R}^3$.

8.3. Glyphs for Higher-Order Tensors

The visualization of higher-order tensors has been addressed in generalized diffusion tensor magnetic resonance imaging, which is introduced in Section 2.2.6. In this context, *generalized Reynolds glyphs* are the only glyph-based visualization technique [149, 95]. Let S be the unit sphere (unit circle in $\mathbb{R}^2$) and J the homogeneous function of the tensor. Then, these glyphs are formed by the set of points

$$\{J(\mathbf{u})\,\mathbf{u} \,|\, \mathbf{u} \in S\} \tag{8.6}$$

which depicts the value of the homogeneous form in each direction. Generalized Reynolds glyphs have a round shape around their maxima, which makes their exact direction difficult to see. To compensate this problem, Hlawitschka and Scheuermann [95] suggest to add arrows that point to the maxima. In this section, we present an alternative glyph, which indicates maxima through sharp peaks in its shape.

8.3.1. Higher-Order Tensor Glyphs with Maximum Enhancement

The diffusion ellipsoid is accepted as the standard glyph for second-order tensors, but does not coincide with the Reynolds glyph for $l = 2$. Özarslan and Mareci [149] argue

[1] `http://www.netlib.org/lapack/`

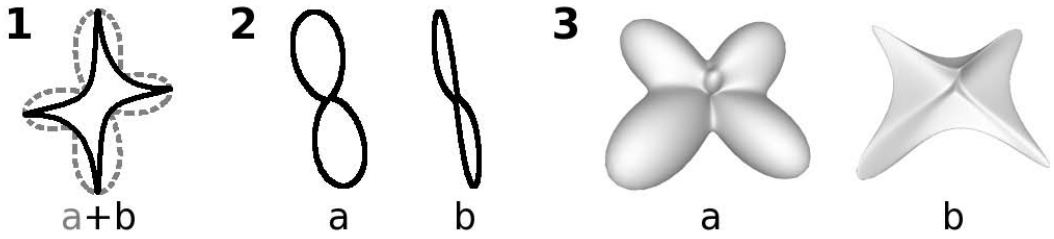


Figure 8.2.: Three structure tensors of order six, visualized with Reynolds glyphs (a) and our new HOME glyphs (b), which visually emphasize the maxima.

that ellipsoids, written as $\mathbf{u}^{\mathrm{T}}\mathbf{J}^{-1}\mathbf{u} = c$, do not have an obvious generalization, since it is unclear how to define the inverse of a higher-order tensor. However, the tensor ellipsoid can be constructed in an alternative manner, by transforming the unit sphere under the linear mapping induced by the tensor [110]. Thus, we generalize it to higher order by taking the inner tensor-vector product $(l-1)$ times, until a vector is left. In the notation of Section 2.1.5, the surface of our glyph is given by the points

$$\{\mathcal{J} \cdot^{l-1} \mathbf{u} \,|\, \mathbf{u} \in S\} \tag{8.7}$$

We name these shapes higher-order maximum enhanced (HOME) glyphs, since they emphasize the maxima, at the cost of a smoother shape around the minima. Their efficiency for the visualization of positive definite tensors is guaranteed by three properties:

1. At stationary points of the homogeneous form, the points on the Reynolds glyph and the HOME glyph coincide. At these points, their distance from the origin equals the value of the homogeneous form.

2. When the homogeneous form has an extremum, the distance of the shape to the origin is extremal both for the Reynolds and the HOME glyph.

3. When the homogeneous form has an extremum, the signed curvature of the Reynolds glyph is smaller than the one of the HOME glyph. Thus, maxima (positive curvature) appear more pronounced in the HOME glyph, while minima (negative curvature) are more pronounced in the Reynolds glyph.

Figure 8.2 demonstrates the stated properties by overlaying a Reynolds glyph (a, dashed and gray in Subfigure 1) with a HOME glyph (b), and presents additional examples, both in 2D and 3D. Unlike the Reynolds glyph, the HOME glyph in Subfigure 2 reveals at first glance that the displayed tensor is not axially symmetric.

Since we are more interested in the maxima than in the minima of the contrast function, we use HOME glyphs throughout this chapter.

8.3.2. Analysis of the HOME glyph

For simplicity, our formal proof is limited to the two-dimensional case. The intuition behind the proof lies in the fact that, at non-stationary points, the product in Equation (8.7)

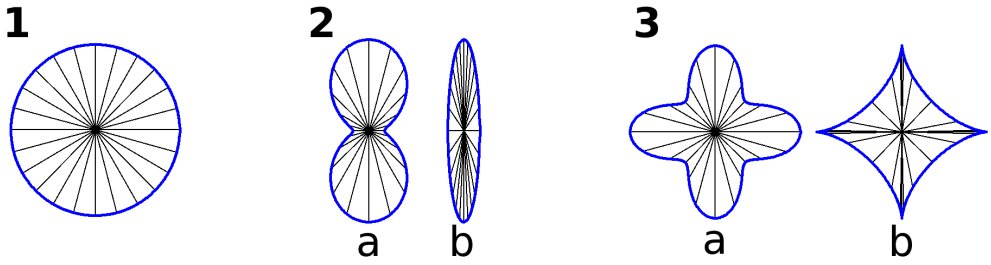


Figure 8.3.: The Reynolds glyph (2a and 3a) preserves the angles between vectors that are distributed uniformly over the unit circle (1). In the ellipse and the HOME glyph (2b and 3b), the vectors are deflected towards maxima.

deflects the vector towards a maximum in the homogeneous form. This fact is the basis of the power method for finding the largest eigenvector of a matrix, as well as its generalization to supersymmetric tensors with a definite homogeneous form [121]. It is illustrated in Figure 8.3 for a matrix (Subfigure 2) and a higher-order tensor (Subfigure 3).

Assume a positive definite tensor $\mathcal{F}$. Given the representation of its homogeneous form on the unit circle as a Fourier series $f(\phi)$ (Section A.2), the Reynolds glyph can be written as a parametric curve $\mathbf{r}(\phi)$ with $\phi \in [0, 2\pi)$. As demonstrated in Equation (2.13), the repeated tensor-vector product $\mathcal{F} \cdot^{l-1} \mathbf{v}$ equals the gradient $\nabla F(\mathbf{v})$ of the homogeneous form, scaled by tensor order l. We re-write this gradient in terms of $f(\phi)$ and its derivative $f'(\phi)$ to obtain a parametric form $\mathbf{h}(\phi)$ of the HOME glyph:

$$\mathbf{r}(\phi) = \begin{pmatrix} f(\phi)\cos(\phi) \\ f(\phi)\sin(\phi) \end{pmatrix} \qquad \mathbf{h}(\phi) = \begin{pmatrix} f(\phi)\cos(\phi) - f'(\phi)\sin(\phi)/l \\ f(\phi)\sin(\phi) + f'(\phi)\cos(\phi)/l \end{pmatrix} \tag{8.8}$$

Let k_r and k_h be the signed curvatures of $\mathbf{r}(\phi)$ and $\mathbf{h}(\phi)$, respectively. Then, the three properties of the HOME glyph stated in the previous section are formalized as:

1. If $f'(\phi) = 0$, $\mathbf{r}(\phi) = \mathbf{h}(\phi)$ and $\|\mathbf{r}(\phi)\| = f(\phi)$.
2. If $f'(\phi) = 0$, $\|\mathbf{r}(\phi)\|' = \|\mathbf{h}(\phi)\|' = 0$.
3. If $f'(\phi) = 0$ and $f''(\phi) \neq 0$, $k_r < k_h$.

The first statement follows directly from Equation (8.8). The second statement involves first derivatives of $\mathbf{r}(\phi)$ and $\mathbf{h}(\phi)$. In case that $f'(\phi) = 0$, they simplify to:

$$\mathbf{r}'(\phi) = \begin{pmatrix} -f(\phi)\sin(\phi) \\ f(\phi)\cos(\phi) \end{pmatrix} \qquad \mathbf{h}'(\phi) = \begin{pmatrix} -f(\phi)\sin(\phi) - f''(\phi)\sin(\phi)/l \\ f(\phi)\cos(\phi) + f''(\phi)\cos(\phi)/l \end{pmatrix} \tag{8.9}$$

From Equations (8.8) and (8.9), it is easily verified that if $f'(\phi) = 0$, $\mathbf{r}(\phi) \cdot \mathbf{r}'(\phi) = 0$ and $\mathbf{h}(\phi) \cdot \mathbf{h}'(\phi) = 0$. This proves statement two. The signed curvatures k_r and k_h are computed from first and second derivatives. If $f'(\phi) = 0$, they are given as:

$$k_r(\phi) = \frac{f(\phi) - f''(\phi)}{f^2(\phi)} \qquad k_h(\phi) = \frac{f^2(\phi) + \frac{3-l}{l} f(\phi) f''(\phi) + \frac{2-l}{l^2} f''^2(\phi)}{|f(\phi) + \frac{1}{l} f''(\phi)|^3} \tag{8.10}$$

In a first step, we show that for a higher-order structure tensor $\mathcal{F}$, $f(\phi)+\frac{1}{l}f''(\phi) \geq 0$: In an appropriate coordinate frame, the homogeneous form of a rank-1 tensor can be written as $f(\phi) = \lambda \cos^l(\phi)$ (cf. Equation (A.3)), so $f''(\phi) = \lambda\, l(l-1)\cos^{l-2}(\phi)\sin^2(\phi) - \lambda\, l \cos^4(\phi)$. Thus, the condition $f(\phi)+\frac{1}{l}f''(\phi) \geq 0$ is true for all ϕ when $\lambda \geq 0$ and order l is even. By linearity of the derivative, the same condition holds for HOSTs, since they are defined as a convex combination of rank-1 tensors with non-negative λ. In the following, we ignore the limit $f(\phi) + \frac{1}{l}f''(\phi) \to 0$, where $k_h(\phi) \to \infty$.

We may now omit the absolute value sign in k_h, and transform Equation (8.10) to specify the following equivalent condition for $k_r < k_h$:

$$\frac{1-2l}{l^2} f^2(\phi) f''^2(\phi) + \frac{1-3l}{l^3} f(\phi) f''^3(\phi) - \frac{1}{l^3} f''^4(\phi) < 0 \tag{8.11}$$

For $f''(\phi) > 0$, all terms in Equation (8.11) are negative, so statement three follows directly. For $f''(\phi) < 0$, we exploit the condition $f(\phi) + \frac{1}{l}f''(\phi) > 0$ to write $f''(\phi) = -\alpha l f(\phi)$ with $\alpha \in (0, 1)$. With this substitution, Equation (8.11) holds if

$$l\alpha^2(-\alpha^2 + 3\alpha - 2) + \alpha^2 - \alpha^3 < 0 \tag{8.12}$$

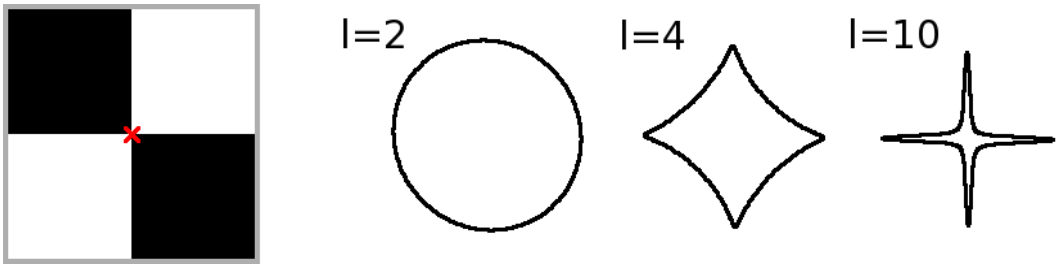
Since $\alpha^2 - \alpha^3 > 0$ and $l > 1$, $\alpha^2 - \alpha^3 < l\alpha^2(1-\alpha)$. Thus, Equation (8.12) is true if $-\alpha^2 + 2\alpha - 1 < 0$, which is verified for the given range of α using simple calculus. This concludes the proof of statement three.

8.4. Results

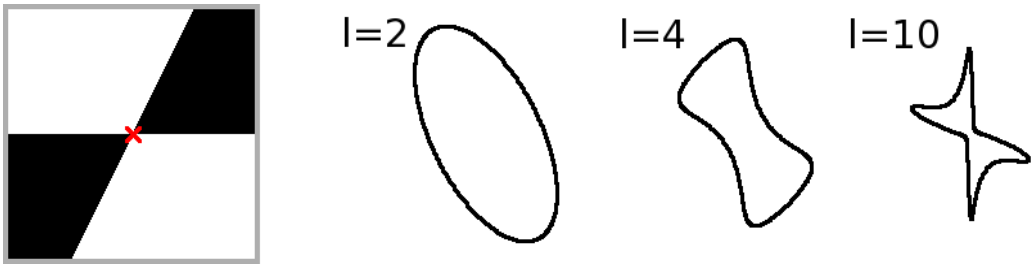
We will now present some experiments to confirm that higher-order structure tensors indeed give a more accurate representation of junctions and multivalued images. Our first experiment uses simple junctions in synthetic grayscale images. Derivatives are calculated by convolution with a derivative-of-Gaussian filter ($\sigma = 0.7$). After HOSTs of different order l have been computed, their information is propagated to a local neighborhood by convolution with a Gaussian kernel ($\rho = 1.4$).

Figure 8.4 shows the test images, with the position of the displayed structure tensor marked by a cross. The results show that a HOST of order $l = 4$ is sufficient to represent two edges that cross orthogonally (a), while the traditional structure tensor ($l = 2$) does not distinguish any particular direction. In the non-orthogonal case (b), the traditional model indicates a principal direction which does not correspond to any gradient found in the image. At the same time, the HOME glyph of order four gives an impression of the involved directions. Similar to the example from Section 8.2.3, the canonical decomposition could be used to find approximations of the gradient directions, but a clear separation of the maxima in the contrast profile requires higher order.

The second experiment is based on a natural color image. Derivatives are now calculated channel-wise and according to the conventional generalization to multi-channel images, the HOSTs of the red, green, and blue color channels are added. For illustration, we do not propagate the structure information ($\rho = 0$) in this case and also show the gradients of the individual color channels in Figure 8.5. Again, the structure tensor of order four



(a) Orthogonal edges are clearly distinguished with order $l = 4$.



(b) For non-orthogonal edges, higher orders provide separate maxima in the homogeneous form.

Figure 8.4.: Two junctions in grayscale images and the corresponding structure tensors. For orders $l > 2$, the directions of the meeting edges can be represented.

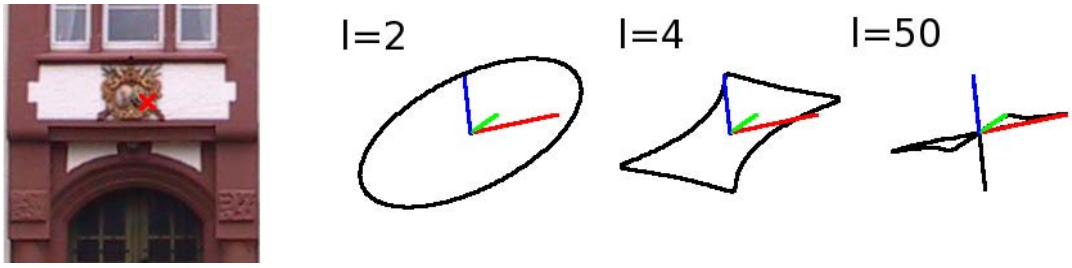
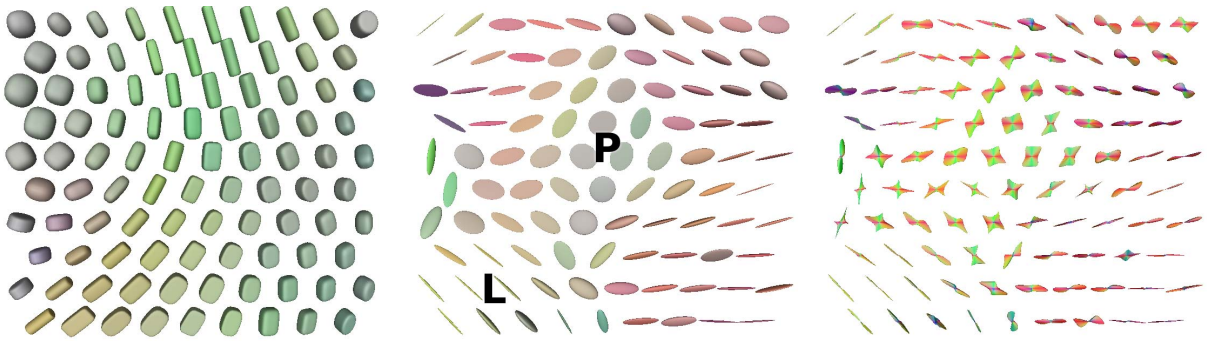


Figure 8.5.: In a color image, the channel-wise gradients may point into different directions. Higher-order structure tensors can be used to model this situation accurately.



(a) Diffusion tensor data (b) Order-2 structure tensors (c) Order-6 structure tensors

Figure 8.6.: The traditional structure tensors (b) from a diffusion tensor field (a) indicate whether the gradients of the tensor channels agree (L) or not (P). Order-6 HOSTs (c) give a detailed impression of the involved directions.

gives a much better impression of the range of directions than the traditional model. To demonstrate the feasibility of going to very high tensor orders, we also present the representation with $l = 50$.

In a similar manner, we computed structure tensors from a three-dimensional diffusion tensor dataset. In this case, both the given image data and the derived descriptors of local structure are tensors. To reduce confusion that might arise from this fact, a detail of the data itself is visualized with superquadric glyphs in Figure 8.6 (a), while the resulting second- and sixth-order structure tensors are shown as ellipsoids and HOME glyphs in (b) and (c), respectively. A large radius of the glyphs in (b) and (c) indicates a strong gradient in the tensor field shown in (a).

In the area marked with an "L", the traditional structure tensors in Figure 8.6 (b) have a linear shape, and are orthogonal to the diffusion tensors in (a). This is due to the fact that in that region, the diffusion tensors keep their orientation, but vary in shape orthogonal to it. A bending of the fiber bundle leads to planar structure tensors in the area marked with a "P": The shape of the diffusion tensors still varies orthogonal to the fiber direction, but additionally, the orientation changes along the path. Distinct peaks in the higher-order structure tensors in (c) resolve these two different sources of variability in the tensor data.

8.5. Conclusion and Future Work

In this chapter, we have shown how higher-order tensors can be used to represent the average of orientations in greater detail than it is possible using traditional second-order structure tensors. We have introduced the notions, definitions and mathematical tools required to work with such higher-order structure tensors (HOSTs). For visualization, we have proposed the higher-order maximum enhancing (HOME) glyph, a generalization of the standard tensor ellipsoid, and we have formally proven three crucial properties of it in the two-dimensional case. Our results establish HOSTs as meaningful features that

represent the local variability of diffusion tensor fields, and HOME glyphs as a way to visualize them effectively.

A number of interesting directions for future work fall outside the scope of this dissertation: In [194], we demonstrated a step towards fully integrated detection of edges and junctions in grayscale images, based on HOSTs. Moreover, higher-order tensors hold the potential to steer image diffusion more precisely than standard models of anisotropy [234], and to allow for finer distinctions in structure tensor-based texture segmentation [177]. Finally, it appears worthwhile to investigate if invariants of higher-order structure tensors [171] can help to distinguish different classes of local neighborhoods.

Part III.

Fiber Tracking

9. Visual Integration of Diffusion MRI and Structural MRI

A significant amount of research has been devoted to the development of fiber tracking algorithms, which try to reconstruct white matter tracts from diffusion MRI data (cf. Section 2.2.5). Moreover, several approaches have been suggested for efficient, hardware-accelerated rendering of the resulting streamtubes [208, 172, 139, 161].

However, showing streamtubes alone is not very informative: Usually, the viewer is interested in understanding the spatial relation of a tract to anatomical landmarks like the gyri and sulci of the cortical surface, or to anomalies like tumors. Such anatomical features are better captured by other measurement protocols, like T_1-weighted MRI.

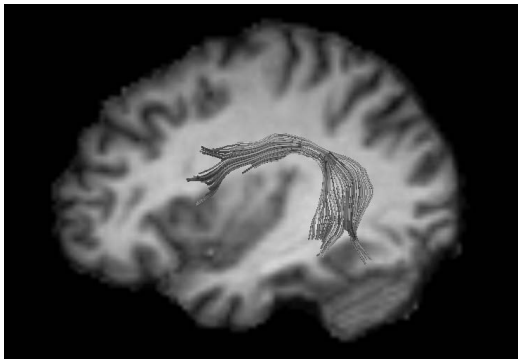
The contribution of our work [188] is to propose a method that puts streamlines from DT-MRI data into context with a coregistered T_1-weighted dataset, such that the spatial relation between both becomes apparent. The technique is based on a data-driven deformation of geometry and has been inspired by a method for anatomical fiber preparation, known as Klingler dissection [133]. It works automatically, but its GPU-based implementation allows for additional, intuitive interaction.

This chapter is organized as follows: First, we give a more detailed motivation (Section 9.1), review related work (Section 9.2), and give an overview of the steps which are necessary to perform a virtual Klingler dissection (Section 9.3). Then, we provide details on the use of deformed geometry for visualization (Section 9.4) and describe implementation issues (Section 9.5). Finally, we present and discuss a number of results (Section 9.6), before we conclude the paper and point out directions for future work (Section 9.7).

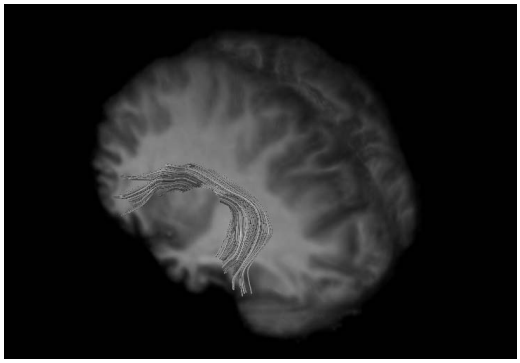
9.1. Motivation

At the current state of the art, T_1 data is coregistered with the diffusion dataset and the streamlines are combined with the anatomical data using standard techniques like slice images [185] or volume renderings with clipping boxes [146].

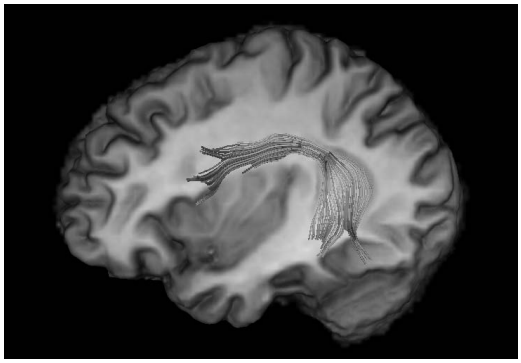
Slice images are the simplest way to put fiber tracts into context. However, they do not convey the three-dimensional shape of structures in the T_1 data (Figure 9.1 (a)). Volume rendering indicates the location of the clipping plane with respect to the surface of the brain, but hardly improves the perception of structures within the plane itself (Figure 9.1 (b)). In fact, since we are mostly cutting through opaque structures, the result from the volume rendering would appear very similar to Figure 9.1 (a) if we had chosen the same viewpoint. Both methods suffer from the fact that the streamlines are not visually connected to the T_1 data. Even when the clipping plane is placed as close as possible to the tract, the streamlines frequently appear to float in mid-air between



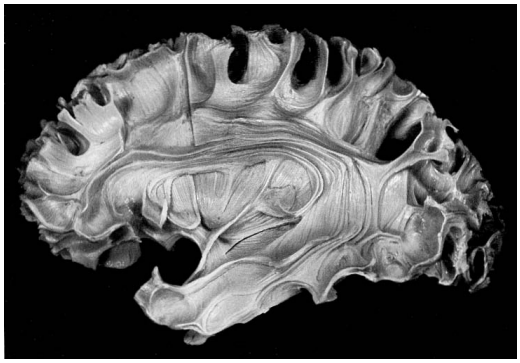
(a) Fiber tracts with context from slice image



(b) Context from direct volume rendering



(c) Virtual Klingler dissection



(d) Example of a Klingler dissection, from [133]

Figure 9.1.: Compared to slice images (a) and direct volume rendering (b), the proposed method (c) both relates fiber tracts more clearly to the anatomy and gives a more plastic impression of the cut T_1 volume.

the T_1 rendering and the viewer, and despite the shading, their exact three-dimensional trajectory is difficult to assess.

When looking at anatomical textbooks, we found the illustrations of white matter tracts produced by Ludwig and Klingler [133] particularly expressive. In Klingler's method for fiber tract dissection, the brain first undergoes a preparation process which includes repeated freezing to spread the fibers apart. Afterwards, it is possible to carefully scratch away tissue from one side to follow the course of fiber bundles. This leads to a relief-like surface in which the desired tract is naturally surrounded by its anatomical context (cf. Figure 9.1 (d)). Our method mimics this process by deforming a cutting plane through the T_1 data, similar to the way in which the final surface in a Klingler dissection is formed by scratching away tissue. Thus, we refer to it as "virtual Klingler dissection".

The example result in Figure 9.1 (c) shows the superior longitudinal fasciculus in its anatomical context: Through the deformed geometry, structures in the T_1 data appear more plastic than with standard methods. Moreover, streamlines produce a visible dent and are rendered more transparently where they are close to the surface, which visually connects them to the T_1 rendering and supports perception of their trajectory. When

comparing our result to the real Klingler dissection, one should bear in mind that it is based on an MRI measurement of limited resolution, not on a photograph of a post-mortem preparation, and that the cerebellum (at the bottom right of our image) has been removed by Ludwig and Klingler.

9.2. Related Work

To our knowledge, our work is the first to specifically address the problem of combining anatomical T_1 data and fiber tracts from DT-MRI data in a way that makes it easy to appreciate the relationship between both. Catani et al. [40] have presented a work with a similar title as ours, "virtual in vivo interactive dissection of white matter fasciculi", but with an entirely different focus: They demonstrate how to reconstruct a number of major fiber tracts from DT-MRI data which agree with the result of postmortem studies. To provide context for their streamtube renderings, they employ simple slice images of fractional anisotropy maps, derived from the DT-MRI data itself.

At the core of our method is the data-driven deformation of a surface. In computer vision, level set methods [198, 148] are a standard tool for this task. Since there is no need to handle topological changes in our context, we preferred a faster and simpler approach based on displacing vertices along a vector field. In graphics, vector fields have previously been used for mesh deformation by von Funck et al. [227]. However, their goal is interactive modeling, not visualization, and there are significant differences in how the vector fields are defined. Among others, their vector fields are analytically defined from a small number of parameters, while ours rely on a sampled representation of volume data.

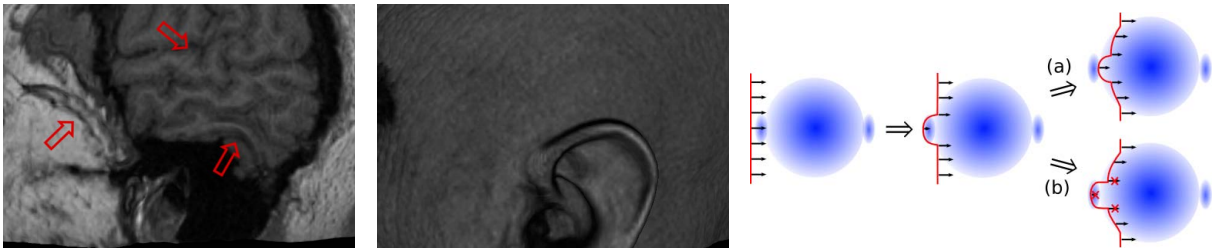
Deformations which mimic the effect of anatomical dissections have recently been studied by Correa et al. [48], but they have focused on illustrative visualization of single datasets, and their method is based on volume deformation, while ours works on explicit geometry.

The relation of our deformation approach to standard methods like isosurfacing [132] and direct volume rendering [62] will be discussed in detail in Section 9.4.1.

9.3. Principles of Virtual Klingler Dissection

In this work, we assume that the fibers of interest have been extracted using any of the established tractography methods and are given as input. The tracts shown in our experiments have been generated by the tensorlines algorithm [237]. To put them into context with anatomical T_1 data, our method takes the following steps:

1. Cut the brain along a plane which is aligned with the streamlines.
2. Deform the plane such that streamlines behind it are revealed and that it adapts to features in a coregistered T_1 data set.
3. Volume texture the plane with the T_1 data and render the streamlines on top of it, using variable transparency to convey their proximity.



(a) Artifacts in a surface at integration time $t = 40$.

(b) Same setting with threshold on total resistance.

(c) Schematic illustration of the situation in (a) and (b).

Figure 9.2.: Pushing a surface through structures like skull and blood vessels can lead to shading artifacts (a) which are avoided by limiting the total resistance (b).

These three steps execute fully automatically in less than a second and in many cases, they already yield a satisfactory rendering. However, our implementation additionally allows the user to modify the result, both by adjusting parameters and by direct interaction with the surface.

9.4. Visualization by Data-Driven Surface Deformation

Inspired by the way in which surfaces evolve in the course of a dissection, we suggest a novel metaphor for volume visualization, in which the volume is thought to possess mechanical resistance. When the initial cutting plane is moved in some direction, the spatially varying resistance deforms the geometry. Similar to direct volume rendering (DVR), in which a transfer function assigns optical properties like color and opacity to materials in the volume, the resistance of the material is defined as a function of the data. Details on the specific transfer function used in a virtual Klingler dissection will be given in Section 9.5.2.

From a scalar resistance measure $r \in [0, 1]$, we define an effective velocity $\mathbf{v}$, parallel to the original surface normal, at which a massless particle may traverse the volume. Its magnitude is given by $\|\mathbf{v}\| = 1-r$. Thus, it is dual to the resistance, just like transparency is dual to opacity in DVR. Formulating the problem in terms of a velocity field allows us to employ a standard tool for flow visualization: One way to visualize a 3D flow is to release a surface into the fluid at some instant, to let it move with the flow, and to observe the evolution of the resulting *time surface* with time t. Our deforming geometry is given as such a time surface, with the velocity defined by the transfer function.

Like an undeformed cutting plane, the final surface is textured with the local values of the volume. However, this only results in an expressive rendering when the shape of the generated surface has a clear connection to its texture. After large integration times, when the surface has been pushed through significant structures, this may no longer be the case: Figure 9.2 (a) presents a closeup of a human head, in which the surface was shaped by its way through skin, skull, and blood vessels. Since the resulting features are not related to the displayed brain tissue, they appear as shading artifacts.

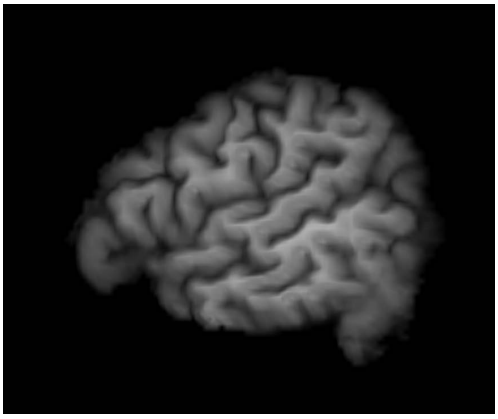
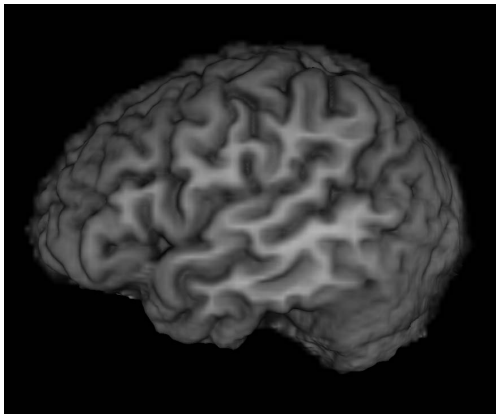
(a)  (b) 

Figure 9.3.: Unlike a simple 2D embossment filter (a), our method reveals structures behind the original plane (b).

To avoid this problem, we limit the total resistance which a surface point may overcome, i.e., a point on the time surface is stopped when the integral $\int_0^s r(u)\,du$ along its path reaches a threshold θ_r. We have fixed this resistance threshold empirically at $\theta_r = 0.7$. Figure 9.2 (b) demonstrates that the threshold avoids the artifacts by stopping the surface on the skin layer. Figure 9.2 (c) illustrates the integration process in both cases: Crossed out vectors denote velocities which are ignored because of the threshold.

9.4.1. Relation to Standard Methods

Even though the deformation process creates geometry from volumetric data, our method is more closely related to direct volume rendering [62] than it is to surface extraction methods like isosurfacing [132]. It is the goal of surface extraction to reconstruct the geometry of some object of interest, like the contour of an organ or a tumor. Thus, rendering the extracted shape alone already provides an expressive visualization. In contrast to this, our deformed geometry is only used to select and shade a part of the volume. Note that this process does not simply result in an improved shading of the original plane (which could be achieved using bump mapping or 2D image filters), but reveals parts of the volume which are behind it (cf. Figure 9.3).

It deserves further discussion whether introducing geometry which does not correspond to a boundary in the volume improves its visualization or rather leads to a false perception of shape. The effectiveness of a Klingler dissection does not suffer from the fact that the resulting surface is shaped by the hands of an anatomist rather than a natural boundary in the living brain. However, a dissection is created by an expert, while our surface deformation happens automatically, based on properties of the volume. In this respect, a more suitable analogy is sandblasting an object which is composed of different materials: Since softer materials erode faster than harder ones, this turns a planar surface into a relief. Even though the evolving surface is not a natural boundary, we can expect its shape to reflect the structure of the object.

Experimentally, we found the resistance threshold described in the previous section essential for obtaining artifact-free results. Interestingly, this threshold creates a theoretical link to thin slab volume rendering [248]. Optical models for direct volume rendering typically include an absorption term which attenuates light intensity $I(s)$ depending on a spatially varying extinction coefficient $\tau(u)$ and distance s [136]:

$$I(s) = I_0 \cdot \mathrm{e}^{-\int_0^s \tau(u)\,du} \tag{9.1}$$

Assume we volume render the slab from which we created the deformed geometry using an orthographic projection perpendicular to the slab and using our resistance term $r(u)$ as the extinction coefficient $\tau(u)$. Then, the distance s at which we stop the surface deformation because of the resistance threshold, $\int_0^s r(u)\,du = \theta_r$, coincides with the point at which transparency in the volume rendering reaches $T(s) = \mathrm{e}^{-\theta_r}$. In particular, the selected threshold $\theta_r = 0.7$ corresponds to transparency $T = 0.5$. This means that in the limit, our deformed geometry converges to a surface of constant accumulated opacity. It is interesting to note that even though such surfaces have so far not been extracted explicitly, previous work [175] has demonstrated how the points of accumulated opacity $A = 0.95$ can be used to separate meaningful layers in certain types of data.

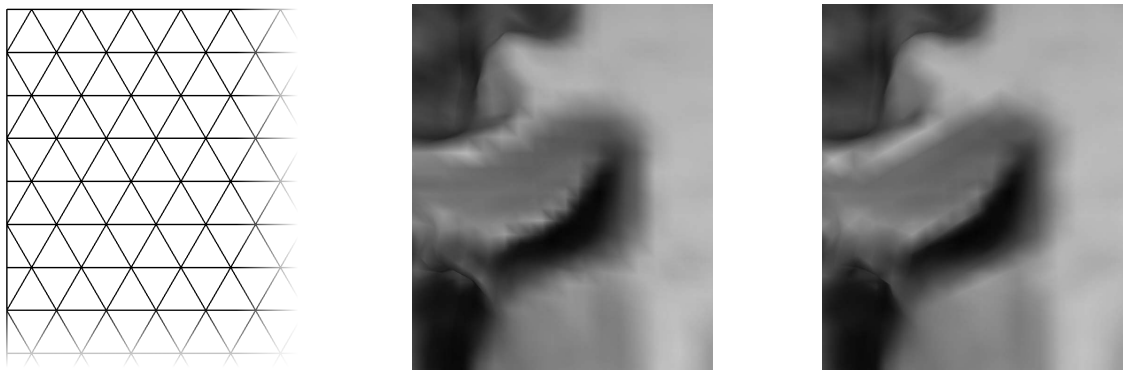
However, this insight only establishes a similarity to volume ray casting, not an equivalence: Our surface deformation is much faster than a full evaluation of the volume rendering integral, since the integration does not have to be performed per-pixel and does not involve a lighting computation in each step. Also, the visual appearance differs, as observed in Figure 9.1 (c): Since we do not perform a compositing, the surface has a clear appearance, while the volume rendering looks more blurry. Moreover, the lighting based on the normal of the surface produces a plastic impression, while the volume rendering has a flat look, which is typical of interfaces to clipping geometry, due to the fact that correct lighting in such areas needs to employ the normal of the geometry rather than the gradient of the volume [238].

9.4.2. Mesh-based Implementation

When the surface is represented as a triangular mesh, it is deformed by considering its vertices as massless particles and letting them move with the flow $\mathbf{v}$, as described by Equation (2.36). We performed numerical integration both using fixed stepsize Euler integration and an adaptive fifth-order Runge-Kutta scheme [168], set for 0.5% accuracy in the final result. In our experiments, both methods achieved similar efficiency and visual results. To speed up the computation, we also implemented a GPU solver similar to the one described by Krüger et al. [124].

In strongly deformed regions, it can become necessary to refine the sampling of the mesh. In order to decide whether an edge should be split, we define an edge weight w. Let $\mathbf{x}_{1,2}$ be the endpoints of an edge with normals $\mathbf{n}_{1,2}$. Then,

$$w = \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{\max(\mathbf{n}_1 \cdot \mathbf{n}_2, 0.5)} \tag{9.2}$$



(a) Grid of equilateral triangles (b) Result with rectangular grid (c) Result with grid in (a)

Figure 9.4.: Compared to a rectangular grid, the sampling scheme in (a) achieves comparable quality with less vertices. (c) uses 14% less vertices than (b)

Taking into account normals results in a smoother shading, while the maximum operator avoids infinite refinement around sharp features. All edges whose weight w is larger than a threshold θ_s are split, in decreasing order of w. Like von Funck et al. [227], we split edges on the undeformed mesh, and integrate the new vertices along the vector field. After five rounds of refinement, we collapse all edges in M' whose weight is below a second threshold θ_c (e.g., $\theta_c = 0.2 \cdot \theta_s$) to remove triangles which are no longer needed.

In our implementation, mesh refinement is performed on the CPU, based on a directed edges data structure [36], and rendering is done with the fixed-function graphics pipeline. However, numerical integration is still accelerated by the GPU. In this case, each vertex is rendered to an off-screen buffer as a point, and the results are read back to change vertex positions in the CPU-based data structure.

9.4.3. Implementation as Height Field

Since we are only considering planar starting geometry in this work and restrict ourselves to deformations in orthogonal direction, the results can alternatively be represented as height values over the original surface. It is possible to store the height field as a texture in graphics memory and to use vertex buffers along with the programmable vertex units of modern GPUs to rapidly transform it into textured geometry [137].

Regular grids simplify implementation, but rectangular grids introduce a bias towards interpolation artifacts along the diagonals. To achieve a homogeneous sampling, we use a grid that consists of equilateral triangles in both the mesh-based and the heightfield implementation (cf. Figure 9.4). In order to represent the heightfield as a rectangular matrix, we use the following formulas.

Given a slice of $n_a \times n_b$ data points that we would like to sample with edge length l, we create a matrix of dimensions $m_a \times m_b$, where

$$m_a = \left\lceil \frac{n_a - 1}{l} + \frac{1}{2} \right\rceil + 1 \qquad m_b = \left\lceil \frac{n_b - 1}{\sqrt{3/4} \cdot l} \right\rceil + 1$$

Slice Size	Vertices	fps GPU	fps CPU
93×116	49 648 / *27 120*	36.2 / *13.3*	6.0 / *7.3*
256×156	103 680 / *44 521*	18.5 / *7.8*	2.6 / *3.7*
256×256	170 112 / *66 998*	11.7 / *6.7*	1.7 / *2.4*

Table 9.1.: Surface deformation and rendering can be done in real time. Results of the mesh-based approach are in italics

Then, the coordinates (x, y) corresponding to matrix entry (i, j) are given by

$$x = \underset{[0,n_a-1]}{\text{clamp}} \begin{cases} l \cdot (i - 0.5) & \text{if } j \text{ even} \\ l \cdot i & \text{if } j \text{ odd} \end{cases}$$
$$y = \underset{[0,n_b-1]}{\text{clamp}} \sqrt{3/4} \cdot l \cdot j$$

If $m_a \neq \lceil (n_a - 1)/l \rceil + 1$, this formulation leads to degenerate triangles along the right boundary, which could be removed by discarding entries $(m_a - 1, j)$ for odd j. However, given that they do not contribute to the final rendering, it is safe to ignore them.

Table 9.1 compares the performance of both implementations on a 2 GHz Athlon 64 with a GeForce 6600 graphics board. Reported timings include surface deformation to integration time $t = 10$ (which was used for all presented examples), normal estimation, and rendering on a 1300×1000 viewport. Results of the mesh-based approach are given in italics, those of the height field are in normal print. In the GPU-based implementation, the user may change the starting geometry and observe the modified final result in real time.

Since the height field implementation currently does not involve a refinement step, it has to employ a much finer sampling to avoid shading artifacts. However, our results indicate that the overhead from remeshing and GPU–CPU communication clearly outweigh the reduced vertex count when the GPU is used for integration. When all processing is done on the CPU, using an adaptive mesh is slightly more efficient.

9.5. Implementing the Individual Stages

9.5.1. Finding a Suitable Cutting Plane

The initial cutting plane should be well-aligned with the streamlines to provide good context. Moreover, physicians are very much used to looking at axis-aligned slices, so cutting planes which are close to an axis-aligned view appear more natural to them and should be preferred.

Given an evenly-spaced discretization of the streamlines, we can reduce them to their vertices to reduce the problem of finding an initial cutting plane to the standard task of fitting a plane to a point cloud. However, the given streamlines will not in general be closely aligned to a plane, so we need to employ a robust estimator which is tolerant

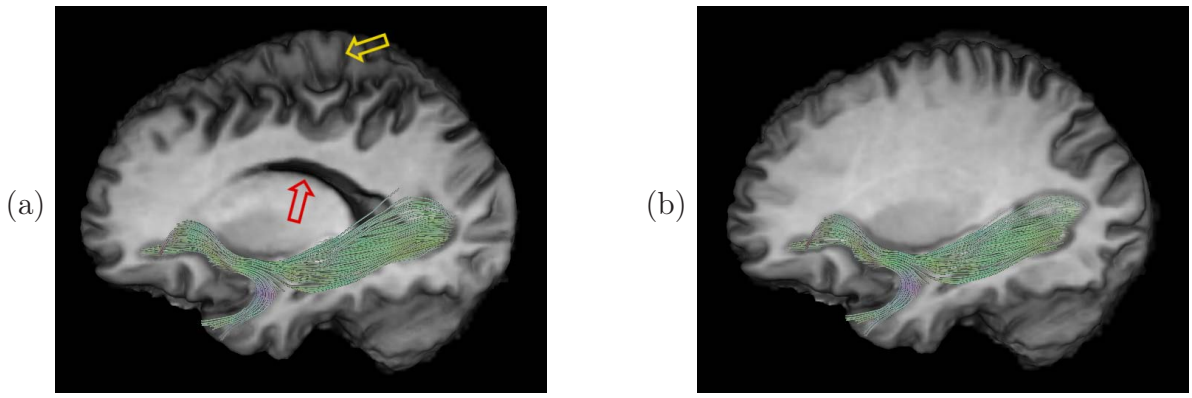


Figure 9.5.: Our robust estimator (b) is less likely to propose unusable views than a simple least squares approach (a).

against gross outliers. In computer vision, random sample consensus (RANSAC) [73] is a popular tool for such tasks. It repeatedly uses a minimum set of random samples to parameterize the model (i.e., three points in case of a plane), estimates the quality of the fit by counting the number of points that are within a predefined distance to the resulting model, and stores the best result. Once a sufficiently good initial estimate has been found, the least squares problem is solved on the inlier points only.

This simple procedure relies on the fact that it will sooner or later draw three inliers from the point cloud and can use the resulting plane to filter out outliers. Moreover, it easily allows us to integrate the preference for axis-aligned views by evaluating planes by a score $S = C \cdot \max_i |\mathbf{n} \cdot \mathbf{e}_i|$, where C is the number of points near the plane, $\mathbf{n}$ is the surface normal, and $\mathbf{e}_i$ are the axes. To ensure interactive response, we let RANSAC run for a fixed period of time (0.5 s) and use the best result so far.

Figure 9.5 presents an example where our modified robust estimator is crucial for obtaining a useful result. In Subfigure (a), a tractography of the inferior fronto-occipital fasciculus and the uncinate fasciculus is shown in context of a plane which has been chosen based on a simple least squares fit on all vertices. The tract is visible, but an expert would find the specific plane, which cuts diagonally through parts of the ventricle (red arrow) and through a part of the opposite hemisphere (yellow arrow) confusing rather than helpful. In contrast, the result of our robust estimator in Subfigure (b) is more closely aligned to a standard sagittal view. If the user is still not entirely satisfied with the suggested plane, she may move and rotate it manually.

9.5.2. Deforming the Geometry

Deformation of the original cutting plane follows two goals: First, the surface should be retracted when streamlines are in its vicinity. This is akin to the way an anatomist would follow the course of a fiber tract in a Klingler dissection: It reveals fibers which would otherwise be occluded and introduces dents in the surface which give visual cues about the immediate proximity between T_1 data and streamlines. Second, a surface whose curvature

enhances the appearance of features in the anatomical data is preferred over a flat one.

To pursue the first goal, a voxel-wise streamline density ρ is derived from the given tracts. It is approximated by normalizing the length of each streamline segment by the volume of one voxel and counting it towards the density of the voxel which contains the midpoint of the segment. The resulting field is convolved with a narrow Gaussian kernel to ensure a smooth deformation. Resistance $r = 1$ should be assigned to $\rho(u) = 0$, while r should tend to zero for $\rho(u) \to \infty$. This is accomplished by taking the difference of unity and the scaled arc tangent of $\rho(u)$.

A second transfer function g is used to take influence of the T_1 data into account. Our implementation lets the user define g as an arbitrary piecewise linear function of scalar value $f(u)$, but all demonstrated results use a simple linear mapping of f to $[0, 1]$. The combined resistance $r(u)$, which is used to deform the surface within the framework of Section 9.4, allows it to move when either of the individual terms indicates low resistance:

$$r(u) = g\left(f(u)\right) \cdot \left[1 - \frac{2}{\pi} \arctan\left(\lambda \cdot \rho(u)\right)\right] \tag{9.3}$$

To obtain a meaningful quantity, the absolute streamline density $\rho(u)$ has to be normalized by the seed point density ρ_s, which is an arbitrary parameter in fiber tracking. Additionally, it depends on the distance of the surface to surrounding fibers how much it has to deform to reveal them. The scaling parameter λ takes care of both facts. Consider the streamline vertices in some corridor around the surface and let σ be their standard deviation from the surface. Then, the following choice of λ reveals streamlines within 3σ in areas where $\rho(u) = \rho_s$:

$$\lambda = \frac{1}{\rho_s} \tan\left(\frac{\pi}{2} \frac{3\sigma}{3\sigma + \theta_r}\right) \tag{9.4}$$

If the user is not entirely satisfied with the deformation, she can alter the suggested settings of g and λ. Moreover, the surface can be further deformed interactively, by clicking and moving the mouse over it. In this case, integration is continued with resistance threshold θ_r disabled, but resistance r still in effect. To keep the deformation local, integration time t decreases with distance from the surface point below the mouse pointer. For example, this intuitive tool allowed us to transform Figure 9.6 (a), where the automatic deformation had failed to reveal a small part of a tract in a region of low streamline density, to Figure 9.6 (b), which resolves this problem, within a few seconds.

9.5.3. Distance Cueing

Depth cueing is a standard computer graphics technique which supports depth perception by blending object colors with the background, depending on the distance from the viewer. We adopt this idea to visually connect the streamlines with the T_1 surface by blending them with the surface when they come close to it. This is done by assigning an opacity value which decreases linearly with distance to the surface when it is less than a threshold $\theta_d = 5\,\text{mm}$.

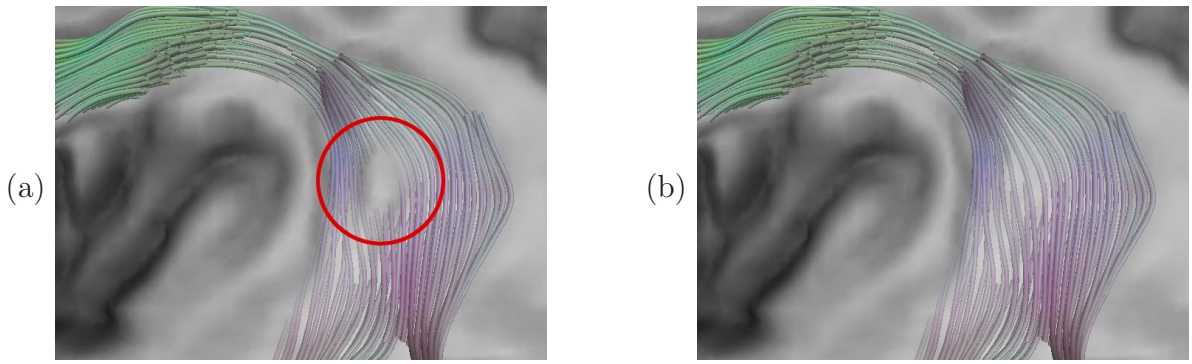


Figure 9.6.: In some cases, the T_1 surface may occlude part of a fiber tract even after deformation (a). Such problems are easily resolved by a local, interactive deformation (b).

The approximate distance of a point to the surface is found by computing the orthogonal distance to the initial plane and subtracting the interpolated height value from the height field representation. If a mesh representation is used, it is first converted into a height field by rendering it to an off-screen buffer and reading back the OpenGL depth buffer.

9.6. Results and Discussion

The cutting planes suggested by our program have been used without modification in Figures 9.5 (b), 9.6, 9.7, 9.8, and 9.9. To test the robustness of the estimator, we tried it on two highly non-planar inputs. In both cases, it successfully ignored part of the streamlines to produce a result which was at least sensible for the rest of them: For fibers from the corpus callosum, which has a saddle-shaped geometry, the estimator suggested a plane that provides good context for the left half (Figure 9.7) and clearly shows how the tract passes above the ventricle and projects to the cortical surface. In a tractography of both cingulum bundles, the estimator chose the vertices of the right bundle as inliers and generated suitable context for them. In both cases, a least squares approach produced a result near the mid-sagittal plane, at a high distance to almost all streamlines.

Within this project, our experience with the novel visualization metaphor suggested in Section 9.4 has been encouraging. With the resistance threshold θ_r enabled, we have not observed any cases in which the deformed geometry would have introduced artifacts or a false perception of shape, and even though the interpretation as a surface of constant opacity only holds when viewed in perpendicular direction, we found that the surface can be rotated without losing its expressiveness. For example, Figure 9.8 shows that changing the viewpoint can help to understand how the cingulum bundle runs over the top of the corpus callosum (red arrow). Note that such an exploration is not limited to the framerates reported in Table 9.1, since it does not involve a re-integration. For direct comparison, Subfigure (c) presents a clipped volume rendering from the same viewpoint.

The resistance term suggested in Equation (9.3) proved effective for our application. In

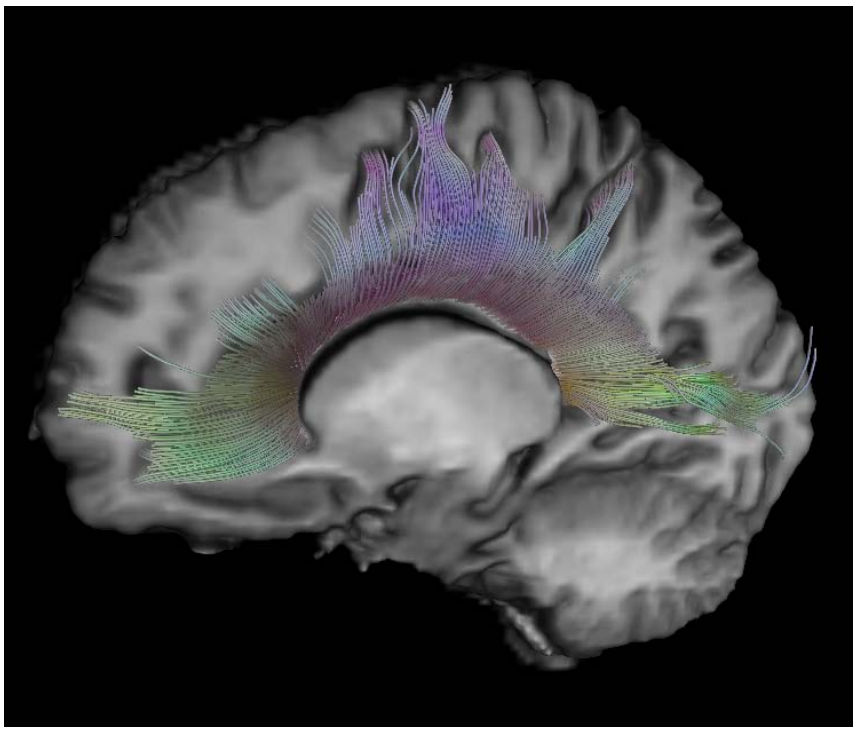
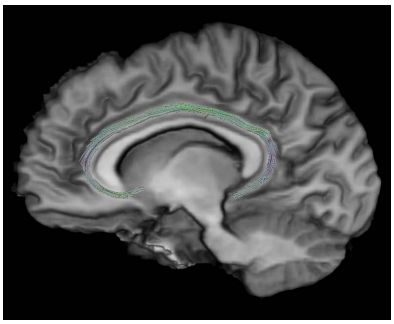
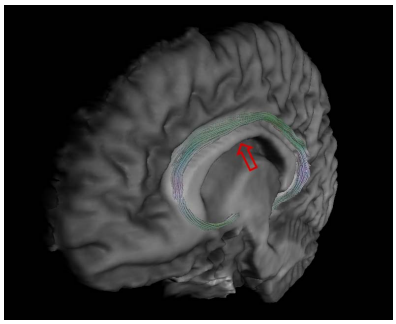


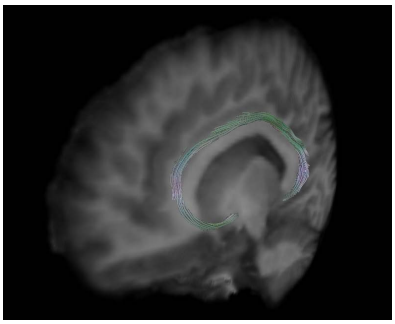
Figure 9.7.: Despite the non-planar shape of the corpus callosum, fitting a cutting plane produces a sensible result.



(a) Frontal view on the cingulum bundle

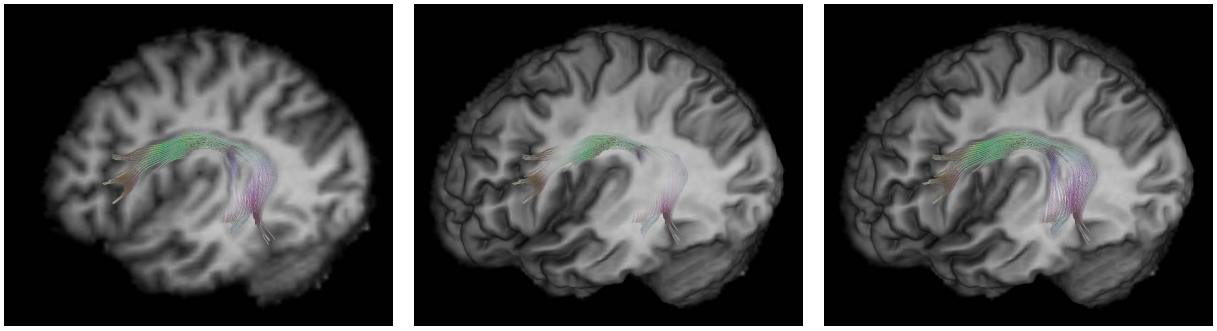


(b) Rotated view on the same surface



(c) Clipped DVR for comparison

Figure 9.8.: Rotating the created geometry interactively (from (a) to (b)) supports the spatial impression. Due to its flat appearance, rotating a clipped volume rendering is less helpful (c).



(a) Deformation based on fibers only

(b) Deformation based on T_1 data only

(c) Proposed combined method

Figure 9.9.: Deforming the cutting plane based on streamline density (a) or T_1 value (b) alone is less effective in conveying their relation than a combination (c).

particular, we found the combination of a streamline-based and a T_1-based resistance to be superior to a deformation which only depends on one of the input datasets. Figure 9.9 illustrates this by showing results based on streamline density (a) and T_1 value (b) only, as well as the proposed combined term (c).

The specified defaults for the scalar transfer function g and the streamline density weight λ were appropriate in most cases. Only in Figure 9.9 (c), they failed to reveal a small portion of the tract, which was easily fixed interactively (cf. closeup in Figure 9.6). Overall, our framework produces usable results automatically, but makes it easy for the user to refine them to her liking.

9.7. Conclusion and Future Work

Understanding the spatial relation of streamlines from DT-MRI to anatomical structures in T_1 data is an important aspect in the visual analysis of fiber tracking results. In this work, we proposed a fused visualization method, in which both types of data interact to reveal their spatial relation. We presented an efficient implementation and results on four different fiber tracts in the human brain. Our results have been validated by a domain expert. Additionally, a physician who is working with DT-MRI data, but not affiliated with our team has confirmed that our renderings convey an impression of the relative fiber positions, while standard methods fail to connect the streamlines to their context.

Part of our contribution lies in the way in which established methods, like RANSAC or depth cueing, are modified and combined to solve a specific problem. However, the data-driven surface deformation, which is at the core of our method, is a novel visualization tool in itself. Despite initial doubts about the appropriateness of creating "artificial" geometry for visualization, the valid and expressive results achieved in our experiments support the arguments which are in favor of such an approach. We are confident that the method from Section 9.4 can help to create more plastic renderings in other situations where the flat appearance caused by clipping geometry may be undesired.

To improve the virtual Klingler dissection further, it would be interesting to consider non-planar starting surfaces to fit the complex geometry of some neuronal pathways even more closely.

10. Higher-Order Tensors for ODF Analysis

Fiber tracking reconstructs the pathways of nerve fiber bundles through an integration process that requires repeated estimation of a principal fiber direction from the measured data. In this dissertation, such tractography methods were used to define topological features (probabilistic tracking, Chapter 3) and to visualize likely nerve fiber pathways directly (deterministic tracking, Chapters 5 and 9).

It is a well-known limitation of the diffusion tensor model that it can only be used to estimate fiber directions in voxels that contain a single predominant bundle. The multimodal orientation distribution functions (ODFs) of q-ball and spherical deconvolution methods offer more flexible models to capture more complex configurations. Prior to our work, fiber tracking in ODF fields used ODF maxima to estimate fiber directions.

The contribution of our work [189] is to point out that in order to get reliable results, extraction of fiber directions from ODFs has to be treated explicitly as an inverse problem. We propose to model estimation of k individual tracts from an ODF as a rank-k approximation of its higher order tensor representation. Based on this model, we develop a practical algorithm that addresses the inverse problem and demonstrate that it produces much better results than simply using ODF maxima.

The chapter is organized as follows: Section 10.1 provides a more detailed motivation, and Section 10.2 discusses related work. Section 10.3 explains our choice of rank-k approximations for addressing the fiber extraction problem and presents the resulting algorithm. Finally, Section 10.4 demonstrates the advantages of our approach both on synthetic and real data, before Section 10.5 summarizes our results and discusses possible directions for future work.

10.1. Motivation

When a voxel contains a single coherent tract, the diffusion tensor model [13] has proven sufficient to infer the dominant fiber direction. Unfortunately, in cases of crossing, touching, fanning or bending fiber configurations, the diffusion tensor may become degenerate (i.e., have two larger eigenvalues of similar magnitude, such that the principal direction is ill-defined) or have an apparently well-defined principal direction which is no longer aligned with any real fiber direction (cf. Figures 10.1 (a) and (b)). Such problems are estimated to affect around one third of all white matter voxels [18].

To gain more information about such voxels, techniques with a higher angular resolution have been proposed. Two of them are q-ball imaging [219] and spherical deconvolution [213], which use a more flexible orientation distribution function (ODF) to indicate multimodal fiber distributions. An overview of these methods and a comparison to the more traditional diffusion tensor model is given in Section 2.2.

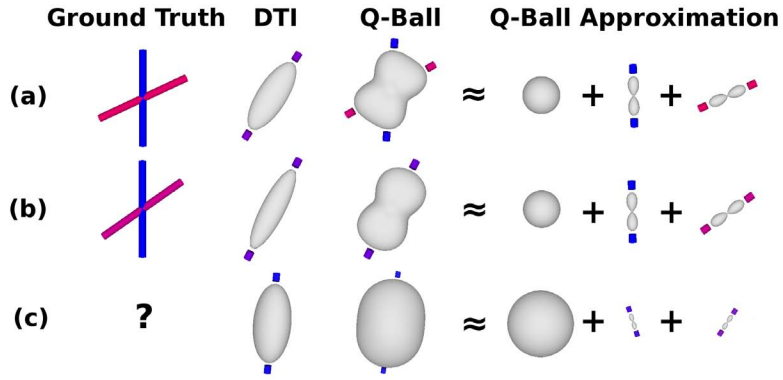


Figure 10.1.: Both DT-MRI and ODF maxima may provide misleading estimates of the underlying fiber structure. Decomposing ODFs in higher-order tensor representation results in much more reliable estimates of fiber crossings.

Orientation distribution functions are continuous functions on the sphere. In many cases, it is desirable to extract discrete principal directions from them. In visualization, the most common application which relies on such directions is fiber tracking [85, 96, 199], but they also play a role in assessing the accuracy of reconstruction schemes [213, 92, 249, 58], as well as for visualization through glyphs [95, 58] and color maps [199].

As a simple ad-hoc solution, it has become common to treat ODF maxima as approximate principal directions. However, when adding peaks of finite width in linear models like q-ball and spherical deconvolution, they will generally interfere. Figure 10.2 uses two simple one-dimensional examples to illustrate the resulting effects: In Figure 10.2 (a), the maxima of the sum are shifted with respect to the original peaks. The distance is marked by the red line segments in the plot on the right. In Figure 10.2 (b), the weaker maximum is masked by the stronger one in the sum.

Consequently, there are many situations in which the number and orientation of ODF maxima deviate from the real tracts: This is demonstrated in Figure 10.1 (a), which shows a q-ball of a synthetic 65° fiber crossing for which inferring principal directions from ODF maxima underestimates the included angle by approximately 10°. Reducing the angle to 55° (Figure 10.1 (b)) even leads to a single maximum, which is not aligned with any of the fiber directions. In this case, the direction estimated from the q-ball profile is not better than the one from traditional DT-MRI. Such effects are known [213] and have been demonstrated in high detail [249].

Our work is founded on the observation that extracting individual fibers from an ODF is an inverse problem of its own: Given the sum of an unknown number of peaks, the task is to obtain a set of peaks whose sum approximates the observed ODF. Clearly, this problem does not have a unique solution. For example, from the ODF in Figure 10.1 (b) alone, it is not clear whether the voxel contains two (or more) crossing tracts or only a single one which causes a broad peak, e.g., because of fanning structures. On the other hand, it has been demonstrated that even a multimodal ODF does not necessarily indicate multiple tracts, but may be due to strong bending [219].

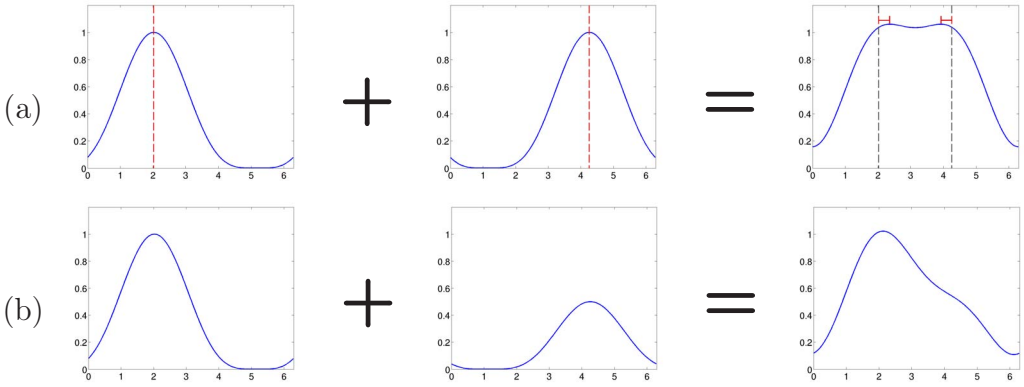


Figure 10.2.: Adding peaks of finite width may shift (a) or mask (b) original maxima.

In this chapter, we demonstrate that even though this problem has no unique solution, a plausible one can be found by making some simple assumptions about the number and shape of single fiber peaks. In particular, we address the case of few crossing bundles, which are assumed to each cause a narrow peak in the ODF. We do this by converting the spherical harmonic series which is commonly used to describe ODFs into a higher-order tensor representation and approximating it with a set of rank-1 tensors, which model the individual peaks.

In synthetic test cases, our method yields more accurate results than a simple maximum extraction (cf. Figures 10.1 (a) and (b)). In real data, it provides plausible and reproducible results. For example, Figure 10.1 (c) presents a q-ball from a voxel in which transcallosal fibers are known to intersect the corona radiata [85]. Unlike a maximum search, our method estimates two fiber bundles, whose orientations agree with the expected directions.

10.2. Related Work

Several previous works [2, 220, 123, 20, 159, 18] have used k-tensor models to resolve crossing fiber configurations. Similarly, Bayesian inference was used to estimate the parameters of a mixture of Bingham distributions [108]. In both cases, the measured signal is modeled directly, using several second-order tensors. Our work differs from this conceptually in that we extract discrete directions from pre-computed q-ball or spherical deconvolution models. Moreover, we build on different mathematical methods, since we use rank-k *higher-order* tensors to approximate existing models. In practice, our approach proved to deal reliably even with three-fiber crossings (cf. Sections 10.4.3 and 10.4.5), while no k-tensor models have been presented which can estimate more than $k = 2$ crossing tracts.

Our method is based on a higher-order tensor decomposition. In the context of diffusion tensor processing, higher-order tensors have previously been analyzed by Basser and Pajevic [14] using spectral decomposition and by Kindlmann et al. [113] based on invariance gradients and rotation tangents. However, both works are concerned with fourth-order covariance tensors that arise from the DT-MRI model and do not address HARDI.

Part of the related work was developed in parallel and was not available at the time we submitted our own approach [189]: The work by Bloy and Verma [26] is similar to ours in that it uses higher-order tensors to model ODFs. However, it employs a different generalization of eigenvectors to higher-order tensors, namely, the Z-eigenvectors from Equation (2.14). This work essentially describes an alternative way to compute the stationary points of the ODF, so it shares the fundamental problems of maximum extraction that will be discussed in the remainder of this chapter.

The method by Yeh et al. [247] does not employ higher-order tensors, but is similar to our own approach algorithmically: In both cases, fiber tracts that have been identified are subtracted out for further analysis. Unlike their approach, our algorithm iterates the optimization once the desired number of tracts has been found. Our experiments indicate that this step improves results even further. However, the extended abstract [247] does not provide enough information to reproduce the described experiments and to perform a more detailed comparison.

Jayachandra et al. [101] have proposed to treat fourth-order diffusion tensors as endomorphisms on the space of second-order tensors. They use an approach equivalent to the one in [14] to find six eigenvalues and corresponding second-order "eigentensors" of these linear maps and claim that each of them represents a separate fiber compartment. However, the only fiber tracking experiment presented in [101] reconstructs a tract that was previously found by standard tractography on second-order tensors. When we implemented the method and tested it on synthetic data with a single dominant direction, the principal eigenvector of the largest eigentensor indeed coincided with the direction of the tract. However, we were not able to resolve the simple 90° two-fiber crossing shown in Figure 2.7, and [101] does not present any quantitative results that would demonstrate the ability of this method to estimate crossing fiber tracts.

10.3. Tensor Approximation for Inferring Fiber Directions

Section 10.3.1 will introduce the fundamental idea of our approach, and the formal model used to separate individual fiber contributions. Then, Section 10.3.2 describes how to find rank-1 tensor approximations, which is an important task within our method. Finally, Section 10.3.3 provides the complete algorithm and a listing in pseudocode.

10.3.1. Rank-1 Tensors as Fiber Terms

The homogeneous form $T(\mathbf{v})$ of a symmetric order-l rank-1 tensor

$$\mathcal{T}(s, \mathbf{u}) = s \cdot \underbrace{\mathbf{u} \otimes \cdots \otimes \mathbf{u}}_{l \text{ terms}} \tag{10.1}$$

defined from a scalar s and a real unit vector $\mathbf{u}$ provides a sharp, non-oscillating, non-negative, axially symmetric peak of height s at $\mathbf{u}$. Its sharpness grows with order l, reflecting the higher angular resolution provided by tensors of higher order. We choose symmetric rank-1 tensors as suitable and computationally convenient models of the narrow

peaks into which we would like to decompose the ODF. It is safe to make this choice, since other analytic, heuristic, or empirical models of a single fiber ODF can be converted to the assumed rank-1 tensor model in a simple preprocessing step. Examples of this will be given in Sections 10.4.1 and 10.4.2.

The best rank-1 approximation $\widetilde{\mathcal{T}}(s, \mathbf{u})$ of a tensor $\mathcal{T}$ in the sense of minimizing the residual norm $\|\mathcal{T} - \widetilde{\mathcal{T}}(s, \mathbf{u})\|$ is given by the vector $\mathbf{u}$ which maximizes the absolute value of its homogeneous form T and the scalar $s = T(\mathbf{u})$ [53]. For second-order tensors, this corresponds to the principal eigenvector and -value. In this sense, using rank-1 tensors as fiber models is a generalization of the traditional way of interpreting second-order diffusion tensors. In the higher-order case, the current practice of selecting ODF maxima can be interpreted as finding an optimal fit for each tract independently from the others, and the bias caused by this in case of crossing fibers can be considered a consequence of the fact that the sum of k non-orthogonal, locally optimal rank-1 approximations generally does not yield an optimal rank-k approximation [121].

The idea behind the proposed method is to improve a fiber estimate by refining the corresponding rank-k tensor approximation of the ODF. This takes into account the interference between non-orthogonal peaks, which is ignored by simple maxima extraction. Unfortunately, there are no methods which find the best rank-k approximation for tensors of order $l > 2$, and there even exist tensors which do not possess an optimal rank-k approximation (cf. Section 2.1.7). Moreover, for tensors over $\mathbb{R}^3$, no algorithms exist to compute the canonical decomposition into a minimal number r of rank-1 terms (cf. Section 2.1.6).

Despite these obstacles, we demonstrate that even potentially suboptimal rank-k approximations allow for a much improved interpretation of ODFs, and we propose a stable and efficient algorithm to compute them.

10.3.2. Finding Rank-1 Approximations

Computing the best rank-1 approximation to a higher-order tensor is equivalent to maximizing the absolute value of its homogeneous form on the unit sphere. For this task, there exists a generalization of the power method for finding the largest eigenvector of a matrix [53]. Unfortunately, its supersymmetric variant is only guaranteed to converge for tensors whose induced homogeneous form is either non-negative or non-positive [121]. While q-ball ODFs are non-negative by definition, neither the residuals which occur in our algorithm nor results from spherical deconvolution have this property.

Therefore, we employ a gradient descent technique with Armijo stepsize [7], as described in [229]. Even though a similar fixed stepsize algorithm has previously been used for fiber tracking in q-ball data [96], the adaptive stepsize proved critical to guarantee convergence in our experiments.

10.3.3. A Practical Algorithm to Resolve Crossings

The goal of our algorithm will be to approximate the higher-order tensor representation of the ODF with an optional isotropic part and several rank-1 terms which represent individual fiber peaks. A small approximation residual will account for noise and factors outside

the model. The isotropic part is needed in case of q-ball data to capture the "ambient" part of the diffusion, which cannot be assigned to any specific fiber compartment. For spherical deconvolution data, it may be omitted.

The proposed algorithm works iteratively: Initially, the full input ODF is assigned to the residual $\mathcal{R}$. Any subsequent step reduces the residual norm $\|\mathcal{R}\|$, until convergence is reached. The isotropic part of $\mathcal{R}$ is found by computing the mean of its homogeneous form, $\mu = \text{mean}(R)$, and multiplying it with the isotropic tensor $\mathcal{I}$ whose homogeneous form is identically one. Since the spherical harmonic Y_0^0 is a constant function, closed formulas for both μ and $\mathcal{I}$ are found from the matrix that relates spherical harmonic coefficients to tensor components (cf. Appendix A.3).

The key part of the problem is to find a rank-k approximation of $\mathcal{R}$, where k is the desired number of fiber tracts $\mathcal{F}_i$. For this, our algorithm does an iterative local optimization, which repeatedly optimizes each rank-1 term, while keeping all others fixed. When adding a new term, this involves a rank-1 approximation $\text{approx}(\mathcal{R})$ of $\mathcal{R}$, as described in Section 10.3.2. To improve an existing rank-1 term $\mathcal{F}_i$, an approximation of $\mathcal{R} + \mathcal{F}_i$ is sought. In this case, we start a gradient descent at the previous optimum $\mathbf{u}$, denoted $\text{refine}(\mathbf{u}, \mathcal{R} + \mathcal{F}_i)$. This procedure is repeated until the residual norm no longer changes significantly. The final result may not be an unconstrained optimum, but is locally optimal in the sense that it cannot be improved by more than some small ϵ by changing any of the terms individually. Experimental results indicate that this weak notion of optimality is sufficient to provide a remarkable enhancement over simple maximum extraction, which does not take into account interference between fiber terms at all.

It is non-trivial to decide how many fibers one should look for in a given ODF. Previous authors [3, 108, 18] have approached this with computationally costly statistical methods. To allow for a faster tracking process, our algorithm employs a simple heuristic that was found effective in noisy synthetic data: A rank-$(k+1)$ approximation is accepted if it reduces the residual norm to at least $\theta_{\text{norm}} \in [0, 1]$ times the residual norm of the rank-k approximation (i.e., its contribution to explaining the ODF in terms of fiber peaks is reasonably large) and if the ratio $\max_i(\|\mathcal{F}_i^{(n)}\|)/\min_i(\|\mathcal{F}_i^{(n)}\|)$ of the largest to the smallest fiber term is below a second threshold θ_{ratio} (i.e., the smallest fiber term is not assumed to be noise). Otherwise, we output the previous rank-k approximation and terminate the algorithm.

Based on the experiments in Section 10.4.4, we set $\theta_{\text{ratio}} = 4$ for the transition from a single-fiber to a two-fiber model and $\theta_{\text{ratio}} = 3$ for going from two to three fibers. θ_{norm} was fixed empirically at $\theta_{\text{norm}} = 0.9$ for synthetic data, $\theta_{\text{norm}} = 0.98$ for real data. Additionally, the user may specify a maximum number $F_{\max}$ of fiber tracts. For easier reproduction, Table 10.1 summarizes the algorithm as annotated pseudocode.

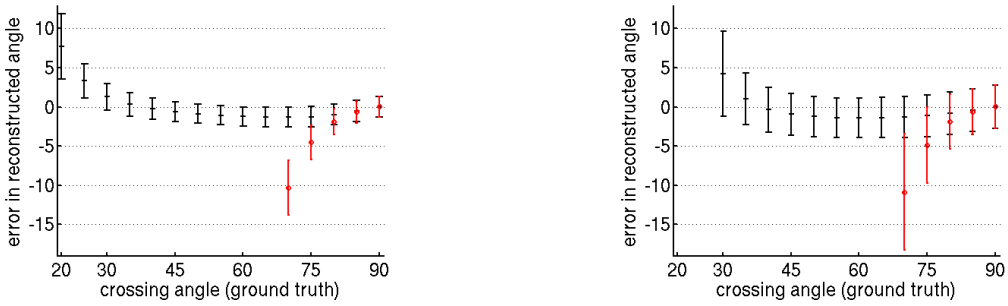
10.4. Results

First, we will concentrate on experiments on synthetic data, which allow us to validate our method against ground truth. On synthetic data, we contrast q-ball reconstruction (Section 10.4.1) with spherical deconvolution (Section 10.4.2), and confirm that our algo-

	INPUT AND PARAMETERS
$\mathcal{O}_{\text{in}}$	orientation distribution function to be analyzed
$k_{\max}$	maximum number of fibers to extract
	OUTPUT
$\widehat{\mathcal{F}}_i$	final fiber term i
	IMPORTANT VARIABLES
$\mathcal{F}_i^{(n)}$	fiber term i at inner iteration n
$\mathbf{u}_i^{(n)}$	vector part of $\mathcal{F}_i^{(n)}$
k	current number of fiber terms
$\mathcal{R}, \mathcal{R}^{(n)}$	current residual, residual before inner iteration n
	CONSTANTS
$\mathcal{I}$	isotropic tensor (homogeneous form identically one)
ϵ	small scalar value greater than zero

$\mathcal{R} := \mathcal{R}^{(1)} := \mathcal{O}_{\text{in}}$; $k := 1$; $n := 1$;
`repeat` // outer iteration
  `repeat` // inner iteration
    `if` $\mathcal{O}_{\text{in}}$ `is a q-ball ODF then`
      $\mathcal{R} := \mathcal{R} - \text{mean}(R) \cdot \mathcal{I}$; // improve isotropic estimate
    `end`
    `for` $i := 1 \ldots k$ `do` // improve all k fiber terms
      `if` $\mathcal{F}_i^{(n)}$ `is defined then`
        $\mathcal{F}_i^{(n+1)} := \text{refine}(\mathbf{u}_i^{(n)}, \mathcal{R} + \mathcal{F}_i^{(n)})$;
        $\mathcal{R} := \mathcal{R} + \mathcal{F}_i^{(n)} - \mathcal{F}_i^{(n+1)}$;
      `else`
        $\mathcal{F}_i^{(n+1)} := \text{approx}(\mathcal{R})$;
        $\mathcal{R} := \mathcal{R} - \mathcal{F}_i^{(n+1)}$;
      `end`
    `done`
    $n := n + 1$; $\mathcal{R}^{(n)} := \mathcal{R}$;
  `until` $\|\mathcal{R}^{(n)}\| > (1 - \epsilon) \cdot \|\mathcal{R}^{(n-1)}\|$ // test convergence
  `if` accept$(\mathcal{F}_i^{(n)})$ `then` // accept k fiber terms
    `for` $i := 1 \ldots k$ `do` // store output candidates
      $\widehat{\mathcal{F}}_i := \mathcal{F}_i^{(n)}$;
    `done`
    $k := k + 1$;
  `end`
`until` $k > k_{\max}$ `or not` accept$(\mathcal{F}_i^{(n)})$

Table 10.1.: Pseudocode of the proposed algorithm for estimating fiber tracts. All function names are explained in the text.



(a) SNR_0=40 (b) SNR_0=20

Figure 10.3.: In a q-ball model ($l = 4$) of a two-fiber crossing, our approximation (black) estimates fibers with less bias and over a far wider range than maximum extraction (red).

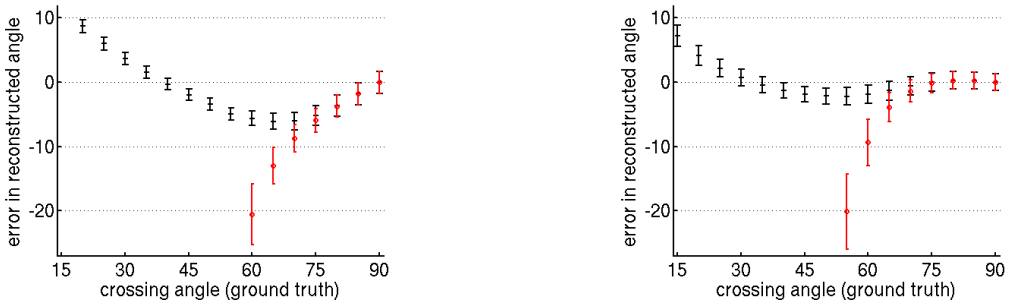
rithm reliably reconstructs three-fiber crossings (Section 10.4.3) and remains stable under varying volume fractions (Section 10.4.4). Then, we turn to a real dataset, and demonstrate that tensor approximation gives clear advantages over maximum extraction for fiber tracking (Section 10.4.5).

10.4.1. Synthetic Q-Ball Data

We have generated synthetic diffusion-weighted MRI measurements $S(\mathbf{g})$ according to Equation (2.20), with an apparent diffusivity function $d(\mathbf{g})$ that results from two Gaussian compartments with equal volume fractions and fractional anisotropy FA = 0.87. This model is commonly used to simulate crossing fiber populations for validation purposes [219, 92, 249]. As measurement parameters, we selected 60 gradient directions based on electrostatic repulsion [107] and a b-value of $b = 3\,000\,\mathrm{s/mm}^2$ (as in [213, 92, 58]).

Rician noise at two levels ($\mathrm{SNR}_0 = 40$ and $\mathrm{SNR}_0 = 20$) was added to the signals $S(\mathbf{g})$. Note that SNR_0 refers to the signal-to-noise ratio in the unweighted data. The resulting SNR in diffusion-weighted images is lower and depends on the exact gradient direction and fiber setup. For example, in a simple one-fiber experiment, $\mathrm{SNR}_0 = 40$ leads to an effective SNR between 22 (perpendicular to the fiber) and 0.25 (along the fiber). Starting from 90°, we decreased the angle between the simulated tracts in steps of 5°, until our method failed to resolve them reliably. For each fiber configuration and each noise level, we took 1 000 samples.

In a first experiment, we estimated q-balls of order $l = 4$ from the synthetic data, using the analytic solution of the Funk-Radon transform and Laplace-Beltrami regularization with smoothing parameter $\lambda = 0.004$ [58]. For comparison with the approximation results, we computed the discrete maxima of the resulting ODFs using a refined icosahedral tessellation of the sphere and improved their accuracy via gradient descent (cf. Section 10.3.2). In case of more than two maxima, we selected the largest ones. In the approximation, we set $k_{\max} = 2$. The included angle of the estimated fibers was computed and compared to the ground truth.



(a) before sharpening

(b) after sharpening

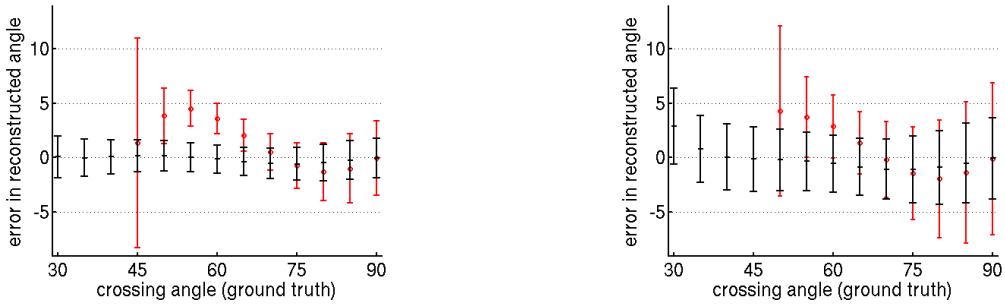
Figure 10.4.: Larger bias in a q-ball model of order $l = 6$ indicates that rank-1 tensors are a less suitable peak model. A heuristic sharpening reduces the problem.

Figure 10.3 presents mean and standard deviation of the reconstruction error for maximum finding (red, circles) and tensor approximation (black, crosses). As previously reported [249], maxima exhibit a clear bias starting at around 80°, which is greatly reduced in the approximation results. Moreover, the approximation reliably reconstructs crossings far beyond the point at which the individual maxima have merged. This can be explained by the similarity of the approximation to a deconvolution: Effectively, the approximation finds a discrete set of delta distributions whose convolution with the kernel defined by the homogeneous form of a rank-1 tensor approximates the q-ball.

The effective angular resolution of analytic q-ball imaging is affected both by the measurement process and the reconstruction [92]: The angular resolution of diffusion measurements is limited by the employed b-value. Additionally, truncating the spherical harmonic expansion introduces a point spread function in the reconstruction, which becomes narrower for increasing order l. If the latter effect dominates, it should be appropriate to decrease the width of the assumed fiber peak, as is done when modeling it as an order-l rank-1 tensor. Unfortunately, the results on q-balls with order $l = 6$ ($\mathrm{SNR}_0 = 40$) in Figure 10.4 (a) suggest that this is not the case: Compared to Figure 10.3 (a), the bias of the approximation is much higher, indicating that an order-6 rank-1 tensor is a less suitable model of a single fiber peak than the wider order-4 rank-1 tensor.

As proposed in Section 10.3.1, we can reduce this bias by mapping a more suitable model M^s to the model M^a assumed by our algorithm. This is done by deconvolving the ODF with M^s, followed by a convolution with M^a. In the spherical harmonic basis, this is a simple operation: An axially symmetric model M is described by a single scalar value m_h per harmonic order h and the result of convolving it with a spherical harmonic series with coefficients c_j is given by $c'_j = m_h \cdot c_j$, where h is the harmonic order of c_j (cf. Appendix A.3).

As a proof of concept, we let M^s be the order-4 rank-1 tensor, which gave usable results in the previous experiment. M^a is the order-6 rank-1 tensor. Then, the values m^a_h and m^s_h are found as the spherical harmonic coefficients c_j of the corresponding rank-1 tensors. The resulting conversion factors $m_h = m^a_h / m^s_h$ are given as $m_h = [5/7, 5/6, 105/77, 1]$ for $h = [0, 2, 4, 6]$, respectively. Since $m^s_6 = 0$, we do not perform a deconvolution on harmonic



(a) SNR_0=40 (b) SNR_0=20

Figure 10.5.: Compared to Figure 10.4 (b), spherical deconvolution with order $l = 6$ reduces bias of the approximation method (black) further.

order $h = 6$. Figure 10.4 (b) shows that this heuristic sharpening reduces the bias both for the approximation and for maximum finding.

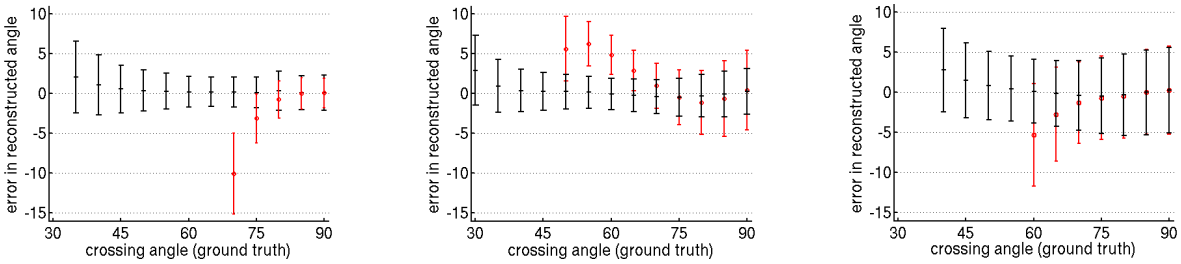
From this experiment, we conclude that reliably extracting fibers from q-ball ODFs generally requires explicit modeling of the single fiber response, e.g., via an additional deconvolution step. Since using an empirical estimate of the appropriate deconvolution kernel (as in [58]) would be conceptually very similar to spherical deconvolution, we will concentrate on the latter technique in the remainder of this work.

10.4.2. Spherical Deconvolution

The similarity between tensor approximation and spherical deconvolution which has been discussed above does not imply that one could replace the other. Existing methods for spherical deconvolution yield a continuous ODF, and we will show that approximating it by rank-1 terms has advantages over taking maxima when discrete directions are desired. On the other hand, it has been emphasized that continuous fiber distributions hold more information than just the principal directions (e.g., evidence on the amount and orientation of fiber spread) [213] and there have been initial attempts to exploit it [199].

For our experiments, we have implemented spherical deconvolution as described in [213]. When setting the response function R, one explicitly specifies the shape of the peak that will result from the deconvolution of a training sample. For this, Tournier et al. use a truncated spherical harmonic representation of a delta distribution (cf. appendix of [212]). As an alternative, we employed the peak described by a rank-1 tensor, whose spherical harmonic coefficients c_j are found as above.

Since they are non-oscillating, rank-1 peaks give smoother ODFs. Their non-negativity reduces the undesired non-physical negative lobes. They also produced best results in combination with our tensor approximation, which explicitly assumes the rank-1 model. When finding ODF maxima, the optimal peak shape depended on tensor order: At order $l = 6$, the ringing in the truncated delta peak leads to a bias which oscillates for varying crossing angles, but remains tolerable over a wider range. For order $l = 4$, the ringing is so strong that a rank-1 tensor gives more accurate results also for maximum finding.



(a) order $l = 4$, SNR=40 (b) order $l = 6$, SNR=40 (c) order $l = 6$, SNR=20

Figure 10.6.: Three-fiber crossings are reliably resolved by decomposing spherical deconvolution ODFs, even at low orders and under higher levels of noise.

To be as fair as possible to both methods, presented results use rank-1 peaks for maximum extraction at $l = 4$ and for tensor approximation, but truncated delta peaks for maximum finding at $l = 6$. Figure 10.5 presents results from spherical deconvolution of the same data as above, with $l = 6$. Again, the approximation (black) exhibits less bias and is applicable in a wider range than maximum extraction (red).

10.4.3. Three-Fiber Crossings

Resolving crossings of three fiber bundles has proven difficult for many previous approaches: Tuch [220] reported that k-tensor estimation becomes unstable for $k = 3$. Bergmann et al. [20] discuss general k, but only present results for $k = 2$. Both Kreher et al. [123] and Peled et al. [159] explicitly restrict themselves to the two-fiber case.

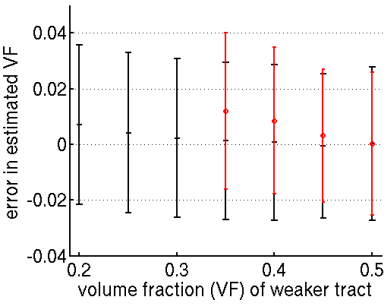
To investigate this more difficult case, we generated data from three Gaussian fiber compartments. The respective principal axes were chosen such that their endpoints form an equilateral triangle and any pair of them includes angle α. Again, α was decreased gradually, starting from 90°. All other measurement parameters were chosen as above. ODFs were reconstructed using spherical deconvolution with orders $l = 4$ and $l = 6$.

Figure 10.6 shows that the rank-k approximation remains stable for $k = 3$ and allows one to resolve three-fiber crossings even at low orders and relatively small angles ($\mathrm{SNR}_0 = 40$). At the higher noise level ($\mathrm{SNR}_0 = 20$, $l = 6$), it remained possible to reliably reconstruct three-fiber crossings down to $\alpha = 40°$ (Figure 10.6 (c)).

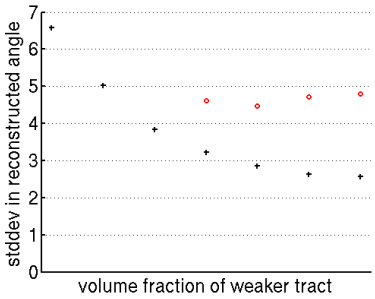
In all our experiments, we have found that under low noise ($\mathrm{SNR}_0 = 40$), models of order $l = 6$ could reliably represent angles which were impossible to reconstruct from noisier data ($\mathrm{SNR}_0 = 20$). From this observation, we conclude that under realistic measurement conditions, noise rather than model complexity will be the limiting factor for angular resolution already at order $l = 6$.

10.4.4. Estimating Volume Fractions and Fiber Number

To test the stability of our method under varying volume fractions, we fixed a two-fiber crossing at 60° and gradually decreased the volume fraction of the weaker tract, until



(a) Error in estimated VF



(b) Error in estimated angle

Figure 10.7.: Under varying volume fractions (VF), ODF approximation (black) reconstructs the weaker tract over a wider range than maximum finding (red).

	$\mathrm{SNR}_0 = 40$			$\mathrm{SNR}_0 = 20$		
	Fiber count estimated by our heuristic					
	1	2	3	1	2	3
1 tract	1000	0	0	998	2	0
2 tracts	0	1000	0	0	999	1
3 tracts	0	0	1000	0	1	999
	Number of above-average ODF maxima					
	1	2	≥ 3	1	2	≥ 3
1 tract	1000	0	0	992	8	0
2 tracts	163	837	0	203	797	0
3 tracts	88	458	454	89	504	407

Table 10.2.: In a fiber detection experiment, our heuristic proved to be a more reliable indicator of fiber number than above-average ODF maxima.

reconstruction failed (spherical deconvolution, $\mathrm{SNR}_0 = 20$, $l = 6$). Both the relative magnitude of maxima and the relative magnitude of the fiber terms, $\|\mathcal{F}_{\mathrm{weak}}\|/(\|\mathcal{F}_{\mathrm{weak}}\| + \|\mathcal{F}_{\mathrm{strong}}\|)$, gave usable estimates of relative volume fractions (cf. Figure 10.7 (a)).

While maximum extraction found the weaker fiber in less than 50% of the cases already for ratio $0.3 : 0.7$, the approximation still worked below $0.2 : 0.8$. However, the standard deviation of the estimated angle became so high that we do not report it for smaller values (Figure 10.7 (b)). A three-fiber crossing at 50° with one dominant tract at ratio $0.6 : 0.2 : 0.2$ was still correctly reconstructed by the approximation in 85% of the cases.

In a final experiment on synthetic data, we validated the heuristic which determines the fiber number in our algorithm (cf. Section 10.3.3). Classification was tested on 1 000 voxels each with a single fiber, two fibers, and three fibers, respectively. Directions were chosen uniformly at random such that all included angles were above 45°, all volume fractions were equal. Under these conditions, our classification proved reliable, even under noise (Table 10.2). For comparison, we specify the number of ODF maxima whose magnitude

is above the ODF mean, which was taken as an estimate of the fiber number in previous works [57].

The estimation of both volume fractions and the number of fiber compartments is more challenging in the presence of fiber spread in real data. However, it is reassuring that the approximation improves upon maximum extraction also in this respect.

10.4.5. Improvement in Fiber Tracking

We tested our algorithm on the real-world dataset described in Section 1.2. At the employed b-value ($b = 1\,000\,\mathrm{s/mm^2}$), only very few crossings are resolved as individual maxima in unsharpened q-ball data, so we used filtered spherical deconvolution for a fair comparison of the approximation method against maximum extraction. As in [213], a DT-MRI model was first estimated and the average fractional anisotropy (FA) of the 300 voxels with highest FA was computed. Based on this, the deconvolution kernel R was set from the synthetic signal of a prolate tensor with the same FA, using noise attenuation vector $\underline{\beta} = [1\ 1\ 1\ 0.6]$ (cf. [213]).

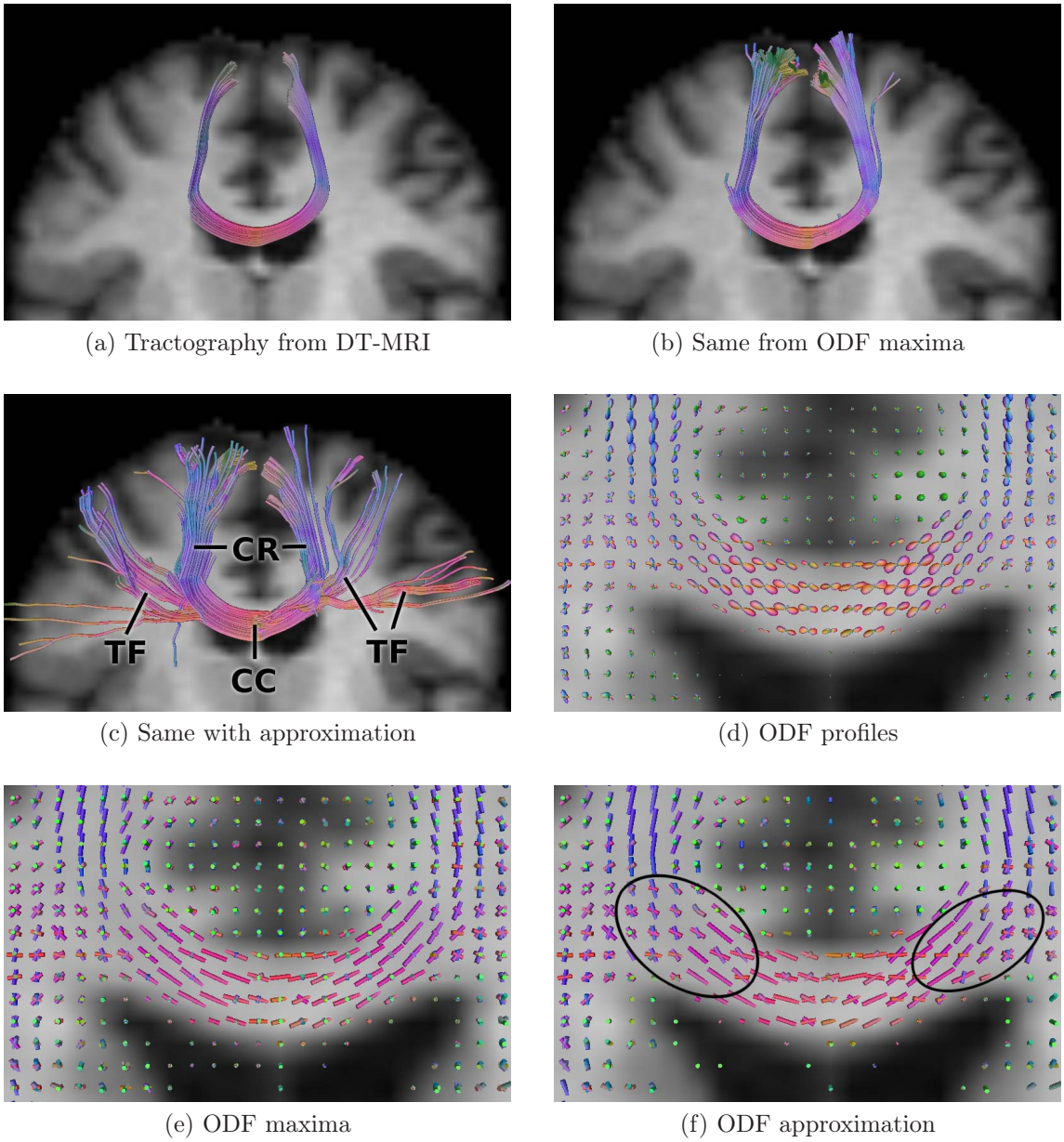
In a first experiment, we tried to track the lateral *transcallosal fibers* (TF) that run through the *corpus callosum* (CC) (cf. Figure 10.8 (c)). It is known that DT-MRI tractography (Figure 10.8 (a)) only captures the dominant U-shaped *callosal radiation* (CR) [228]. The ability to find the lateral fibers with high angular resolution imaging has previously been demonstrated using simple streamline tracking along the most collinear ODF maximum on high b-value diffusion spectrum data [85] and using probabilistic tractography on sharpened q-balls [58].

We have tried to reproduce the tract using a deterministic higher-order tensor tracking algorithm, similar to [95]. However, we additionally allowed for tract splitting in cases where a second maximum was found within 30° of the current tracking direction. Since q-ball estimation, spherical deconvolution, and the conversion between spherical harmonics and tensor coefficients are all linear operations, one may equivalently interpolate diffusion-weighted images, spherical harmonic coefficients, or tensor components. For efficiency, we used component-wise trilinear interpolation of the tensor field before extracting maxima.

With seed points in the *corpus callosum*, the lateral fibers were not found when following ODF maxima (Figure 10.8 (b)). When using approximation results instead, the tract could be reconstructed (c). A visual comparison of maxima (e) to approximation results (f) reveals that the latter separates the diverging transcallosal fibers earlier and reconstructs their crossing with fibers from the *internal capsule* more reliably (cf. ellipses in (f)), which facilitates tracking. In all figures, a slice of co-registered T_1 data is shown for context.

Figure 10.9 confirms that this result can be reproduced along a large part of the *corpus callosum.* Here, only fibers which leave a corridor of $\pm\,20\,\mathrm{mm}$ around the mid-sagittal plane have been colored to visually emphasize the *transcallosal fibers.* Note that the results include a part of the *inferior fronto-occipital fasciculus* (in green, most notable in (a)), which gets connected to the tractography of the *corpus callosum* through partial voluming effects.

The low number of parameters in order-4 models makes them particularly attractive for experiments with a relatively low number of measurements. Due to the limited measure-



(a) Tractography from DT-MRI

(b) Same from ODF maxima

(c) Same with approximation

(d) ODF profiles

(e) ODF maxima

(f) ODF approximation

Figure 10.8.: Tensor approximation (c) allows a deterministic tracking method to reconstruct the *transcallosal fibers* (TF) from seeds in the *corpus callosum* (CC). This is neither achieved by a DT-MRI tractography (a), nor by following maxima in the spherical deconvolution ODFs (b).

(a) Tracts from DT-MRI (b) Tracts from ODF maxima (c) Tracts from ODF approximation

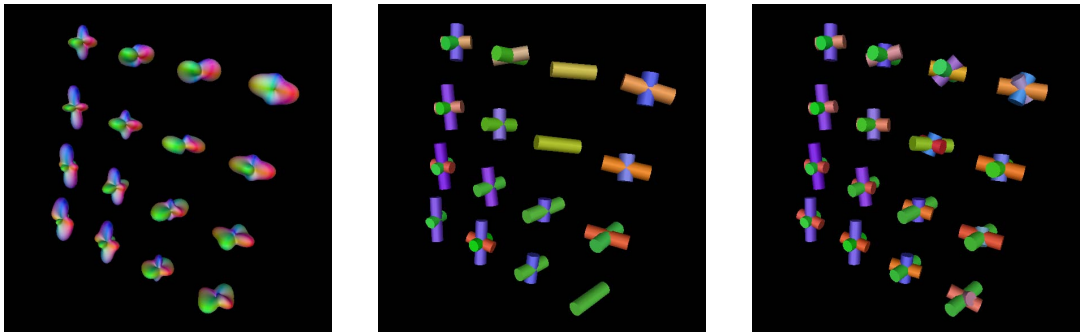
Figure 10.9.: Tractography of the full *corpus callosum*, shown from above. Only the approximation result (c) reliably includes *transcallosal fibers*.

ment time feasible in clinical practice, there has been some interest in resolving crossing tracts in such settings (e.g., [20, 159]). Figure 10.10 illustrates the advantages of tensor approximation for order $l = 4$. It presents 4×4 voxels from a coronal slice of the three-fiber crossing between *cortico-spinal tract* (blue), *transcallosal fibers* (red), and *superior longitudinal fasciculus* (green). Since many peaks have merged (cf. the ODF profiles in (a)), the extracted maxima in (b) miss one or two of the crossing tracts in most voxels. In comparison, the approximation in (c) reconstructs them more reliably, allowing all three tracts to be tracked through the crossing region (e), while the red *transcallocal fibers* are mostly blocked at the crossing when using ODF maxima (d). To avoid visual clutter, tracts have been terminated when leaving a small region of interest.

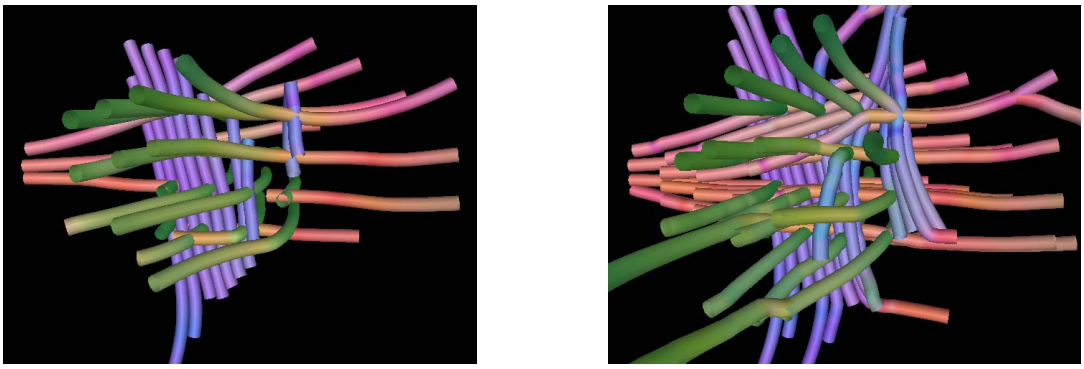
In a final experiment on real data, we have seeded the tractography within the *decussation of the superior cerebellar peduncle* (dscp), a location in the brainstem where parts of the *superior cerebellar peduncle* (scp) cross to the opposite hemisphere. Due to partial voluming, this region also contains a part of the adjacent *corticospinal / corticopontine tract* (cst/cpt). Figure 10.11 compares results of maximum tracking and tensor approximation in a view from posterior / superior. Again, the approximation makes tracking through this complex configuration more reliable. In particular, a part of the cst/cpt is reconstructed.

10.4.6. Efficiency

In our experiments, the runtime of the approximation method was on the same order as the maximum search, despite the better results. For example, decomposing all 254 578 ODFs in the real-world dataset took 10.5 s for $l = 4$, 18.5 s for $l = 6$, while maxima extraction took 9.5 s and 23.9 s, respectively, on a regular 2 GHz workstation. Tracking

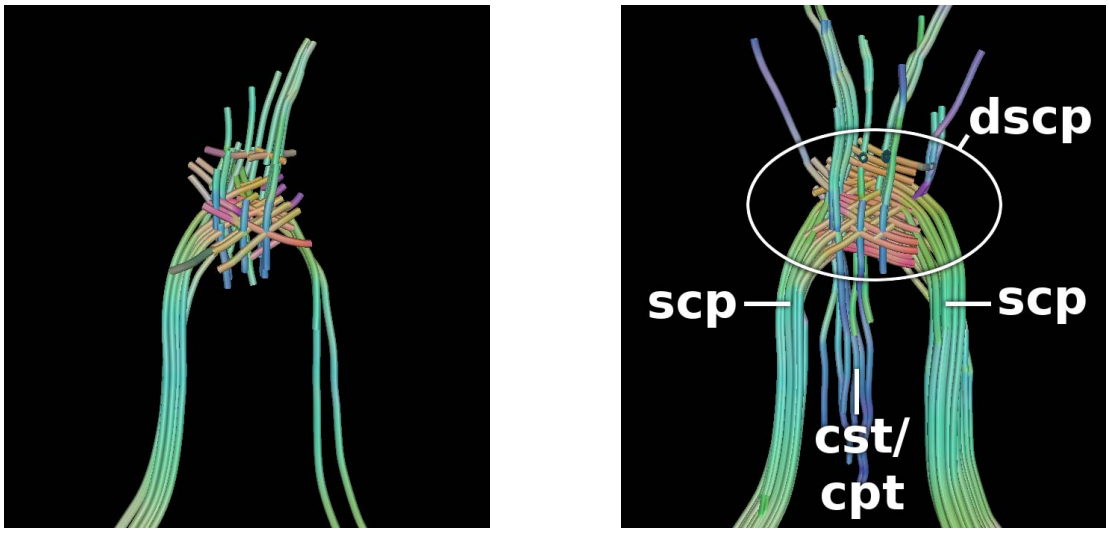


(a) ODF profiles (b) Maxima (c) Approximation



(d) Tracts from maxima (e) Tracts from approximation

Figure 10.10.: Tensor approximation allows for tracking through the triple crossing of *cortico-spinal tract* (blue), *corpus callosum* (red) and *superior longitudinal fasciculus* (green) even at $b = 1\,000$ and $l = 4$.



(a) Tracts from maxima (b) Tracts from approximation

Figure 10.11.: Reliable tracking through the *decussation of the superior cerebellar peduncle* (dscp) requires tensor approximation ($l = 6$).

typically took longer with the approximation, due to the higher number of fibers which were found (e.g., 1.2 s for Figure 10.8 (b) vs. 1.8 s for Figure 10.8 (c)). These results are obtained when taking 321 samples on the hemisphere to extract discrete local maxima and refining them using gradient descent. To initialize the rank-1 approximations, we found a much coarser sampling sufficient, since we are not interested in obtaining all local maxima. In experiments on a synthetic 60° crossing, accuracy hardly increased when using more than 10 samples on the hemisphere for initializing the gradient descent. To leave a safety margin, our current implementation employs 30 isotropic samples.

10.5. Conclusion and Future Work

In many visualization methods for q-ball and spherical deconvolution data, finding the directions of crossing fibers from a continuous orientation distribution function (ODF) is a crucial step. In this chapter, we have shown that taking into account the interference between the signals from different tracts by approximating the ODF as a sum of individual fiber peaks provides estimates of much higher accuracy than the previous practice of extracting maxima.

Previous authors have proposed to go to high harmonic orders (up to $l = 12$) in spherical deconvolution, to reduce bias in fiber estimates, and to represent close tracts by individual ODF maxima. Since this increases the number of model parameters above the number of measurements which are typically available, it requires a non-linear, constrained ODF reconstruction, which involves additional, heuristic parameters [212]. In contrast, we have demonstrated that the angular resolution of q-ball or spherical deconvolution models is not reached when crossing fibers are no longer separated by individual maxima. Rather, a subsequent approximation step reliably reconstructs fibers over a considerable angular range, at greatly reduced bias. In our synthetic experiments, angular resolution was bounded by assumed measurement noise rather than model resolution already for order $l = 6$.

Addressing the problem of separating crossing fibers via tensor approximation has been motivated by the great utility of the eigenvector decomposition in DT-MRI. While higher-order tensors have previously been considered as an alternative to spherical harmonics in the context of diffusion imaging [149, 152, 95], the proposed use of rank-1 tensors and rank-k tensor approximation is new. It provides a useful way to formalize the problem and allowed us to find an efficient algorithm for crossing fiber estimation, which draws on existing techniques for rank-1 approximation. It is our hope that the link between the analysis of orientation distribution functions and recent efforts in multilinear algebra (e.g., [46, 54, 229]) will become even more useful as more results become available in this interesting area of research.

Evaluating the algorithm against ground truth in synthetic data has shown that it resolves crossing fibers more effectively than existing methods and experiments on real data have given plausible results. However, several aspects of the inverse problem in diffusion-weighted imaging remain challenging: In many cases, it is difficult to find the correct number of fibers in a voxel and to tell crossings from diverging or bending configurations.

Moreover, low-rank approximations are affected by noise, since they solve an underdetermined inverse problem. As future work, one might try to derive formal guarantees for the quality of the approximation obtained by our algorithm, to quantify the confidence of fiber estimates, and to explore both coherence over spatial neighborhoods and prior knowledge to increase it.

11. Conclusion

11.1. Contributions

This dissertation has contributed a variety of methods for the feature-based visualization of DW-MRI data. In the feature definitions, the emphasis was on applications of diffusion imaging in neuroscience:

- *Probabilistic tractography* is widely used in neuroscience [119, 17, 6, 18, 45], and methods for visual analysis and illustration are needed. Chapter 3 presents one of the first approaches to visualizing the results of probabilistic tractography.
- Based on *edge detectors* like the ones discussed in Chapter 6, *segmentation algorithms* like the one from Chapter 7 provide a tool for the quantitative analysis of DW-MRI data. In the past, they have been used both as an alternative to fiber tracking [104, 84], and to segment the thalamic nuclei [242, 105].
- The *virtual Klingler dissection* in Chapter 9 was developed in cooperation with a domain scientist, modeled on an anatomical fiber preparation technique [133], and is currently being extended for practical use [196].
- *Estimating fiber directions* from high angular resolution diffusion imaging, as it is done in Chapter 10, is currently an active topic of research in the medical imaging and neuroscience communities [2, 220, 213, 123, 20, 108, 159, 212, 18, 247].

11.1.1. Improvements over Previous Work

Even though we concentrated on developing novel concepts, our research has also integrated and improved upon several previous works:

- The applicability of *standard tensor topology* [56, 93, 254, 255] to data from DT-MRI has been explored in Chapter 3. To the best of our knowledge, Chapter 4 presents the first application in which degenerate lines in a tensor field have a clear interpretation, namely, as boundary curves of extremal surfaces.
- Chapter 4 clarifies the topological properties of *height crease surfaces*, which have been ignored in previous works [63, 115, 178, 209, 157].
- In Chapter 5, *DT-MRI streamsurfaces* [251, 204, 223] have been shown to be mathematically ill-defined and their extraction was shown to be unstable. Planarity ridge surfaces have been proposed as a replacement to illustrate regions of planar diffusion.

- As part of Chapter 6, the *invariant gradients framework* [113] has been reformulated and extended. Moreover, the *Log-Euclidean edge detector* [9] has been analyzed and a source of artifacts has been corrected.

- As a consequence of Chapter 10, the insight that *finding the directions of fiber bundles* from an orientation distribution function (ODF) should be treated as an inverse problem has replaced the belief that these directions can simply be found by localizing ODF maxima [85, 249, 212, 96, 26, 58].

11.1.2. Relevance Beyond DW-MRI

The visualization of data from DW-MRI was the starting point and the clear focus of our work. However, some of our contributions are also relevant outside of this field:

- To the best of our knowledge, Chapter 3 presents the first *fuzzy topological visualization* which conveys the degree of uncertainty to the user. This could give an impulse to integrating uncertainty in other fields in which topological methods are used.

- The *crease surface* definition studied in Chapter 4 can be applied whenever one would like to extract a surface along which a given function takes on large values. Existing applications outside medical image analysis include flow visualization [178] and digital geometry processing [209].

- The watershed-based algorithm for *mesh segmentation* in Chapter 7 can be used whenever it is possible to specify edge weights that indicate whether or not the adjacent vertices should be grouped together.

- The *higher-order maximum enhancing (HOME)* glyph from Chapter 8 can be used for all positive definite supersymmetric higher-order tensors.

- The *higher-order structure tensor (HOST)* from the same chapter has the potential to be used for image diffusion and texture segmentation. The algorithm for low-rank approximation of higher-order tensors in Chapter 10 could also be used to extract discrete principle directions from HOSTs in $\mathbb{R}^3$, for example to implement higher-order tensor voting [142].

11.2. Future Work

Possible directions of future research have been identified at the end of the individual chapters. The topics on which we plan to concentrate fall into two categories: The processing and visualization of HARDI data, and the use of higher-order tensors in a range of applications which includes, but is not limited to, diffusion imaging.

11.2.1. HARDI Processing and Visualization

Chapter 10 has established the low-rank approximation of higher-order tensors as a way to obtain more reliable streamline visualizations from q-ball and spherical deconvolution data. One of our future goals will be to extend this idea to a point at which it can be used for quantitative studies in neuroscience and medicine. Beside the mathematical problems discussed in Section 10.5, we will address the minimum requirements which higher-order tensor models impose on data acquisition, and perform a careful comparison to alternative methods for the analysis of high-angular resolution data. We consider it important that these results are brought to the attention of domain scientists by publishing them outside the visualization community.

Existing visualization research has concentrated on the DT-MRI model. At the same time, domain scientists have developed an increasing interest in HARDI, and in combining diffusion measurements with other imaging modalities, like structural MRI or functional MRI (fMRI). In the future, we plan to develop visualization methods that will support the understanding of large-scale structures in HARDI data, and help to see relations between different types of data. We expect that feature extraction strategies like the ones presented in Chapters 3, 5 and 9 of this dissertation will be particularly successful in dealing with the high information density involved in these tasks. In addition, we plan to increase our emphasis on methods for interactive exploration.

11.2.2. Higher-Order Tensors in Computer Vision

Second-order structure and image diffusion tensors play a central role in a number of image processing and computer vision tasks, including filtering and segmentation. In Chapter 8, we have demonstrated that higher-order tensors can provide a more detailed description of local image structure, both for processing multi-valued data, and at junctions and corners. We will look into using higher-order tensors to define a scale space which better preserves relevant image structures. The results of this research can be used for scale space analysis of features like crease surfaces (Chapter 4) in HARDI data, but at the same time, they will be useful for general types of images and volume data.

With the prior work on invariant gradients [113] and our own contribution on the use of eigenvalue derivatives (Chapter 6), the structure of boundaries in DT-MRI fields is now well understood. We plan to extend these approaches to investigate different types of boundaries in HARDI data, and work towards establishing HARDI segmentation as a suitable alternative to fiber tracking in cases where quantitative results are desired.

A. Homogeneous Forms and Fourier Series / Spherical Harmonics

This appendix describes three details which are relevant to the practical implementation of the methods in Chapters 8 and 10. Section A.1 pertains memory-efficient storage of the coefficients of a supersymmetric tensor, while Sections A.2 and A.3 show how to relate homogeneous forms to Fourier series and spherical harmonics, respectively.

A.1. Efficient Representation of Supersymmetric Tensors

An order-l tensor based on an n-dimensional vector space has n^l components. Storing them is very memory-consuming already for moderate l. However, supersymmetry reduces the number of independent channels to $N = \binom{n+l-1}{l}$, which means merely linear growth for $n = 2$ ($N = l + 1$) and quadratic growth for $n = 3$ ($N = (l + 1)(l + 2)/2$).

We store the non-redundant components of a tensor $\mathcal{T}$ in a zero-based linear array with elements $\mathcal{T}_i$, sorted in lexicographic order (in $\mathbb{R}^3$, $[T_{1111}, T_{1112}, T_{1113}, T_{1122}, \ldots]$). Let $\nu_{i,k} \in \{0, 1, \ldots, l\}$ denote the number of times $k \in \{1, 2, \ldots, n\}$ appears as an index of the i-th element. The multiplicity of element i, denoted μ_i, is the number of times it appears as a component of the original tensor. For $n = 2$, $\mu_i = \binom{l}{\nu_{i,1}}$, for $n = 3$, $\mu_i = \binom{l}{\nu_{i,1}}\binom{l-\nu_{i,1}}{\nu_{i,2}}$. In this notation, the homogeneous form from Equation (2.10) can be evaluated as

$$T(\mathbf{v}) = \sum_{i=0}^{N-1} \mu_i \mathcal{T}_i v_1^{\nu_{i,1}} v_2^{\nu_{i,2}} \cdots v_n^{\nu_{i,n}} \tag{A.1}$$

A.2. Tensors in $\mathbb{R}^2$ and Fourier Series

Consider a Fourier Series, truncated after order l:

$$f(\phi) = \frac{1}{2} a_0 + \sum_{k=1}^{l} a_k \cos(k\phi) + \sum_{k=1}^{l} b_k \sin(k\phi) \tag{A.2}$$

Setting $a_k := b_k := 0$ for odd k leaves a $(l + 1)$-dimensional function space. For $n = 2$, Equation (A.1) can be rewritten in polar coordinates:

$$T(\phi) = \sum_{i=0}^{l} \mathcal{T}_i \binom{l}{i} \cos^{l-i}\phi \sin^i \phi \tag{A.3}$$

Let us regard $\mathcal{T}_i$ as coefficients and $\binom{l}{i}\cos^{l-i}\phi\sin^i\phi$ as basis functions. We will now show that these basis functions span the same space as the truncated Fourier Series. The proof is by induction on order l and constructive in the sense that it implies a recursive method to find the change-of-basis matrix.

Let $\{\mathbf{f}_k\}$ denote the basis functions of a truncated Fourier Series in which only even multiples of ϕ are allowed:

$$\mathbf{f}_k := \begin{cases} 0.5 & \text{if } k = 0 \\ \cos((k+1)\phi) & \text{if } k \text{ odd} \\ \sin(k\phi) & \text{if } k \text{ even } (k \neq 0) \end{cases}$$

Likewise, $\mathbf{t}^l_k$ is the k-th basis function of an order-l tensor:

$$\mathbf{t}^l_k := \binom{l}{k}\cos^{l-k}\phi\sin^k\phi$$

For $l = 0$, both the Fourier Series and the tensor basis represent constant functions and $\mathbf{f}_0 = 0.5\mathbf{t}^0_0$. Assume that the functions that can be represented using $\{\mathbf{f}_k\}$ with $k \leq l$ are equivalent to the functions represented by $\{\mathbf{t}^l_k\}$. Further, assume that we know how to express the Fourier basis in terms of the tensor basis. Then, we can show that the same assumption also holds for $l + 2$: Observe that

$$\begin{aligned} \cos^{l-i}\phi\sin^i\phi &= \left(\cos^2\phi + \sin^2\phi\right)\cos^{l-i}\phi\sin^i\phi \\ &= \cos^{l+2-i}\phi\sin^i\phi + \cos^{l-i}\phi\sin^{i+2}\phi \end{aligned}$$

and that the latter functions are proportional to functions in $\{\mathbf{t}^{l+2}_k\}$. Thus, we can express the first $l + 1$ Fourier basis functions in terms of $\{\mathbf{t}^{l+2}_k\}$ by replacing each occurence of $\mathbf{t}^l_k$ in their known representation by

$$\mathbf{t}^l_k = \frac{(l+2-k)(l+1-k)}{(l+2)(l+1)}\mathbf{t}^{l+2}_k + \frac{(k+2)(k+1)}{(l+2)(l+1)}\mathbf{t}^{l+2}_{k+2}$$

It remains to be shown how to express $\mathbf{f}_{l+1}$ and $\mathbf{f}_{l+2}$ in terms of $\{\mathbf{t}^{l+2}_k\}$. For this, we use trigonometric identities for multiple angles:

$$\begin{aligned} \mathbf{f}_{l+1} = \cos((l+2)\phi) = \sum_{i=0}^{l/2+1}(-1)^i\binom{l+2}{2i}\cos^{l+2-2i}\phi\sin^{2i}\phi &= \sum_{i=0}^{l/2+1}(-1)^i\mathbf{t}^{l+2}_{2i} \\ \mathbf{f}_{l+2} = \sin((l+2)\phi) = \sum_{i=0}^{l/2}(-1)^i\binom{l+2}{2i+1}\cos^{l+1-2i}\phi\sin^{2i+1}\phi &= \sum_{i=0}^{l/2}(-1)^i\mathbf{t}^{l+2}_{2i+1} \end{aligned}$$

■

For reference, Table A.1 presents the relations for $l = 2$ and $l = 4$.

$l=2$	$a_0 = \mathcal{T}_0 + \mathcal{T}_2$	$a_2 = \frac{1}{2}\mathcal{T}_0 - \frac{1}{2}\mathcal{T}_2$	$b_2 = \mathcal{T}_1$
$l=4$	$a_0 = \frac{3}{4}\mathcal{T}_0 + \frac{3}{2}\mathcal{T}_2 + \frac{3}{4}\mathcal{T}_4$	$a_2 = \frac{1}{2}\mathcal{T}_0 - \frac{1}{2}\mathcal{T}_4$	$b_2 = \mathcal{T}_1 + \mathcal{T}_3$
		$a_4 = \frac{1}{8}\mathcal{T}_0 - \frac{3}{4}\mathcal{T}_2 + \frac{1}{8}\mathcal{T}_4$	$b_4 = \frac{1}{2}\mathcal{T}_1 - \frac{1}{2}\mathcal{T}_3$

Table A.1.: Relation of Fourier coefficients and tensor components for orders $l = 2$ and $l = 4$. A method to compute these relations for general l is given in the text.

A.3. Tensors in $\mathbb{R}^3$ and Spherical Harmonics

Spherical harmonics form an orthonormal basis for complex functions on the unit sphere, much like the Fourier series offers an orthonormal basis over an interval in Cartesian space. The spherical harmonic Y_l^m for order l and phase factor $m \leq l$ is given as

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

where P_l^m is an associated Legendre polynomial and i is the imaginary unit. Since ODFs are real-valued and exhibit antipodal symmetry, we employ the restricted basis used in [57]. For $h = 0, 2, 4, \ldots, l$ and $m = -h, \ldots, 0, \ldots, h$, $j := (h^2 + h + 2)/2 + m - 1$ and

$$Y_j := \begin{cases} \sqrt{2} \cdot \mathrm{Re}(Y_h^m) & \text{if } -h \leq m < 0 \\ Y_h^0 & \text{if } m = 0 \\ \sqrt{2} \cdot \mathrm{Img}(Y_h^m) & \text{if } 0 < m \leq h \end{cases} \tag{A.4}$$

where Re and Img denote real and imaginary parts, respectively. For a spherical harmonics series up to even order l, this results in $(l+1)(l+2)/2$ terms.

Via their homogeneous forms $T(\mathbf{v})$, higher-order tensors $\mathcal{T}$ can be used to provide an alternative representation of functions on the sphere. It is equivalent to spherical harmonics in the sense that any function $T(\mathbf{v})$ in the form of Equation (2.10) can be re-written as a linear combination of spherical harmonics, $T = \sum_j c_j Y_j$, with Y_j from Equation (A.4), and vice versa [149]. A matrix that relates $\mathcal{T}_{i_1 i_2 \ldots i_l}$ to c_j is found by writing $\mathbf{v}$ in spherical coordinates and solving

$$c_j = \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\pi} T(\mathbf{v}(\theta, \phi))\, Y_j(\theta, \phi)\, \sin\theta \, d\theta \, d\phi$$

symbolically for each j, which is simplified by software like Maple or Mathematica.

Bibliography

[1] M. Adachi. *Embeddings and Immersions*. American Mathematical Society, 1993.

[2] A. L. Alexander, K. M. Hasan, M. Lazar, J. S. Tsuruda, and D. L. Parker. Analysis of partial volume effects in diffusion-tensor MRI. *Magnetic Resonance in Medicine*, 45:770–780, 2001.

[3] D. C. Alexander, G. J. Barker, and S. R. Arridge. Detection and modeling of non-gaussian apparent diffusion coefficient profiles in human brain data. *Magnetic Resonance in Medicine*, 48:331–340, 2002.

[4] A. W. Anderson. Theoretical analysis of the effects of noise on diffusion tensor imaging. *Magnetic Resonance in Medicine*, 46(6):1174–1188, 2001.

[5] A. W. Anderson. Measurement of fiber orientation distributions using high angular resolution diffusion imaging. *Magnetic Resonance in Medicine*, 54(5):1194–1206, 2005.

[6] A. Anwander, M. Tittgemeyer, D. von Cramon, A. Friederici, and T. Knösche. Connectivity-based parcellation of broca's area. *Cerebral Cortex*, 17(4):816–825, 2007.

[7] L. Armijo. Minimization of functions having lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16(1):1–3, 1966.

[8] S. Arseneau and J. R. Cooperstock. An improved representation of junctions through asymmetric tensor diffusion. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. V. Nefian, M. Gopi, V. Pascucci, J. Zara, J. Molineros, H. Theisel, and T. Malzbender, editors, *Advances in Visual Computing*, volume 4291 of *Lecture Notes in Computer Science*, pages 363–372. Springer, 2006.

[9] V. Arsigny, P. Fillard, X. Pennec, and N. Ayache. Log-euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic Resonance in Medicine*, 56(2):411–421, 2006.

[10] Y. Assaf and P. J. Basser. Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain. *NeuroImage*, 27(1):48–58, 2005.

[11] M. M. Bahn. Invariant and orthonormal scalar measures derived from magnetic resonance diffusion tensor imaging. *Journal of Magnetic Resonance*, 141:68–77, 1999.

[12] P. J. Basser and D. K. Jones. Diffusion-tensor MRI: theory, experimental design and data analysis – a technical review. *NMR in Biomedicine*, 15(7–8):456–467, 2002.

[13] P. J. Basser, J. Mattiello, and D. L. Bihan. Estimation of the effective self-diffusion tensor from the NMR spin echo. *Journal of Magnetic Resonance*, B(103):247–254, 1994.

[14] P. J. Basser and S. Pajevic. Spectral decomposition of a 4th-order covariance tensor: Applications to diffusion tensor MRI. *Signal Processing*, 87:220–236, 2007.

[15] P. J. Basser, S. Pajevic, C. Pierpaoli, J. Duda, and A. Aldroubi. In vivo fiber tractography using DT-MRI data. *Magnetic Resonance in Medicine*, 44:625–632, 2000.

[16] P. J. Basser and C. Pierpaoli. Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. *Journal of Magnetic Resonance*, B(111):209–219, 1996.

[17] T. Behrens, M. Woolrich, M. Jenkinson, H. Johansen-Berg, R. Nunes, S. Clare, P. Matthews, J. Brady, and S. Smith. Characterization and propagation of uncertainty in diffusion-weighted MR imaging. *Magnetic Resonance in Medicine*, 50:1077–1088, 2003.

[18] T. E. J. Behrens, H. Johansen-Berg, S. Jbabdi, M. F. S. Rushworth, and M. W. Woolrich. Probabilistic diffusion tractography with multiple fibre orientations: What can we gain? *NeuroImage*, 34:144–155, 2007.

[19] A. G. Belyaev, E. V. Anoshkina, and T. L. Kunii. Ridges, ravines and singularities. In A. T. Fomenko and T. L. Kunii, editors, *Topological Modeling for Visualization*, chapter 18. Springer, 1997.

[20] Ø. Bergmann, G. Kindlmann, A. Lundervold, and C.-F. Westin. Diffusion k-tensor estimation from Q-Ball imaging using discretized principal axes. In *Proc. Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 4191 of *LNCS*, pages 268–275, 2006.

[21] J. Bigün, T. Bigün, and K. Nilsson. Recognition by symmetry derivatives and the generalized structure tensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(12):1590–1605, 2004.

[22] J. Bigün, G. Granlund, and J. Wiklund. Multidimensional orientation estimation with applications to texture analysis and optical flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):775–790, 1991.

[23] J. Bigün and G. H. Granlund. Optimal orientation detection of linear symmetry. In *Proc. IEEE International Conference on Computer Vision*, pages 433–438, 1987.

[24] J. Blaas, C. P. Botha, B. Peters, F. M. Vos, and F. H. Post. Fast and reproducible fiber bundle selection in DTI visualization. In C. Silva, E. Gröller, and H. Rushmeier, editors, *Proc. IEEE Visualization 2005*, pages 59–64, October 2005.

[25] F. Bloch. Nuclear induction. *Physical Review*, 70(7–8):460–474, 1946.

[26] L. Bloy and R. Verma. On computing the underlying fiber directions from the diffusion orientation distribution function. In D. N. Metaxas, L. Axel, G. Fichtinger, and G. Székely, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 5241 of *LNCS*, pages 1–8. Springer, 2008.

[27] H. Blum and R. N. Nagel. Shape description using weighted symmetric axis features. *Pattern Recognition*, 10:167–180, 1978.

[28] G. E. Bredon. *Topology and Geometry*. Springer, 1993.

[29] T. Brox and J. Weickert. A TV flow based local scale measure for texture discrimination. In T. Pajdla and J. Matas, editors, *Proc. European Conference on Computer Vision (ECCV)*, volume 3022 of *LNCS*, pages 578–590. Springer, 2004.

[30] T. Brox and J. Weickert. Level set segmentation with multiple regions. *IEEE Transactions on Image Processing*, 15(10):3213–3218, October 2006.

[31] T. Brox, J. Weickert, B. Burgeth, and P. Mrázek. Nonlinear structure tensors. *Image and Vision Computing*, 24(1):41–55, 2006.

[32] A. Brun, H. Knutsson, H. J. Park, M. E. Shenton, and C.-F. Westin. Clustering fiber tracts using normalized cuts. In C. Barillot, D. Haynor, and P. Hellier, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 3216 of *Lecture Notes in Computer Science*, pages 368–375. Springer, 2004.

[33] B. Budiansky. Tensors. In C. E. Pearson, editor, *Handbook of Applied Mathematics*, chapter 4, pages 179–225. Van Nostrand Reinhold, 2nd edition, 1990.

[34] B. Cabral and L. C. Leedom. Imaging vector fields using line integral convolution. In *Proc. ACM SIGGRAPH*, pages 263–270, 1993.

[35] P. T. Callaghan, C. D. Eccles, and Y. Xia. NMR microscopy of dynamic displacements: k-space and q-space imaging. *Journal of Physics E*, 21(8):820–822, 1988.

[36] S. Campagna, L. Kobbelt, and H.-P. Seidel. Directed edges – a scalable representation for triangle meshes. *Journal of Graphics Tools*, 3(4):1–11, 1998.

[37] J.-F. Cardoso. Super-symmetric decomposition of the fourth-order cumulant tensor. blind identification of more sources than sensors. In *IEEE International Conference on Acoustic Speech Signal Processing (ICASSP)*, pages 3109–3112, 1991.

[38] H. Y. Carr and E. M. Purcell. Effects of diffusion on free precession in nuclear magnetic resonance experiments. *Physical Review*, 94(3):630–638, 1954.

[39] J. D. Carroll and J.-J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35(3):283–319, 1970.

[40] M. Catani, R. J. Howard, S. Pajevic, and D. K. Jones. Virtual in vivo interactive dissection of white matter fasciculi in the human brain. *NeuroImage*, 17:77–94, 2002.

[41] J. Y. Chang, K. M. Lee, and S. U. Lee. Multiview normal field integration using level set methods. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[42] L. Chen and N. D. Georganas. An efficient and robust algorithm for 3D mesh segmentation. *Multimedia Tools and Applications*, 29(2):109–125, 2006.

[43] W. Chen, S. Zhang, S. Correia, and D. S. Ebert. Abstractive representation and exploration of hierarchically clustered diffusion tensor fiber tracts. *Computer Graphics Forum (Proc. EuroVis)*, 27(3):1071–1078, 2008.

[44] T.-S. Chow. Matrices and linear algebra. In C. E. Pearson, editor, *Handbook of Applied Mathematics*, chapter 16, pages 878–927. Van Nostrand Reinhold, 2nd edition, 1990.

[45] M. X. Cohen, J.-C. Schoene-Bake, C. E. Elger, and B. Weber. Connectivity-based segregation of the human striatum predicts personality characteristics. *Nature Neuroscience*, 12(1):32–34, 2009.

[46] P. Comon, G. Golub, L.-H. Lim, and B. Mourrain. Symmetric tensors and symmetric tensor rank. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1254–1279, 2008.

[47] P. Comon and B. Mourrain. Decomposition of quantics in sums of powers of linear forms. *Signal Processing*, 53(2):96–107, September 1996.

[48] C. D. Correa, D. Silver, and M. Chen. Feature aligned volume manipulation for illustration and visualization. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 12(5):1069–1076, 2006.

[49] J. Crank. *The Mathematics of Diffusion.* Oxford University Press, 2nd edition, 1975.

[50] J. C. Criscione, J. D. Humphrey, A. S. Douglas, and W. C. Hunter. An invariant basis for natural strain which yields orthogonal stress response terms in isotropic hyperelasticity. *Journal of the Mechanics and Physics of Solids*, 48:2445–2465, 2000.

[51] J. Damon. Generic structure of two-dimensional images under gaussian blurring. *SIAM Journal on Applied Mathematics*, 59(1):97–138, 1998.

[52] J. Dauguet, S. Peled, V. Berezovskii, T. Delzescaux, S. K. Warfield, R. Born, and C.-F. Westin. 3D histological reconstruction of fiber tracts and direct comparison with diffusion tensor MRI tractography. In R. Larsen, M. Nielsen, and J. Sporring, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 4190 of *LNCS*, pages 109–116. Springer, 2006.

[53] L. De Lathauwer, B. De Moor, and J. Vandewalle. On the best rank-1 and rank-$(r_1, r_2, \ldots, r_n)$ approximation of higher-order tensors. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1324–1342, 2000.

[54] V. de Silva and L.-H. Lim. Tensor rank and the ill-posedness of the best low-rank approximation problem. *SIAM Journal on Matrix Analysis and Applications*, 30(3):1084–1127, 2008.

[55] T. Delmarcelle and L. Hesselink. Visualizing second-order tensor fields with hyperstreamlines. *IEEE Computer Graphics and Applications*, 13(4):25–33, 1993.

[56] T. Delmarcelle and L. Hesselink. The topology of symmetric, second-order tensor fields. In R. D. Bergeron and A. E. Kaufman, editors, *Proc. IEEE Visualization*, pages 140–147, 1994.

[57] M. Descoteaux, E. Angelino, S. Fitzgibbons, and R. Deriche. Regularized, fast, and robust analytical Q-Ball imaging. *Magnetic Resonance in Medicine*, 58:497–510, 2007.

[58] M. Descoteaux, R. Deriche, T. R. Knösche, and A. Anwander. Deterministic and probabilistic tractography based on complex fibre orientation distributions. *IEEE Transactions on Medical Imaging*, 28(2):269–286, 2009.

[59] S. Di Zenzo. A note on the gradient of a multi-image. *Computer Vision, Graphics, and Image Processing*, 33:116–125, 1986.

[60] Z. Ding, J. C. Gore, and A. W. Anderson. Classification and quantification of neuronal fiber pathways using diffusion tensor MRI. *Magnetic Resonance in Medicine*, 49:716–721, 2003.

[61] C. T. J. Dodson and T. Poston. *Tensor Geometry*, volume 130 of *Graduate Texts in Mathematics*. Springer, 2nd edition, 1997.

[62] R. A. Drebin, L. Carpenter, and P. Hanrahan. Volume rendering. In *Proc. ACM SIGGRAPH*, pages 65–74, 1988.

[63] D. Eberly. *Ridges in Image and Data Analysis*, volume 7 of *Computational Imaging and Vision*. Kluwer Academic Publishers, 1996.

[64] D. Eberly, R. Gardner, B. Morse, S. Pizer, and C. Scharlach. Ridges for image analysis. *Journal of Mathematical Imaging and Vision*, 4:353–373, 1994.

[65] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

[66] F. Enders, N. Sauber, D. Merhof, P. Hastreiter, C. Nimsky, and M. Stamminger. Visualization of white matter tracts with wrapped streamlines. In C. Silva, E. Gröller, and H. Rushmeier, editors, *Proc. IEEE Visualization*, pages 51–58, 2005.

[67] D. B. Ennis and G. Kindlmann. Orthogonal tensor invariants and the analysis of diffusion tensor magnetic resonance images. *Magnetic Resonance in Medicine*, 55(1):136–146, 2006.

[68] D. B. Ennis, G. Kindlmann, I. Rodriguez, P. A. Helm, and E. R. McVeigh. Visualization of tensor fields using superquadric glyphs. *Magnetic Resonance in Medicine*, 53(1):169–176, 2005.

[69] C. Feddern, J. Weickert, B. Burgeth, and M. Welk. Curvature-driven PDE methods for matrix-valued images. *International Journal of Computer Vision*, 69(1):93–107, 2006.

[70] A. Fick. Über Diffusion. *Annalen der Physik*, 170(1):59–86, 1855.

[71] P. Fillard, X. Pennec, V. Arsigny, and N. Ayache. Clinical DT-MRI estimation, smoothing and fiber tracking with log-euclidean metrics. *IEEE Transactions on Medical Imaging*, 26(11):1472–1482, 2007.

[72] P. A. Firby and C. F. Gardiner. *Surface topology*. Horwood, 1982.

[73] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[74] W. Förstner and E. Gülch. A fast operator for detection and precise location of distinct points, corners and centres of circular features. In *International Society for Photogrammetry and Remote Sensing (ISPRS) Intercomission Conference on Fast Processing of Photogrammetric Data*, pages 281–305, Interlaken, 1987.

[75] L. R. Frank. Anisotropy in high angular resolution diffusion-weighted MRI. *Magnetic Resonance in Medicine*, 45:935–939, 2001.

[76] L. R. Frank. Characterization of anisotropy in high angular resolution diffusion-weighted MRI. *Magnetic Resonance in Medicine*, 47:1083–1099, 2002.

[77] R. T. Frankot and R. Chellappa. A method for enforcing integrability in shape from shading algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(4):439–451, 1988.

[78] A. D. Friederici, J. Bahlmann, S. Heim, R. I. Schubotz, and A. Anwander. The brain differentiates human and non-human grammars: Functional localization and structural connectivity. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, 103(7):2458–2463, February 2006.

[79] J. D. Furst and S. M. Pizer. Marching ridges. In *Proc. IASTED International Conference on Signal and Image Processing*, pages 22–26, 2001.

[80] J. D. Furst, S. M. Pizer, and D. H. Eberly. Marching cores: A method for extracting cores from 3D medical images. In *Workshop on Mathematical Methods in Biomedical Image Analysis*, pages 124–130, 1996.

[81] I. Galić, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, and H.-P. Seidel. Image compression with anisotropic diffusion. *Journal of Mathematical Imaging and Vision*, 31(2–3):255–269, 2008.

[82] S. Gumhold and W. Straßer. Real time compression of triangle mesh connectivity. In *Proc. ACM SIGGRAPH*, pages 133–140, 1998.

[83] P. Hagmann, L. Cammoun, X. Gigandet, R. Meuli, C. J. Honey, and V. J. Wedeen. Mapping the structural core of human cerebral cortex. *PLoS Biology*, 6(7):1479–1493, 2008.

[84] P. Hagmann, L. Jonasson, T. Deffieux, R. Meuli, J.-P. Thiran, and V. J. Wedeen. Fibertract segmentation in position orientation space from high angular resolution diffusion MRI. *NeuroImage*, 32(2):665–675, 2006.

[85] P. Hagmann, T. G. Reese, W.-Y. I. Tseng, R. Meuli, J.-P. Thiran, and V. J. Wedeen. Diffusion spectrum imaging tractography in complex cerebral white matter: an investigation of the centrum semiovale. In *Proc. International Society of Magnetic Resonance in Medicine (ISMRM)*, page 623, 2004.

[86] E. L. Hahn. Spin echoes. *Physical Review*, 80(4):580–594, 1950.

[87] R. M. Haralick. Ridges and valleys on digital images. *Computer Vision, Graphics, and Image Processing*, 22:28–38, 1983.

[88] R. A. Harshman. Foundations of the parafac procedure: Models and conditions for an "explanatory" multimodal factor analysis. *UCLA Working Papers in Phonetics*, 16:1–84, 1970.

[89] D. M. Healy, H. Hendriks, and P. T. Kim. Spherical deconvolution. *Journal of Multivariate Analysis*, 67:1–22, 1998.

[90] J. Helman and L. Hesselink. Representation and display of vector field topology in fluid flow data sets. *Computer*, 22(8):27–36, 1989.

[91] M. Herberthson, A. Brun, and H. Knutsson. Pairs of orientations in the plane. In *Proceedings of the SSBA Symposium on Image Analysis*, pages 97–100, Umeå, Sweden, March 2006. SSBA.

[92] C. P. Hess, P. Mukherjee, E. T. Han, D. Xu, and D. B. Vigneron. Q-Ball reconstruction of multimodal fiber orientations using the spherical harmonic basis. *Magnetic Resonance in Medicine*, 56:104–117, 2006.

[93] L. Hesselink, Y. Levy, and Y. Lavin. The topology of symmetric, second-order 3D tensor fields. *IEEE Transactions on Visualization and Computer Graphics*, 3(1):1–11, 1997.

[94] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *Journal of Mathematics and Physics*, 6(1):164–189, 1927.

[95] M. Hlawitschka and G. Scheuermann. HOT-lines: Tracking lines in higher order tensor fields. In C. Silva, E. Gröller, and H. Rushmeier, editors, *Proc. IEEE Visualization*, pages 27–34, 2005.

[96] M. Hlawitschka, G. Scheuermann, A. Anwander, T. Knösche, M. Tittgemeyer, and B. Hamann. Tensor lines in tensor fields of arbitrary order. In G. Bebis et al., editor, *Advances in Visual Computing (Proc. ISVC'07)*, volume 4841 of *LNCS*, pages 341–350. Springer, 2007.

[97] U. Höhle and E. P. Klement, editors. *Non-classical logics and their applications to fuzzy subsets*, volume 32 of *Theory and decision library. B: mathematical and statistical methods*. Kluwer, 1995.

[98] M. A. Horsfield and D. K. Jones. Applications of diffusion-weighted and diffusion tensor MRI to white matter diseases – a review. *NMR in Biomedicine*, 15(7–8):570–577, 2002.

[99] V. Interrante and C. Grosch. Visualizing 3D flow. *IEEE Computer Graphics and Applications*, 18(4):49–53, 1998.

[100] T. J. Jankun-Kelly and K. Mehta. Superellipsoid-based, real symmetric traceless tensor glyphs motivated by nematic liquid crystal alignment visualization. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 12(5):1197–1204, 2006.

[101] M. R. Jayachandra, N. Rehbein, C. Herweh, and S. Heiland. Fiber tracking of human brain using fourth-order tensor and high angular resolution diffusion imaging. *Magnetic Resonance in Medicine*, 60(5):1207–1217, 2008.

[102] B. Jeremić, G. Scheuermann, J. Frey, Z. Yang, B. Hamann, K. I. Joy, and H. Hagen. Tensor visualizations in computational geomechanics. *International Journal for Numerical and Analytical Methods in Geomechanics*, 26:925–944, 2002.

[103] H. Johansen-Berg, T. Behrens, M. Robson, I. Drobnjak, M. Rushworth, J. Brady, S. Smith, D. Higham, and P. Matthews. Changes in connectivity profiles define functionally distinct regions in human medial frontal cortex. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, 101(36):13335–13340, 2004.

[104] L. Jonasson, X. Bresson, P. Hagmann, O. Cuisenaire, R. Meuli, and J.-P. Thiran. White matter fiber tract segmentation in DT-MRI using geometric flows. *Medical Image Analysis*, 9:223–236, 2005.

[105] L. Jonasson, P. Hagmann, C. Pollo, X. Bresson, C. R. Wilson, R. Meuli, and J.-P. Thiran. A level set method for segmentation of the thalamus and its nuclei in DT-MRI. *Signal Processing*, 87(2):309–321, 2007.

[106] D. K. Jones. The effect of gradient sampling schemes on measures derived from diffusion tensor MRI: A monte carlo study. *Magnetic Resonance in Medicine*, 51(4):807–815, 2004.

[107] D. K. Jones, M. A. Horsfield, and A. Simmons. Optimal strategies for measuring diffusion in anisotropic systems by magnetic resonance imaging. *Magnetic Resonance in Medicine*, 42:515–525, 1999.

[108] E. Kaden, T. R. Knösche, and A. Anwander. Parametric spherical deconvolution: Inferring anatomical connectivity using diffusion MR imaging. *NeuroImage*, 37:474–488, 2007.

[109] T. Kato. *Perturbation theory for linear operators*, volume 132 of *Die Grundlehren der mathematischen Wissenschaften*. Springer, 2nd edition, 1976.

[110] G. Kindlmann. Superquadric tensor glyphs. In *Eurographics/IEEE Symposium on Visualization (SymVis)*, pages 147–154, 2004.

[111] G. Kindlmann. *Visualization and Analysis of Diffusion Tensor Fields*. PhD thesis, School of Computing, University of Utah, September 2004.

[112] G. Kindlmann. Tensor invariants and their gradients. In J. Weickert and H. Hagen, editors, *Visualization and Processing of Tensor Fields*, pages 215–224. Springer, 2006.

[113] G. Kindlmann, D. Ennis, R. Whitaker, and C.-F. Westin. Diffusion tensor analysis with invariant gradients and rotation tangents. *IEEE Transactions on Medical Imaging*, 26(11):1483–1499, 2007.

[114] G. Kindlmann, X. Tricoche, and C.-F. Westin. Anisotropy creases delineate white matter structure in diffusion tensor MRI. In R. Larsen, M. Nielsen, and J. Sporring, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 4190 of *LNCS*, pages 126–133. Springer, 2006.

[115] G. Kindlmann, X. Tricoche, and C.-F. Westin. Delineating white matter structure in diffusion tensor MRI with anisotropy creases. *Medical Image Analysis*, 11(5):492–502, 2007.

[116] G. Kindlmann, D. Weinstein, and D. Hart. Strategies for direct volume rendering of diffusion tensor fields. *IEEE Transactions on Visualization and Computer Graphics*, 6(2):124–138, April 2000.

[117] G. Kindlmann and C.-F. Westin. Diffusion tensor visualization with glyph packing. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 12(5):1329–1335, 2006.

[118] J. C. Klein, T. E. J. Behrens, M. D. Robson, C. E. Mackay, D. J. Higham, and H. Johansen-Berg. Connectivity-based parcellation of human cortex using diffusion MRI: Establishing reproducibility, validity and observer independence in BA 44/45 and SMA/pre-SMA. *NeuroImage*, 34(1):204–211, 2007.

[119] M. A. Koch, D. G. Norris, and M. Hund-Georgiadis. An investigation of functional and anatomical connectivity using magnetic resonance imaging. *NeuroImage*, 16:241–250, 2002.

[120] J. J. Koenderink and A. J. van Doorn. Local features of smooth shapes: Ridges and courses. In B. C. Vemuri, editor, *Geometric Methods in Computer Vision II*, volume 2031 of *Proc. of SPIE*, pages 2–13, 1993.

[121] E. Kofidis and P. A. Regalia. On the best rank-1 approximation of higher-order supersymmetric tensors. *SIAM Journal on Matrix Analysis and Applications*, 23(3):863–884, 2002.

[122] U. Köthe. Edge and junction detection with an improved structure tensor. In B. Michaelis and G. Krell, editors, *Pattern Recognition. 25th DAGM Symposium*, volume 2781 of *Lecture Notes in Computer Science*, pages 25–32. Springer, 2003.

[123] B. Kreher, J. Schneider, I. Mader, E. Martin, J. Hennig, and K. Il'yasov. Multitensor approach for analysis and tracking of complex fiber configurations. *Magnetic Resonance in Medicine*, 54:1216–1225, 2005.

[124] J. Krüger, P. Kipfer, P. Kondratieva, and R. Westermann. A particle system for interactive visualization of 3D flows. *IEEE Transactions on Visualization and Computer Graphics*, 11(6):744–756, 2005.

[125] D. Le Bihan, E. Breton, D. Lallemand, P. Grenier, E. Cabanis, and M. Laval-Jeantet. MR imaging of intravoxel incoherent motions: Application to diffusion and perfusion in neurologic disorders. *Radiology*, 161(2):401–407, 1986.

[126] D. Le Bihan, S. ichi Urayama, T. Aso, T. Hanakawa, and H. Fukuyama. Direct and fast detection of neuronal activation in the human brain with diffusion MRI.

Proceedings of the National Academy of Sciences of the United States of America (PNAS), 103(21):8263–8268, 2006.

[127] D. Le Bihan, R. Turner, and P. Douek. Is water diffusion restricted in human brain white matter? An echo-planar NMR imaging study. *Neuroreport*, 4(7):887–890, 1993.

[128] K. O. Lim and J. A. Helpern. Neuropsychiatric applications of DTI – a review. *NMR in Biomedicine*, 15(7–8):587–593, 2002.

[129] C.-P. Lin, W.-Y. I. Tseng, H.-C. Cheng, and J.-H. Chen. Validation of diffusion tensor magnetic resonance axonal fiber imaging with registered manganese-enhanced optic tracts. *NeuroImage*, 14(5):1035–1047, 2001.

[130] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision*, 30(2):117–154, 1998.

[131] C. Liu, R. Bammer, B. Acar, and M. E. Moseley. Characterizing non-gaussian diffusion by using generalized diffusion tensors. *Magnetic Resonance in Medicine*, 51(5):924–937, 2004.

[132] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. In *Proc. ACM SIGGRAPH*, pages 163–169, 1987.

[133] E. Ludwig and L. Klingler. *Atlas cerebri humani*. S. Karger AG, Basel, 1956.

[134] J. R. Magnus and H. Neudecker. *Matrix Differential Calculus with Applications in Statistics and Econometrics*. Wiley, rev. edition, 1998.

[135] A. P. Mangan and R. T. Whitaker. Partitioning 3D surface meshes using watershed segmentation. *IEEE Transactions on Visualization and Computer Graphics*, 5(4):308–321, 1999.

[136] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995.

[137] M. McGuire and P. Sibley. A heightfield on an isometric grid. Technical Report CS-05-14, Department of Computer Science at Brown University, 2005.

[138] D. Merhof, M. Sonntag, F. Enders, V. P. Hastreiter, R. Fahlbusch, C. Nimsky, and G. Greiner. Visualization of diffusion tensor data using evenly spaced streamlines. In *Vision, Modelling and Visualization (VMV)*, pages 79–86, 2005.

[139] D. Merhof, M. Sonntag, F. Enders, C. Nimsky, P. Hastreiter, and G. Greiner. Hybrid visualization for white matter tracts using triangle strips and point sprites. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 12(5):1181–1188, 2006.

[140] H. Miura and S. Kida. Identification of tubular vortices in turbulence. *Journal of the Physical Society of Japan*, 66(5):1331–1334, 1997.

[141] M. Moakher. The algebra of fourth-order tensors with application to diffusion MRI. In D. H. Laidlaw and J. Weickert, editors, *Visualization and Processing of Tensor Fields: Advances and Perspectives*. Springer, 2009. In press.

[142] P. Mordohai and G. G. Medioni. *Tensor voting: a perceptual organization approach to computer vision and machine learning*. Morgan & Claypool, 2007.

[143] S. Mori, B. J. Crain, V. P. Chacko, and P. C. M. van Zijl. Three-dimensional tracking of axonal projections in the brain by magnetic resonance imaging. *Annals of Neurology*, 45(2):265–269, 1999.

[144] S. Mori and P. C. van Zijl. Fiber tracking: principles and strategies – a technical review. *NMR in Biomedicine*, 15:468–480, 2002.

[145] B. S. Morse. *Computation of Object Cores from Grey-Level Images*. PhD thesis, University of North Carolina at Chapel Hill, 1994.

[146] C. Nimsky, O. Ganslandt, F. Enders, D. Merhof, T. Hammen, and M. Buchfelder. Visualization strategies for major white matter tracts for intraoperative use. *International Journal of Computer Assisted Radiology and Surgery*, 1(1):13–22, 2006.

[147] L. O'Donnell, W. E. L. Grimson, and C.-F. Westin. Interface detection in diffusion tensor MRI. In C. Barillot, D. Haynor, and P. Hellier, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 3216 of *LNCS*, pages 360–367. Springer, 2004.

[148] S. Osher and R. Fedkiw. *Level Set Methods and Dynamic Implicit Surfaces*, volume 153 of *Applied Mathematical Sciences*. Springer, 2003.

[149] E. Özarslan and T. Mareci. Generalized diffusion tensor imaging and analytical relationships between diffusion tensor imaging and high angular resolution diffusion imaging. *Magnetic Resonance in Medicine*, 50:955–965, 2003.

[150] E. Özarslan, T. M. Shepherd, B. C. Vemuri, S. J. Blackband, and T. H. Mareci. Fast orientation mapping from HARDI. In J. Duncan and G. Gerig, editors, *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 3749 of *LNCS*, pages 156–163. Springer, 2005.

[151] E. Özarslan, B. C. Vemuri, and T. H. Mareci. Fiber orientation mapping using generalized diffusion tensor imaging. In *Proc. IEEE International Symposium on Biomedical Imaging*, pages 1036–1039, 2004.

[152] E. Özarslan, B. C. Vemuri, and T. H. Mareci. Generalized scalar measures for diffusion MRI using trace, variance, and entropy. *Magnetic Resonance in Medicine*, 53:866–876, 2005.

[153] D. L. Page, A. F. Koschan, and M. A. Abidi. Perception-based 3D triangle mesh segmentation using fast marching watersheds. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 27–32, 2003.

[154] S. Pajevic, A. Aldroubi, and P. J. Basser. A continuous tensor field approximation of discrete DT-MRI data for extracting microstructural and architectural features of tissue. *Journal of Magnetic Resonance*, 154:85–100, 2002.

[155] S. Pajevic and C. Pierpaoli. Color schemes to represent the orientation of anisotropic tissues from diffusion tensor data: application to white matter fiber tract mapping in the human brain. *Magnetic Resonance in Medicine*, 42(3):526–540, 1999.

[156] R. Peikert and M. Roth. The "parallel vectors" operator - a vector field visualization primitive. In D. S. Ebert, M. Gross, and B. Hamann, editors, *Proc. IEEE Visualization 1999*, pages 263–270, 1999.

[157] R. Peikert and F. Sadlo. Height ridge computation and filtering for visualization. In I. Fujishiro, H. Li, and K.-L. Ma, editors, *Proc. Pacific Vis 2008*, pages 119–126, 2008.

[158] M. M. Peixoto. Structural stability on two-dimensional manifolds. *Topology*, 1(2):101–121, 1962.

[159] S. Peled, O. Friman, F. Jolesz, and C.-F. Westin. Geometrically constrained two-tensor model for crossing tracts in DWI. *Magnetic Resonance Imaging*, 24(9):1263–1270, 2006.

[160] M. Perrin, Y. Cointepas, A. Cachia, C. Poupon, B. Thirion, D. Rivière, P. Cathier, V. El Kouby, A. Constantinesco, D. Le Bihan, and J.-F. Mangin. Connectivity-based parcellation of the cortical mantle using q-ball diffusion imaging. *International Journal of Biomedical Imaging*, 2008, 2008.

[161] V. Petrovic, J. Fallon, and F. Kuester. Visualizing whole-brain DTI tractography with GPU-based tuboids and LoD management. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 13(6):1488–1495, 2007.

[162] B. T. Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975.

[163] C. Pierpaoli, A. Barnett, S. Pajevic, R. Chen, L. Penix, A. Virta, and P. Basser. Water diffusion changes in Wallerian degeneration and their dependence on white matter architecture. *NeuroImage*, 13(6):1174–1185, 2001.

[164] C. Pierpaoli, P. Jezzard, P. J. Basser, A. Barnett, and G. Di Chiro. Diffusion tensor MR imaging of the human brain. *Radiology*, 201(3):637–648, 1996.

[165] S. M. Pizer, C. A. Burbeck, J. M. Coggins, D. S. Fritsch, and B. S. Morse. Object shape before boundary shape: Scale-space medial axes. *Journal of Mathematical Imaging and Vision*, 4:303–313, 1994.

[166] F. H. Post, B. Vrolijk, H. Hauser, R. S. Laramee, and H. Doleisch. The state of the art in flow visualisation: Feature extraction and tracking. *Computer Graphics Forum*, 22(4):775–792, 2003.

[167] B. Preim and D. Bartz. *Visualization in Medicine. Theory, Algorithms, and Applications*. Morgan Kaufmann, 2007.

[168] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C++: The Art of Scientific Computing, Second Edition*. Cambridge Univ. Press, 2002.

[169] A. Pressley. *Elementary differential geometry*. Springer, 2001.

[170] L. Qi. Eigenvalues of a real supersymmetric tensor. *Journal of Symbolic Computation*, 40:1302–1324, 2005.

[171] L. Qi. Eigenvalues and invariants of tensors. *Journal of Mathematical Analysis and Applications*, 325(2):1363–1377, 2007.

[172] G. Reina, K. Bidmon, F. Enders, P. Hastreiter, and T. Ertl. GPU-based hyperstreamlines for diffusion tensor imaging. In T. Ertl, K. Joy, and B. Santos, editors, *Proc. Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis)*, pages 35–42, 2006.

[173] M. E. Rettmann, X. Han, C. Hu, and J. L. Prince. Automated sulcal segmentation using watersheds on the cortical surface. *NeuroImage*, 15(2):329–344, 2002.

[174] C. Rezk-Salama, P. Hastreiter, C. Teitzel, and T. Ertl. Interactive exploration of volume line integral convolution based on 3D-texture mapping. In D. Ebert, M. Gross, and B. Hamann, editors, *Proc. IEEE Visualization*, pages 233–240, 1999.

[175] C. Rezk-Salama and A. Kolb. Opacity peeling for direct volume rendering. *Computer Graphics Forum (Proc. Eurographics)*, 25(3):597–606, 2006.

[176] R. J. Rost. *OpenGL shading manual*. Addison-Wesley, 2nd edition, 2006.

[177] M. Rousson, T. Brox, and R. Deriche. Active unsupervised texture segmentation on a diffusion based feature space. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 699–706, Madison, Wisconsin, USA, June 2003.

[178] F. Sadlo and R. Peikert. Efficient visualization of lagrangian coherent structures by filtered AMR ridge extraction. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 13(6):1456–1463, 2007.

[179] J. Sahner, T. Weinkauf, and H.-C. Hege. Galilean invariant extraction and iconic representation of vortex core lines. In K. Brodlie, D. Duke, and K. Joy, editors, *Proc. Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis)*, pages 151–160, 2005.

[180] J. Sahner, T. Weinkauf, N. Teuber, and H.-C. Hege. Vortex and strain skeletons in eulerian and lagrangian frames. *IEEE Transactions on Visualization and Computer Graphics*, 13(5):980–990, 2007.

[181] H. Sakuma, Y. Nomura, K. Takeda, T. Tagami, T. Nakagawa, Y. Tamagawa, Y. Ishii, and T. Tsukamoto. Adult and neonatal human brain: Diffusional anisotropy and myelination with diffusion-weighted MR imaging. *Radiology*, 180(1):229–233, 1991.

[182] T. Salzbrunn and G. Scheuermann. Streamline predicates. *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1601–1612, 2006.

[183] G. Scheuermann, B. Hamann, K. I. Joy, and W. Kollmann. Visualizing local vector field topology. *Journal of Electronic Imaging*, 9(4):356–367, October 2000.

[184] G. Scheuermann and X. Tricoche. Topological methods in flow visualization. In C. Johnson and C. Hansen, editors, *The Visualization Handbook*, pages 341–356. Academic Press, 2004.

[185] J. D. Schmahmann, D. N. Pandya, R. Wang, G. Dai, H. E. D'Arceuil, A. J. de Crespigny, and V. J. Wedeen. Association fibre pathways of the brain: parallel observations from diffusion spectrum imaging and autoradiography. *Brain*, 130(3):630–653, 2007.

[186] E. Schmidt. Zur Theorie der linearen und nichtlinearen Integralgleichungen. *Mathematische Annalen*, 63(4):433–476, 1907.

[187] T. Schultz, B. Burgeth, and J. Weickert. Flexible segmentation and smoothing of DT-MRI fields through a customizable structure tensor. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, P. Remagnino, A. V. Nefian, M. Gopi, V. Pascucci, J. Zara, J. Molineros, H. Theisel, and T. Malzbender, editors, *Advances in Visual Computing (Proc. ISVC)*, volume 4291 of *LNCS*, pages 455–464, 2006.

[188] T. Schultz, N. Sauber, A. Anwander, H. Theisel, and H.-P. Seidel. Virtual Klingler dissection: Putting fibers into context. *Computer Graphics Forum (Proc. EuroVis)*, 27(3):1063–1070, 2008.

[189] T. Schultz and H.-P. Seidel. Estimating crossing fibers: A tensor decomposition approach. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 14(6):1635–1642, 2008.

[190] T. Schultz and H.-P. Seidel. Using eigenvalue derivatives for edge detection in DT-MRI data. In G. Rigoll, editor, *Pattern Recognition (Proc. DAGM)*, volume 5096 of *LNCS*, pages 193–202. Springer, 2008.

[191] T. Schultz, H. Theisel, and H.-P. Seidel. Segmentation of DT-MRI anisotropy isosurfaces. In K. Museth, T. Möller, and A. Ynnerman, editors, *Proc. Eurographics/IEEE-VGTC Symposium on Visualization (EuroVis) 2007*, pages 187–194. Eurographics, 2007.

[192] T. Schultz, H. Theisel, and H.-P. Seidel. Topological visualization of brain diffusion MRI data. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 13(6):1496–1503, 2007.

[193] T. Schultz, H. Theisel, and H.-P. Seidel. Crease surfaces: From theory to extraction and application to diffusion tensor MRI. *IEEE Transactions on Visualization and Computer Graphics*, 2009. To appear. Preprint posted online 16 April 2009.

[194] T. Schultz, J. Weickert, and H.-P. Seidel. A higher-order structure tensor. In D. H. Laidlaw and J. Weickert, editors, *Visualization and Processing of Tensor Fields – Advances and Perspectives*, pages 263–280. Springer, 2009.

[195] M. Schünke, E. Schulte, and U. Schumacher. *Prometheus: LernAtlas der Anatomie. Kopf und Neuroanatomie.* Thieme, 2006.

[196] R. Schurade. Visualisierung von Nervenfaserflächen im menschlichen Gehirn aus DTI Daten. Master's thesis, Fakultät für Mathematik und Informatik, Universität Leipzig, 2009.

[197] Scientific Computing and Imaging Institute (SCI). BioTensor: A SCIRun power app for processing and visualizing diffusion tensor images, 2006.

[198] J. A. Sethian. *Level Set Methods and Fast Marching Methods.* Cambridge University Press, 2nd edition, 2001.

[199] K. K. Seunarine, P. A. Cook, M. G. Hall, K. V. Embleton, G. J. M. Parker, and D. C. Alexander. Exploiting peak anisotropy for tracking through complex structures. In *Proc. IEEE Workshop Mathematical Methods in Biomedical Image Analysis (MMBIA)*, pages 1–8, 2007.

[200] H.-W. Shen, C. R. Johnson, and K.-L. Ma. Visualizing vector fields using line integral convolution and dye advection. In *Proc. Symposium on Volume Visualization*, pages 63–70, 1996.

[201] J. S. Shimony, A. Z. Snyder, N. Lori, and T. E. Conturo. Automated fuzzy clustering of neuronal pathways in diffusion tensor tracking. In *Proc. International Society of Magnetic Resonance in Medicine (ISMRM)*, volume 10, 2002.

[202] S. Smale. Structurally stable systems are not dense. *American Journal of Mathematics*, 88(2):491–496, 1966.

[203] S. M. Smith, M. Jenkinson, H. Johansen-Berg, D. Rueckert, T. E. Nichols, C. E. Mackay, K. E. Watkins, O. Ciccarelli, M. Z. Cader, P. M. Matthews, and T. E. J. Behrens. Tract-based spatial statistics: Voxelwise analysis of multi-subject diffusion data. *NeuroImage*, 31(4):1487–1505, 2006.

[204] R. Sondershaus and S. Gumhold. Meshing of diffusion surfaces for point-based tensor field visualization. In *Proc. 12th International Meshing Roundtable (IMR)*, pages 177–188, 2003.

[205] C. H. Sotak. The role of diffusion tensor imaging in the evaluation of ischemic brain injury: a review. *NMR in Biomedicine*, 15(7–8):561–569, 2002.

[206] E. Stejskal and J. Tanner. Spin diffusion measurements: Spin echoes in the presence of a time-dependent field gradient. *Journal of Chemical Physics*, 42:288–292, 1965.

[207] G. W. Stewart. On the early history of the singular value decomposition. *SIAM Review*, 35(4):551–566, 1993.

[208] C. Stoll, S. Gumhold, and H.-P. Seidel. Visualization with stylized line primitives. In C. Silva, E. Gröller, and H. Rushmeier, editors, *Proc. IEEE Visualization*, pages 695–702, 2005.

[209] J. Süßmuth and G. Greiner. Ridge based curve and surface reconstruction. In A. Belyaev and M. Garland, editors, *Eurographics Symposium on Geometry Processing*, pages 243–251, 2007.

[210] G. Taubin. Estimating the tensor of curvature of a surface from a polyhedral approximation. In *Proc. Fifth International Conference on Computer Vision (ICCV)*, pages 902–907, 1995.

[211] H. Theisel and H.-P. Seidel. Feature flow fields. In G.-P. Bonneau, S. Hahmann, and C. D. Hansen, editors, *Proc. Eurographics/IEEE-TCVG Symposium on Visualization (VisSym)*, pages 141–148, 2003.

[212] J.-D. Tournier, F. Calamante, and A. Connelly. Robust determination of the fibre orientation distribution in diffusion MRI: Non-negativity constrained super-resolved spherical deconvolution. *NeuroImage*, 35:1459–1472, 2007.

[213] J.-D. Tournier, F. Calamante, D. G. Gadian, and A. Connelly. Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. *NeuroImage*, 23:1176–1185, 2004.

[214] X. Tricoche. *Vector and Tensor Field Topology Simplification, Tracking, and Visualization*. PhD thesis, Fachbereich Informatik, Universität Kaiserslautern, 2002.

[215] X. Tricoche, G. Kindlmann, and C.-F. Westin. Invariant crease lines for topological and structural analysis of tensor fields. *IEEE Transactions on Visualization and Computer Graphics (Proc. IEEE Visualization)*, 14(6):1627–1634, 2008.

[216] X. Tricoche, X. Zheng, and A. Pang. Visualizing the topology of symmetric, second-order, time-varying two-dimensional tensor fields. In J. Weickert and H. Hagen, editors, *Visualization and Processing of Tensor Fields*, pages 225–240. Springer, 2006.

[217] D. Tschumperlé and R. Deriche. Variational frameworks for DT-MRI estimation, regularization and visualization. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 116–121, 2003.

[218] D. Tschumperlé and R. Deriche. Vector-valued image regularization with PDE's: A common framework for different applications. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2003)*, pages 651–656, 2003.

[219] D. S. Tuch. Q-Ball imaging. *Magnetic Resonance in Medicine*, 52:1358–1372, 2004.

[220] D. S. Tuch, T. G. Reese, M. R. Wiegell, N. Makris, J. W. Belliveau, and V. J. Wedeen. High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. *Magnetic Resonance in Medicine*, 48:577–582, 2002.

[221] D. S. Tuch, T. G. Reese, M. R. Wiegell, and V. J. Wedeen. Diffusion MRI of complex neural architecture. *Neuron*, 40(5):885–895, 2003.

[222] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966.

[223] A. Vilanova, G. Berenschot, and C. van Pul. DTI visualization with stream surfaces and evenly-spaced volume seeding. In O. Deussen, C. Hansen, D. A. Keim, and D. Saupe, editors, *Proc. Joint Eurographics – IEEE TCVG Symposium on Visualization (VisSym)*, pages 173–182, 2004.

[224] A. Vilanova, S. Zhang, G. Kindlmann, and D. H. Laidlaw. An introduction to visualization of diffusion tensor imaging and its applications. In J. Weickert and H. Hagen, editors, *Visualization and Processing of Tensor Fields*, pages 121–153. Springer, 2006.

[225] F. Vivodtzev, L. Linsen, G.-P. Bonneau, B. Hamann, K. I. Joy, and B. A. Olshausen. Hierarchical isosurface segmentation based on discrete curvature. In G.-P. Bonneau, S. Hahmann, and C. D. Hansen, editors, *Proceedings of the symposium on Data visualisation (VisSym)*, pages 249–258, 2003.

[226] E. A. H. von dem Hagen and R. M. Henkelman. Orientational diffusion reflects fiber structure. *Magnetic Resonance in Medicine*, 48:454–459, 2002.

[227] W. von Funck, H. Theisel, and H.-P. Seidel. Vector field based shape deformations. *ACM Transactions on Graphics (Proc. ACM SIGGRAPH)*, 25(3):1118–1125, 2006.

[228] S. Wakana, H. Jiang, L. M. Nagae-Poetscher, P. C. M. van Zijl, and S. Mori. Fiber tract-based atlas of human white matter anatomy. *Radiology*, 230:77–87, 2004.

[229] Y. Wang and L. Qi. On the successive supersymmetric rank-1 decomposition of higher-order supersymmetric tensors. *Numerical Linear Algebra with Applications*, 14:503–519, 2007.

[230] Z. Wang, B. C. Vemuri, Y. Chen, and T. Mareci. A constrained variational principle for direct estimation and smoothing of the diffusion tensor field from complex DWI. *IEEE Transactions on Medical Imaging*, 23(8):930–939, 2004.

[231] D. S. Watkins. *Fundamentals of Matrix Computations*. Pure and Applied Mathematics. Wiley, 2nd edition, 2005.

[232] S. Webb. *The Physics of Medical Imaging*. CRC Press, 1988.

[233] V. J. Wedeen, T. G. Reese, D. S. Tuch, M. R. Weigel, J.-G. Dou, R. M. Weisskoff, and D. Chessler. Mapping fiber orientation spectra in cerebral white matter with Fourier-transform diffusion MRI. In *Proc. International Society for Magnetic Resonance in Medicine*, volume 8, page 82, 2000.

[234] J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner, Stuttgart, 1998.

[235] J. Weickert and T. Brox. Diffusion and regularization of vector- and matrix-valued images. In M. Nashed and O. Scherzer, editors, *Inverse Problems, Image Analysis, and Medical Imaging*, volume 313 of *Contemporary Mathematics*, pages 251–268. AMS, 2002.

[236] T. Weinkauf. *Extraction of Topological Structures in 2D and 3D Vector Fields*. PhD thesis, Fakultät für Informatik, Otto-von-Guericke-Universität Magdeburg, 2008.

[237] D. Weinstein, G. Kindlmann, and E. Lundberg. Tensorlines: advection-diffusion based propagation through diffusion tensor fields. In *Proc. IEEE Visualization*, pages 249–253, 1999.

[238] D. Weiskopf, K. Engel, and T. Ertl. Interactive clipping techniques for texture-based volume visualization and volume shading. *IEEE Transactions on Visualization and Computer Graphics*, 9(3):298–312, 2003.

[239] C.-F. Westin, S. Maier, B. Khidhir, P. Everett, F. Jolesz, and R. Kikinis. Image processing for diffusion tensor magnetic resonance imaging. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 1679 of *LNCS*, pages 441–452. Springer, 1999.

[240] C.-F. Westin, S. Maier, H. Mamata, A. Nabavi, F. Jolesz, and R. Kikinis. Processing and visualization for diffusion tensor MRI. *Medical Image Analysis*, 6:93–108, 2002.

[241] C.-F. Westin, S. Peled, H. Gudbjartsson, R. Kikinis, and F. A. Jolesz. Geometrical diffusion measures for MRI from tensor basis analysis. In *Proc. International Society for Magnetic Resonance in Medicine*, page 1742, 1997.

[242] M. R. Wiegell, D. Tuch, H. B. Larsson, and V. J. Wedeen. Automatic segmentation of thalamic nuclei from diffusion tensor magnetic resonance imaging. *NeuroImage*, 19:391–401, 2003.

[243] S. Wiggins. *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Springer, 2nd edition, 2003.

[244] D. M. Wimberger, T. P. Roberts, A. J. Barkovich, L. M. Prayer, M. E. Moseley, and J. Kucharczyk. Identification of "premyelination" by diffusion-weighted MRI. *Journal of Computer Assisted Tomography*, 19(1):28–33, 1995.

[245] T. Wischgoll and J. Meyer. Locating closed hyperstreamlines in second order tensor fields. In J. Weickert and H. Hagen, editors, *Visualization and Processing of Tensor Fields*, pages 257–267. Springer, 2006.

[246] B. Wünsche. Advanced texturing techniques for the effective visualization of neuroanatomy from diffusion tensor imaging data. In *Proc. Asia-Pacific Bioinformatics Conference (APBC)*, pages 303–308, 2004.

[247] F.-C. Yeh, V. J. Wedeen, and W.-Y. I. Tseng. A recursive algorithm to decompose orientation distribution function and resolve intra-voxel fiber directions. In *Proc. International Society for Magnetic Resonance in Medicine (ISMRM)*, page 40, 2008.

[248] S. Y. Yen, S. Napel, and G. D. Rubin. Fast sliding thin slab volume visualization. In *Proc. Symposium on Volume Visualization*, pages 79–86, 1996.

[249] W. Zhan and Y. Yang. How accurately can the diffusion profiles indicate multiple fiber orientations? a study on general fiber crossings in diffusion MRI. *Journal of Magnetic Resonance*, 183:193–202, 2006.

[250] S. Zhang, S. Correia, and D. H. Laidlaw. Identifying white-matter fiber bundles in DTI data using an automated proximity-based fiber-clustering method. *IEEE Transactions on Visualization and Computer Graphics*, 14(5):1044–1053, 2008.

[251] S. Zhang, C. Demiralp, and D. H. Laidlaw. Visualizing diffusion tensor MR images using streamtubes and streamsurfaces. *IEEE Transactions on Visualization and Computer Graphics*, 9(4):454–462, 2003.

[252] S. Zhang, D. H. Laidlaw, and G. Kindlmann. Diffusion tensor MRI visualization. In C. D. Hansen and C. R. Johnson, editors, *The Visualization Handbook*, pages 327–340. Elsevier, 2005.

[253] X. Zheng and A. Pang. HyperLIC. In *Proc. IEEE Visualization*, pages 249–256, 2003.

[254] X. Zheng, B. Parlett, and A. Pang. Topological lines in 3D tensor fields and discriminant hessian factorization. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):395–407, 2005.

[255] X. Zheng, B. Parlett, and A. Pang. Topological structures of 3D tensor fields. In *IEEE Visualization*, pages 551–558, 2005.

[256] X. Zheng, X. Tricoche, and A. Pang. Degenerate 3D tensors. In J. Weickert and H. Hagen, editors, *Visualization and Processing of Tensor Fields*, pages 241–256. Springer, 2006.

[257] L. Zhukov, K. Museth, D. Breen, R. Whitaker, and A. H. Barr. Level set segmentation and modeling of DT-MRI human brain data. *Journal of Electronic Imaging*, 12:125–133, 2003.