Mathematics and Visualization

Thomas Schultz · Evren Özarslan
Ingrid Hotz *Editors*

Modeling, Analysis, and Visualization of Anisotropy

Springer

Mathematics and Visualization

Series editors
Hans-Christian Hege
David Hoffman
Christopher R. Johnson
Konrad Polthier
Martin Rumpf

More information about this series at http://www.springer.com/series/4562

Thomas Schultz • Evren Özarslan • Ingrid Hotz
Editors

Modeling, Analysis, and Visualization of Anisotropy

Springer

Editors
Thomas Schultz
Institute of Computer Science
University of Bonn
Bonn, Germany

Evren Özarslan
Department of Biomedical Engineering (IMT)
Linköping University
Linköping, Sweden

Ingrid Hotz
Department of Science & Technology
Linköping University
Norrköping, Sweden

ISSN 1612-3786 ISSN 2197-666X (electronic)
Mathematics and Visualization
ISBN 978-3-319-61357-4 ISBN 978-3-319-61358-1 (eBook)
DOI 10.1007/978-3-319-61358-1

Library of Congress Control Number: 2017951143

Mathematics Subject Classification (2010): 68-06, 15A69, 68U10, 92C55, 74E10

© Springer International Publishing AG 2017
This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.
The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Cover illustration: The cover illustration was kindly provided by Roxana Bujack, TU Kaiserslautern, and is related to Chapter 3, “Moment Invariants for Multi-Dimensional Data”. We would like to thank the FAnToM development team, Leipzig University, for providing the software for the visualization.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Starting from the entire Universe and proceeding all the way down to the atomic scale, physical quantities often take on different values when observed along different directions. Directional dependence (anisotropy) of data is thus paramount to numerous scientific disciplines such as chemistry, material science, astrophysics, neuroscience, and medical imaging. In each of these disciplines, a wide array of works employs tensors and other mathematical constructs to represent anisotropy. The focus of this book is on the modeling, processing, and visualization of anisotropy, regardless of the context in which it emerges. As such, it differs substantially from conventional reference works, which tend to be centered on a particular application.

This multidisciplinary book is the sixth in a series that aims to foster scientific exchange between communities that employ tensors and other higher-order representations of directionally dependent data. A significant portion of the chapters were co-authored by the participants of the workshop titled "Multidisciplinary Approaches to Multivalued Data: Modeling, Visualization, Analysis," which was held in Dagstuhl, Germany, in April 2016. However, the book does not gather the proceedings of the workshop; some contributions deviate from the presentations at the workshop, and we have invited several completely new chapters within the scope of the workshop.

The first four chapters of the book make up Part I, which focuses on characterizing the features of and visualizing tensor fields. In turn, Part II includes a collection of chapters on the processing of this data. Here, diffusion anisotropy continues to be a topic of interest. More fundamental aspects of the topic are addressed in Part III, while some important applications are presented in Part IV. Part V, which is the final part, highlights nascent approaches inspired by machine learning.

Visualization and analysis provide the necessary basis for understanding and exploring data, which is especially true and challenging when it comes to anisotropic or higher-order entities. Part I of this book presents different strategies for the structural analysis of tensor data, which serve as basis for condensed multiscale visualization. The methods range from topological analysis, to statistical moments, to the analysis of the gradient of tensors in materials. Chapter 1 starts with

theoretical considerations regarding the stability of topological structures with respect to perturbations in the tensor field. The result is a hierarchy of features that supports multiscale analysis. Chapter 2 provides a more applied perspective on tensor topology and its interpretations in the context of solid mechanics simulations. A different approach to structure is followed in Chap. 3, which uses moment invariants as descriptors of the tensor field. These descriptors are used to detect patterns in a data set independent of its orientation and scale. The last chapter of Part I investigates the gradients of (symmetric) second-order tensor fields, which are tensors of third order. Here, special emphasis is put on stress gradients in structural mechanics, for which glyph-based visualization is proposed.

Various imaging methods represent important sources of anisotropic data and produce voxelized data or unstructured point clouds. The analysis and processing of such data are the topic of Part II of this book. The first step in analyzing discrete data is the choice of an appropriate interpolation schema, which is the topic of Chap. 5. It provides a survey of interpolation schemes, including a detailed analysis of their respective properties. Chapter 6 proposes a general framework for fundamental morphological operations, such as dilation and erosion for matrix fields. The next two chapters deal with the analysis of point clouds: Chap. 7 proposes a robust extraction of geometrical shapes like curves and surfaces from point clouds, even in noisy scenarios, and introduces an advanced tensor voting technique for blood vessel analysis. Chapter 8, which is the last chapter, uses a feature classification approach to extract and reconstruct the point-sampled geometry of topographic data captured using airborne LiDAR technology.

Measuring structural anisotropy within locally oriented media is a challenging problem with enormous implications for assessing the state of neural tissues. Magnetic resonance techniques provide a powerful though indirect means of noninvasively measuring such anisotropy by characterizing the orientational preference of water diffusion. In Chap. 9 of Part III, the authors illustrate how anisotropic information can be obtained via diffusion MR. Various anisotropy metrics obtained through state-of-the-art signal-based and multicompartmental models are reviewed and compared. The rather large voxels afforded by typical MR imaging studies complicate the determination of local (microscopic) anisotropy when there is significant heterogeneity within the voxel. In a subsequent chapter, the notion of microscopic anisotropy and recent developments concerning its measurement via traditional as well as novel MR pulse sequences are reviewed. In Chap. 11, estimation of the single-diffusion tensor model in each voxel is revisited from a Bayesian perspective by incorporating heteroscedastic noise (i.e., the variance does not have to be constant but can also change throughout the samples). In the last chapter, intra-voxel heterogeneity is modeled by means of a multicompartmental model for diffusion MRI, employing a combination of tensor-valued non-central Wishart distributions.

Diffusion MRI fiber tractography remains the only available technique for mapping the brain's structural connectivity in vivo. The diverse contributions in Part IV demonstrate that it also remains a vital and active research topic. Chapter 13 introduces a novel method for obtaining a geometric representation of

tract boundaries from streamline-based tractography via a curve similarity metric, while Chap. 14 contributes to the validation of two widely used software packages for fiber tractography by investigating their test-retest reliability. Chapter 15, which is the final chapter, presents a pipeline for automated fiber bundle segmentation and quantification, which is specifically targeted toward the assessment of brain maturity in preterm neonates.

In recent years, the broad use of deep learning approaches has significantly enhanced the state of the art in neighboring fields, such as computer vision. However, there has been relatively little work on adapting such techniques to tensor fields or other mathematical representations of anisotropy. Accordingly, Part V of our book includes two contributions addressing this timely field of research. The first explores the feasibility of using a convolutional neural network (CNN) to map strain tensors to an associated scalar value, while the second uses CNNs to reconstruct fiber orientation distribution functions from diffusion MRI, which continues to work well even when the number of measurements provided as input is reduced.

We wish to thank the board and staff at Schloss Dagstuhl for their time and effort in facilitating the workshop, which fostered scientific exchange and ultimately led to the creation of this book. Special thanks go to the editors of the Springer Mathematics and Visualization book series for their valued consideration and support. The book would of course never have been possible without the high-quality manuscripts submitted by the authors. Last but not least, we are indebted to all reviewers, whose incisive comments greatly improved many of the chapters.

We certainly hope that this book will be a valuable resource for those who work on multidirectional data and inspirational in the development of new models as well as analysis and visualization techniques, thus advancing the state of the art in studies involving anisotropy.

Bonn, Germany — Thomas Schultz
Linköping, Sweden — Evren Özarslan
Norrköping, Sweden — Ingrid Hotz
March 2017

Contents

Part I
Features and Visualization

Robustness for 2D Symmetric Tensor Field Topology

Bei Wang and Ingrid Hotz

Abstract Topological feature analysis is a powerful instrument to understand the essential structure of a dataset. For such an instrument to be useful in applications, however, it is important to provide some importance measure for the extracted features that copes with the high feature density and discriminates spurious from important structures. Although such measures have been developed for scalar and vector fields, similar concepts are scarce, if not nonexistent, for tensor fields. In particular, the notion of robustness has been proven to successfully quantify the stability of topological features in scalar and vector fields. Intuitively, robustness measures the minimum amount of perturbation to the field that is necessary to cancel its critical points.

This chapter provides a mathematical foundation for the construction of a feature hierarchy for 2D symmetric tensor field topology by extending the concept of robustness, which paves new ways for feature tracking and feature simplification of tensor field data. One essential ingredient is the choice of an appropriate metric to measure the perturbation of tensor fields. Such a metric must be well-aligned with the concept of robustness while still providing some meaningful physical interpretation. A second important ingredient is the index of a degenerate point of tensor fields, which is revisited and reformulated rigorously in the language of degree theory.

1 Introduction

As a linear approximation of physical phenomena, tensors play an important role in numerous engineering, physics, and medical applications. Examples include various descriptors of stress at a point in a continuous medium under load or the diffusion characteristics of water molecules in fibrous media. Tensors provide a powerful and

B. Wang
University of Utah, Salt Lake City, UT, USA
e-mail: beiwang@sci.utah.edu

I. Hotz (✉)
Linköping University, Norrköping, Sweden
e-mail: ingrid.hotz@liu.se

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_1

simple language to describe anisotropic phenomena for which scalars and vectors are not sufficient, but the analysis of tensor fields is a complex and challenging task. Therefore, visualization becomes a crucial capability to support the understanding of tensor fields. See [18] for a survey on the analysis and visualization of second-order tensors.

In this chapter, we are especially interested in a structural characterization of symmetric second-order tensor fields using topological methods, which can form the basis of advanced analysis and visualization methods. Roughly speaking, tensor field topology segments the tensor field into regions of equivalent tensor line behavior. Conceptually, it is closely related to the vector field topology. Degenerate points in tensor fields take the role of critical points in vector fields, and tensor lines correspond to streamlines. However, despite these parallels, there are also many differences.

First, whereas critical points in vector fields behave as sources and sinks and separatrices can be interpreted as material boundaries of flows, the topological features of tensor fields often do not have a direct physical meaning. Degenerate points are points of high symmetry with isotropic behavior and thus might be considered as being especially boring. However, they play an important role from a structural point of view, as they are points where the eigenvector field is not uniquely specified and thus not necessarily continuous.

Second, there are also major structural differences in comparison to vector fields, because eigenvector fields have no specified orientation. In the 2D case, they exhibit a rotational symmetry with a rotational angle of π. As such, they are a special case of N-symmetric direction fields [17], which are important for many applications in geometry processing and texture design. For example, the eigenvector fields of the curvature tensor have been used for the purpose of quadrangular re-meshing, where degenerate points are mesh vertices with distinct valency [1, 16]. A related application is the synthesis of textures, for example, by defining the stroke directions as an eigenvector field of some tensor field, where degenerated points account for points with non-trivial texture characteristics [3, 30]. For both applications, it is essential to have control over the number of degenerate points. Furthermore, in tensor field analysis, it can also be beneficial to have control over not only the degenerated points but also their cancellation for feature-preserving interpolation and smoothing [15, 24].

While tensor field topology has attracted the most attention in geometric applications, it was introduced along with the vector field topology in visualization applications by Delmarcelle [9] and Tricoche [26]. Since the introduction of tensor field topology, theoretical and application-driven advancement has been slow for several reasons (in contrast to vector field topology): the lack of theory for 3D tensor fields, the complexity of the resulting topological structures, and the challenge of a direct interpretation of such structures in the application domain. However, there has been some recent, encouraging effort by Zhang et al. [31] concerning a theory for 3D tensor fields and the application of stress tensor field analysis. To further develop tensor field topology as a useful analysis tool, we are convinced that a major requirement is to find a mathematically rigorous way to cope

with the high density of the extracted topological features, even in the setting of 2D tensor fields, where a large number of structures originate from extended isotropic regions and are very sensitive to small changes in the data.

The most important requirement in applications is a *stable* topological skeleton representing the core structure of the data. For all the above-mentioned applications, tensor field topology can provide a means for the controlled manipulation and simplification of data. In this work, we introduce a measure for the stability of degenerate points with respect to small perturbations of the field. The measure is based on the notion of robustness and well group theory, which has already been successfully applied to vector fields. We extend this concept to 2D symmetric second-order tensor fields to lay the foundation for a discriminative analysis of essential and spurious features.

The work presented in this chapter paves the way for a complete framework of tensor field simplification based on robustness. We generalize the theory of robustness to the space of analytical tensor fields. In particular, we discuss the appropriate metrics for measuring the perturbations of tensor fields, but a few challenges remain. First, we need to develop efficient and stable algorithms to generate a hierarchical scheme among degenerate points. Second, the actual simplification of the tensor field using the hierarchical representation and cancellation of degenerate points is technically non-trivial. In this chapter, we focus on the first part by providing the necessary foundation for the following steps.

Our main contributions are threefold: First, we interpret the notion of tensor index under the setting of degree theory; Second, we define tensor field perturbations and make precise connections between such perturbations with the perturbations of bidirectional vector fields; Third, we generalize the notion of robustness to the study of tensor field topology.

This chapter is structured as follows: After reviewing relevant work in Sect. 2, we provide a brief description of well group theory and robustness for vector fields in Sect. 3. Then in Sect. 4, we reformulate some technical background in tensor field topology in a way that is compatible with robustness, by introducing the bidirectional vector field and an anisotropy vector field. The anisotropy vector field then provides the basis for Sect. 5 in which the notion of robustness is extended to the tensor field setting.

2 Related Work

Tensor Field Topology Previous research has examined the extraction, simplification, and visualization of the topology of symmetric second order tensor fields on which this work builds. The introduction of topological methods to the structural analysis of tensor fields goes back to Delmarcelle [9]. In correspondence to vector field topology, Delmarcelle has defined a topological skeleton, consisting of degenerate points and separatrices, which are tensor lines connecting the degenerate points. Delmarcelle has mainly been concerned with the characterization of

degenerate points in two-dimensional fields. Therefore he also provided a definition for the index of a critical point. Tricoche et al. [27] built on these ideas by developing algorithms to apply the concept of topological skeleton to real data. A central question of their work is the simplification of the tensor field topology and tracking it over time. They succeeded in simplifying the field, but the algorithm contains many parameters and is very complex. A robust extraction and classification algorithm for degenerate points has been presented by Hotz et al. [15]. Their method is based on edge labeling using an eigenvector-based interpolation. This work has been extended by Auer et al. [2] to cope with the challenge of discontinuities of tensor fields on triangulated surfaces. While the characteristics of the tensor field topology for two-dimensional fields are similar to the vector field topology, it is in general not possible to define a global vector field with the same topological structure. It is possible, however, to define a vector field whose critical points are located in the same positions as the degenerate points of the tensor field by duplicating their indexes. This idea has been used by Zhang et al. [30] for constructing a simplified tensor field for texture generation. Our method follows a similar line of thought but goes a step further by defining an isometric mapping of the tensor field to a vector field.

Robustness for Vector Fields In terms of vector field topology, topological methods have been employed extensively to extract features such as critical points and separatrices for vector field visualization [19] and simplification [8]. Motivated by hierarchical simplification of vector fields, the topological notion of *robustness* has been used to rank the critical points by measures of their stability. Robustness is closely related to the notion of *persistence* [10]. Introduced via the algebraic concept of well diagram and well group theory [7, 11, 12], it quantifies the stability of critical points with respect to the minimum amount of perturbation in the fields required to remove them. Robustness has been shown to be very useful for the analysis and visualization of 2D and 3D vector fields [23, 28]. In particular, it is the core concept behind simplifying a 2D vector field with a hierarchical scheme that is independent of the topological skeleton [22]; and it leads to the first ever 3D vector field simplification, based on critical point cancellation [20]. Measures of robustness also lead to a fresh interpretation of critical point tracking [21]: Stable critical points can be tracked more easily and more accurately in the time-varying setting.

In this paper, we extend the notion of robustness to the study of tensor field topology. We would like to rely on such a notion to develop novel, scalable, and mathematically rigorous ways to understand tensor field data, especially questions pertaining to their structural stability. We believe that robustness holds the key to increase the interpretability of tensor field data, and may lead to a new line of research that spans feature extraction, feature tracking, and feature simplification of tensor fields.

3 Preliminaries on Robustness for Vector Fields

In this section, we briefly review the relevant technical background of robustness for 2D vector field such as critical points, degrees, indices, well groups and well diagrams. These concepts are important for developing and understanding the extensions of robustness for the tensor field.

Critical Point and Sublevel Set Let $f : \mathbb{R}^2 \to \mathbb{R}^2$ be a continuous vector field. A *critical point* of f is a zero of the field, i.e., $f(x) = 0$. Define $f_0 : \mathbb{R}^2 \to \mathbb{R}$ as the vector magnitude of f, $f_0(x) = ||f(x)||_2$, for all $x \in \mathbb{R}^2$. Let $\mathbb{F}_r$ denote the sublevel set of f_0, $\mathbb{F}_r = f_0^{-1}(-\infty, r]$, that is, all points in the domain with a magnitude up to r. In particular, $\mathbb{F}_0 = f^{-1}(0)$ is the set of critical points. A value $r > 0$ is a *regular value* of f_0 if $\mathbb{F}_r$ is a 2-manifold, and for all sufficiently small $\epsilon > 0, f_0^{-1}[r-\epsilon, r+\epsilon]$ retracts to $f_0^{-1}(r)$; otherwise it is a *critical value*. We assume that f_0 has a finite number of critical values and f contains a finite number of isolated critical points. Figure 1 gives an example of a 2D vector field f with four critical points (Fig. 1 left) and the regions in the domain enclosed by colored contour lines of f_0 (Fig. 1 middle) illustrate sublevel sets of f_0 at critical values.

Degree and Index Suppose x is an isolated critical point of f. For a 2D vector field, the *degree* of x equals its index, which in turn corresponds to the winding number of a simple closed curve on the plane around x. Formally, fix the local coordinates near x and pick a closed disk D that encloses x in its interior and contains no other critical points. Then the *index* of x (w.r.t. f), $\mathrm{I}_f(x)$, or equivalently the *(local) degree* of f at x, denoted as $\deg(f\ |_x)$, is the degree of the mapping $u : \partial D \to \mathbb{S}^1$ that associates ∂D (the boundary of D) to the circle, given by $u(z) = f(z)/|f(z)|$ (u is sometimes referred to as the *Gauss* map). It is shown that isolated first-order critical points have an index of ± 1: a saddle has an index of -1 and non-saddles have an index of $+1$. In Fig. 1 right, x_2, x_4 are saddles of index -1 whereas x_1 and x_3 have index $+1$.

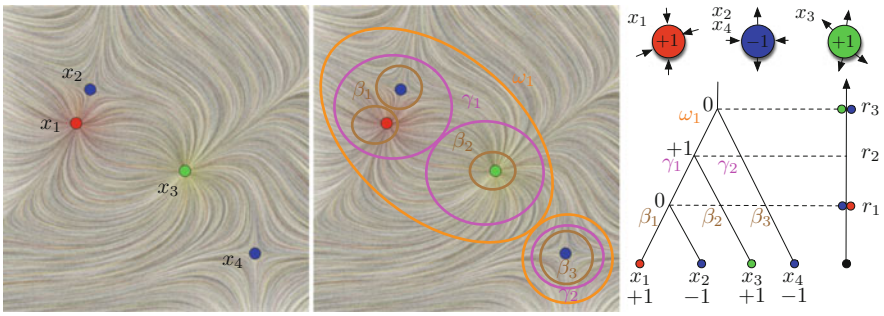


Fig. 1 Figure recreated from [28] showing the merge tree for a continuous 2D vector field example. From *left to right*: vector fields f, relations among components of $\mathbb{F}_r$ (for $r \geq 0$), and the augmented merge tree. f contains four critical points, a red sink x_1, a green source x_3, and two blue saddles x_2 and x_4. We use β, γ, ω, etc., to represent components of the sublevel sets

Let $C \subseteq \mathbb{F}_r$ be a path-connected component of $\mathbb{F}_r$. Consider $\{x_1, x_2, \ldots, x_n\}$ to be the set of critical points in C. Then the degree of f restricted to ∂C is the sum of the degrees of f at the x_i, $\deg(f\ |_{\partial C}) = \sum_{i=1}^{n} \deg(f\ |_{x_i})$. For notational convenience, when f is fixed, we abuse the notation by defining the degree of C as $\deg(C) := \deg(f\ |_{\partial C})$. For example, in Fig. 1 middle, component β_1 (representing a sublevel set of f_0) is of degree 0 as it contains critical points x_1 and x_2 with opposite degrees.

Poincaré-Hopf Theorem for Vector Fields We review the Poincaré-Hopf theorem in the setting of a 2D vector field. A particularly useful corollary for 2D vector field simplification is that if a region $C \subset \mathbb{R}^2$ has degree zero, it is possible to replace the vector field inside C with a vector field free of critical points.

Theorem 1 (Poincaré-Hopf theorem) *Let $\mathbb{M}$ be a smooth compact 2-manifold. Let f be a vector field on $\mathbb{M}$ with finitely many isolated critical points. For $\mathbb{M}$ with boundary, f points in the outward normal direction along the boundary. Then the sum of the indices of the critical points is equal to the Euler characteristic of the manifold: $\sum_i \mathrm{I}_f(x_i) = \chi(\mathbb{M})$.*

Well Group Given a mapping $f : \mathbb{X} \to \mathbb{Y}$ and a subspace $A \subseteq \mathbb{Y}$, the well group theory [7, 11, 12] studies the robustness of the homology of the pre-image of A, $f^{-1}(A)$ with respect to perturbations of the mapping f. Roughly speaking, the homology of a topological space measures its topological features, where the rank of the 0-, 1- and 2-dimensional homology groups corresponds to the number of connected components, tunnels, and voids, respectively. Here we review the well group theory in the setting of a 2D vector field $f : \mathbb{R}^2 \to \mathbb{R}^2$ where $A = 0$, and correspondingly study the stable property of the critical points ($f^{-1}(0)$) of f [7].

Let $f, h : \mathbb{R}^2 \to \mathbb{R}^2$ be two continuous 2D vector fields. Define the distance between the two mappings as $\mathrm{d}(f, h) = \sup_{x \in \mathbb{R}^2} ||f(x) - h(x)||_2$. We say a continuous mapping h is an *r-perturbation* of f, if $\mathrm{d}(f, h) \leq r$. In other words, for each point $x \in \mathbb{R}^2$, the point $h(x)$ lies within a disk of radius r centered at $f(x)$. See Fig. 2.

If h is an r-perturbation of f, then $h^{-1}(0)$ is a subspace of $\mathbb{F}_r$, that is, we have an inclusion $h^{-1}(0) \subseteq \mathbb{F}_r$. The connected components of $h^{-1}(0)$ generate a vector space that is the 0-dimensional homology group of $h^{-1}(0)$, denoted as $\mathsf{H}(h^{-1}(0))$. Similarly, we have the 0-dimensional homology group of $\mathbb{F}_r$, denoted as $\mathsf{H}(\mathbb{F}_r)$. The subspace relation $h^{-1}(0) \subseteq \mathbb{F}_r$ induces a linear map $j_h : \mathsf{H}(h^{-1}(0)) \to \mathsf{H}(\mathbb{F}_r)$

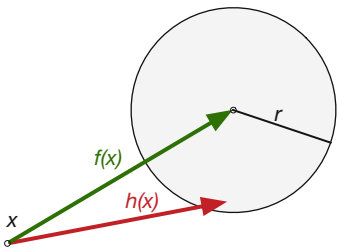


Fig. 2 Geometric interpretation of an r-perturbation of a vector field at a point x in the domain

between the two vector spaces. The *well group*, $\mathsf{U}(r)$, as first studied in [12], is the subgroup of $\mathsf{H}(\mathbb{F}_r)$ whose elements belong to the image of each j_h for all r-perturbation h of f. That is,

$$\mathsf{U}(r) = \bigcap_h \operatorname{im} j_h. \tag{1}$$

Assuming a finite number of critical points, the rank of $\mathsf{U}(0)$ is the number of critical points of f. For values $r < s$, $\mathbb{F}_r \subseteq \mathbb{F}_s$ inducing a linear map $\mathsf{f}_r^s : \mathsf{H}(\mathbb{F}_r) \to \mathsf{H}(\mathbb{F}_s)$ between the two homology groups. It can be shown that $\mathsf{U}(s) \subseteq \mathsf{f}_r^s(\mathsf{U}(r))$, for $r \leq s$. Therefore the rank of the well group decreases monotonically as r increases. The following lemma suggests an algorithm to compute the rank of the well groups.

Lemma 1 (Lemma 3, [7]) *If r is a regular value of f_0, then the rank of the well group $\mathsf{U}(r)$ is the number of connected components $C \subseteq \mathbb{F}_r$ such that* $\deg(C) \neq 0$.

Well Diagram A point r belongs to the *well diagram* of f_0, $\mathrm{Dgm}(f_0)$, with multiplicity k if the rank of the well group drops by k at r [7]. For reasons of stability, the point 0 is counted with infinite multiplicity. The point ∞ is counted with multiplicity k if for all sufficiently large values of r, the rank of $\mathsf{U}(r)$ is k. The well diagram contains a multi-set of points (infinitely many points at 0 and a finite number of nonzero points) on the extended real line, $\bar{\mathbb{R}} = \mathbb{R} \cup \{\pm\infty\}$, where each point in $\mathrm{Dgm}(f_0)$ is either a 0, a positive real number, or ∞.

We therefore consider each point in the well diagram as a measure of how *resistant* a homology class of $f^{-1}(0)$ is against perturbations of the mapping f [12]. Recall that f_0 has finitely many critical values that can be indexed consecutively as $\{r_i\}_i$ (where $0 = r_0 < r_1 < r_2 < \cdots < r_l$), $\mathsf{U}(r_i) \subseteq \mathsf{F}(r_i) := \mathsf{H}(\mathbb{F}_{r_i})$ are the corresponding well groups. We define the mapping $\mathsf{f}_0^j : \mathsf{F}(0) \to \mathsf{F}(r_j)$. A homology class α in the well group $\mathsf{U}(0)$ *dies* at r_j if $\mathsf{f}_0^i(\alpha)$ is a nonzero class in $\mathsf{U}(r_i)$; and either $\mathsf{f}_0^j(\alpha) = 0$, or $\mathsf{f}_0^j(\alpha) \notin \mathsf{U}(r_j)$, for each $i < j$. The *robustness* of a class α in $\mathsf{U}(0)$ is the value at which the class dies [12].

As shown in the example of Fig. 1 right, each critical point of f generates a class in $\mathsf{U}(0)$, denoted as α_1, α_2, α_3, and α_4 (corresponding to critical points x_1 to x_4, respectively). At r_1, two classes α_1 and α_2 die, and therefore they have a robustness of r_1. Similarly, α_3 and α_4 die at r_3, with a corresponding robustness of r_3. In terms of the well diagram, the well group $\mathsf{U}(r_1)$ drops in rank by two because there is an r_1-perturbation of f such that there are only two zeros. Therefore two points are in the well diagram at r_1. Similarly, two points are in the well diagram at r_3 because the well group drops its rank by two.

Robustness of Critical Points In the setting of 2D vector fields, the robustness of a critical point x_i can be described by the robustness of the class α_i in $\mathsf{U}(0)$ that it generates[1]. Therefore in our example (Fig. 1), points x_1, x_2, x_3, and x_4 have robustness r_1, r_1, r_3 and r_3 respectively.

[1] We rely on this definition to describe the robustness of a critical point x_i, even though the critical point is only a particularly chosen generator of the class α_i in $\mathsf{U}(0)$.

To compute the robustness of critical points in f, we construct an *augmented merge tree* of f_0 that tracks the (connected) components of $\mathbb{F}_r$ together with their degree information as they appear and merge by increasing r from 0. A leaf node represents the creation of a component at a local minima of f_0 and an internal node represents the merging of components. See [6, 28] for algorithmic details. The *robustness* of a critical point is the height of its lowest degree zero ancestor in the merge tree. To illustrate the construction, we show a 2D example recreated from [28] in Fig. 1. By definition, the critical points x_1 and x_2 have robustness r_1, whereas x_3 and x_4 have robustness r_3. Such a topological notion quantifies the stability of a critical point with respect to perturbations of the vector fields. Intuitively, if a critical point x has robustness r, then it can be canceled with a $(r + \delta)$-perturbation, but not with any $(r - \delta)$-perturbation, for $\delta > 0$ arbitrarily small.

Given the above machineries, the properties associated with robustness for critical points are direct consequences of Lemma 2 and Lemma 3. Their original proof sketches can be found in the supplementary material of [28]. These proofs, which are similar to the proof of Lemma 1, are revisited in Sect. 5.3 for completeness (in the setting of a specific type of vector field).

Lemma 2 (Nonzero Degree Component for Vector Field Perturbation, Corollary 1.2 in [28] supplement) *Let r be a regular value of f_0 and C a connected component of $\mathbb{F}_r$ such that* $\deg(C) \neq 0$. *Then for any δ-perturbation h of f, where $\delta < r$, the sum of the degrees of the critical points in $h^{-1}(0) \cap C$ is* $\deg(C)$.

Lemma 3 (Zero Degree Component for Vector Field Perturbation, Corollary 1.1 in [28] supplement) *Let r be a regular value of f_0 and C a connected component of $\mathbb{F}_r$ such that* $\deg(C) = 0$. *Then, there exists an r-perturbation h of f such that h has no critical points in C, $h^{-1}(0) \cap C = \emptyset$. In addition, h equals f except possibly within the interior of C.*

In the example of Fig. 1 right, x_1 has a robustness of r_1, Lemma 3 implies that there exists an $(r_1 + \delta)$-perturbation (for an arbitrarily small $\delta > 0$) that can cancel x_1 by locally modifying the connected component $C \subseteq \mathbb{F}_{r_1+\delta}$ containing it.

4 Tensor Fields and Bidirectional Anisotropy Vector Fields

For the remainder of this paper, we consider 2D symmetric second-order tensor fields. In this section, we establish the necessary foundations for introducing a robustness measure for the degenerate points of tensor fields. We introduce the notion of *bidirectional anisotropy vectors*, which serves two purposes. First, we use it to define the notion of perturbations of tensor fields for our setting (Sect. 4.2). Second, this notion will be central for the definition of the tensor index under the setting of degree theory (Sect. 5.1).

We start by summarizing some basic concepts of tensor field topology in Sect. 4.1. For a complete introduction, we refer the reader to the work by Delmarcelle [9] or Trichoche [25]. Then we introduce the notion of bidirectional

anisotropy vector fields in Sect. 4.2 and discuss its relation with respect to the space of deviators. Finally, we establish an isometry from the space of deviators to the anisotropy vector field in Sect. 4.3.

4.1 Background in Tensor Field Topology

The topology of a 2D symmetric second-order tensor field is defined as the topology of one of the two eigenvector fields [9]. The degenerate points constitute the basic ingredient of the tensor field topology and play a role similar to that for the critical points (zeros) for vector fields.

2D Symmetric Second-Order Tensor Fields In our setting, a *tensor* T is a linear operator that associates any vector v to another vector $u = Tv$, where v and u are vectors in the Euclidean vector space $\mathbb{R}^2$. In this work, we restrict ourselves to symmetric tensors. A *tensor field* $\mathbf{T}$ assigns to each position $x = (x_1, x_2) \in \mathbb{R}^2$ a symmetric tensor $\mathbf{T}(x) = T$. Let $\mathscr{T}$ denote the space of 2D symmetric second-order tensors over $\mathbb{R}^2$. In matrix form, with respect to a given basis of $\mathbb{R}^2$, a tensor field $\mathbf{T}$ is defined as

$$\mathbf{T} : \mathbb{R}^2 \to \mathscr{T}, \mathbf{T}(x) = T = \begin{bmatrix} t_{11} & t_{12} \\ t_{12} & t_{22} \end{bmatrix}. \tag{2}$$

The tensor T at x is fully specified by two orthogonal eigenvectors v_i at x and its two associated real eigenvalues λ_i, defined by the eigenvector equation $Tv_i = \lambda_i v_i$ (for $i \in \{1, 2\}$) with $v_i \in \mathbb{R}^2$ and $v_i \neq 0$. By imposing an ordering of $\lambda_1 \geq \lambda_2$, the normalized eigenvectors e_1 (resp. e_2) associated with λ_1 (resp. λ_2) are referred to as the *major* (resp. *minor*) eigenvectors.

Degenerate Points At points x where the eigenvalues of $\mathbf{T}(x)$ are different $\lambda_1 \neq \lambda_2$, the eigenspace of λ_i (for $i \in \{1, 2\}$) is the union of the zero vector and the set of all eigenvectors corresponding to eigenvalue λ_i, which is a one-dimensional subspace of $\mathbb{R}^2$. Such points are considered *non-degenerate points* of the tensor field $\mathbf{T}$. At these points, the tensor can then be expressed as

$$T = \lambda_1 e_1 \otimes e_1 + \lambda_2 e_2 \otimes e_2 \tag{3}$$

where $\otimes$ denotes the tensor product of the normalized eigenvectors e_i. For point $x_0 \in \mathbb{R}^2$ where $\lambda_1(x_0) = \lambda_2(x_0) = \lambda$, its associated tensor is proportional to the unit tensor and the corresponding eigenspace is the entire vector space $\mathbb{R}^2$. Its matrix representation is independent from the frame of reference, given as

$$T(x_0) = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}.$$

The points x_0 are called *degenerate points*. In the following sections, we assume that these points are isolated points in $\mathbb{R}^2$, which is usually the case. While the degenerate points are isotropic points exhibiting a high symmetry, they are structurally the most important features for the eigenvector fields.

Real Projective Line Before we proceed, we need the notion of real projective line and the homeomorphism between the real projective line and the circle. The real projective line, denoted as $\mathbb{RP}^1$ (or $\mathbb{P}^1$ for short), can be thought of as the set of lines through the origin of $\mathbb{R}^2$, formally $\mathbb{P}^1 := (\mathbb{R}^2 \setminus \{0\})/\sim$, for the equivalence relation $x \sim y$ iff $x = cy$ for some nonzero $c \in \mathbb{R}$. We sketch the proof below for $\mathbb{P}^1$ being homeomorphic to a circle $\mathbb{S}^1$, via $\mathbb{P}^1 \simeq (\mathbb{S}^1/\sim) \simeq \mathbb{S}^1$.

The quotient topology of a real projective line can be described by the mapping $\eta : \mathbb{R}^2 \setminus \{0\} \to \mathbb{P}^1$ that sends a point $x \in \mathbb{R}^2 \setminus \{0\}$ to its equivalent class $[x]$. η is surjective and has the property that $\eta(x) = \eta(y)$ iff $x \sim y$. Restricting such a mapping to $\mathbb{S}^1$, we obtain a mapping $\eta|_{\mathbb{S}^1} : \mathbb{S}^1 \to \mathbb{P}^1$ that identifies the two antipodal points. We now consider $\mathbb{S}^1$ as $\{z \in \mathbb{C} \mid ||z|| = 1\}$. Then we have $\eta|_{\mathbb{S}^1}(z) = [z]$. It is easy to show that $\eta|_{\mathbb{S}^1}$ defines a homeomorphism between $\mathbb{P}^1$ and $\mathbb{S}^1/\sim$ (where $\sim$ describes the equivalence of $z \sim -z$) since $\eta|_{\mathbb{S}^1}$ has the property that $U \subset (\mathbb{S}^1/\sim)$ is open (w.r.t. quotient topology on $\mathbb{S}^1/\sim$) iff $(\eta|_{\mathbb{S}^1})^{-1}(U)$ is open in $\mathbb{R}^2 \setminus \{0\}$.

Now consider the mapping $\theta : \mathbb{S}^1 \to \mathbb{S}^1$ defined as $\theta(z) = z^2$. θ is a continuous surjective function such that $\theta(z) = \theta(-z)$. Following the universal property (of quotient topology[2]), there exists a unique continuous homeomorphism $\phi : (\mathbb{S}^1/\sim) \to \mathbb{S}^1$ by having θ descending to the quotient. Therefore $(\mathbb{S}^1/\sim) \simeq \mathbb{S}^1$.

Eigenvector Fields In the following section, we describe the construction of an eigenvector field associated with the tensor field $\mathbf{T}$. As described before, a real 2D symmetric tensor T at x has two (not necessarily distinct) real eigenvalues $\lambda_1 \geq \lambda_2$ with associated eigenvectors v_1 and v_2. It is important to note that neither norm nor orientation is defined for the eigenvectors via the eigenvector equation, that is, if v_i is an eigenvector, then so is cv_i for any nonzero $c \in \mathbb{R}$. The normalized eigenvectors are denoted as e_i (for $i \in \{1, 2\}$), where $e_i \in \mathbb{S}^1$. This point of view is reflected through the interpretation of an eigenvector as elements of the real projective line. Thus we define the two eigenvector fields as the mapping $\boldsymbol{\psi}_i : \mathbb{R}^2 \to \mathbb{P}^1$ (for $i \in \{1, 2\}$), referred to as the major and minor eigenvector fields, respectively:

$$\boldsymbol{\psi}_i : \mathbb{R}^2 \to \mathbb{P}^1, x \mapsto \begin{cases} [e_i] \text{ if } \lambda_1 \neq \lambda_2 \\ [e_0] \text{ for degenerate points, if } \lambda_1 = \lambda_2 \end{cases} \tag{4}$$

$[e_0]$ is an arbitrarily chosen element of $\mathbb{P}^1$. Note that the eigenvector field is not continuous in degenerate points, and in general it is not possible to define $[e_0]$ such

[2]The quotient space $\mathbb{X}/\sim$ together with the quotient map $q : \mathbb{X} \to (\mathbb{X}/\sim)$ is characterized by the following universal property: If $g : \mathbb{X} \to \mathbb{Z}$ is a continuous map such that $a \sim b$ implies $g(a) = g(b)$ for all a and b in $\mathbb{X}$, then there exists a unique continuous map $f : (\mathbb{X}/\sim) \to \mathbb{Z}$ such that $g = f \circ q$. We say that g descends to the quotient.

that it becomes continuous. From now on, we restrict our attention to the major eigenvector field, referred to as the *eigenvector field* of $\mathbf{T}$, denoted as $\boldsymbol{\psi} := \boldsymbol{\psi}_1$, as the minor eigenvectors are always orthogonal and do not provide additional information in the 2D case.

The eigenvector fields and the degenerate points constitute the basic ingredients of the topological structure of a tensor field, and they build the basics for the bidirectional anisotropy vector fields that will be defined in Sect. 4.2.

4.2 Space of Bidirectional Anisotropy Vectors

In this section, we define the space of *bidirectional anisotropy vectors* equipped with a distance measure that is based on the L_2 norm of vectors. A comparison with the commonly used distance measure for tensors using the Frobenius norm shows that this space is topologically equivalent to the space of deviators $\mathscr{D}$. The bidirectional anisotropy vectors constitute a step toward the definition of an *anisotropy vector field* later used in the study of robustness.

Bidirectional Anisotropy Vectors We define bidirectional anisotropy vectors as bidirectional vectors ω whose direction is defined by the equivalence class of the major eigenvector $[e_1]$ and a norm given by the tensor anisotropy A (e.g., $A = |\lambda_1 - \lambda_2|$). Formally, we consider these vectors as elements of $\mathbb{P}^1 \times \mathbb{R}_{\geq 0}$. Degenerate points, that is, points with zero anisotropy and an undefined major eigenvector, are represented as the zero vectors.

Let $\mathscr{T}$ be the space of 2D symmetric tensors over $\mathbb{R}^2$. For each tensor $T \in \mathscr{T}$, we define the bidirectional anisotropy vector by the following mapping (Fig. 3):

$$\begin{aligned} &\Omega : \mathscr{T} \to \mathbb{P}^1 \times \mathbb{R}_{\geq 0} \\ &\quad T \mapsto \Omega(T) = \omega = \begin{cases} ([e_1], A) \text{ if } \lambda_1 \neq \lambda_2 \\ ([e_0], 0) \text{ for degenerate points, if } \lambda_1 = \lambda_2 \end{cases} \end{aligned} \tag{5}$$

The space $\mathbb{P}^1 \times \mathbb{R}_{\geq 0}$ can also be interpreted as $(\mathbb{R}^2/\sim)$, for the equivalence relation $x \sim y$ iff $x = -y$. In this setting, ω is equal to the equivalence class $[Ae_1] = \{v, -v\}$ consisting of the two vectors $v = Ae_1 \in \mathbb{R}^2$ and $-v = -Ae_1 \in \mathbb{R}^2$ with $e_1 \in [e_1]$.

Distance Measure We now define a distance measure between two bidirectional vectors $\omega = \{v, -v\}$ and $\omega' = \{v', -v'\}$ with vector representatives v and v', respectively. See Fig. 4 for an illustration:

$$d(\omega', \omega) = \min(\|v - v'\|_2, \|v + v'\|_2). \tag{6}$$

Theorem 2 *The distance measure defined in Eq. (6) is a metric on the space of bidirectional vectors.*

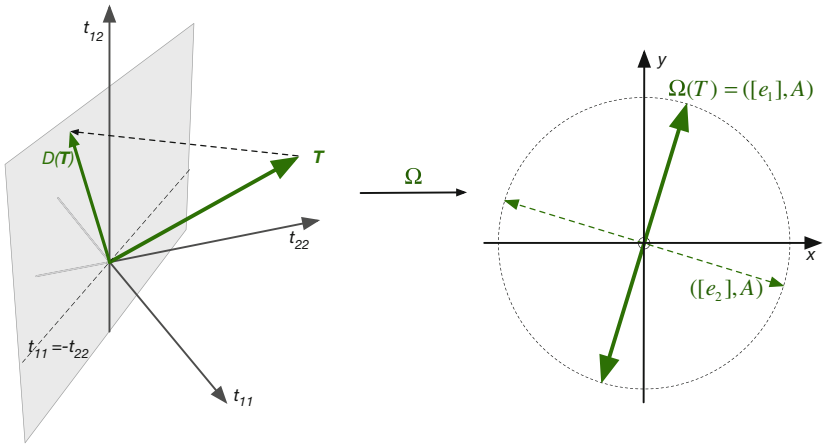


Fig. 3 Mapping of a tensor T to the bidirectional anisotropy vector defined by its anisotropy A and the eigenvector $[e_1]$. The *space on the right* represents the vector space spanned by the three independent components of the tensor. The *gray plane on the left* highlights the subspace of traceless tensors

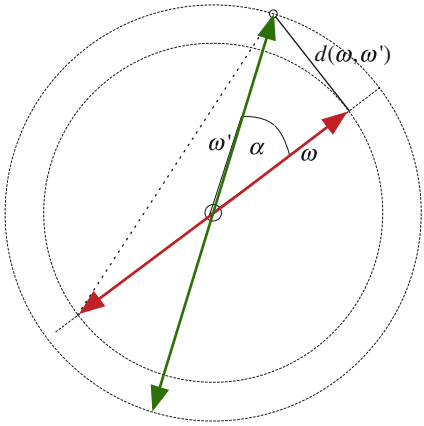


Fig. 4 The distance between two bidirectional vectors ω and ω' defined as the minimal distance between the members of their equivalence classes

Proof The expression in Eq. (6) is obviously independent of the arbitrarily chosen representatives v and v'. It is also obviously symmetric and non-negative. Therefore we have: $\mathrm{d}(\omega, \omega') = 0 \Leftrightarrow \min(\|v - v'\|_2, \|v + v'\|_2) = 0 \Leftrightarrow v = v'$ or $v = -v' \Leftrightarrow \omega = \omega'$. Furthermore the triangle inequality is satisfied (see Appendix 6 for derivations). Thus Eq. (6) defines a metric on the space of bidirectional vector fields. □

Space of Deviators The space of bidirectional anisotropy vectors is closely related to the space of deviatoric tensors $\mathscr{D}$. A deviator D is the traceless or anisotropic part of a tensor T:

$$D = T - \frac{tr(T)}{2} I, \tag{7}$$

where I represents the unit tensor. The space of 2D symmetric deviatoric tensors $\mathscr{D}$ is a subspace of the set of 2D symmetric tensors $\mathscr{T}$ (see Fig. 3). The eigenvectors of D coincide with the eigenvectors of T. Thus the deviator field has the same topology as the original tensor field. Its eigenvalues are $\delta_1 = -\delta_2 = \frac{1}{2}(\lambda_1 - \lambda_2)$. The most commonly used norm in $\mathscr{T}$ is the Frobenius norm. For the deviator, the Frobenius norm is $\|D\|_F = \frac{1}{2}|\lambda_1 - \lambda_2|$, which corresponds to an anisotropy measure (shear stress) that is typically used for failure analysis in mechanical engineering and will be used as the anisotropy measure A below, that is, let $A = |\lambda_1 - \lambda_2|$. Based on the Frobenius norm, degenerate points are the points x_0 at which $\|D(x_0)\|_F = 0$. The Frobenius norm therefore induces a metric on $\mathscr{D}$, that is, for $D, D' \in \mathscr{D}$:

$$d_F(D, D') = \|D - D'\|_F. \tag{8}$$

Deviator and Bidirectional Vectors If we restrict the mapping $\Omega|$ defined in Eq. (5) to the space of deviatoric tensors $\mathscr{D}$, the resulting mapping $\Omega|_{\mathscr{D}}$ is one-to-one. The inverse mapping is then defined by

$$\begin{aligned} (\Omega|_{\mathscr{D}})^{-1} : \mathbb{P}^1 \times \mathbb{R}_{\geq 0} &\to \mathscr{D} \\ \omega = [Ae] &\mapsto D = \frac{\sqrt{A}}{2}\ e \otimes e - \frac{\sqrt{A}}{2}\ e_\perp \otimes e_\perp. \end{aligned} \tag{9}$$

Here $e_\perp$ represents a normalized vector orthogonal to e. It can be seen immediately that this expression is independent of the sign of the representative vector e and thus is well-defined. For $\omega = [0e_0]$, Eq. (9) results in a zero tensor that is independent of the chosen vector e_0.

Theorem 3 *For the above defined metric Eq. (6) on the space of bidirectional anisotropy vectors and the Frobenius metric Eq. (8) on the space of deviators, we have*

$$\Omega(D') \in B_r(\Omega(D)) \Rightarrow D' \in B_{r'}(D) \tag{10}$$

with $r' = \sqrt{5}r$. *For the opposite direction, we have*

$$D' \in B_{r'}(D) \Rightarrow \Omega(D') \in B_r(\Omega(D)). \tag{11}$$

Thus the mapping defined in Eq. (5) is continuous, and the space of tensor deviators and the bidirectional anisotropy vectors are topologically equivalent.

Proof Let D and D' be two symmetric, traceless 2D tensors with major eigenvalues $(\frac{1}{\sqrt{2}}\lambda, \frac{-1}{\sqrt{2}}\lambda)$ and $(\frac{1}{\sqrt{2}}\mu, \frac{-1}{\sqrt{2}}\mu)$, respectively, as well as their corresponding eigenvectors $[e_i]$ and $[f_i]$, for $i = 1, 2$. Their norms are given by $\|D\|_F^2 = \lambda^2$ and $\|D'\|_F^2 = \mu^2$. The corresponding bidirectional anisotropy vectors are defined as $\omega = \Omega(D) = [Ae_1]$ with $A = \lambda$ and $\omega' = \Omega(D') = [A'f_1]$ with $A' = \mu$.

In order to compare the Frobenius distance between deviators and the distance between bidirectional anisotropy vectors, we first bring them into similar forms. Therefore we decompose the distance into two parts (see Appendix 6 for a derivation).

$$\begin{aligned} d_F^2(D, D') &= \|D - D'\|_F^2 = \|D\|_F^2 + \|D'\|_F^2 - 2(D : D') \\ &= (\lambda - \mu)^2 + 4\lambda\mu \ \sin^2\alpha \end{aligned} \tag{12}$$

where $D : D'$ is the inner product of the tensors and α is the angle between the major eigenvectors. One can interpret this decomposition as having a shape-related part $(\lambda - \mu)^2$ and a direction-related part $4\lambda\mu \ \sin^2\alpha$. A similar decomposition has been proposed by Zhang et al. [29] for the comparison of normalized tensors.

The distance defined between the bidirectional anisotropy vectors (Eq. (6)) is based on the L_2 distance between vectors. Therefore we will now express the L_2 distance between two vectors with length A and A' accordingly (see Appendix 6 for a derivation):

$$\begin{aligned} d^2(\omega, \omega') &= (A - A')^2 + 4A\,A' \ \sin^2(\alpha/2) \\ &= (\lambda - \mu)^2 + 4\lambda\mu \ \sin^2(\alpha/2). \end{aligned} \tag{13}$$

As in Eq. (12), we can interpret Eq. (13) as having a shape-related part and a distance-related part. The shape-related parts in Eqs. (12) and (13) are identical; however, the direction-related parts differ with respect to the angles.

Now let $\Omega(D') \in B_r(\Omega(D))$ be a bidirectional vector in the r-ball of $\Omega(D)$ for some value $r \in \mathbb{R}_{>0}$, which means

$$d^2(\Omega(D), \Omega(D')) = (\lambda - \mu)^2 + 4\lambda\mu \ \sin^2(\alpha/2) \leq r^2.$$

It follows that $(\lambda - \mu)^2 \leq r^2$ and $4\lambda\mu \ \sin^2(\alpha/2) \leq r^2$. From this we can derive an upper limit for the Frobenius distance of the two tensors D' and D. Combining the relation $\sin\alpha = 2\sin(\alpha/2)\cos(\alpha/2)$ and the fact that $\alpha \in [0, \pi/2]$, we have $\sin\alpha \leq 2\sin(\alpha/2)$. It follows for the deviators:

$$d_F^2(D, D') = \underbrace{(\lambda - \mu)^2}_{\leq r^2} + \underbrace{4\lambda\mu \ \sin^2\alpha}_{\leq 4\lambda\mu(4\sin^2(\alpha/2)) \leq 4r^2} \leq 5r^2 \Rightarrow D' \in B_{\sqrt{5}r}(D).$$

The opposite direction is trivially satisfied, since $\sin^2(\alpha/2) \leq \sin^2\alpha$ for all $\alpha \in [0, \pi/2]$, and $d^2(\Omega(D), \Omega(D')) \leq d_F^2(D, D')$. ∎

Bidirectional Anisotropy Vector Field In accordance with the tensor field, we now define a bidirectional anisotropy vector field.

A *bidirectional anisotropy vector field* $\boldsymbol{\omega}$ assigns to each position $x = (x_1, x_2) \in \mathbb{R}^2$ a bidirectional anisotropy vector ω. The map Ω can be used to convert the tensor field $\mathbf{T}$ into a bidirectional anisotropy vector field $\boldsymbol{\omega}(x) = \Omega(\mathbf{T}(x)) = (\Omega \circ \mathbf{T})(x)$. If the tensor field is continuous, then the bidirectional anisotropy vector field is also continuous, as it is a concatenation of two continuous mappings.

4.3 The Anisotropy Vector Field

In the following section, we define an *anisotropy vector field* $\tilde{\boldsymbol{\omega}}$ as a mapping from $\mathbb{R}^2$ to $\mathbb{S}^1 \times \mathbb{R}_{\geq 0}$. An element in $\mathbb{S}^1 \times \mathbb{R}_{\geq 0}$ can be understood as a vector in $\mathbb{R}^2$ represented in polar coordinates. Such a vector field $\tilde{\boldsymbol{\omega}}$ serves two purposes. First, we use it to specify the perturbation of a tensor field. Second, we use it to define the tensor index following the degree theory.

To define anisotropy vectors, we first define a mapping $\tilde{\Omega}$ from the space of tensors $\mathscr{T}$ to $\mathbb{S}^1 \times \mathbb{R}_{\geq 0}$ by lifting the first part of the mapping Ω from $\mathbb{P}^1$ to its covering space $\mathbb{S}^1$ using the mapping $\phi : \mathbb{P}^1 \to \mathbb{S}^1$ defined in Sect. 4.1. According to Eq. (5), we define

$$\begin{aligned} \tilde{\Omega} : \mathscr{T} &\to \mathbb{S}^1 \times \mathbb{R}_{\geq 0} \\ T &\mapsto \tilde{\Omega}(T) = \tilde{\omega} = ((\phi \times Id) \circ \Omega) = Ae_1^2. \end{aligned} \tag{14}$$

Here $e_1 \in \mathbb{C}$ is an eigenvector representative of $[e_1]$ considered as a complex number. It can be easily seen that $Ae_1^2 = A(-e_1)^2$ is independent of the choice of the representative.

Theorem 4 *The above defined mapping (Eq. (14)) restricted to the space of deviators $\tilde{\Omega}|_{\mathscr{D}}$ is an isometry with respect to the L_2-norm in $\mathbb{R}^2$ and the Frobenius norm in $\mathscr{D}^2$.*

Proof The proof follows directly from Eqs. (12) and (13). Let D and D' be two symmetric, traceless 2D tensors defined as above. We have

$$d^2(\tilde{\Omega}(D), \tilde{\Omega}(D')) = (\lambda - \mu')^2 + 4\lambda\mu \ \sin^2((2\alpha)/2) = d_F^2(D, D'),$$

since squaring a complex number doubles the angle. □

We would like to point out that thus defined vectors are less appropriate for geometric representations of the tensor and their directions are not directly correlated to the principal directions of the tensor. The explicit direction depends on the frame of reference chosen for the representation of the complex numbers (see also Sect. 4.4).

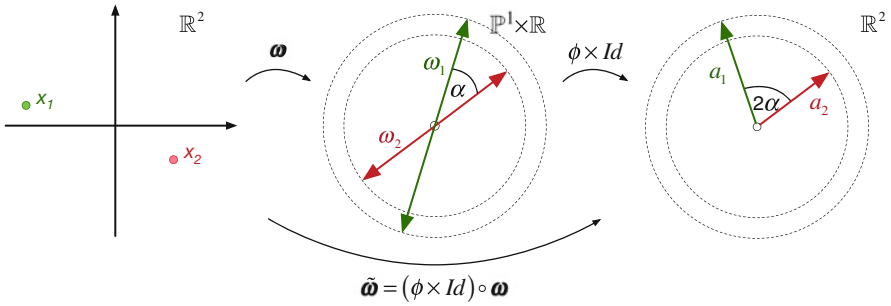


Fig. 5 The concatenation of the mapping defined by the tensor field and the homomorphism ϕ between $\mathbb{P}$ and $\mathbb{S}$ defined in Sect. 4.1 is a continuous mapping from $\mathbb{R}^2$ to $\mathbb{R}^2$. It defines a vector field on $\mathbb{R}^2$

Anisotropy Vector Field With these definitions, we can define the *anisotropy vector field* $\tilde{\boldsymbol{\omega}}$, which serves as the basis for the application of the concept of robustness to tensor fields. $\tilde{\boldsymbol{\omega}}$ assigns to each position $x \in \mathbb{R}^2$ a vector $\tilde{\omega}$. Thereby the map $\tilde{\Omega}$ is used to convert the tensor field $\mathbf{T}$ into a vector field $\tilde{\boldsymbol{\omega}}(x) = \tilde{\Omega}(\mathbf{T}(x)) = (\tilde{\Omega} \circ \mathbf{T})(x)$. If the tensor field is continuous, then the anisotropy vector field is also continuous as a concatenation of two continuous mappings (see Fig. 5).

$$\tilde{\boldsymbol{\omega}} : \mathbb{R}^2 \to \mathbb{S}^1 \times \mathbb{R}_{\geq 0} (\simeq \mathbb{R}^2), \tilde{\boldsymbol{\omega}}(x) = ((\phi \times Id) \circ \boldsymbol{\omega})(x) \tag{15}$$

Therefore $Id : \mathbb{R} \to \mathbb{R}$ is the identity map.

4.4 Notes on the Topology of the Anisotropy Vector Field

When looking at the vector field derived from the tensor field in Sect. 4.3, an obvious question is how its vector field topology relates to the tensor topology of the original tensor field. From the construction of the anisotropy vector field, it is clear that its critical points, zeros of the vector field, coincide with the degenerate points of the tensor field.

These points, however, constitute only a part of the topology. The second essential part is the connecting separatrices. For the vector case, these are the integral lines of the vector field. For tensor fields, the separatrices are tensor lines, which follow one eigenvector field. The structure in the vicinity of the critical points is characterized by its index (compare to Sect. 5.1). In our setting, the index of the tensor field degenerate points and the index of the anisotropy vector field are related by the degree two mapping ϕ defined in 4.1. Thus a wedge point in the tensor field (tensor index $+1/2$) is mapped to sources/sinks (vector index $+1$) and trisectors (tenors index $-1/2$) are mapped to saddle points (vector index -1). In general a degenerate point of tensor index i is mapped to a critical point of index $2i$ (Fig. 6).

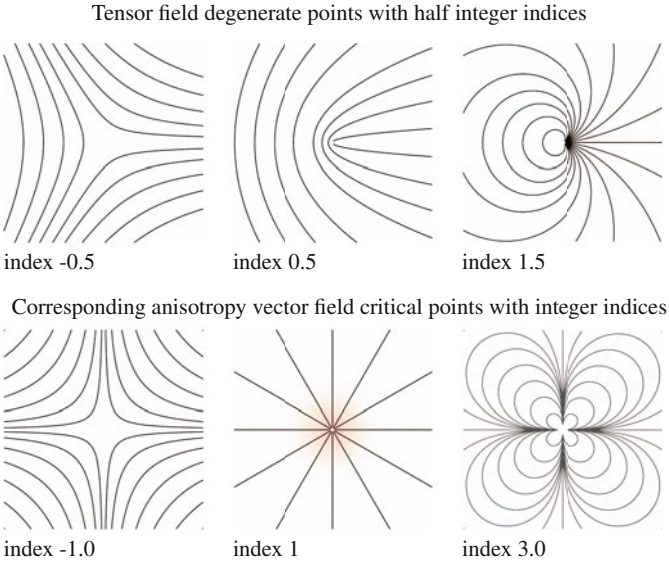


Fig. 6 Change of the structure of the field when mapping the tensor field to the anisotropic vector field. Examples for isolated degenerate points

This mapping gives rise to a very distinct structure in which different critical points will be connected. Integral lines in the vector field do not coincide with the integral lines for the tensor field.

What is important, however, for our discussion is that the stability of the critical points and the degenerated points in terms of robustness is the same.

5 Robustness for Tensor Fields

Similar to vector field topology, one of the major challenges in tensor field topology is the complexity of the topological structure. A large part of the topological structure originates from extended isotropic regions and such a structure is very sensitive to small changes in the data. Therefore we would like to have a *stable* topological skeleton representing the core structure of the data. Previous attempts to simplify the tensor field topology have relied on heuristics that lack a clean mathematical framework. Motivated by the notion of robustness based on the well group theory for the vector fields, we extend such a concept to 2D symmetric second-order tensor fields. In this section, we connect the indexes of degenerate points with the degree theory in Sect. 5.1, define tensor field perturbation in Sect. 5.2, and generalize robustness to tensor field topology in Sect. 5.3. Our main contributions are threefold: We interpret the notion of tensor index under the setting of degree theory; we define tensor field perturbations and make precise connections between such perturbations with the perturbations of bidirectional anisotropy vector fields; and we generalize the notion of robustness to tensor field topology.

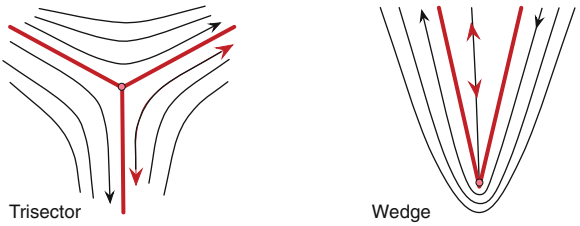


Fig. 7 Basic structure of eigenvector fields in the vicinity of degenerate points. It can easily be seen that it is not possible to orient the tensor lines in a continuous way

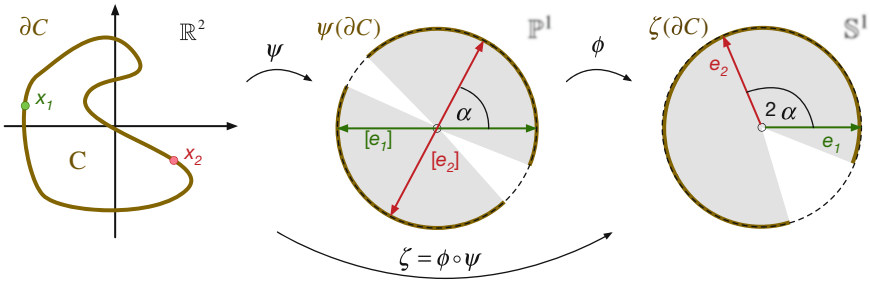


Fig. 8 The mapping $\xi = \phi \circ \psi$ defines a continuous mapping from $\mathbb{R}^2$ to $\mathbb{S}^1$, which corresponds to lifting the mapping ϕ from $\mathbb{P}^1$ to the covering space $\mathbb{S}^1$

5.1 Indexes of Degenerate Points and Degree Theory

Index of Degenerate Points Similar to the zeros of vector fields, we also consider the notion of index for these degenerate points. Delmarcelle [9] defines the index of a degenerate point $x \in \mathbb{R}^2$ as the number of "half-windings" an eigenvector performs when moved along a simple closed curve (i.e., a Jordan curve) enclosing the degenerate point. For linear fields, the structure of the eigenvector fields surrounding the degenerate points follows two characteristic patterns depending on their indexes (see Fig. 7).

Connection to Degree Theory The above definition of an index by Delmarcelle follows a geometric point of view considering the number of "half-windings" of the eigenvectors. There is also, however, a close connection between the index of a degenerate point and the degree of a mapping as defined in algebraic topology. Since the degree plays an important role in the theory of robustness, we revisit the concept in Sect. 3 and provide here a formulation in terms of degree theory. The line of thought is similar to that of Trichoche ([25], page 55).

Consider a tensor field T defined on an orientable surface $\mathbb{M}$ (here $\mathbb{M} = \mathbb{R}^2$), and suppose all degenerate points are isolated and finite in number. We would like to associate (via the theory of Hopf [14]) an index with each x of $\mathbb{M}$. We built a continuous mapping ξ by lifting the eigenvector field ψ to its covering space $\mathbb{S}^1$ (see Fig. 8). That is, $\xi = \phi \circ \psi$, where $\psi : \mathbb{R}^2 \to \mathbb{P}^1$ and $\phi : \mathbb{P}^1 \to \mathbb{S}^1$, that is, $\xi(x) = [e_1]^2$, where $[e_1]$ is a generator.

For the definition of the index of a degenerate point x, we consider the boundary of a region C enclosing x, that is, the curve ∂C with no other degenerate points in its interior. We define the index of x, $I_T(x)$ to be $\frac{1}{2}\deg(\xi|_{\partial C})$.

Poincaré-Hopf Theorem for Tensor Fields Delmarcelle has provided a tensor field equivalence of the Poincaré-Hopf theorem ([9], page 163).

Theorem 5 (Theorem 15, [9]) *The tensor index of a 2D orientable surface* $\mathbb{M}$ *relative to a tangent tensor field* T *with a finite number of degenerate points on* $\mathbb{M}$ *is equal to the Euler characteristic of* $\mathbb{M}$*. That is,* $\mathrm{I}_T(\mathbb{M}) = \sum_i \mathrm{I}_T(x_i) = \chi(\mathbb{M})$.

According to Hopf's result [14], whenever the continuous field of directions tangent to $\mathbb{M}$ is not zero at more than finitely many points x_i, we always have the above theorem [5].

5.2 r-Perturbation of Anisotropy Vector Field

Suppose we have two anisotropy vector fields f and h, derived from tensor fields $\mathbf{T}$ and $\mathbf{T}'$, respectively, that is, $f = \tilde{\boldsymbol{\omega}}$ and $h = \tilde{\boldsymbol{\omega}}'$. We define the distance between the two as

$$\mathrm{d}(f, h) = \sup_{x \in \mathbb{R}^2} ||f(x) - h(x)||_2.$$

We say a continuous mapping h is an *r-perturbation* of f, if $\mathrm{d}(f, h) \leq r$. In other words, for each point $x \in \mathbb{R}^2$, the point $h(x)$ lies within a disk of radius r centered at $f(x)$. See Fig. 2 for a geometric interpretation of an r-perturbation of the anisotropy vector field at a point x in the domain.

5.3 Robustness of Degenerate Points

Converting a tensor field $\mathbf{T}$ to its corresponding anisotropy vector field f greatly simplifies the extension of robustness from the vector field to the setting of the tensor field. First, the degenerate points of $\mathbf{T}$ correspond to the critical points of f; therefore f has no critical points in a path-connected region $C \subset \mathbb{R}^2$ iff $\mathbf{T}$ has no degenerate points in C. Second, the index of a degenerate point in $\mathbf{T}$ is half the degree of its corresponding critical point in f. Third, the r-perturbation of f relates to the perturbation of $\mathbf{T}$ via its projection $\mathbf{D}$ in a quantifiable way; an r-perturbation in f corresponds to an r-perturbation in $\mathbf{D}$.

We have conjectured that the robustness of degenerate points x for tensor fields $\mathbf{T}$ would resemble the robustness of its corresponding critical point f for the anisotropy

vector fields. Recall that, by definition, f is an anisotropy vector field, $f : \mathbb{R}^2 \to \mathbb{R}^2$, $f_0 = ||f||_2 : \mathbb{R}^2 \to \mathbb{R}$, $\mathbb{F}_r = f_0^{-1}(-\infty, r]$. Let h be another anisotropy vector field $h : \mathbb{R}^2 \to \mathbb{R}^2$. We would prove the following lemmas, whose proofs are identical to the proofs used for results in [28] (Corollary 1.1 and Corollary 1.2 in the supplemental material) with respect to vector field perturbation. We include the proofs here for completeness.

Lemma 4 (Nonzero Degree Component for Tensor Field Perturbation) *Let r be a regular value of f_0 and C a connected component of $\mathbb{F}_r$ such that* $\deg(C) \neq 0$. *Then for any δ-perturbation h of f, where $\delta < r$, the sum of the degrees of the critical points in $h^{-1}(0) \cap C$ is* $\deg(C)$.

Proof Before we illustrate the details of the proof, we need to provide a rigorous definition of the degree of a mapping.

Let $C \subseteq \mathbb{F}_r$ be a path-connected component of $\mathbb{F}_r$. Function f restricted to C, denoted $f|_C : (C, \partial C) \to (B_r, \partial B_r)$, maps C to the closed ball B_r of radius r centered at the origin, where ∂ is the boundary operator. $f|_C$ induces a homomorphism on the homology level, $f_*|_C : \mathsf{H}(C, \partial C) \to \mathsf{H}(B_r, \partial B_r)$. Let μ_C and μ_{B_r} be the generators of $\mathsf{H}(C, \partial C)$ and $\mathsf{H}(Br, \partial B_r)$, respectively. The *degree* of C (more precisely the degree of $f|_C$), $\deg(C) = \deg(f|_C)$, is the unique integer such that $f_*|_C(\mu_C) = \deg(C) \cdot \mu_{B_r}$. Furthermore we have the function restricted to the boundary, that is, $f|_{\partial C} : \partial C \to \mathbb{S}^1$. It was shown that $\deg(f|_C) = \deg(f|_{\partial C})$ ([7], Lemma 1).

Consider the following diagram for any δ-perturbation h of f, where $\delta < r$:

$$\begin{array}{ccc} \mathsf{H}(C, \partial C) & \xrightarrow{i_*} & \mathsf{H}(C, C - h^{-1}(0)) \\ \downarrow f_*|_C & & \downarrow h_*|_0 \\ \mathsf{H}(B_r, \partial B_r) & \xrightarrow{j_*} & \mathsf{H}(B_r, B_r - \{0\}). \end{array} \tag{16}$$

i_* and j_* are homomorphisms induced by space-level inclusions $i : (C, \partial C) \to (C, C - h^{-1}(0))$ and $j : (B_r, \partial B_r) \to (B_r, B_r - \{0\})$. j_* is also an isomorphism. The vertical maps $f_*|_C$ and $h_*|_0$ are induced by f and h with restrictions, respectively. Therefore the diagram commutes.

Suppose r is a regular value and $\deg(C) \neq 0$. Then by commutativity, the sum of degrees of the critical points in $h^{-1}(0) \cap C$ is $\deg(C)$. ∎

Lemma 5 (Zero Degree Component for Tensor Field Perturbation) *Let r be a regular value of f_0 and C a connected component of $\mathbb{F}_r$ such that* $\deg(C) = 0$. *Then there exists an r-perturbation h of f such that h has no degenerate points in C, $h^{-1}(0) \cap C = \emptyset$. In addition, h equals f except possibly within the interior of C.*

Proof The proof follows the commutative diagram above (Eq. (16)) for any r-perturbation h of f. Suppose r is a regular value. Then well groups $\mathsf{U}(r - \delta)$ and

$\mathsf{U}(r+\delta)$ are isomorphic for all sufficiently small $\delta > 0$. Suppose $\deg(C) = \deg(f|_C) = \deg(f|_{\partial C}) = 0$. Then following the Hopf Extension Theorem ([13], page 145), if the function $f|_{\partial C} : \partial C \to \mathbb{S}^1$ has degree zero, then f can be extended to a globally defined map $g : C \to \mathbb{S}^1$ such that g equals f when both are restricted to ∂C. Now we define a perturbation $h : \mathbb{R}^2 \to \mathbb{R}^2$ such that $h = 0.5 \cdot f + 0.5 \cdot g$. h is the midpoint on a straight line homotopy between f and g. By definition $\mathrm{d}(h,f) \leq r$, so h is an r-perturbation of f. In addition, $h^{-1}(0) \cap C$ is empty. □

Remark One important aspect of well group theory is that the well group is defined to be the intersection of the images of j_h for *all* r-perturbation h of f (Eq. (1)). Given f as an anisotropy vector field, we introduce an r-perturbation h of f. We would need to make sure that *any* such h is itself a valid anisotropy vector field. That is, for any r-perturbation h of f, there exists a corresponding tensor field $\mathbf{T}$ from which an anisotropy vector field h can be derived. This is true based on derivations in Sect. 4.

6 Discussion

There are a few challenges in extending our framework to a 3D symmetric tensor field. The notion of deviator can be generalized to 3D, but the notion of anisotropy vector field does not generalize to 3D. The lack of such a notion poses a challenge in studying robustness for 3D symmetric tensor field topology via transformation of the data to the anisotropy vector field. We suspect a possible solution is to define perturbations with respect to the bidirectional anisotropy vector field derived from eigenvector fields.

An important contribution of this paper is the conversion from a tensor field $\mathbf{T}$ to its corresponding anisotropy vector field f. There is a one-to-one correspondence between the degenerate points of $\mathbf{T}$ and the critical points of f. However, as shown in Fig. 6, the topology of $\mathbf{T}$ and that of f obviously do not agree. Understanding their differences and the consequences will be an interesting direction.

The main motivation of extending robustness to a 2D symmetric tensor field is that it would lead to simplification schemes for tensor field data. In general, topology-based simplification techniques pair the topological features for simplification via the computation of topological skeleton, which can be numerically unstable. In contrast, the proposed robustness-based method is independent of the topological skeleton and, thus, is insensitive to numerical error.

Appendix A: Notations

Symbol	Description
$\mathscr{T}$	Space of 2D symmetric second-order tensors
$\mathscr{D} \subset \mathscr{T}$	Space of 2D symmetric second-order deviators
$\mathbb{RP}^1$ or $\mathbb{P}^1$	Real projective line
T, D	Tensors
$\mathbf{T}, \mathbf{D} : \mathbb{R}^2 \to \mathscr{T}$	Tensor fields
v_i	Eigenvectors
e_i	Normalized (unit) eigenvectors
λ_i, μ_i	Eigenvalues
A (e.g. $= \lvert\lambda_1 - \lambda_2\rvert$)	Anisotropy measure
$[e_i] \in \mathbb{P}^1$	Equivalence class of unit eigenvectors
$\boldsymbol{\psi}_i : \mathbb{R}^2 \to \mathbb{P}^1, i = 1, 2,$	Major ($i = 1$) and minor ($i = 2$) eigenvector fields (direction fields, no magnitude)
$\boldsymbol{\psi} = \boldsymbol{\psi}_1 : \mathbb{R}^2 \to \mathbb{P}^1$	(Major) eigenvector field
ω	Bidirectional anisotropy vector
$\Omega : \mathscr{T} \to \mathbb{P}^1 \times \mathbb{R}_{\geq 0}$	Mapping that assigns a bidirectional anisotropy vector to the tensor
$\boldsymbol{\omega} : \mathbb{R}^2 \to \mathbb{P}^1 \times \mathbb{R}_{\geq 0}$	Bidirectional anisotropy vector field (direction fields with magnitude, not a traditional vector field)
$\tilde{\Omega} : \mathscr{T} \to \mathbb{R}^2$	Mapping that assigns an anisotropy vector to a tensor
$\tilde{\boldsymbol{\omega}} : \mathbb{R}^2 \to \mathbb{R}^2$	Anisotropy vector field (traditional vector field)
$\tilde{\omega}(x) \in \mathbb{R}^2$	Anisotropy vector
$\phi : (\mathbb{P}^1 \simeq (\mathbb{S}^1/\sim)) \to \mathbb{S}^1$	Degree 2 mapping
$\xi = \phi \circ \psi : \mathbb{R}^2 \to \mathbb{S}^1$	Mapping used for defining the index of degenerate points

Appendix B: Triangle Inequality for the Distance Measure Between Bidirectional Anisotropy Vectors (Eq. (6))

Let ω, ω' and ω'' be bidirectional anisotropy vectors as defined in Eq. (5) with vector representatives v, w, and u respectively. Recall the distance measure is defined as $d(\omega, \omega') = \min(\|v - w\|_2, \|v + w\|_2)$. Therefore we have (Fig. 9):

$$
\begin{aligned}
&\mathrm{d}(\omega, \omega') + \mathrm{d}(\omega', \omega'') \\
&\quad = \min(\quad \|v - w\|_2, \|v + w\|_2) + \min(\|w - u\|_2, \|w + u\|_2) \\
&\quad \geq \min(\quad \|v - w\|_2 + \|w - u\|_2, \|v - w\|_2 + \|w + u\|_2, \\
&\qquad\qquad \|v + w\|_2 + \|w - u\|_2, \|v + w\|_2 + \|w + u\|_2) \\
&\quad = \min(\quad \|v - w\|_2 + \|w - u\|_2, \|v - w\|_2 + \|w - (-u)\|_2, \\
&\qquad\qquad \|v - (-w)\|_2 + \|(-w) - (-u)\|_2, \|v - (-w)\|_2 - \|(-w) - u\|_2) \\
&\quad \geq \min(\quad \|v - u\|_2, \|v - (-u)\|_2, \|v - (-u)\|_2, \|v - u\|_2) \\
&\quad = \min(\quad \|v - u\|_2, \|v + u\|_2) \\
&= \mathrm{d}(\omega, \omega'')
\end{aligned}
$$

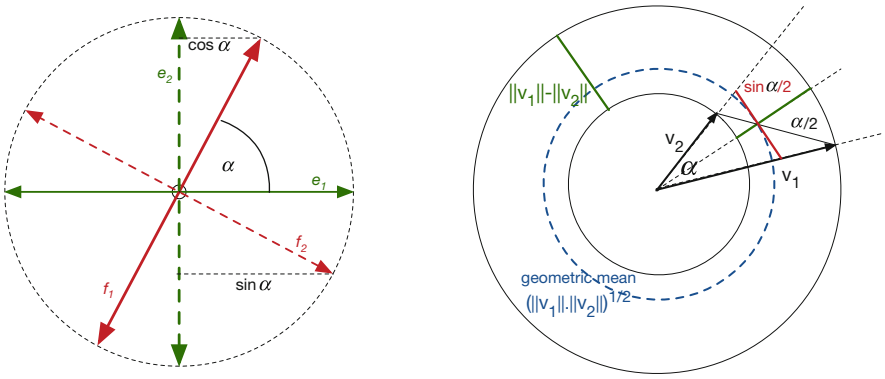


Fig. 9 *Left*: eigenvectors of D and D' and angle α by definition. *Right*: geometric interpretation of the vector distance decomposed in radial and directional parts

Appendix C: Derivations for Eqs. (12) and (13)

The inner product of two symmetric tensors T and T' is defined as $T : T' = \sum_{ij} t_{ij} t'_{ij}$. It can be expressed in terms of eigenvectors and eigenvalues $\sum_{ks} \lambda_k \mu_s (e_k \cdot f_s)^2$ (see e.g., [4]). Here $\cdot$ denotes the standard scalar product of vectors. For 2D deviatoric tensors D and D' with eigenvalues $(\frac{1}{\sqrt{2}}\lambda, \frac{-1}{\sqrt{2}}\lambda)$ and $(\frac{1}{\sqrt{2}}\mu, \frac{-1}{\sqrt{2}}\mu)$, respectively, and their corresponding eigenvectors e_i and f_i (for $i = 1, 2$), this yields

$$\begin{aligned}(D : D') &= \frac{1}{2}\lambda\mu \left((e_1 \cdot f_1)^2 - (e_1 \cdot f_2)^2 - (e_2 \cdot f_1)^2 + (e_2 \cdot f_2)^2\right) \\ &= \lambda\mu \left(\cos^2\alpha - \sin^2\alpha\right)\end{aligned}$$

$$\begin{aligned}d_F^2(D, D') = \|D - D'\|_F^2 &= \|D\|^2 + \|D'\|^2 - 2(D : D') \\ &= \|D\|^2 + \|D'\|^2 - 2\lambda\mu \left(\cos^2\alpha - \sin^2\alpha\right) \\ &= \lambda^2 + \mu^2 - 2\lambda\mu \left(1 - 2\sin^2\alpha\right) \\ &= (\lambda - \mu)^2 - 4\lambda\mu \sin^2\alpha\end{aligned}$$

A similar construction for 2D vectors v_1 and v_2 using the trigonometric equality $1 - \cos(2\beta) = 2\sin^2(\beta)$ gives:

$$\begin{aligned}d^2(v_1, v_2) &= \|v_1\|^2 + \|v_2\|^2 - 2(v_1 \dot{v}_2) \\ &= (\|v_1\| + \|v_2\|)^2 - 2\|v_1\| \, \|v_2\| - 2(v_1 \cdot v_2) \\ &= (\|v_1\| + \|v_2\|)^2 - 2\|v_1\| \, \|v_2\| \, (1 - 2\cos\alpha) \\ &= (\|v_1\| + \|v_2\|)^2 - 4\|v_1\| \, \|v_2\| \sin^2(\alpha/2)\end{aligned}$$

References

1. Alliez, P., Cohen-Steiner, D., Devillers, O., Levy, B., Desbrun, M.: Anisotropic polygonal remeshing. In: Siggraph '03 - ACM Transactions on Graphics (TOG), vol. 22, issue 3, pp. 485–493 (2003)
2. Auer, C., Hotz, I.: Complete tensor field topology on 2D triangulated manifolds embedded in 3D. Comput. Graph. Forum **30**(3), 831–840 (2011)
3. Auer, C., Stripf, C., Kratz, A., Hotz, I.: Glyph- and texture-based visualization of segmented tensor fields. In: International Conference on Information Visualization Theory and Applications (IVAPP'12), (2012)
4. Basser, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. J. Magn. Reson. Imaging **111**(3), 209–219 (1996)
5. Berger, M.: Geometry Revealed: A Jacob's Ladder to Modern Higher Geometry. Springer, Heidelberg (2010)
6. Carr, H., Snoeyink, J., Axen, U.: Computing contour trees in all dimensions. In: SODA - Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 918–926 (2000)
7. Chazal, F., Patel, A., Skraba, P.: Computing well diagrams for vector fields on R^n. Appl. Math. Lett. **25**(11), 1725–1728 (2012)
8. Chen, G., Mischaikow, K., Laramee, R., Pilarczyk, P., Zhang, E.: Vector field editing and periodic orbit extraction using Morse decomposition. IEEE Trans. Vis. Comput. Graph. **13**(4), 769–785 (2007)
9. Delmarcelle, T.: The visualization of second-order tensor fields. Ph.D. thesis, Stanford University (1994)
10. Edelsbrunner, H., Letscher, D., Zomorodian, A.: Topological persistence and simplification. Discret. Comput. Geom. **28**, 511–533 (2002)
11. Edelsbrunner, H., Morozov, D., Patel, A.: The stability of the apparent contour of an orientable 2-manifold. In: Topological Methods in Data Analysis and Visualization, pp. 27–41. Springer, Heidelberg (2010)
12. Edelsbrunner, H., Morozov, D., Patel, A.: Quantifying transversality by measuring the robustness of intersections. Found. Comput. Math. **11**, 345–361 (2011)
13. Guillemin, V., Pollack, A.: Differential Topology. Prentice-Hall, Englewood Cliffs (1974)
14. Hopf, H.: Vektorfelder in n-dimensionalen Mannigfaltigkeiten. Math. Ann. **96**, 225–250 (1926)
15. Hotz, I., Sreevalsan-Nair, J., Hagen, H., Hamann, B.: Tensor field reconstruction based on eigenvector and eigenvalue interpolation. In: Hagen, H. (ed.) Scientific Visualization: Advanced Concepts. Dagstuhl Follow-Ups, vol. 1, pp. 110–123. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Wadern (2010)
16. Kälberer, F., Nieser, M., Polthier, K.: Quadcover–surface parameterization using branched coverings. Comput. Graph. Forum **26**(3), 375–384 (2007)
17. Knöppel, F., Crane, K., Pinkall, U., Schröder, P.: Globally optimal direction fields. ACM Trans. Graph. **32**(4) (2013)
18. Kratz, A., Auer, C., Stommel, M., Hotz, I.: Visualization and analysis of second-order tensors: moving beyond the symmetric positive-definite case. Comput. Graph. Forum–State Art Rep. **32**(1), 49–74 (2013)
19. McLoughlin, T., Laramee, R.S., Peikert, R., Post, F.H., Chen, M.: Over two decades of integration-based, geometric flow visualization. Comput. Graph. Forum **29**(6), 1807–1829 (2010)
20. Skraba, P., Rosen, P., Wang, B., Chen, G., Bhatia, H., Pascucci, V.: Critical point cancellation in 3D vector fields: Robustness and discussion. IEEE Trans. Vis. Comput. Graph. **22**(6), 1683–1693 (2016)
21. Skraba, P., Wang, B.: Interpreting feature tracking through the lens of robustness. In: Topological Methods in Data Analysis and Visualization III. Springer, Cham (2014)

22. Skraba, P., Wang, B., Chen, G., Rosen, P.: 2D vector field simplification based on robustness. In: Proceedings of IEEE Pacific Visualization Symposium (2014)
23. Skraba, P., Wang, B., Chen, G., Rosen, P.: Robustness-based simplification of 2D steady and unsteady vector fields. IEEE Trans. Vis. Comput. Graph. **21**(8), 930–944 (2015)
24. Sreevalsan-Nair, J., Auer, C., Hamann, B., Hotz, I.: Eigenvector-based interpolation and segmentation of 2D tensor fields. In: Topological Methods in Data Analysis and Visualization: Theory, Algorithms, and Applications (TopoInVis'09). Mathematics and Visualization. Springer, Heidelberg (2011)
25. Tricoche, X.: Vector and tensor field topology simplification, tracking and visualization. Ph.D. thesis, University of Kaiserslautern (2002)
26. Tricoche, X., Kindlmann, G., Westin, C.F.: Invariant crease lines for topological and structural analysis of tensor fields. IEEE Trans. Vis. Comput. Graph. **14**(6), 1627–1634 (2008)
27. Tricoche, X., Scheuermann, G., Hagen, H., Clauss, S.: Vector and tensor field topology simplification on irregular grids. In: Ebert, D., Favre, J.M., Peikert, R. (eds.) VisSym '01: Proceedings of the Symposium on Data Visualization, pp. 107–116. Springer, Berlin (2001)
28. Wang, B., Rosen, P., Skraba, P., Bhatia, H., Pascucci, V.: Visualizing robustness of critical points for 2D time-varying vector fields. Comput. Graph. Forum **32**(2), 221–230 (2013)
29. Zhang, C., Schultz, T., Lawonn, K., Eisemann, E., Vilanova, A.: Glyph-based comparative visualization for diffusion tensor fields. IEEE Trans. Vis. Comput. Graph. **22**(1), 797–806 (2016)
30. Zhang, E., Hays, J., Turk, G.: Interactive tensor field design and visualization on surfaces. IEEE Trans. Vis. Comput. Graph. **13**(1), 94–107 (2007)
31. Zhang, Y., Palacios, J., Zhang, E.: Topology of 3D linear symmetric tensor fields. In: Hotz, I., Schultz, T. (eds.) Visualization and Processing of Higher Order Descriptors for Multi-Valued Data (Dagstuhl'14), pp. 73–92. Springer, Berlin (2015)

Applying 2D Tensor Field Topology to Solid Mechanics Simulations

Yue Zhang, Xiaofei Gao, and Eugene Zhang

Abstract There has been much work in the topological analysis of symmetric tensor fields, both in 2D and 3D. However, there has been relatively little work in the physical interpretations of the topological analysis, such as why wedges and trisectors appear in stress and strain tensors. In this chapter, we explore the physical meanings of degenerate points and describe some results made during our initial investigation.

1 Introduction

The analysis of symmetric tensor fields has seen much advance in the last two decades. Topology-based tensor field analysis has found applications in not only scientific visualization, but also computer graphics [20] and geometry processing [1, 20].

Delmarcelle and Hesselink [6] study the singularities in a 2D symmetric tensor field, which they term *degenerate points*, i.e., points in the domain where the tensor field has two identical eigenvalues (i.e., *degeneracy*). Delmarcelle and Hesselink point out that there are two types of degenerate points, *wedges* and *trisectors*, which they classify using a descriptor that they introduce.

Both wedges and trisectors represent directional discontinuities in the tensor field. However, why are there only two types of fundamental degenerate points

Y. Zhang (✉)
School of Electrical Engineering and Computer Science, 3117 Kelley Engineering Center, Oregon State University, Corvallis, OR 97331, USA
e-mail: zhangyue@oregonstate.edu

X. Gao
School of Electrical Engineering and Computer Science, 1148 Kelley Engineering Center, Oregon State University, Corvallis, OR 97331, USA
e-mail: gaoxia@oregonstate.edu

E. Zhang
School of Electrical Engineering and Computer Science, 2111 Kelley Engineering Center, Oregon State University, Corvallis, OR 97331, USA
e-mail: zhange@eecs.oregonstate.edu

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_2

in a 2D tensor field and why they appear in some locations? To the best of our knowledge, these questions are well understood only for the curvature tensor, which describes the bending of surfaces. For example, wedges tend to appear at the tips of protrusions in the surface, while trisectors tend to appear at the joints of the surface [2].

While the above interpretation of the degenerate points is well understood in shape modeling, it is not clear how to adapt this interpretation to the stress tensor and strain tensor in solid mechanics. The curvature tensor naturally relates to the distribution of the Gaussian curvature of surfaces considered and of the degenerate points in the tensor field. In solid and fluid mechanics, the study of tensor field topology is relatively under-utilized.

In this book chapter, we describe our ongoing effort and some initial results in applying 2D symmetric tensor field topology to the stress tensor fields. Our approach includes the generation of simulation scenarios with controlled shapes, material properties, external forces, and boundary conditions. The occurrence of the degenerate points becomes a function of the physical properties of the shapes studied. In addition, we present an enhanced topological description for 2D symmetric tensor fields, based on the concepts of isotropy index and deviator variability index.

Section 2 reviews past research in symmetric tensor field visualization. In Sect. 3 we review relevant mathematical background about tensor fields. In Sect. 4 we describe our approach. We present our enhanced topological descriptors for 2D symmetric tensor fields in Sect. 5 before concluding in Sect. 6.

2 Previous Work

There has been much work on the topic of 2D and 3D tensor fields for medical imaging, scientific visualization, and geometry modeling. We refer the readers to the recent survey by Kratz et al. [10]. Here we only refer to the research most relevant to this chapter.

Most of the earlier research on symmetric tensor field analysis and visualization focused on the *diffusion tensor*, a semi-positive-definite tensor extracted from brain imaging. The main focus is two-folds. First, fibers following the eigenvectors of the diffusion tensor are computed. Second, appropriate glyphs are designed to help the user understand the diffusion tensor. This has led to various measures for the anisotropy in the diffusion tensor, such as the *relative anisotropy* and the *fractional anisotropy* [3]. Unfortunately these measures do not distinguish between the linear and planar types of tensors. Westin et al. [18] overcome this by modeling the anisotropy using three coefficients that measure the linearity, planarity, and sphericalness of a tensor, respectively. The aforementioned measures are designed for semi-positive-definite tensors, such as the diffusion tensor. We refer interested readers to the book [5] and the survey by Zhang et al. [22] on research related to this area.

At the same time, there have been a number of approaches to visualize 2D and 3D symmetric tensor fields. Delmarcelle and Hesselink [6] introduce the notion of hyperstreamlines for the visualization of 2D and 3D symmetric tensor fields. Zheng and Pang [24] visualize hyperstreamlines by adapting the well-known Line Integral Convolution (LIC) method of Cabral and Leedom [4] to symmetric tensor fields which they term *HyperLIC* [24]. Zheng and Pang also deform an object to demonstrate the deformation tensor [23]. These visualization techniques have been later used for geomechanics data sets [13]. Hotz et al. [9] use an physically-driven approach to produce LIC like visualization. One of the fundamental differences between the diffusion tensor and the other symmetric tensors from mechanics (stress, strain, symmetric part of the velocity gradient tensor) is that the former is semi-positive-definite (no negative eigenvalues) while the latter can have both semi-positive and negative eigenvalues. Schultz and Kindlmann [15] extend ellipsoidal glyphs that are traditionally used for semi-positive-definite tensors to *superquadric glyphs* which can be used for general symmetric tensors.

Delmarcelle and Hesselink [6, 7] introduce the topology of 2D symmetric tensor fields as well as conduct some preliminary studies on 3D symmetric tensors in the context of flow analysis. Hesselink et al. later extend this work to 3D symmetric tensor fields [8] and study the degeneracies in such fields. Zheng and Pang [25] point out that triple degeneracy, i.e., a tensor with three equal eigenvalues, cannot be extracted in a numerically stable fashion. They further show that double degeneracies, i.e., only two equal eigenvalues, form lines in the domain. In this work and subsequent research [27], they provide a number of degenerate curve extraction methods based on the analysis of the discriminant function of the tensor field. Furthermore, Zheng et al. [26] point out that near degenerate curves the tensor field exhibits 2D degenerate patterns and define separating surfaces which are extensions of separatrices from 2D symmetric tensor field topology. Tricoche et al. [16] convert the problem of extracting degenerate curves in a 3D tensor field to that of finding the ridge and valley lines of an invariant of the tensor field, thus leading to a more robust extraction algorithm. Tricoche and Scheuermann [17] introduce a topological simplification operation which removes two degenerate points with opposite tensor indexes from the field. Zhang et al. [20] propose an algorithm to perform this *pair cancellation* operation by converting the tensor field to a vector field and reusing similar operations in vector field topological simplification [21].

Finally, a number of researchers have investigated the visualization and analysis of data in solid and fluid mechanics using non-topological approaches, such as the stress classification using Mohr's diagram [11], stress streamlets [19], and machine learning [12].

3 Background on Tensors and Tensor Fields

In this section we review the most relevant background on 2D symmetric tensors and tensor fields.

3.1 Tensors

A K-dimensional (symmetric) tensor $\mathbf{T}$ has K real-valued *eigenvalues*: $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_K$. When all the eigenvalues are non-negative, the tensor is referred to as *semi-positive-definite*. The largest and smallest eigenvalues are referred to as the *major eigenvalue* and *minor eigenvalue*, respectively. When $K = 3$, the middle eigenvalue is referred to as the *medium eigenvalue*. An eigenvector belonging to the major eigenvalue is referred to as a *major eigenvector*. Medium and minor eigenvectors can be defined similarly. Eigenvectors belonging to different eigenvalues are mutually perpendicular.

The trace of a tensor $\mathbf{T} = (\mathbf{T}_{ij})$ is $trace(\mathbf{T}) = \sum_{i=1}^{K} \lambda_i$. $\mathbf{T}$ can be uniquely decomposed as $\mathbf{D} + \mathbf{A}$ where $\mathbf{D} = \frac{trace(\mathbf{T})}{K}\mathbb{I}$ ($\mathbb{I}$ is the K-dimensional identity matrix) and $\mathbf{A} = \mathbf{T} - \mathbf{D}$. The *deviator* $\mathbf{A}$ is a *traceless* tensor, i.e., $trace(\mathbf{A}) = 0$. Note that $\mathbf{T}$ and $\mathbf{A}$ have the same set of eigenvectors. Consequently, the anisotropy in a tensor field can be defined in terms of its deviator tensor field. Another nice property of the set of traceless tensors is that it is closed under matrix addition and scalar multiplication, making it a linear subspace of the set of tensors.

The magnitude of a tensor $\mathbf{T}$ is $||\mathbf{T}|| = \sqrt{\sum_{1 \leq i,j \leq K} T_{ij}^2} = \sqrt{\sum_i^K \lambda_i^2}$, while the determinant is $|\mathbf{T}| = \prod_{i=1}^{K} \lambda_i$. For a traceless tensors $\mathbf{T}$, when $K = 2$ we have $||\mathbf{T}||^2 = -2|\mathbf{T}|$. This observation infers a simplification in tensor invariant calculations when only the traceless portions of the tensors are investigated, and it becomes advantageous to decompose the tensor into the trace and deviator parts.

A tensor is *degenerate* when there are repeating eigenvalues. In this case, there exists at least one eigenvalue whose corresponding eigenvectors form a higher-dimensional space than a line. When $K = 2$ a degenerate tensor must be a multiple of the identity matrix. In 2D, the aforementioned trace-deviator decomposition can turn any tensor into the sum of a degenerate tensor (isotropic) and a non-degenerate tensor (anisotropic). For example, when the tensor is the curvature tensor of two-dimensional manifolds embedded in 3D, the isotropic-deviator decomposition amounts to

$$K = U' \begin{pmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{pmatrix} U \tag{1}$$

where $\kappa_1 \geq \kappa_2$ are the *principal curvatures* and the columns of U are the corresponding *principal curvature directions*. Rewriting the above equation we have

$$\begin{pmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{pmatrix} = \frac{\sqrt{\kappa_1^2 + \kappa_2^2}}{\sqrt{2}} \left[\sin\phi \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \cos\phi \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right] \tag{2}$$

where $\kappa_1^2 + \kappa_2^2 = ||K||^2$ is the *total curvature* and $\phi = \arctan(\frac{\kappa_1+\kappa_2}{\kappa_1-\kappa_2})$ measures the relative strength between the isotropic and anisotropic components in the curvature tensor. Figure 1 shows the canonical shapes corresponding to various representative ϕ values and how ϕ can be used as a classification of surface geometry on the bunny surface.

If all the eigenvalues of a tensor $\mathbf{T}$ are positive, the tensor is *positive-definite*. Examples of semi-positive-definite tensors include the diffusion tensor from medical imaging and the metric tensor from differential geometry. Tensors that are not semi-positive-definite include the curvature tensor from differential geometry and the stress and strain tensors from solid and fluid mechanics.

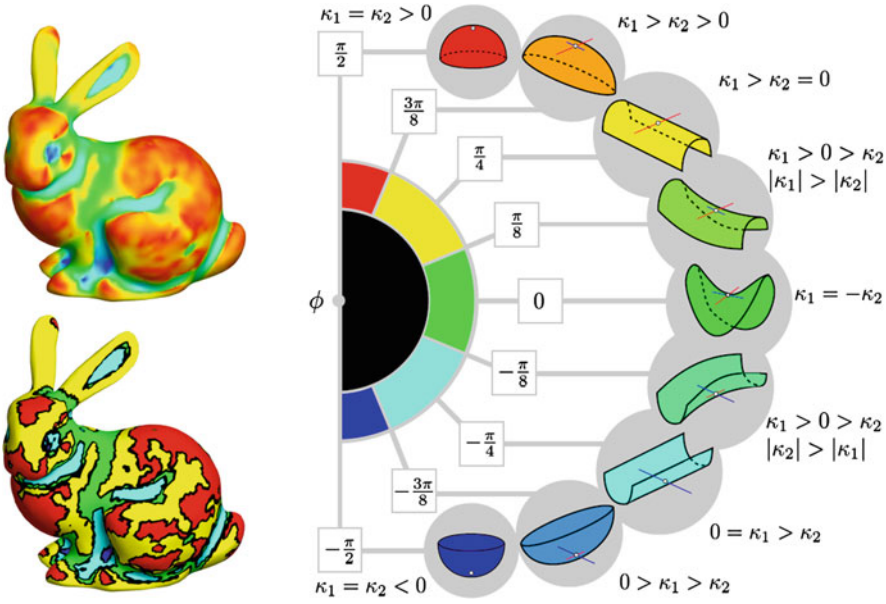


Fig. 1 Surface classification scheme based on the shape index $\phi \in [\pi/2, \pi/2]$ is color mapped to the [*BLUE,RED*] arc in HSV color space: *Left top*: continuous mapping. *Bottom*: binned classification. The legend (*right*) shows surfaces patches which are locally similar to points with given values. This figure is a courtesy of [14], ©2012 IEEE

3.2 Tensor Fields

We now review *tensor fields*, which are tensor-valued functions over some domain $\Omega \subset \mathbb{R}^K$. A tensor field can be thought of as K eigenvector fields, corresponding to the K eigenvalues. A *hyperstreamline* with respect to an eigenvector field $e_i(p)$ is a $3D$ curve that is tangent to e_i everywhere along its path. Two hyperstreamlines belonging to two different eigenvalues can only intersect at the right angle, since eigenvectors belonging to different eigenvalues must be mutually perpendicular.

Hyperstreamlines are usually curves. However, they can occasionally consist of only one point, where there are more than one choice of lines that correspond to the eigenvector field. This is precisely where the tensor field is degenerate. A point $p_0 \in \Omega$ is a *degenerate point* if $\mathbf{T}(p_0)$ is degenerate. The topology of a tensor field consists of its degenerate points.

In $2D$, the set of degenerate points of a tensor field are isolated points under numerically stable configurations, when the topology does not change given sufficiently small perturbation in the tensor field. An isolated degenerate point can be measured by its *tensor index* [20], defined in terms of the *winding number* of one of the eigenvector fields on a loop surrounding the degenerate point. The most fundamental types of degenerate points are *wedges* and *trisectors*, with a tensor index of $\frac{1}{2}$ and $-\frac{1}{2}$, respectively. Let $L\mathbf{T}_{p_0}(p)$ be the local linearization of $\mathbf{T}(p)$ at a degenerate point $p_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$, i.e.,

$$L\mathbf{T}_{p_0}(p) = \begin{pmatrix} a_{11}(x-x_0) + b_{11}(y-y_0) & a_{12}(x-x_0) + b_{12}(y-y_0) \\ a_{12}(x-x_0) + b_{12}(y-y_0) & a_{22}(x-x_0) + b_{22}(y-y_0) \end{pmatrix} \tag{3}$$

Then $\delta = \left| \begin{pmatrix} \frac{a_{11}-a_{22}}{2} & a_{12} \\ \frac{b_{11}-b_{22}}{2} & b_{12} \end{pmatrix} \right|$ is invariant under the change of basis. Moreover, p_0 is a wedge when $\delta > 0$ and a trisector when $\delta < 0$. When $\delta = 0$, p_0 is a higher-order degenerate point. A *major separatrix* is a hyperstreamline emanating from a degenerate point following the major eigenvector field. A *minor separatrix* is defined similarly. The directions in which a separatrix can occur at a degenerate point p_0 can be computed as follows.

Let $v = (x, y)$ be a unit vector. Let $L\mathbf{T}_{p_0}(p)$ be the local linearization at p_0. A major separatrix can leave p_0 in the direction of v if v is parallel to $e_1(p_0, \alpha v)$ for some $\alpha > 0$. Here, $e_1(p_0, \alpha v)$ is a major eigenvector of $L\mathbf{T}_{p_0}(p_0 + \alpha v)$. Note that the $L\mathbf{T}_{p_0}(p)$ is a linear tensor field. Consequently, $L\mathbf{T}_{p_0}(p_0 + \alpha v) = \alpha L\mathbf{T}_{p_0}(p_0 + v)$, and it is sufficient to choose $\alpha = 1$. Similarly, a minor separatrix can leave p_0 in the direction of v if v is parallel to $e_2(p_0, \alpha v)$, a minor eigenvector of $L\mathbf{T}_{p_0}(p_0 + \alpha v)$. Finding either major separatrix or minor separatrix directions leads to a cubic polynomial with either one or three solutions under stable conditions. It is known that around a trisector there must be three solutions, corresponding to three

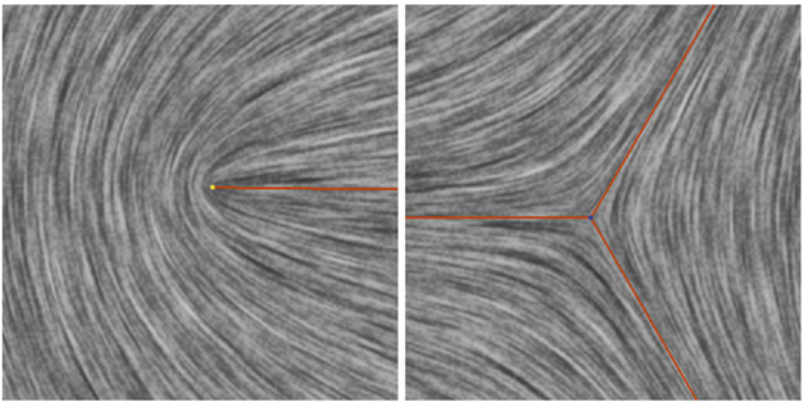


Fig. 2 A wedge (*left*) and a trisector (*right*)

separatrices that divide the neighborhood of p_0 into three sectors, thus the namesake. Around a wedge there can be either one sector or three sectors.

The total tensor index of a continuously tensor field over a two-dimensional manifold is equal to the *Euler characteristic* of the underlying manifold. Consequently, it is not possible to remove one degenerate point. Instead, a pair of degenerate points with opposing tensor indexes (a wedge and trisector pair) must be removed simultaneously [20]. Figure 2 shows a wedge pattern (left) and a trisector pattern (right), respectively.

4 Our Approach

The classification in Eq. (2) is applicable to tensor fields other than the curvature tensor. In this book chapter, we apply it to the stress and strain tensors from 2D solid mechanics simulations.

Our research is inspired by the following question: in the stress tensor, what tensor properties decide the location and type of a degenerate point? To explore this question, we develop a number of simulation scenarios, using a commercial software tool for continuum mechanics. Currently, we are focusing on 2D scenarios. Furthermore, we implement a visualization system which takes as input a 2D symmetric tensor field and computes a number of derived quantities from the tensor field, such as the components of the tensors as well as their traces, determinants, magnitudes, eigenvalues, and eigenvectors. The eigenvectors (either major, or minor, or both) are displayed using texture-based methods [20] while the scalar quantities are displayed using colors. For our initial geometry, we choose a relative simple case, i.e., a square shape with uniform linear elastic material property. After deforming this shape, we obtain nontrivial stress tensors over the domain. In our visualization, the textures and colors can be displayed on the mesh before and

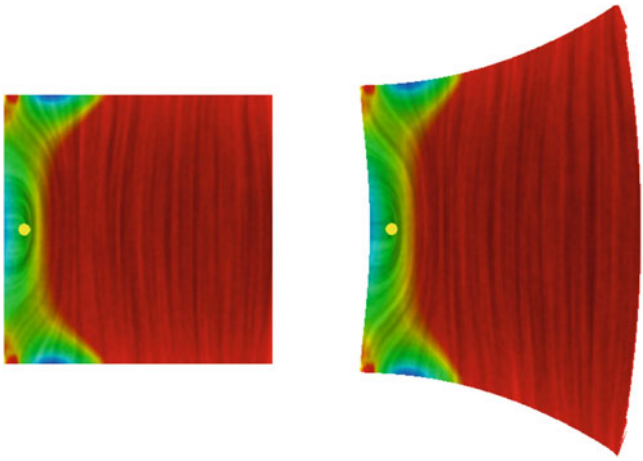


Fig. 3 The visualization of the stress tensor from a simulation in which the left edge is fixed, and the top and bottom edges are pulled to expand the shape. No boundary condition is applied to the right edge. The colors show the trace based on the rainbow map: *red* (high), *yellow* (medium high), *green* (zero), *cyan* (medium low), and *blue* (low). The textures show the major eigenvectors. In the *left*, the object is undeformed, while in the *right* it is deformed

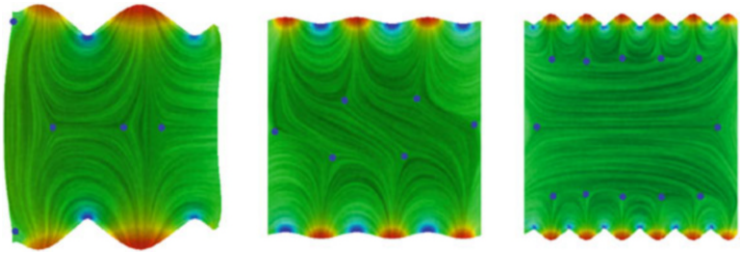


Fig. 4 Visualization of three deformed shapes from the same initial shape by varying the boundary conditions. Even though different amplitudes and locations of the boundary conditions lead to different topology, a possible pattern of when the trisectors (shown here for this family of scenarios) seems to exist

after the deformation. Figure 3 shows one example of our visualization applied to a simulated scenario before and after the deformation. Figure 4 shows different tensor field topology for three such simulation scenarios with the same initial domain and homogeneous material properties but different boundary conditions and external force configurations.

Our study of these data sets has led to a new tensor classification approach as well as some observations.

5 Tensor Classification

Equation (2) is applicable to not only the curvature tensor in geometry processing, but also the stress and strain tensors in solid and fluid mechanics. In this case, degenerate tensors correspond to points where the stress is isostropic, with either a positive trace (pressure) or a negative trace (pressure). Degenerate points in the stress tensor are therefore where isotropic stress occurs, i.e., only hydrostatic pressure and no shear.

This straightforward application of 2D symmetric tensor field topology to stress tensors has two shortcomings. First, pure shear locations, where no hydrostatic pressure exists, are not included as important features in tensor field topology. Second, the type of the degenerate points (wedge or trisector) is not considered. To address this, we introduce an enhanced version of the tensor field topology.

As mentioned earlier, there is not a curvature tensor quantity in stress tensor fields and this hinders developing topology over these fields. To divide the material region into meaningful topological domains, we explore various measurement criteria. Here, we consider the following two quantities: we consider the following two quantities: *isotropy index* ϕ, and *deviator variation index* δ. The isotropy index ϕ is the same as the shape index in Eq. (2). Next, we describe the deviator variation index.

Given a symmetric tensor field $T(x, y)$ defined on a domain $D \subset \mathbb{R}^2$, the deviator of $T(x, y)$ introduces a map $\eta : D \rightarrow \mathbf{F}$ where $\mathbf{F} = \{\begin{pmatrix} a & b \\ b & -a \end{pmatrix} \| a, b \in \mathbb{R}^2\}$ is the set of 2D traceless, symmetric tensors. We define the anisotropic variation index at a point $(x_0, y_0) \in D$ as:

$$\delta(x_0, y_0) = \lim_{A(\Gamma)\rightarrow 0} \frac{A(\eta(\Gamma))}{A(\Gamma)} \tag{4}$$

where Γ is a region enclosing (x_0, y_0), $\eta(\Gamma)$ is the image of Γ under the map η, and $A(K)$ is the *signed* area of K, which in this case refers to Γ and $\eta(\Gamma)$, respectively. This quantity measures the spatial variability of the deviatoric stress tensor, which also shows spatial variability in the (major or minor) eigenvector fields around (x_0, y_0). Note that $\delta(x_0, y_0)$ can be negative, indicating Γ and $\eta(\Gamma)$ are oppositely oriented.

Assuming $T(x, y)$ is sufficiently differentiable, it can be shown that

$$\delta(x_0, y_0) = \left| \begin{pmatrix} \frac{1}{2}(\frac{\partial T_{11}}{\partial x}(x_0, y_0) - \frac{\partial T_{22}}{\partial x}(x_0, y_0)) & \frac{\partial T_{12}}{\partial x}(x_0, y_0) \\ \frac{1}{2}(\frac{\partial T_{11}}{\partial y}(x_0, y_0) - \frac{\partial T_{22}}{\partial y}(x_0, y_0)) & \frac{\partial T_{12}}{\partial y}(x_0, y_0) \end{pmatrix} \right| \tag{5}$$

Notice that δ is exactly the same quantity that Delmarcelle and Hesselink [6] used to classify a degenerate point in a 2D symmetric tensor field. In their analysis, the sign of the deviator variation index is used to classify a degenerate point (positive for wedges and negative for trisectors). We wish to point out that the absolute

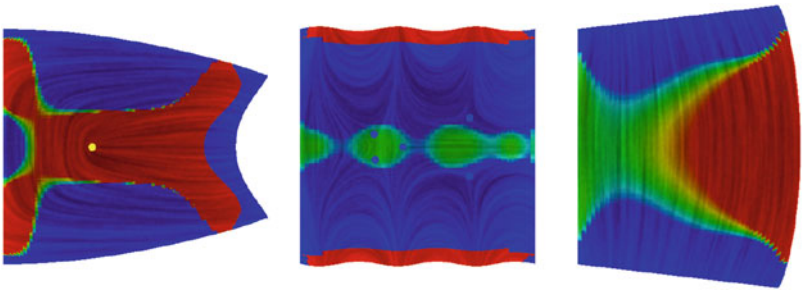


Fig. 5 Three simulation scenarios visualized using the deviator variability index. As expected, the deviator variability index is positive for wedges (e.g., (*left*)) and negative for trisectors (e.g., (*middle*)). However, the absolute value of the deviator variability index is a useful tool to measure the importance of degenerate points (*middle*: the two trisectors near the right edge have a higher index value than those of the three trisectors near the left edge, indicating great variability in the eigenvectors around the degenerate points). Moreover, even in the absence of degenerate points (*right*), the spatial variation of the deviator variability index is also a characterization of the tensor field itself

value of the deviator variation index is also important as it measures how spatially varying the eigenvector fields are around the point of interest, which can be either a degenerate point or a regular point.

Figure 5 demonstrates this with three examples. This leads to the following characterization of a 2D stress tensor field, based on the magnitude, the isotropy index, and the deviator variability index.

In addition to the enhanced description of tensor fields, we also propose to add the set of zero isotropy index points and the set of zero deviator variability index points to 2D tensor field features. For convenience, we refer to a zero isotropy index point as a pure shear point and a zero deviator variability index point as a transition point. Note that the latter is somewhat a misuse of the term transition point, which refers to a structurally unstable degenerate point. Not all points that have a zero deviator variability index is a degenerate point. However, for convenience we will overload the term in the remainder of this book chapter. The sets of pure shear points and transition points are both curves under structurally stable conditions. Together, they divide the domain into four types of regions, which we refer to as (1) expansion wedge region, (2) expansion trisector region, (3) compression wedge region, and (4) compression trisector region. Again, note that we use wedge and trisector in the names of these regions for convenience, instead of terms such as positive deviator variability index and negative deviator variability index. The names do not suggest that there must be any degenerate point in any of these regions (Fig. 6).

Figure 7 shows the partitions of the domain by the set of pure shear points (left) and the transition points (right) of a simulation.

It is interesting that when we set up the boundary conditions for our simulations, the expected number, location, and type of degenerate points have often deviated from the actual outcome. This highlights the need to understand the physical

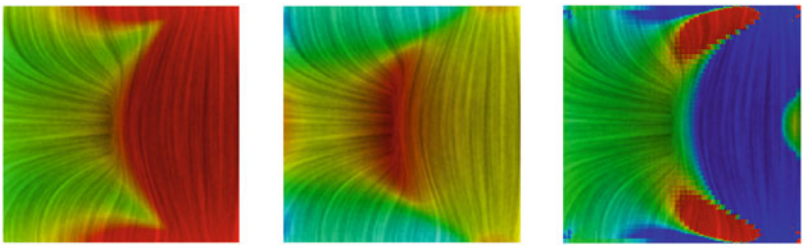


Fig. 6 The tensor magnitude (*left*), isotropy index (*middle*), and deviator variability index (*right*) of a simulated scenario. Note that the three descriptors together provide more insight into the tensor field

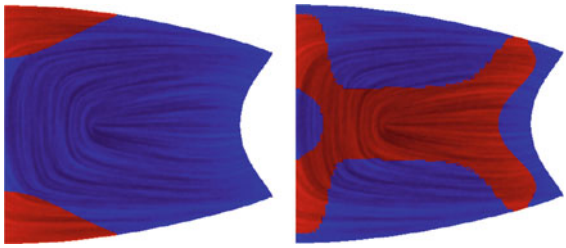


Fig. 7 The sign functions of the isotropy index (*left*) and the deviator variability index (*right*) of a simulated scenario. The boundaries between *red and blue regions* in both figures correspond to pure shear points (*left*) and zero deviator variability points (*right*), respectively. These two sets of points form curves, which, collectively divide the domain in four types of regions

interpretations of degenerate points, such as why they appear in certain locations. In addition, the knowledge can help domain scientists control the number, type, and location of degenerate points when setting up the simulation.

6 Conclusion

The notion of tensor field topology originated from fluid and solid mechanics [6]. Yet, after years of research in tensor field topology, its uses in engineering application are still rather limited. One of the main difficulties is the lack of physical interpretation of tensor field topology in these applications. Inspired by this, we have started the exploration of the physical interpretations of tensor field topology in terms of stress tensors. Our approach includes the generation of a large number of simulation scenarios with controlled shapes and material properties and external force as well as topological analysis of the resulting stress tensor fields. As a first step of this exploration, we enhanced existing 2D symmetric tensor fields by including the set of pure shear points and the set of transition points, which divide the domain into four types of regions. At part of this, we point

out the geometric meaning of the deviator variability index, which was limited to classifying degenerate points.

For future work, we plan to adapt our analysis to 2D asymmetric tensor fields and 3D symmetric tensor fields.

References

1. Alliez, P., Cohen-Steiner, D., Devillers, O., Lévy, B., Desbrun, M.: Anisotropic polygonal remeshing. ACM Trans. Graph. (SIGGRAPH 2003) **22**(3), 485–493 (2003)
2. Alliez, P., Meyer, M., Desbrun, M.: Interactive geometry remeshing. In: Proceedings of the 29th annual conference on Computer graphics and interactive techniques, SIGGRAPH '02, pp. 347–354. ACM, New York (2002). doi:10.1145/566570.566588. http://doi.acm.org/10.1145/566570.566588
3. Basser, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. J. Magn. Reson. B **111**(3), 209–219 (1996)
4. Cabral, B., Leedom, L.C.: Imaging vector fields using line integral convolution. In: Poceedings of ACM SIGGRAPH 1993, Annual Conference Series, pp. 263–272 (1993)
5. Cammoun, L., Castano-Moraga, C.A., Munoz-Moreno, E., Sosa-Cabrera, D., Acar, B., Rodriguez-Florido, M., Brun, A., Knutsson, H., Thiran, J., Aja-Fernandez, S., de Luis Garcia, R., Tao, D., Li, X.: Tensors in Image Processing and Computer vision. Advances in Pattern Recognition. Springer, London (2009). http://www.springer.com/computer/computer+imaging/book/978-1-84882-298-6
6. Delmarcelle, T., Hesselink, L.: Visualizing second-order tensor fields with hyperstream lines. IEEE Comput. Graph. Appl. **13**(4), 25–33 (1993)
7. Delmarcelle, T., Hesselink, L.: The topology of symmetric, second-order tensor fields. In: Proceedings IEEE Visualization '94 (1994)
8. Hesselink, L., Levy, Y., Lavin, Y.: The topology of symmetric, second-order 3D tensor fields. IEEE Trans. Vis. Comput. Graph. **3**(1), 1–11 (1997)
9. Hotz, I., Feng, L., Hagen, H., Hamann, B., Joy, K., Jeremic, B.: Physically based methods for tensor field visualization. In: Proceedings of the Conference on Visualization '04, VIS '04, pp. 123–130. IEEE Computer Society, Washington (2004). doi:10.1109/VISUAL.2004.80. http://dx.doi.org/10.1109/VISUAL.2004.80
10. Kratz, A., Auer, C., Stommel, M., Hotz, I.: Visualization and analysis of second-order tensors: Moving beyond the symmetric positive-definite case. Comput. Graph. Forum **32**(1), 49–74 (2013). http://dblp.uni-trier.de/db/journals/cgf/cgf32.html#KratzASH13
11. Kratz, A., Meyer, B., Hotz, I.: A visual approach to analysis of stress tensor fields. In: Scientific Visualization: Interactions, Features, Metaphors, pp. 188–211 (2011). Dagstuhl Publishing, Saarbrücken. doi: 10.4230/DFU.Vol2.SciViz.2011.188. http://dx.doi.org/10.4230/DFU.Vol2.SciViz.2011.188
12. Maries, A., Haque, M.A., Yilmaz, S.L., Nik, M.B., Marai, G.: Interactive exploration of stress tensors used in computational turbulent combustion. In: Laidlaw, D., Villanova, A. (eds.) New Developments in the Visualization and Processing of Tensor Fields. Springer, Heidelberg (2012)
13. Neeman, A., Jeremic, B., Pang, A.: Visualizing tensor fields in geomechanics. In: IEEE Visualization, p. 5 (2005)
14. Nieser, M., Palacios, J., Polthier, K., Zhang, E.: Hexagonal global parameterization of arbitrary surfaces. IEEE Trans. Vis. Comput. Graph. **18**(6), 865–878 (2012). doi: 10.1109/TVCG.2011.118. http://dx.doi.org/10.1109/TVCG.2011.118
15. Schultz, T., Kindlmann, G.L.: Superquadric glyphs for symmetric second-order tensors. IEEE Trans. Vis. Comput. Graph. **16**(6), 1595–1604 (2010)

16. Tricoche, X., Kindlmann, G., Westin, C.F.: Invariant crease lines for topological and structural analysis of tensor fields. IEEE Trans. Vis. Comput. Graph. **14**(6), 1627–1634 (2008). doi:http://doi.ieeecomputersociety.org/10.1109/TVCG.2008.148
17. Tricoche, X., Scheuermann, G.: Topology simplification of symmetric, second-order 2D tensor fields. In: Geometric Modeling Methods in Scientific Visualization (2003)
18. Westin, C.F., Peled, S., Gudbjartsson, H., Kikinis, R., Jolesz, F.A.: Geometrical diffusion measures for MRI from tensor basis analysis. In: ISMRM '97, p. 1742. Vancouver, Canada (1997)
19. Wiebel, A., Koch, S., Scheuermann, G.: Glyphs for Non-Linear Vector Field Singularities, pp. 177–190. Springer, Berlin (2012). doi:10.1007/978-3-642-23175-9_12. http://dx.doi.org/10.1007/978-3-642-23175-9_12
20. Zhang, E., Hays, J., Turk, G.: Interactive tensor field design and visualization on surfaces. IEEE Trans. Vis. Comput. Graph. **13**(1), 94–107 (2007)
21. Zhang, E., Mischaikow, K., Turk, G.: Vector field design on surfaces. ACM Trans. Graph. **25**(4), 1294–1326 (2006)
22. Zhang, S., Kindlmann, G., Laidlaw, D.H.: Diffusion tensor MRI visualization. In: Visualization Handbook. Academic Press, London (2004). http://www.cs.brown.edu/research/vis/docs/pdf/Zhang-2004-DTM.pdf
23. Zheng, X., Pang, A.: Volume deformation for tensor visualization. In: IEEE Visualization, pp. 379–386 (2002)
24. Zheng, X., Pang, A.: Hyperlic. In: Proceeding IEEE Visualization, pp. 249–256 (2003)
25. Zheng, X., Pang, A.: Topological lines in 3D tensor fields. In: Proceedings IEEE Visualization 2004, VIS '04, pp. 313–320. IEEE Computer Society, Washington (2004). doi:10.1109/VISUAL.2004.105. http://dx.doi.org/10.1109/VISUAL.2004.105
26. Zheng, X., Parlett, B., Pang, A.: Topological structures of 3D tensor fields. In: Proceedings IEEE Visualization 2005, pp. 551–558 (2005)
27. Zheng, X., Parlett, B.N., Pang, A.: Topological lines in 3D tensor fields and discriminant hessian factorization. IEEE Trans. Vis. Comput. Graph. **11**(4), 395–407 (2005)

Moment Invariants for Multi-Dimensional Data

Roxana Bujack and Hans Hagen

Abstract Moment invariants have long been successfully used for pattern matching in scalar fields. By their means, features can be detected in a data set independent of their exact orientation, position, and scale. Their recent extension to vector fields was the first step towards rotation invariant pattern detection in multi-dimensional data.

In this paper, we propose an algorithm that extends the normalization approach to tensor fields of arbitrary rank in two and three dimensions.

1 Introduction

Tensor fields play an important role in the study of many physical phenomena. Earthquakes, volcanoes, diffusion, or deformation can all be described using tensor fields. Higher derivatives of scalar and vector fields also form tensors. In contrast to their lower rank counterparts, tools for the analysis of higher rank tensor fields are not as well developed. In particular, the visualization of three-dimensional tensor fields can suffer from a clutter. Any given visualization element may occlude elements behind it. The question of what is included or omitted in a visualization is very important and can potentially impact the scientific understanding. Pattern detection can help to address this problem by reducing the areas that are drawn to the locations of features that are of importance to the analyst. In this paper, we suggest an algorithm for rotation invariant pattern detection for tensor fields of arbitrary rank.

Rotation invariance is a critical requirement of pattern detection to minimize the set of unique patterns required in a search. Moment invariants allow one to achieve rotation invariance without the need for point to point correlations.

Moments are the projections of a function with respect to a function space basis. We can think of them as the coordinates that represent the pattern. They can then be used to construct moment invariants - values that do not change under certain

R. Bujack (✉) • H. Hagen
Technical University Kaiserslautern, Kaiserslautern, Germany
e-mail: bujack@cs.uni-kl.de; hagen@cs.uni-kl.de

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_3

transformations. In this paper, we concentrate on orthogonal transformation and isotropic scaling.

There are two main approaches to the construction of moment invariants: normalization and the definition of a generator [17]. For *normalization*, a standard position is defined by demanding certain moments to assume fixed values and all functions are transformed to match it. Then the remaining moments form a complete and independent set of moment invariants. The *generator* approach relies on defining an explicit rule on how to combine the moments in a way that suppresses the alignment information, usually by multiplication and addition.

The main disadvantage of the normalization approach is that it is unstable if the moment chosen for normalization should become zero. On the other hand, the main disadvantage of the generator approach is the difficulty in finding and proving the existence of an independent moment. Depending on the application, one method may be more effective than the other.

Both approaches have been applied to generate moment invariants for scalar fields and recently also to vector fields. To the authors' best knowledge, to date, moment invariants for matrix fields or higher rank tensor fields have only been presented using a generator approach [25].

In this paper, we present an algorithm that constructs moment invariants using normalization for two- and three-dimensional tensor fields of arbitrary rank. For scalar fields, the zeroth and first order moments are usually used for the normalization with respect to translation and scaling. This is why the standard position with respect to orientation is generally chosen to be the Jordan normal form of the second order moments, which is related to the principal axes of the covariance matrix. For vector fields, Bujack et al. [5] use the Schur form of the first order moments. This is also of second rank. For higher rank tensor fields, the first order moments are already of a rank higher than two. Hence, matrix algebra approaches can no longer be applied.

Our solution to this problem is to use tensor contractions that produce first rank tensors from higher rank moment tensors. These first rank tensors behave like vectors and can easily assume a standard position. A similar approach has been used in [10] to generate a normalizer for 2D affine transformations and in [9, 27] for 3D scalar functions. We extend this idea to generate rotation invariants for tensor valued functions.

The main contributions of this paper are as follows:

- We propose a methodology to apply tensor algebra to the normalization of moments of tensor fields of any order $o \in \mathbb{N}$, including scalar, vector, and matrix fields.
- By producing a complete and independent set of moment invariants, this method can provide a solution for tensor fields where the generator approach fails [25].
- The flexibility inherent in this method also improves on state the art normalization approaches for vector and scalar data by finding solutions where the current techniques fail due to vanishing moments.
- To our knowledge, this is the first time that moment invariants have been computed for a tensor field of second order.

2 Related Work

The first moment invariants were introduced to the image processing society by Hu [21]. The development of moment tensors by Dirilten and Newman [11] extended moment invariants to three-dimensional data. Pinjo, Cyganski, and Orr [27] calculated 3D orientation estimation from moment contraction to first order moments.

Please note that invariants can be constructed not only from moments, but, for example, from derivatives, too [12, 13]. The fundamental theorem of moment invariants [28] guarantees that every algebraic invariant has a moment invariant counterpart.

In his seminal work [14], Flusser presented a calculation rule to generate a complete and independent set for 2D scalar functions. Later, he proved that it further solves the inverse problem in [15].

For three-dimensional functions, the task is much more challenging. One research path goes in the direction of the spherical harmonics. They are an irreducible representation of the rotation group and therefore an adequate basis for the generation of moment invariants. Lo and Don [26], Burel and Henocq [6], Kazhdan et al. [22], Canterakis [8], and Suk et al. [33], use them to construct moment invariants for three-dimensional scalar functions.

A second research path makes use of the tensor contraction method, as described by Dirilten and Newman [11]. While all tensor contractions to zeroth rank are rotationally invariant, it can be difficult to find a complete and independent set. Suk and Flusser propose to calculate all possible zeroth rank contractions from moment tensors up to a given order and then skip the linearly dependent ones in [32]. Higher order dependencies still remain.

Schlemmer et al. [30, 31] generalized the notion of moment invariants to vector fields. Later Bujack et al. [3] provided a normalization method that is flexible with respect to the choice of the normalizer. It leads to a complete set of independent moment invariants for 2D flow fields that is applicable to any pattern. Their extension to three-dimensional vector fields uses the transformation properties of the second rank tensors and their eigenvectors to define a standard position [5]. This method requires the second rank tensor to not vanish.

Langbein and Hagen [25] treat tensor fields of higher rank. They show that the tensor contraction method can be generalized to arbitrary tensor fields. However, their suggested method to reduce the redundancy of the generator approach uses exact calculation. Its application in a discrete setting (i.e., programmatically) has not yet been practically applied. In this paper, we introduce a normalization method for tensor fields which automatically results in an independent and complete set. We demonstrate its utility by applying it to pattern detection in analytic as well as simulation data.

Table 1 State of the art of moment invariants constructed from normalization

Dim.	Data type	Authors	Complete	Independent	Flexible
2D	Scalar	Flusser et al. [17]	✓	✓	–
2D	Vector	Bujack et al. [3]	✓	✓	✓
2D	Tensor	–	–	–	–
3D	Scalar	Cyganski et al. [9]	✓	✓	–
3D	Vector	Bujack et al. [5]	✓	✓	–
3D	Tensor	–	–	–	–

Table 2 State of the art of moment invariants constructed using the generator approach

Dim.	Data type	Authors	Complete	Independent	Flexible
2D	Scalar	Flusser et al. [14]	✓	✓	–
2D	Vector	Schlemmer et al. [29]	–	–	–
2D	Tensor	Langbein et al. [25]	?	(✓)	–
3D	Scalar	Flusser et al. [32]	?	–	?
3D	Vector	Langbein et al. [25]	?	(✓)	–
3D	Tensor	Langbein et al. [25]	?	(✓)	–

The brackets indicate that independence is given only theoretically and the question mark that this property is unknown

The state of the art of moment invariants with respect to the each of the two approaches, the different data types and dimensions are summarized in Tables 1 and 2. The attributes have the following meaning:

- *Complete:* The set is complete if any arbitrary moment invariant can be constructed from it.
- *Independent:* The set is independent if none of its elements can be constructed from its other elements.
- *Flexible:* The set is flexible w.r.t. vanishing moments if it exists for any pattern, meaning it does not rely on any specific moment c_{p_0,q_0} to be non-zero.

3 Theory

We will start by reviewing the theoretical underpinnings of our algorithm so that readers from both the visualization world and the mathematical world start from common ground. We also show how to generate tensors that transform like vectors under rotations and reflections.

3.1 Tensors and Transformations

Tensors are a natural representation of physical quantities that follow specific rules under transformations of the coordinate system. They can be represented as arrays of

numbers relative to a fiducial basis. The rank of a tensor corresponds to the number of indices that we need to identify the different numbers in the array. Scalars are tensors of rank zero, vectors are tensors of rank one, and matrices are tensors of rank two. The interested reader can find exemplary introductions to tensor analysis, in addition to the definitions and lemmata we review in this work in [1] or in [18]. We will make use of the Einstein notation where the summation symbol is dropped in products over the same index. The summation is performed from 1 to d, in which the latter is the underlying dimension. Please note that the theory is valid for arbitrary $d \in \mathbb{N}$. Later, in our experiments, we will work with two- and three-dimensional fields.

Definition 1 A multidimensional array $T^{i_1 \dots i_n}_{j_1 \dots j_m}$ that, under an active transformation by the invertible matrix $A^i_j \in \mathbb{R}^{d \times d}$, behaves as:

$$T'^{i_1 \dots i_n}_{j_1 \dots j_m} = |\det(A^{-1})|^w A^{i_1}_{k_1} \dots A^{i_n}_{k_n} (A^{-1})^{l_1}_{j_1} \dots (A^{-1})^{l_m}_{j_m} T^{k_1 \dots k_n}_{l_1 \dots l_m}, \tag{1}$$

is called a (relative, axial) **tensor** of covariant rank m, contravariant rank n, and weight w. An (absolute) tensor has weight zero.

Remark 1 An active transformation, $x' = Ax$, transforms the field, but not the frame. Thus the coordinate system remains unchanged and a tensor field $T : \mathbb{R}^d \to \mathbb{R}^{d^n \times d^m}$ is transformed via $T'(x') = T(x)$. One can, for example, rotate an object actively. This is in contrast to a passive rotation in which the coordinate system is rotated rather than the object itself.

Example 1 A vector $v \in \mathbb{R}^d$ is an absolute, contravariant, first rank tensor, because it behaves via $v'^i = \sum_{j=1}^d A^i_j v^j$ under active transformations. In Einstein notation, this is written as $v'^i = A^i_j v^j$.

A matrix $M \in \mathbb{R}^{d \times d}$ is a tensor of contravariant rank one and covariant rank 1, because it behaves via $M' = AMA^{-1}$ under active transformations. In Einstein notation, this is written as $M'^i_j = A^i_k (A^{-1})^l_j M^k_l$.

Remark 2 For orthogonal transformations $A \in \mathbb{R}^{d \times d}$, i.e. rotations and reflections, the distinction of the indices into covariant and contravariant ones from Definition 1 is not necessary because they satisfy $A^T = A^{-1}$. Further, the weight can be ignored, because they satisfy $|\det A| = |\det(A^{-1})| = 1$.

Lemma 1 *Let T and $\tilde{T}$ be two relative tensors of covariant rank m, contravariant rank n, and weight w and $\tilde{m}, \tilde{n}, \tilde{w}$ respectively. Then the* product $T \otimes \tilde{T}$ *(also called outer product or tensor product):*

$$(T \otimes \tilde{T})^{i_1 \dots i_n \tilde{i}_1 \dots \tilde{i}_{\tilde{n}}}_{j_1 \dots j_m \tilde{j}_1 \dots \tilde{j}_{\tilde{m}}} := T^{i_1 \dots i_n}_{j_1 \dots j_m} \tilde{T}^{\tilde{i}_1 \dots \tilde{i}_{\tilde{n}}}_{\tilde{j}_1 \dots \tilde{j}_{\tilde{m}}} \tag{2}$$

is a relative tensor of covariant rank $m + \tilde{m}$, contravariant rank $n + \tilde{n}$, and weight $w + \tilde{w}$.

Lemma 2 *Let T be a relative tensor of covariant rank m, contravariant rank n, and weight w. Then the* contraction $\sum_{(i_k,j_l)} T$ *of a covariant index i_k and a contravariant index j_l*

$$(\sum_{(i_k,j_l)} T)^{i_1\dots i_{k-1}i_{k+1}\dots i_n}_{j_1\dots j_{l-1}j_{l+1}\dots j_m} := T^{i_1\dots i_{k-1}\imath i_{k+1}\dots i_n}_{j_1\dots j_{l-1}\imath j_{l+1}\dots j_m} \tag{3}$$

is a relative tensor of covariant rank $m-1$, contravariant rank $n-1$, and weight w.

Remark 3 Please note that $\sum_{(i_k,j_l)}$ is the symbol for the contraction of the two indices i_k and j_l as used in [25]. It is different from the sum $\sum_{i_k,j_l}$ over these indices. In particular, the contracted indices are no longer indices of the contracted tensor.

Example 2 The product of a matrix $M \in \mathbb{R}^{d\times d}$ and a vector $v \in \mathbb{R}^d$ is a tensor product $\otimes$ followed by a contraction $\sum_{(2,1)}$ of the second covariant with the first contravariant index

$$(\sum_{(2,1)}(M \otimes v))^i = (\sum_{(2,1)}(M^i_j v^k))^i = M^i_j v^j. \tag{4}$$

Also, the trace of a matrix M can be written as the contraction by M^i_i.

3.2 Moment Tensors

Dirilten and Newman suggest the use of moment tensors for the construction of moment invariants with respect to orthogonal transforms in [11]. They construct the moment tensors by arranging the moments of each order in a way such that they obey the tensor transformation property (1).

Definition 2 For a scalar function $f : \mathbb{R}^d \to \mathbb{R}$ with compact support, the **moment tensor** ${}^oM$ of order $o \in \mathbb{N}$ takes the shape

$${}^oM^{k_1\dots k_o} = \int_{\mathbb{R}^d} x^{k_1} \dots x^{k_o} f(x)\, \mathrm{d}^d x, \tag{5}$$

with $l \in \{1,\dots,o\}, k_l \in \{1,\dots,d\}$, and x^{k_l} representing the k_l-th component of $x \in \mathbb{R}^d$.

This arrangement of the moments of the same order into arrays simplifies the calculation of their behavior under linear transformations, which is very helpful for the construction of moment invariants. Cyganski et al. [10] use moment tensors to determine the orientation of scalar functions and to normalize with respect to linear transformations. In [9], they present the following important theorem.

Theorem 1 *The moment tensor of order o of a scalar function is a contravariant tensor of rank o and weight -1.*

Langbein et al. [25] have generalized the definition of the moment tensor to tensor valued functions.

Definition 3 For a tensor field $T : \mathbb{R}^d \to \mathbb{R}^{d^n \times d^m}$ with compact support, the **moment tensor** ${}^oM$ of order $o \in \mathbb{N}$ takes the shape

$$ {}^oM = \int_{\mathbb{R}^d} x^{\otimes o} \otimes T(x)\, \mathrm{d}^d x, \tag{6} $$

where $x^{\otimes o}$ denotes the o-th tensor power of the vector x.

The analogy between Definitions 2 and 3 can be seen more easily if we write Definition 3 using the indices

$$ {}^oM^{i_1 \ldots i_n k_1 \ldots k_o}_{j_1 \ldots j_m} = \int_{\mathbb{R}^d} x^{k_1} \ldots x^{k_o} T^{i_1 \ldots i_n}_{j_1 \ldots j_m}(x)\, \mathrm{d}^d x, \tag{7} $$

with $l \in \{1, \ldots, o\}, k_l \in \{1, \ldots, d\}$, and x^{k_l} representing the k_l-th component of $x \in \mathbb{R}^d$.

The following theorem is the main theoretical contribution of this paper. It allows the construction of moment invariants for tensor fields analogously to the ones for scalar fields.

Theorem 2 *The moment tensor of order o of a tensor field of covariant rank m, contravariant rank n and weight w is a tensor of covariant rank m, contravariant rank $n + o$ and weight $w - 1$.*

Proof The vector x is an absolute, contravariant, first rank tensor. Application of Lemma 1 shows that $x^{\otimes o} \otimes T(x)$ is a tensor of covariant rank m, contravariant rank $n + o$ and weight w.

The decrease of the weight by one comes from the integral, because under a change of the integration variable $x' = Ax$, the infinitesimal element is multiplied by the functional determinant $\mathrm{d}^d x' = |\det A|\, \mathrm{d}^d x$.

The following Corollary is the property used by Langbein and Hagen [25] to construct moment invariants. They generate the invariants from moment tensor contractions to scalars.

Corollary 1 *The rank zero contractions of any product of the moment tensors are moment invariants with respect to rotation and reflection.*

Proof It follows from Lemma 1 that any combination of moment tensor factors in a tensor product is a tensor. Contraction of this product to zeroth rank is a tensor because of Lemma 2. According to Definition 1, this zeroth rank tensor satisfies ${}^0M' = {}^0M$ because rotations and reflections satisfy $|\det A| = 1$.

Analogously to the previous corollary, we construct tensors of first and second rank from contractions and products of moment tensors.

Corollary 2 *The rank one contractions of any product of the moment tensors behave like vectors under rotation and reflection.*

Proof Lemmata 1 and 2 guarantee that the first order contractions are tensors. According to Definition 1, it satisfies

$$ {}^1M'^i = A^{i1}_j M^j, \tag{8} $$

because rotations and reflections satisfy $|\det A| = 1$. This corresponds to the classical matrix vector product ${}^1M' = A^1M$.

Corollary 3 *The rank two contractions of any product of the moment tensors behave like matrices under rotation and reflection.*

Proof Lemmata 1 and 2 guarantee that the rank one contractions are tensors. According to Definition 1, it satisfies

$$ {}^2M'^{i_1 i_2} = A^{i_1}_{j_1} A^{i_2}_{j_2} {}^2M^{j_1 j_2}, \tag{9} $$

because rotations and reflections satisfy $|\det A| = 1$. Because they also satisfy $A^T = A^{-1}$. This coincides with the matrix product ${}^2M' = A^2MA^{-1}$.

4 Algorithm and Complexity

In the two-dimensional case, a rotation has one degree of freedom. That means a standard position can be defined using one vector, for example, by demanding this vector to align with the positive real axis.

In the three-dimensional case, we need two vectors to normalize w.r.t. a rotation. As a standard position, we choose the first one to align with the positive x-axis and the second one to lie in the upper half of the x-y-plane, which corresponds to a positive y-coordinate,compare Fig. 1. For the remainder of the paper, we will describe the three-dimensional situation keeping in mind that the second vector will not be needed in 2D.

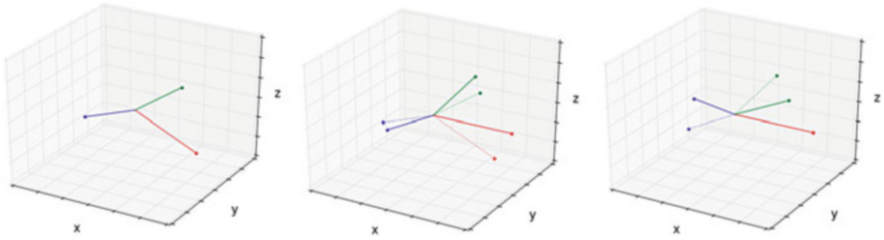


Fig. 1 Illustration os the 3D normalization wrt. rotation. The three vectors represent different normalizer candidates, i.e. first order moment tensor contractions. To maximize robustness, we use the vector with the highest magnitude (*red*) and align it with the positive x-axis. Since, the green vector has the highest magnitude orthogonal to the red one, it is used for the second step and rotated into the x-y-plane

Theoretically, it does not matter which first rank contractions we choose as long as they are the same for the normalization of the pattern and the field. In the case of a scalar field, for example, we could use either the first order moment tensor ${}^1M$ or a contraction of the third order moment tensor $\sum_{(1,2)} {}^3M$ to normalize with respect to rotation. However, for the stability of the algorithm, it is important that the tensors chosen for the normalization are not close to zero nor are linearly dependent. In our scalar field example, let us assume that the pattern of interest has no linear component and ${}^1M = 0$. It is impossible to rotate the zero vector onto the x-axis, which is why it is better to use the contraction $\sum_{(1,2)} {}^3M \neq 0$ for the normalization instead. Even if the tensor is not exactly zero, but numerically small, using it impairs the robustness. Its orientation may change significantly if the field gets disturbed by only small noise, which would lead to a different standard position.

In practice, the moments are calculated up to a maximal given order o, which results in tensors $\{T\}$ up to rank $o + n + m$. To maximize robustness, we construct all possible combinations of tensor products $\{\tilde{T}\}$ not exceeding a given upper rank of $o' > o$ from the tensors $\{T\}$. Then, we calculate all first rank contractions $\{{}^1\tilde{T}\}$ from this set of products $\{\tilde{T}\}$. We allow the user to set the orders o and o'. We choose the vector with maximal magnitude $v_1 := \text{argmax}_{v \in \{{}^1\tilde{T}\}} \|v\|$ for the first normalization step. For the second step, we use the vector that is as orthogonal as possible to the first vector $v_2 := \text{argmax}\, v \in \{{}^1\tilde{T}\} \|v_1 \times v\|$.

Using only the first rank contractions has one important potential issue. If all odd ranked moment tensors are zero, we cannot construct any non-zero first rank contraction because the contraction always decreases the rank by two. To solve this problem, we make use of Corollary 3 and also generate all second rank tensors $\{{}^2\tilde{T}\}$ from the products $\{\tilde{T}\}$. These behave like matrices and so, if they have eigenvectors, these are also possible vectors that could be used to assume the standard position. We restrict ourselves to the symmetric parts of the second rank tensors in order to guarantee the existence of real valued eigenvectors.

The stability of an eigenvector is determined by how distinguishable its corresponding eigenvalue is from the remaining eigenvalues. For example, a small perturbation of the matrix $\text{diag}(1, \epsilon, -\epsilon)$ with small ϵ could change the order of the two smaller eigenvalues. That changes the corresponding eigenvectors and the result of the normalization. On the other hand, the eigenvector that belongs to 1 is far more robust. Therefore, we weigh the vector with the minimal distance of its eigenvalue to the others. Let v_i be the eigenvectors that belong to the eigenvalues λ_i of the symmetric part of a second order contraction $\{{}^2\tilde{T}\}$. Then, we add the vectors $\tilde{v}_i = v_i(\min_{j \neq i} |\lambda_i - \lambda_j|)$ to the set of first order contractions $\{{}^1\tilde{T}\}$. This increases the chances of finding a robust standard position. In our example, we would add $(1 - \epsilon, 0, 0)^T$, $(0, 2\epsilon, 0)^T$, and $(0, 0, 2\epsilon)^T$.

Please note that in contrast to real vectors, eigenvectors do not have a direction. For any eigenvector v, $-v$ is also an eigenvector. We therefore must keep track of two standard positions if an eigenvector is chosen as a normalizer. We do this by storing a second set of normalized moment tensors for the pattern.

Algorithm 1: Moment normalization for tensor fields.

1: Calculate the moment tensors $\{T\}$ of the pattern up to order o, (5).
2: From the tensors $\{T\}$, calculate all possible products $\{\tilde{T}\}$ up to order o', (2).
3: From the products $\{\tilde{T}\}$, calculate all possible contractions to first $\{^1\tilde{T}\}$ and second rank $\{^2\tilde{T}\}$, (3).
4: Compute all eigenvectors $\{v\}$ of the symmetric part of the second rank contractions $\{^2\tilde{T}\}$.
5: Weigh the eigenvectors $v_i \in \{v\}$ by the difference of the corresponding eigenvalues $\tilde{v}_i := v_i(\min_{j\neq i} |\lambda_i - \lambda_j|)$ and add the result to the set $\{^1\tilde{T}\}$.
6: Chose v_1 as the contraction or eigenvector with the biggest norm, i.e. $v_1 := \operatorname{argmax}_{v\in\{^1\tilde{T}\}} \|v\|$.
7: Chose v_2 as the contraction or eigenvector with the highest component orthogonal to v_1, i.e. $v_2 := \operatorname{argmax} v \in \{^1\tilde{T}\}\|v_1 \times v\|$.
8: R_1 is the rotation matrix around the axis $v_1 + (1, 0, 0)^T$ by the angle $\alpha_1 = \pi$.
9: R_2 is the rotation matrix around v_1 by the angle $\alpha_2 = -\operatorname{atan2}((R_1 v_2)^3, (R_1 v_2)^2)$.
10: Rotate the moment tensors by $R = R_2 R_1$ using the transformation rule (1).

In summary, the consecutive steps for the construction of a complete and independent set of moment invariants for tensor fields of arbitrary rank can be found in Algorithm 1. It selects the most robust rank one tensors and determines the rotation that puts them into the standard position as illustrated in Fig. 1.

In the two-dimensional case, the rotation matrix R is simply the one by the angle $\alpha_1 = -\operatorname{atan2}(v_1^2, v_1^1)$ and the second step is not necessary. The upper indices indicate the vector component.

Please note that the normalization step is only performed on the moments using Eq. (1), and not on the field or the pattern themselves. The algorithm does not require any interpolation or sampling.

For the actual pattern detection, we perform the normalization on the pattern, normalize the moments of the field using the same contractions, and determine the similarity using the reciprocal of the Euclidean distance between the moments of the pattern and the field at each position. This process is described in more detail in [4].

In 2D, arbitrary reflections can be generated from the concatenation of a rotation and a reflection at the x-axis. It can be seen from (1) that it leaves the magnitude of each moment $T^{i_1\dots i_n}_{j_1\dots j_m}$ unchanged but causes a sign change in a component of the tensor if the corresponding indices have an odd number of appearances of twos, which corresponds to the y-direction. This means that we can multiply each component with $(-1)^{\sum_{i_k=2} 1+\sum_{j_l=2} 1}$. If the user wants to normalize with respect to reflections, we proceed as described above for rotations and then simply apply the sign change. For invariance with respect to all orthogonal transformations, we use the maximal similarity of both.

In 3D, the normalization with respect to reflections works analogously. We can combine a rotation with a reflection at the x-y-plane, which leads to a sign change if the number of threes in the indices, corresponding to the z-direction, is odd. Thus we can multiply each component with $(-1)^{\sum_{i_k=3} 1+\sum_{j_l=3} 1}$ and proceed as in the 2D case.

In the remainder of this section, we will briefly discuss the complexity of Algorithm 1. The main computational effort goes into the computation of the moments, because we have to evaluate an integral numerically. Assume, we have a d-dimensional tensor field of rank R with N points and want to find a pattern with M points using moments up to order O. Then, we need to compute a total of $\sum_{o=1}^{O} d^{R+o}$ moments at each of the N points, each of which requires the evaluation of M points for the integration, which leaves us with $NM \sum_{o=1}^{O} d^{R+o}$ operations. Please note that this step is inherent in all moment-based algorithms. Therefore, they all have a comparable runtime. It can be performed in a preprocessing step, because the moments of the field do not depend on the pattern. Also, the moments do not depend on each other, which enables a straight forward a parallel computation.

The steps 2 to 9 in Algorithm 1 need to be performed on the moments of the pattern only, which makes them independent from the size of the dataset and the size of the pattern. The number of operations depends on the order O up to which the moments are computed and O' up to which, we compute the products , which can be considered small compared to the dataset size. For example, we used $O, O' \leq 5$ in our result section.

Each contraction requires d operations for each of the d^{r-2} remaining entries of the tensor. For a tensor of rank r, there are $r(r-1)$ possible contractions. The computation of all contractions to first or second order of a tensor of rank R, takes therefore less than $\sum_{r=1}^{R} d^{r-1} r(r-1)$ operations. This process can be accelerated by removing identical contractions that appear because the order in which indices are contracted does not change the result, for example $\sum_{(1,2)} \sum_{(3,4)} T = \sum_{(1,4)} \sum_{(2,3)}$. We also accelerate this process by only considering non-zero moment tensors for the products and the contractions.

Once, we have determined v_1 and v_2, we can specifically compute only these two products and contractions that are necessary to produce them for each point in the field. The complexity of it depends on the specific v_1 and v_2.

For the orientation into standard position, Eq. (1) needs to be applied to all moment tensors in the field and once to the pattern, which takes $(N+1)(\sum_{o=1}^{O} d^{R+o})^2$ operations. Finally, the comparison of the normalized moments of the pattern to the ones of the field requires another $N(\sum_{o=1}^{O} d^{R+o})^2$ operations.

The computation of the pattern detection tasks in our result section took less than a minute on a laptop.

5 Results

In this section, we apply our algorithm to some use cases to visually demonstrate its effectiveness. We first briefly show how our algorithm improves the normalization of 3D scalar and vector fields by adding more flexibility to the choice of the normalizer thus allowing us to avoid vanishing moments. Then we present results of our method applied to tensor fields in 2D and 3D. We use the first and second derivatives of an

analytic scalar field so that the reader can compare the results. Please note that we do not advocate to use higher rank methods to the derivative in cases, where you could as well apply the lower rank algorithm to the original function. The algorithm is meant for pattern detection tasks in tensor fields, like from diffusion or stress measurements, where lower rank data is not available to describe the phenomenon. We do this here only as an illustrative example.

5.1 3D Scalar

The normalization of 3D scalar functions in previous approaches has been performed using the second rank moment tensor $\Sigma = {}^2M$. The standard position of this symmetric matrix was given by its eigenbasis. That means the rotation that diagonalizes Σ was used as the normalizer. This method is equivalent to aligning the principal axes of the function with the coordinate axes. It fails if Σ does not have three distinguishable eigenvalues. All vectors suggested by Cygansky et al. [9] for normalization rely on the second rank moment tensor, too. As a result, it will not work for patterns without a quadratic component.

Our algorithm is able to compensate for a vanishing Σ by using the contractions of higher rank tensors. To illustrate this, we first consider an analytic use case. Figure 3 shows a cut view of fields that have been generated from different linear combinations of polynomials from first to third degree in x and y. We then generate a scalar field such that, at each position $(i, j, 0)^T \in \mathbb{R}^3, i, j \in \{1, 2, 3\}$, we center a polynomial given by the formula $(x-1)^i + 0.5(y-j)^j$ and modulated with the radial Gaussian $\exp(-4(x-i)^2 - 4(y-j)^2)$, superimposing the resulting nine functions.

In order to create a pattern for our search, we choose a small section in the lower right corner, rotated and reflected it randomly. This pattern, drawn from one of the nine possibilities, has a linear and a cubic component but no quadratic. A volume rendering of the rotated pattern can be seen in Fig. 2. The missing quadratic part causes currently available methods to fail to find a normalizer. Algorithm 1, on the other hand, chooses the first rank tensor $v_1 = \{{}^1M\}$ and the contraction of the first two indices in the third rank tensor $v_2 = \{\sum_{(1,2)} {}^2M\}$ and assumes the corresponding standard position. Applying this normalizer, the search results applied to the polynomials of Fig. 3 are visualized in the same figure by applying a brightness transfer function (shown to the right of the figure). The brighter the polynomial visualization, the closer it is to the actual search pattern. Algorithm 1 successfully detects the occurrence of the target polynomial in the field. Other less similar structures in the field are indicated with decreasing similarity values. We calculated the moments up to third order, i.e. third rank, and normalized with respect to rotation, reflection, and scaling. Please note that the analytic field was chosen such that it does not depend on z in order to simplify the visualization. Also, the algorithm was performed on the complete 3D data and the cutting plane is shown for visualization purposes only. The missing z component is not a simpler special

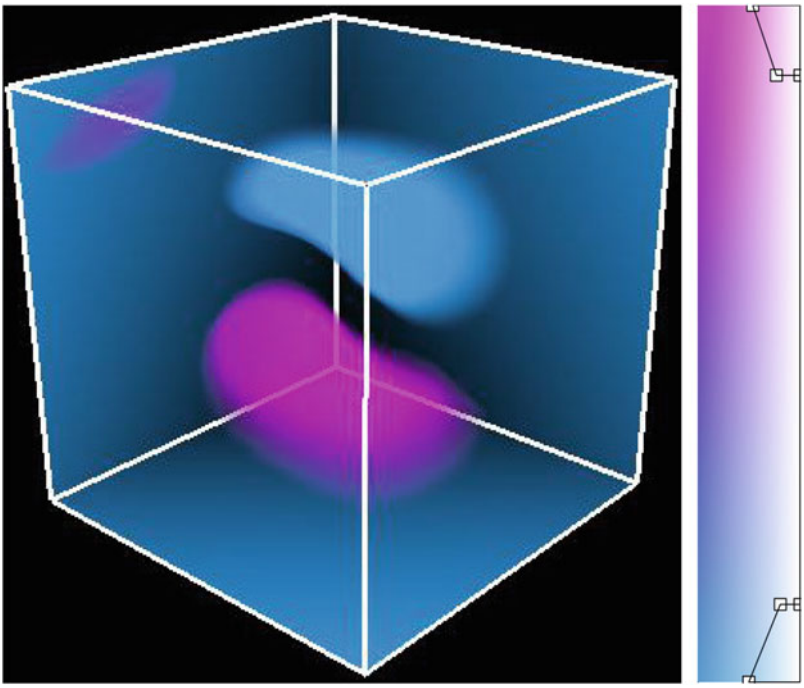


Fig. 2 Volume rendering of the pattern used in Fig. 3

case, but actually more challenging as the algorithm has fewer possible vectors from which to choose.

Please note that the suggested algorithm will coincide with its predecessors if the chosen pattern has a big enough quadratic part. In this case, it is therefore as robust as the former approach. If the pattern lacks this part, the former algorithm will produce an unreliable output, because it will try to determine the orientation of the eigenvectors of a matrix that is numerically zero.

5.2 3D Vector

We now move to the 3D vector case. So far, the normalization of vector field moments makes use of the first order moment tensor $\Sigma = {}^1M$ which again has second rank as in the scalar case. But in contrast to the latter, this matrix is not symmetric for vector fields. Bujack et al. [5] use the Schur form as standard position. The normalizer is the rotation that transforms Σ into an upper triangular matrix. Analogous to the scalar case, this method fails if Σ is zero. The algorithm in this paper is capable of overcoming this issue because of its flexibility. It can use any first rank contraction and is not bound to a tensor of a specific rank.

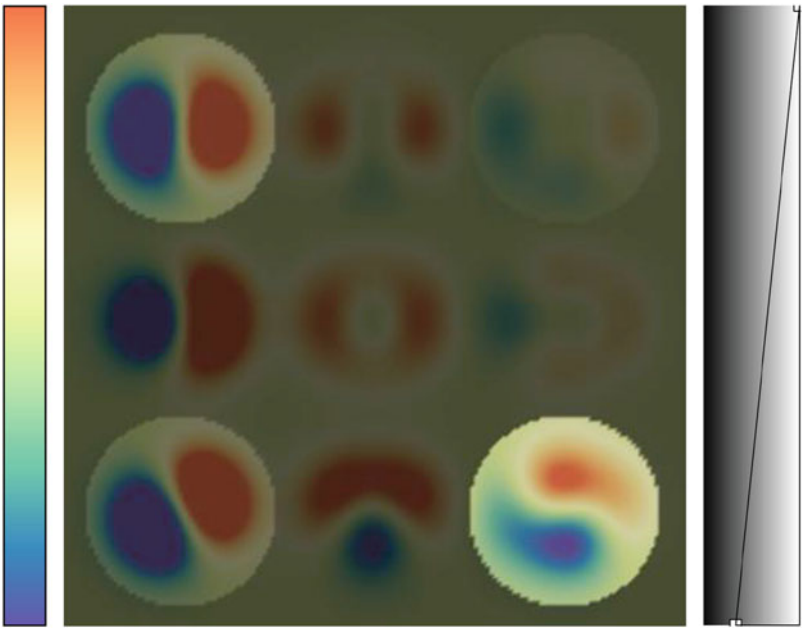


Fig. 3 Color coding of the $z = 0$ plane of the analytic scalar field. The similarity to the pattern from Fig. 2 is encoded using the transfer function on the right. The higher the similarity, the brighter the underlying field

We use the gradient of the scalar field from the preceding section as an example 3D vector field and calculate the moments up to third order, i.e. fourth rank. The randomly rotated and reflected pattern has a constant and a quadratic part. Its linear component now vanishes. Therefore, the second rank moment tensor Σ is zero and the method described in [5] does not yield any result. Algorithm 1 on the other hand retrieves one vector from the first rank tensor of the zeroth order moments $v_1 = {}^0M$ and one from contracting the first and the last index of the third rank tensor of the second order moments $v_2 = \sum_{(1,3)} {}^2M$. Figure 4 shows streamtubes of the randomly rotated and reflected target pattern and Fig. 5 is a visualization of the similarity within the field. The specific similarity values differ from the ones of the scalar field, but the relative order remains the same.

5.3 2D Tensor

For the tensor case, we constructed an analytic pattern shown in Fig. 6 and a corresponding analytic matrix field in Fig. 7, in which we placed the exact copy of the pattern and a squeezed one. Neither occurrence was aligned with the pattern, but

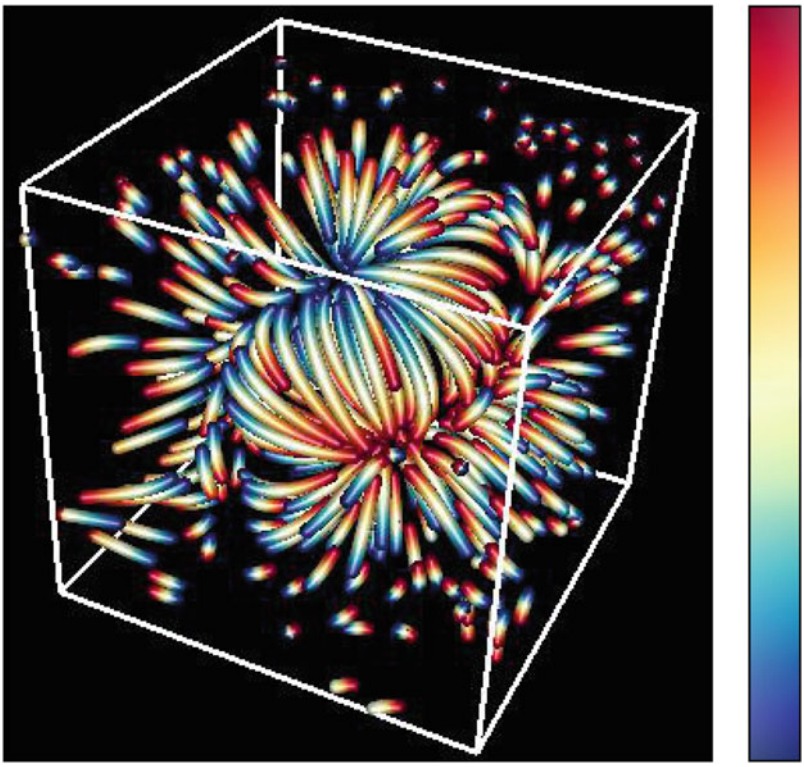


Fig. 4 Stream tubes of the pattern used in Fig. 5. *Red* implies moving forward and *blue* backward in time

the algorithm correctly detected its copy with the highest similarity and the distorted version with lower similarity.

The visualization of the pattern matching Fig. 7 is done analogously to [4]. We draw circles around the local similarity maxima. The color of the circles encodes the similarity and the diameter represents the integration area for the moment calculation that resulted in the maximum. For comparison, we lay the result of our algorithm on top of a tensor LIC [35] image of the field. This technique is based on the classical line integral convolution (LIC) for vector fields [7]. For symmetric matrix fields, the two LIC images that correspond to the directions of the eigenvectors are calculated and then either one of them or a combined image that interweaves both can be shown. More information about tensor visualization can be found in [24].

To demonstrate the applicability of the algorithm in real world applications, we applied our algorithm to the strain tensor field of the fluid dynamics simulation of the von Kármán vortex street from [2]. The strain tensor is the symmetric part of the Jacobian. It describes the separation of neighboring particles [20].

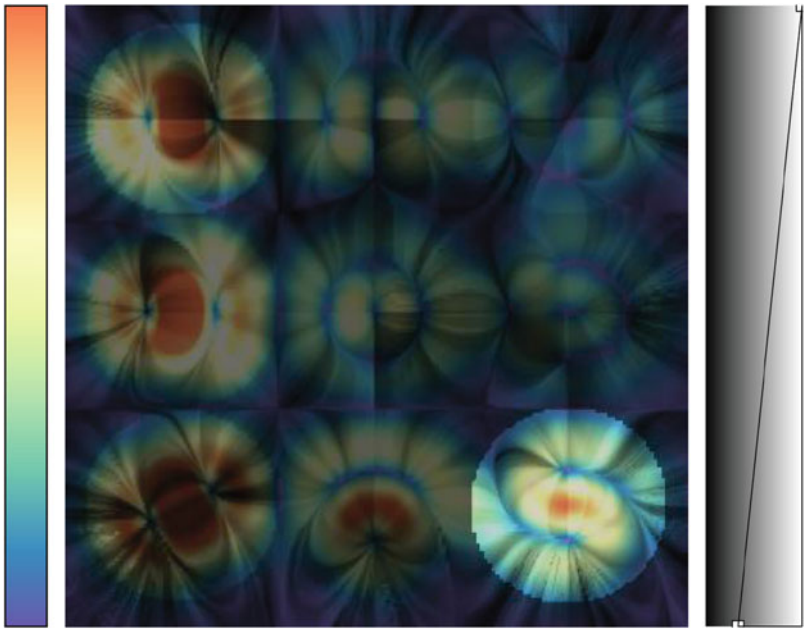


Fig. 5 Cut through the gradient of the analytic scalar field from Fig. 3 at the $z = 0$ plane is visualized with LIC and color encoding the velocity of the vector field. The similarity is encoded using the transfer function on the right. The higher the similarity, the brighter is the underlying field

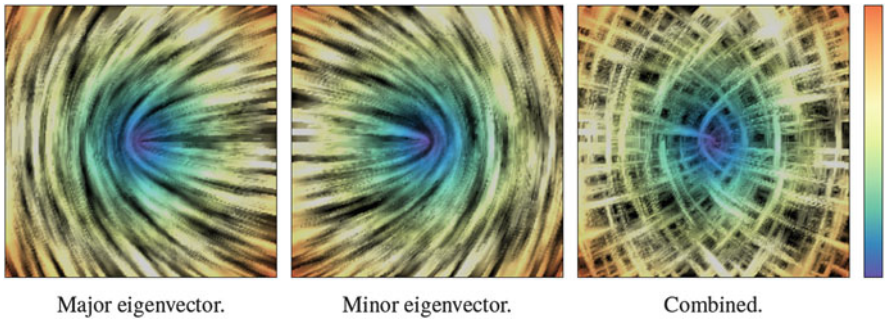


Fig. 6 Tensor LIC of the pattern used in Fig. 7. The color map encodes the Frobenius norm of the matrix

Figures 8, 9, 10, 11 show two pattern detection results visualized with tensor LIC [35]. Differently sized patterns were cut out from the field, randomly rotated, and searched for. We use moment tensors up to third order, i.e. fourth rank. The highest similarity peak corresponds to the location where the pattern was selected from. The repeating matches nicely show the repetitive structure of the vortex street. In Fig. 9, we normalized with respect to rotation, reflection, and scaling. Without

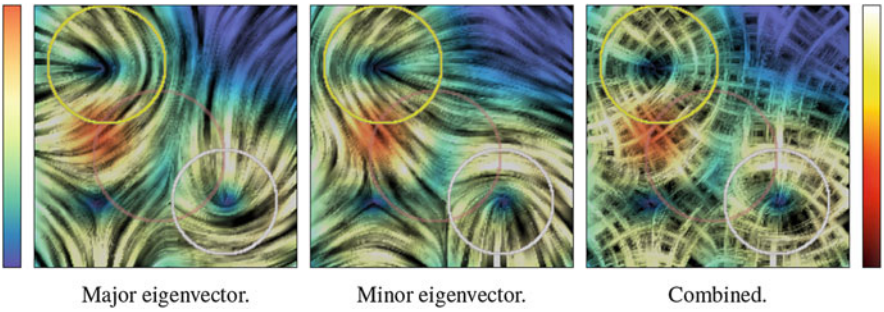


Fig. 7 Output of Algorithm 1 for the pattern from Fig. 6 in a matrix field. The color in the LIC corresponds to the Frobenius norm of the matrix, the color of the circles to the similarity

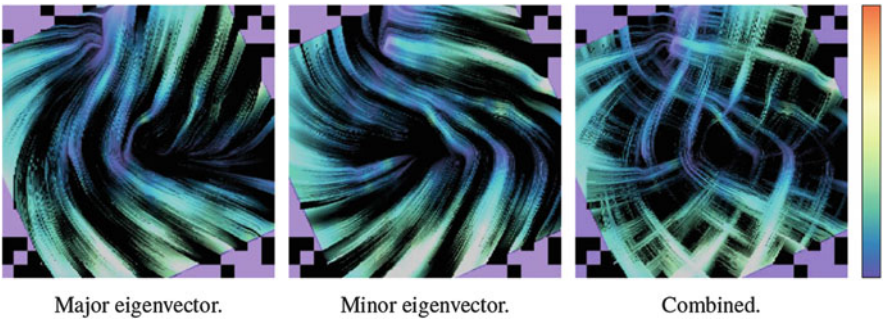


Fig. 8 Tensor LIC of the pattern used in Fig. 9. The color map encodes the Frobenius norm of the strain tensor

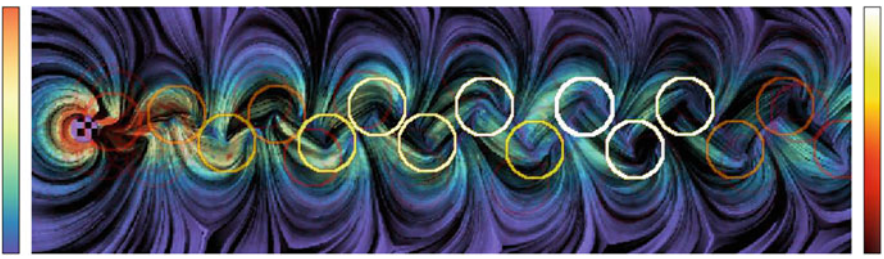


Fig. 9 Output of Algorithm 1 for the pattern from Fig. 8 in the strain tensor field of the von Kármán vortex street laid over the tensor LIC of the major eigenvector. The color in the LIC corresponds to the Frobenius norm of the strain tensor, the color of the circles to the similarity

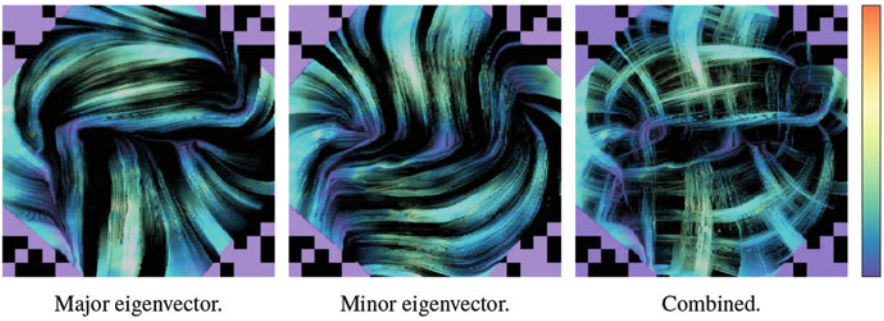


Fig. 10 Tensor LIC of the pattern used in Fig. 11. The color map encodes the Frobenius norm of the strain tensor

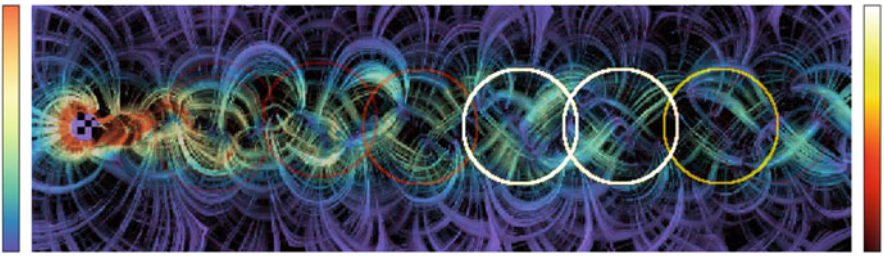


Fig. 11 Output of Algorithm 1 for the pattern from Fig. 10 in the strain tensor field of the von Kármán vortex street laid over the tensor LIC of both eigenvectors. The color in the LIC corresponds to the Frobenius norm of the strain tensor, the color of the circles to the similarity

the reflection enabled, there are no matches on the lower half of the vortex street. In Fig. 11, we normalized with respect to rotation and scaling. If reflection was enabled, additional matches would appear between each two circles.

5.4 3D Tensor

Similar to the 3D vector case, we apply Algorithm 1 to the Hessian, i.e. matrix of the second derivatives, of the scalar field from Fig. 3 to consider the 3D tensor case. We use superquadric tensor glyphs [23] and illuminated tensor lines [19, 34, 36] in directions of all three eigenvectors to visualize the randomly rotated and reflected pattern in Fig. 12. Figure 13 shows the output of the algorithm. We used moments up to second order, i.e. fourth rank, and normalized with respect to rotation, reflection, and scaling. Again, the position moment that was used to create the target pattern is clearly identified as the strongest match and the locations with lower resemblance follow. The visualization of the similarity is done using tensor LIC [35] color coded with the Frobenius norm of the Laplacian.

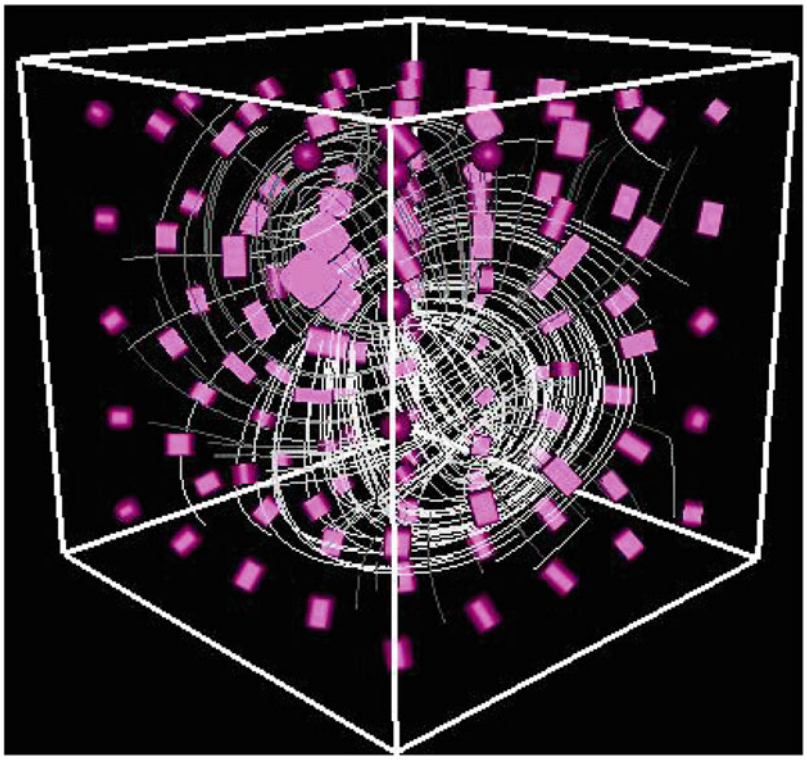


Fig. 12 Superquadric tensor glyphs and illuminated tensor lines of the pattern

6 Discussion

In this paper, we have elucidated the properties of the moment tensors of tensor fields of arbitrary dimension and rank in Theorem 2. We have applied this theoretical result to develop an algorithm for the generation of rotation invariants for two- and three-dimensional tensor fields using moment normalization and demonstrated its applicability to analytical data and simulation use cases. To our knowledge this is the first time that moment invariants have been practically applied to detect patterns in matrix fields. Further, we have shown how the algorithm improves existing algorithms for 3D scalar fields and 3D vector fields.

We would like to emphasize that the work on this topic is far from finished. The existing algorithms, both the normalization and the generator approach, still have some shortcomings. For the generator approach, the question of redundancy has not been completely solved yet and both approaches struggle to find symmetric patterns. For patterns that are not completely rotationally symmetric but show a certain rotational symmetry, all contractions to zeroth, first, and second rank may be zero and hence both the generator and the normalization approach fail. In 2D this

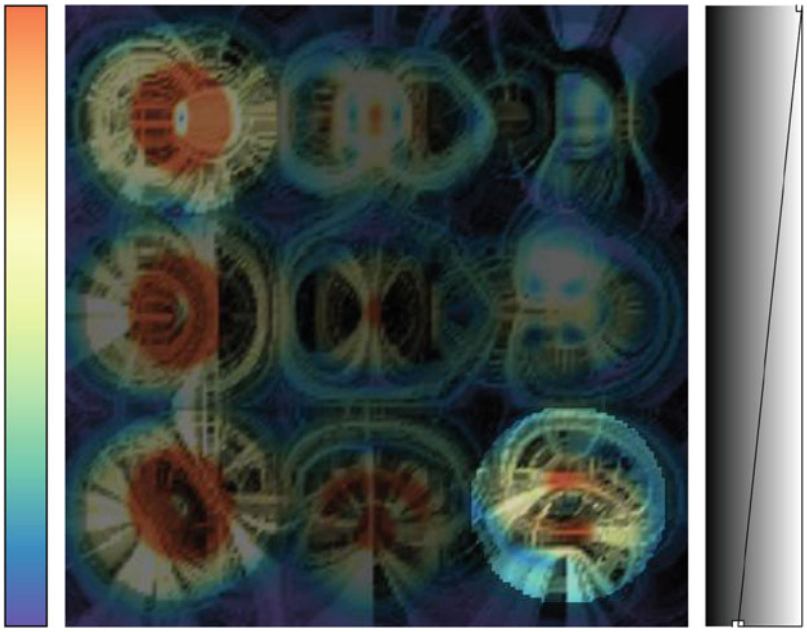


Fig. 13 Cut through the Laplacian of the analytic field from Fig. 3 at the $z = 0$ plane is visualized with tensor LIC. The color bar left encodes the Frobenius norm. The similarity is encoded using the transfer function on the right. The higher the similarity, the brighter is the underlying field

issue has been treated for scalar fields by Flusser and Suk [16], who generated bases specifically for this problem. For vector fields, Bujack et al. [3] solve the problem using a set of multiple standard positions if a rotationally symmetric moment is chosen for the normalization. It needs to be investigated in future work how or if either of these solutions can be generalized to contraction based methods.

Acknowledgements We would like to thank the FAnToM development group from Leipzig University for providing the environment for the visualization of the presented work. Further, we thank Professor Mario Hlawitschka for providing the dataset used in this publication and Terece Turton for help with the writing. This work was partly funded by the German Research Foundation (DFG) as part of the IRTG 2057 "Physical Modeling for Virtual Manufacturing Systems and Processes".

References

1. Bowen, R., Wang, C.: Introduction to Vectors and Tensors. Dover Books on Mathematics. Dover Publications, Mineola (2008)
2. Bujack, R., Hlawitschka, M., Joy, K.I.: Topology-inspired Galilean invariant vector field analysis. In: Proceedings of the IEEE Pacific Visualization Symposium, PacificVis 2016 in Taipei, Taiwan, pp. 72–79 (2016)

3. Bujack, R., Hotz, I., Scheuermann, G., Hitzer, E.: Moment invariants for 2D flow fields using normalization. In: IEEE Pacific Visualization Symposium, PacificVis 2014 in Yokohama, Japan, pp. 41–48 (2014)
4. Bujack, R., Hotz, I., Scheuermann, G., Hitzer, E.: Moment invariants for 2D flow fields via normalization in detail. IEEE Trans. Vis. Comput. Graph. **21**(8), 916–929 (2015)
5. Bujack, R., Kasten, J., Hotz, I., Scheuermann, G., Hitzer, E.: Moment invariants for 3D flow fields via normalization. In: IEEE Pacific Visualization Symposium, PacificVis 2015 in Hangzhou, China (2015)
6. Burel, G., Henocq, H.: 3D invariants and their application to object recognition. Signal Process. **45**(1), 1–22 (1995)
7. Cabral, B., Leedom, L.C.: Imaging vector fields using line integral convolution. In: Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '93, pp. 263–270. ACM (1993)
8. Canterakis, N.: Complete moment invariants and pose determination for orthogonal transformations of 3D objects. In: Mustererkennung 1996, 18. DAGM Symposium, Informatik aktuell, pp. 339–350. Springer (1996)
9. Cyganski, D., Kreda, S.J., Orr, J.A.: Solving for the general linear transformation relating 3-d objects from the minimum moments. In: 1988 Robotics Conferences, pp. 204–211. International Society for Optics and Photonics (1989)
10. Cyganski, D., Orr, J.A.: Applications of tensor theory to object recognition and orientation determination. IEEE Trans. Pattern Anal. Mach. Intell. **7**(6), 662–673 (1985)
11. Dirilten, H., Newman, T.G.: Pattern matching under affine transformations. IEEE Trans. Comput. **100**(3), 314–317 (1977)
12. Florack, L., Romeny, B.T.H., Koenderink, J., Viergever, M.: Cartesian differential invariants in scale-space. J. Math. Imaging Vision **3**(4), 327–348 (1993)
13. Florack, L., Romeny, B.T.H., Koenderink, J.J., Viergever, M.A.: General intensity transformations and differential invariants. J. Math. Imaging Vision **4**(2), 171–187 (1994)
14. Flusser, J.: On the independence of rotation moment invariants. Pattern Recognit. **33**(9), 1405–1410 (2000)
15. Flusser, J.: On the inverse problem of rotation moment invariants. Pattern Recognit. **35**, 3015–3017 (2002)
16. Flusser, J., Suk, T.: Rotation moment invariants for recognition of symmetric objects. IEEE Trans. Image Process. **15**(12), 3784–3790 (2006)
17. Flusser, J., Zitova, B., Suk, T.: Moments and Moment Invariants in Pattern Recognition. John Wiley & Sons, Chichester (2009)
18. Grinfeld, P.: Introduction to Tensor Analysis and the Calculus of Moving Surfaces. Springer, New York (2013)
19. Hlawitschka, M., Scheuermann, G.: Hot-lines: tracking lines in higher order tensor fields. In: Proceedings of IEEE Visualization, pp. 27–34. IEEE (2005)
20. Hotz, I., Feng, L., Hagen, H., Hamann, B., Joy, K., Jeremic, B.: Physically based methods for tensor field visualization. In Proceedings of IEEE Visualization, pp. 123–130. IEEE (2004)
21. Hu, M.-K.: Visual pattern recognition by moment invariants. IRE Trans. Inf. Theory **8**(2), 179–187 (1962)
22. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Symposium on Geometry Processing (2003)
23. Kindlmann, G.: Superquadric tensor glyphs. In: Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization, pp. 147–154. Eurographics Association (2004)
24. Laidlaw, D.H., Weickert, J.: Visualization and Processing of Tensor Fields: Advances and Perspectives. Springer Science & Business Media, Berlin (2009)
25. Langbein, M., Hagen, H.: A generalization of moment invariants on 2D vector fields to tensor fields of arbitrary order and dimension. In: International Symposium on Visual Computing, pp. 1151–1160. Springer, Berlin (2009)
26. Lo, C., Don, H.: 3-D moment forms: their construction and application to object identification and positioning. IEEE Trans. Pattern Anal. Mach. Intell. **11**(10), 1053–1064 (1989)

27. Pinjo, Z., Cyganski, D., Orr, J.A.: Determination of 3-D object orientation from projections. Pattern Recognit. Lett. **3**(5), 351–356 (1985)
28. Reiss, T.H.: The revised fundamental theorem of moment invariants. IEEE Trans. Pattern Anal. Mach. Intell. **13**(8), 830–834 (1991)
29. Schlemmer, M.: Pattern recognition for feature based and comparative visualization. PhD thesis, Universität Kaiserslautern, Germany (2011)
30. Schlemmer, M., Heringer, M., Morr, F., Hotz, I., Hering-Bertram, M., Garth, C., Kollmann, W., Hamann, B., Hagen, H.: Moment invariants for the analysis of 2D flow fields. IEEE Trans. Vis. Comput. Graph. **13**(6), 1743–1750 (2007)
31. Schlemmer, M., Hotz, I., Hamann, B., Hagen, H.: Comparative visualization of two-dimensional flow data using moment invariants. In: Proceedings of Vision, Modeling, and Visualization (VMV'09), vol. 1, pp. 255–264 (2009)
32. Suk, T., Flusser, J.: Tensor method for constructing 3D moment invariants. In: Computer Analysis of Images and Patterns. Lecture Notes in Computer Science, vol. 6855, pp. 212–219. Springer, Berlin (2011)
33. Suk, T., Flusser, J., Boldyš, J.: 3D rotation invariants by complex moments. Pattern Recognit. **48**(11), 3516–3526 (2015)
34. Weinstein, D., Kindlmann, G., Lundberg, E.: Tensorlines: advection-diffusion based propagation through diffusion tensor fields. In: Proceedings of IEEE Visualization, pp. 249–253. IEEE (1999)
35. Zheng, X., Pang, A.: Hyperlic. In: Proceedings of IEEE Visualization, pp. 249–256. IEEE (2003)
36. Zöckler, M., Stalling, D., Hege, H.-C.: Interactive visualization of 3D-vector fields using illuminated stream lines. In: Proceedings of IEEE Visualization, pp. 107–113. IEEE (1996)

Visualizing Gradients of Stress Tensor Fields

Valentin Zobel, Markus Stommel, and Gerik Scheuermann

Abstract In some applications, it is necessary to look into gradients of (symmetric) second order tensor fields. These tensors are of third order. In three-dimensional space, we have 18 independent coefficients at each position, so the visualization of these fields provides a challenge. A particular case are stress gradients in structural mechanics. There are specific situations where the stress gradient is required together with the stress to study material behavior. Since the visualization community lacks methods to show these fields, we look at some preliminary ideas to design appropriate glyphs. We motivate our glyph designs by typical depictions of stress in engineering textbooks.

1 Introduction

The prediction of failure is a crucial question in engineering. To predict failure, usually a structure simulation is performed, which computes the stresses inside the material under a predefined load condition. In a next step, a failure criterion is used to evaluate the resulting stress tensor. Such yield criteria, like the widely-used von Mises stress or the Tsai-Hill criterion, are usually functions depending on the Cauchy stress tensor. The Cauchy stress tensor, in the following only called stress tensor, is an indefinite, symmetric tensor of second order which defines the complete state of stress at a single point. However, there is some evidence that not only the stress at a single point is decisive for failure, but also the stress values in the vicinity, i. e. the gradient of the stress tensor. For example, the stress gradient is used to analyze the effect of stress concentrations caused by notches. The so-called notch effect plays an important role for fatigue of materials, see for example [11] or [12]. In this case, the material is analyzed under a simple load condition, i. e. a

V. Zobel (✉) • G. Scheuermann
Leipzig University, Augustusplatz 10, 04109 Leipzig, Germany
e-mail: zobel@informatik.uni-leipzig.de; scheuermann@informatik.uni-leipzig.de

M. Stommel
TU Dortmund University, Leonhard-Euler-Str. 5, 44227 Dortmund, Germany
e-mail: markus.stommel@tu-dortmund.de

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_4

unidirectional stretching, such that the stress can be reduced to a scalar quantity. Then the derivative in direction perpendicular to the notch surface is considered.

This makes it promising to consider the gradient of the stress tensor for failure prediction also in other situations. But for many technical materials there is no obvious or sufficiently proven way to reduce the stress tensor to a scalar value that describes the material stressing adequate. Thus a thorough understanding of the whole stress gradient is desirable to examine the effects of the stress gradient on material failure. We propose some first visualizations to analyze the whole gradient of the stress tensor. The gradient of a three-dimensional stress tensor consists of 18 independent components, given by the partial derivatives of the 6 components of the symmetric stress tensor. Together with the stress tensor itself there are 24 components at each position which have to be visualized. This makes the visualization of these data very challenging.

In this chapter, we propose glyphs that allow to analyze the stress tensor and its gradient at discrete points. Based on widely used depictions of the stress vector in engineering textbooks, we suggest a glyph that completely visualizes the stress tensor and the gradient. Due to the complexity of this glyph we discuss ways to reduce the shown information in a systematic fashion, which allows for simpler visualizations. For example, if we are interested in the overall change of the tensor in certain directions, we can reduce directional derivatives, which are symmetric second order tensors, to scalar quantities. Similarly, we can focus on the normal stress in certain directions while neglecting the direction of differentiation. Such visualizations allow to highlight different properties of the full gradient of the tensor consecutively.

2 Related Work

There seems to be very few work on the visualization of tensor gradients. Kriz et al. [10] consider gradients of the Cauchy stress tensor. They visualize the gradient at a single point by showing a collection of stress ellipsoids. The principal axes of a stress ellipsoid are defined by the eigenvectors of the stress tensor, their radii are given by the absolute values of the corresponding eigenvalues. One stress ellipsoid is used to represent the stress at the considered position itself, additional ellipsoids show the stress in nearby positions in the direction of the coordinate axes. The user can guess the gradient by comparing these glyphs. Kindlmann et al. analyze the gradient of diffusion tensors in [7] by examining gradients of scalar quantities derived from the tensor. The gradient of a tensor field is characterized by the gradients of three tensor invariants, as well as three spatial gradients of rotation. The gradients of rotation measure the rotational part of the gradient with respect to infinitesimal rotations of the tensor around its eigenvectors. Thus the 18 components of the gradient of a symmetric tensor are decomposed into 6 gradients of scalar quantities. For the visualization the magnitudes of these gradients are shown by grayscale images.

We state some other publications focusing on the visualization of stress tensor fields. For an overview on visualization methods for indefinite tensor fields we refer the reader to a state of the art report [8] and the references therein. In [9] the tensor lines of a stress tensor field, i. e. the field lines of the eigenvector fields, are used to improve the rib structure of components. Ribs are often used to improve the stability and rigidity of components, but the proper placement of ribs is an open question. The tensor lines are considered as major load paths, ribs constructed along tensor lines successfully improved the stability. An approach to improve the stability for components from fiber reinforced polymers is presented in [14]. Fiber orientations are rated according to their ability for supporting the load. Jeremic et al. consider stress tensor fields in a geomechanical context in [5]. In [2] they are used for implant planning in orthopedics.

Since this work uses glyphs to visualize gradients of stress tensor fields, we provide some references to publications on tensor glyphs for second order tensors. A comparison of different types of glyphs is given in [3]. Additionally a glyph focusing on the visualization of the shear stresses is provided. Superquadrics for positive definite tensor fields are introduced in [6], an extension to indefinite tensors is proposed in [13]. The state of the art report [1] gives general design guidelines for glyphs as well as strategies for glyph placement.

3 Theoretical Background

In this section we quickly recap some basics on three-dimensional tensor fields and state the gradient of a tensor field. We also discuss the Cauchy stress tensor and give an interpretation in terms of stress vectors, which serve as a basis for common depictions of the stress tensor in engineering textbooks. Such depictions are the starting point for the visualizations presented in Sect. 4.

3.1 Tensor Fields

In this chapter we are concerned with three-dimensional tensor fields of second order and their gradients, which are tensors of third order. In general, a (covariant) tensor of order r is a map

$$T \;:\; \mathbb{R}^3 \times \cdots \times \mathbb{R}^3 \to \mathbb{R} \;,$$

which takes r copies of $\mathbb{R}^3$ to a real number. Alternatively, a tensor can also be given by its components with respect to a (usually orthonormal) basis $\{e_1, e_2, e_3\}$ of $\mathbb{R}^3$.

The components are given by

$$T_{i_1 \dots i_r} := T(e_{i_1}, \dots, e_{i_r}) \ , \quad 1 \leq i_1, \dots, i_r \leq 3 \ .$$

Consequently, a second order tensor can also be considered as a matrix $T \in \mathbb{R}^{3,3}$. A second order tensor T is called *symmetric* if $T(v, w) = T(w, v)$ for all $v, w \in \mathbb{R}^3$, thus we consider symmetric tensors also as symmetric matrices denoted by Sym(3).

A three-dimensional *tensor field* T of order r defined on a subset $U \subset \mathbb{R}^3$ is a map assigning each $p \in U$ a three-dimensional tensor of order r. A second order tensor field is called symmetric if this condition holds pointwise, this yields a map

$$T \ : \ U \to \mathrm{Sym}(3) \ .$$

The stress tensor fields considered in this chapter are symmetric tensor fields. We assume the tensor field to be differentiable, which is equivalent to the condition that the components of the tensor

$$p \mapsto T_{ij}(p) \ , \quad 1 \leq i, j \leq 3 \ ,$$

form differentiable maps.

3.2 Gradients of Tensor Fields

The set of three-dimensional tensors of order r is isomorphic to $\mathbb{R}^{3^r}$, thus the definition of the derivative applies also for tensor fields. We consider the gradient $\nabla T(p)$ of a second order tensor field T at a point $p \in U$ as a linear map

$$\begin{aligned} \nabla T(p) \ &: \ \mathbb{R}^3 \to \mathbb{R}^{3,3} \ , \\ &\quad v \mapsto (v_1 \partial_1 + v_2 \partial_2 + v_3 \partial_3)\, T(p) \ , \end{aligned} \tag{1}$$

where ∂_i denotes the partial derivative in direction of the basis vector e_i. Thus $[\nabla T(p)]v$ is the directional derivative of T in direction v, which we also denote by $\nabla_v T(p)$. In this way, differentiation adds an additional argument to the second order tensor T, so $\nabla T(p)$ is a third order tensor with components

$$[\nabla T(p)]_{ijk} = \partial_k T_{ij}(p) \ ,$$

i. e. the components of $\nabla T(p)$ are the partial derivatives of the components of $T(p)$.

3.3 Stress Tensor

A load condition causes stresses inside a material and the complete state of stress is given by the Cauchy stress tensor, which defines an indefinite, symmetric tensor field

$$\sigma = \begin{pmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{xy} & \sigma_y & \tau_{yz} \\ \tau_{xz} & \tau_{yz} & \sigma_z \end{pmatrix} .$$

The components σ and τ are called the normal and shear stresses, respectively. The stress tensor is often interpreted in terms of the stress vector, which serves also as a starting point for our visualizations. For a given normal vector n the *stress vector* σn describes the forces acting on a plane perpendicular to n. The projection of the stress vector σn on the normal n is called the normal stress, the projection on the plane perpendicular to n is called the shear stress. Due to the symmetry of σ there is a orthonormal basis of eigenvectors n_1, n_2, n_3, which are called the principal directions. The corresponding eigenvalues $\sigma_1, \sigma_2, \sigma_3$ are called principal stresses. The shear stresses vanish on the planes perpendicular to the eigenvectors n_1, n_2, n_3, there are only normal stresses. Positive eigenvalues indicate tensile stresses, negative eigenvalues indicate compressive stresses.

Stress vectors are often used to depict the stress tensor. Given an orthonormal coordinate system e_1, e_2, e_3, the stress tensor σ is completely described by the three stress vectors $\sigma e_1, \sigma e_2, \sigma e_3$. The three planes perpendicular to e_1, e_2, e_3 define the sides of a cube, which is shown together with the three stress vectors, see Fig. 1 left.

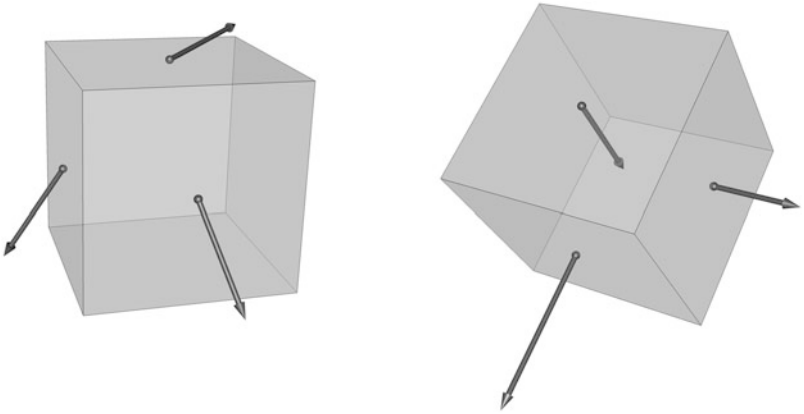


Fig. 1 Widely-used illustration of the stress tensor. The *arrows* depict stress vectors, that is the force acting on the planes given by the sides of the cube. For the right image the cube is oriented according to the principal stress directions, thus there is only normal stress and the stress vectors are perpendicular to the sides of the cube. An *outward pointing arrow* represents tension, an *inward pointing arrow* represents compression

If the eigenvectors of σ are used as coordinate system, the three stress vectors are perpendicular to the sides of the cube, see Fig. 1 right. Note that such an illustration is not unique if there are equal eigenvalues. Similar illustrations of the stress tensor are used in many textbooks of solid mechanics, see for example [4].

4 Glyphs for Gradients of Stress Tensor Fields

Our goal is to visualize the stress tensor σ and its gradient $\nabla\sigma$ at single points of a stress tensor field. From such visualizations we want to gain a thorough understanding of the tensor field at points of high load. The information of the stress tensor and its gradient is given by the linearization L of the stress tensor field at a point p, which is given by

$$\begin{aligned} L \;:\; \mathbb{R}^3 &\to \mathrm{Sym}(3) \;, \\ v &\mapsto \sigma(p) + \nabla_v \sigma(p) \;, \end{aligned} \tag{2}$$

i. e. $L(v)$ is the first order Taylor approximation of σ around p. We will usually omit p in the following formulas, since the point p is secondary for the glyphs suggested in the remainder of this section. As mentioned before, we focus on the stress vector in our visualizations. Based on the linearization L we define a map

$$\begin{aligned} \mathbf{t} \;:\; \mathbb{S}^2 \times \mathbb{S}^2 &\to \mathbb{R}^3 \;, \\ (v, n) &\mapsto \sigma n + \epsilon \nabla_v \sigma n \;, \end{aligned} \tag{3}$$

where $\epsilon > 0$ and $\mathbb{S}^2 \subset \mathbb{R}^3$ denotes the unit sphere. For a given direction of differentiation $v \in \mathbb{S}^2$ and a normal vector $n \in \mathbb{S}^2$, $\mathbf{t}(v, n)$ is the (linearized) change of the stress vector σn for a displacement by ϵ in direction v. The parameter ϵ serves as a scaling of the gradient, it controls the size of the (linearized) neighborhood taken into consideration. A meaningful choice depends on the dimensions of the data. The stress tensor σ and its gradient $\nabla\sigma$ are still completely characterized by the graph of $\mathbf{t}$ which is given by

$$\left\{ (v, n, \mathbf{t}(v, n)) \in \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^3 \mid n \in \mathbb{S}^2, \;\; v \in \mathbb{S}^2 \right\} \;.$$

The glyphs presented in the following can be considered as visualizations of (parts of) the graph of $\mathbf{t}$. Of course, the graph of $\mathbf{t}$ contains redundant information, thus it is not necessary to visualize the whole graph in order to visualize σ and $\nabla\sigma$. The map $\mathbf{t}$ is linear with respect to n and affine linear with respect to v, moreover σ is symmetric. Thus it is possible to show portions of the graph of $\mathbf{t}$ without losing information. But we will also propose glyphs which do not show the complete information of σ and $\nabla\sigma$, with the benefit of easier understandable visualizations.

4.1 Visualizing Gradients of Stress Vectors

The glyphs presented in this section extend the depiction of the stress vectors shown in Fig. 1 with information on the gradient of the stress vectors. Our first glyph uses additional arrows which give information on the partial derivatives of the stress vectors. A variant of this glyph shows all stress vectors that occur in a linearized neighborhood.

4.1.1 Partial Derivatives of Stress Vectors

At first we depict the basis vectors similarly to Fig. 1. We choose a basis of eigenvectors n_1, n_2, n_3 and show a cube with sides defined by the planes normal to n_1, n_2, n_3. The respective stress vectors $\sigma n_1, \sigma n_2, \sigma n_3$ are shown as gray tubes. Since we use a basis of eigenvectors they are perpendicular to the sides of the cube. Now, the variations of the stress vectors in the coordinate directions e_1, e_2, e_3 are depicted as arrows in red, green or blue, respectively. These arrows are the partial derivatives of the stress vector $\epsilon \partial_i \sigma n_j$ for $i, j = 1, \ldots, 3$, while the ϵ adjusts the displacement as in Eq. (3). An example is shown in Fig. 2. Note that the coordinate system e_1, e_2, e_3 for the directional derivatives might be different to the coordinate system defined by the eigenvectors n_1, n_2, n_3. The coordinate system e_1, e_2, e_3 is usually a global coordinate system which is consistent with the application and intuitive for the user. For example, the tensile bars considered in the next section are aligned parallel to the coordinate axes and the stress direction is parallel to e_3.

The proposed visualization of the stress vectors and its partial derivatives contains the subset

$$\{(v, n, \mathbf{t}(v, n)) \mid n = n_1, n_2, n_3, \quad v = 0, e_1, e_2, e_3\}$$

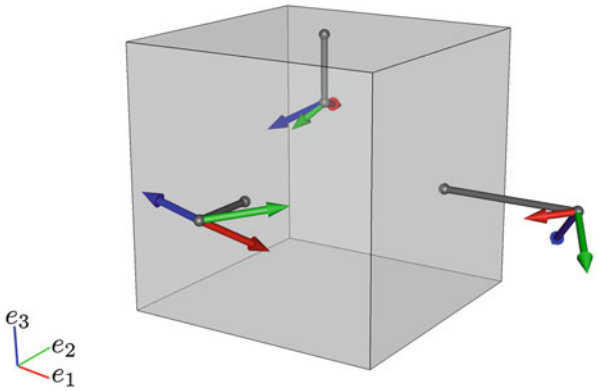


Fig. 2 The *colored arrows* depict directional derivatives of the stress vectors. The *red, green, blue arrows* correspond to a displacement in direction of e_1, e_2, e_3, respectively

of the graph of **t**. Since **t** is linear with respect to n and affine linear with respect to v, this subset describes **t** completely. Equivalently, a stress tensor σ is completely defined by the three stress vectors $\sigma n_1, \sigma n_2, \sigma n_3$ and the gradient $\nabla\sigma$ is completely defined by the partial derivatives of the stress vectors $\partial_i \sigma n_j$ for $i, j = 1, \ldots, 3$. Consequently, all information of σ and $\nabla\sigma$ is included in this visualization. An obvious drawback of this method is that it needs a thorough examination to read all the information. Especially the direction of differentiation, which is encoded in the colors red, green and blue, might not be immediately apparent for many users. This makes it comprehensible to reduce the presented information and, in return, obtain a visualization that is easier to understand.

4.1.2 Envelope of Stress Vectors under Linear Approximation

As a first simplification of the glyph proposed in the previous section we replace the partial derivatives of the stress vectors, which are shown as red, green and blue arrows. Instead of showing single arrows for the partial derivatives of the stress vectors we show the envelope of all directional derivatives of the stress vectors, i. e. $\epsilon\nabla_v \sigma n$ for all $v \in \mathbb{S}^2$. In this way we show all stress vectors that occur in a linearized neighborhood of radius ϵ. An example is shown in Fig. 3. For a fixed normal vector n, the envelope of all directional derivatives is the image of the unit sphere $\mathbb{S}^2$ under the linear transformation

$$\begin{aligned} \epsilon\nabla\sigma n \;\; : \;\; \mathbb{R}^3 &\to \mathbb{R}^3 \\ v &\mapsto \epsilon\nabla_v \sigma n \;\; . \end{aligned}$$

Thus the envelopes form ellipsoids, while the principal axes are defined by the eigenvectors of $\epsilon\nabla\sigma n$ and their radii are given by the absolute values of the

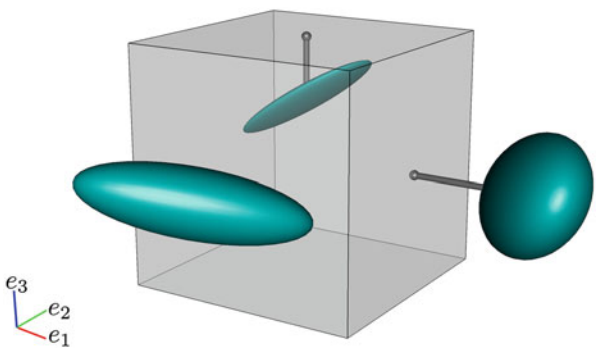


Fig. 3 The ellipsoids depict the envelopes of the stress vectors within a linearized neighborhood, i.e. the ellipsoid marks the end points of all stress vectors occurring in this neighborhood

corresponding eigenvalues. The depicted information can be summarized as

$$\left\{(n, \mathbf{t}(v, n)) \mid n = n_1, n_2, n_3, \quad v \in \mathbb{S}^2\right\} \quad .$$

Note that we lose information on the preimage of v here, but compared to the visualization in Sect. 4.1.1 we show the image of all $v \in \mathbb{S}^2$ rather than only the image of the three basis vectors. So, if one is interested in the change of the stress vectors in all directions, but the particular direction is not important at first, then this visualization might be simpler to read than the visualization in the previous section.

4.2 Reducing Directional Derivatives of Stress Vectors to Scalar Quantities

In the previous section we have visualized the gradient of the stress tensor completely, but the resulting visualizations might be difficult to read. Now we want to reduce the shown information by focussing on special properties of the gradient. The basic idea is to consider a fixed direction of differentiation $v \in \mathbb{S}^2$ and measure the change of the stress vectors for all normal directions $n \in \mathbb{S}^2$. Or, conversely, for a normal direction $n \in \mathbb{S}^2$ we measure the overall change for all directions of differentiation $v \in \mathbb{S}^2$.

To realize this we reduce $\nabla_v \sigma n$ to a scalar quantity, first. An obvious choice is to use the magnitude, i. e. the euclidean norm $\|\nabla_v \sigma n\|$. Another possibility is using only the directional derivative of the normal stress of $\nabla_v \sigma n$ which is given by $\nabla_v n^\top \sigma n = n^\top \nabla_v \sigma n$. In the next step we compute the average, the maximum or the minimum of these scalar quantities for variable n or v, while v or n is fixed, respectively. This yields functions $\mathbb{S}^2 \to \mathbb{R}$ depending on v or n, which are easier to visualize than the complete gradient.

4.2.1 Magnitude of Directional Derivatives of Stress Vectors

At first we consider the magnitude of the directional derivatives of the stress vector. Computing the average/minimum/maximum of $\|\nabla_v \sigma n\|$ for all n or v might seem expensive, but by using some mathematical conversions these quantities can be computed efficiently. Note that we have for any matrix $A \in \mathbb{R}^{3,3}$

$$\int_{\mathbb{S}^2} x^\top A x \, dx = \frac{4\pi}{3} \operatorname{tr} A \quad . \tag{4}$$

Thus, if we use the L^2-norm for averaging $\|\nabla_v \sigma n\|$ over $n \in \mathbb{S}^2$ we obtain

$$\sqrt{\int_{\mathbb{S}^2} \|\nabla_v \sigma n\|^2 \, dn} = \sqrt{\int_{\mathbb{S}^2} n^T (\nabla_v \sigma)^T (\nabla_v \sigma) n \, dn}$$
$$= \sqrt{\frac{4\pi}{3}} \sqrt{\mathrm{tr}\left((\nabla_v \sigma)^\top (\nabla_v \sigma)\right)} = \sqrt{\frac{4\pi}{3}} \|\nabla_v \sigma\|_F \ ,$$

where $\|\nabla_v \sigma\|_F$ denotes the Frobenius norm of $\nabla_v \sigma \in \mathbb{R}^{3,3}$, i. e. the Frobenius norm of the directional derivative of σ in direction v. Similarly, for the average over $v \in \mathbb{S}^2$ we have

$$\sqrt{\int_{\mathbb{S}^2} \|\nabla_v \sigma n\|^2 \, dv} = \sqrt{\frac{4\pi}{3}} \|\nabla \sigma n\|_F \ ,$$

while $\nabla \sigma n$ is the gradient of the stress vector σn. In this way we can compute the average by computing the Frobenius norm of a matrix, rather than computing the integral.

The extrema of $\|\nabla_v \sigma n\|$ for all $n \in \mathbb{S}^2$ or all $v \in \mathbb{S}^2$ can be computed from the following consequence of the Courant-Fischer theorem. For any symmetric matrix A we have

$$\max_{x \in \mathbb{S}^2} x^\top A x = \lambda_{max}(A) \ , \quad \min_{x \in \mathbb{S}^2} x^\top A x = \lambda_{min}(A) \ , \tag{5}$$

while $\lambda_{min}(A)$ and $\lambda_{max}(A)$ denote the smallest and the largest eigenvalue of A, respectively. Now, the maximum of $\|\nabla_v \sigma n\|$ for all $n \in \mathbb{S}^2$ is given by

$$\max_{n \in \mathbb{S}^2} \|\nabla_v \sigma n\| = \sqrt{\max_{n \in \mathbb{S}^2} n^T (\nabla_v \sigma)^\top (\nabla_v \sigma) n} = \sqrt{\lambda_{max}\left((\nabla_v \sigma)^\top (\nabla_v \sigma)\right)} \ ,$$

and the minimum is given by

$$\min_{n \in \mathbb{S}^2} \|\nabla_v \sigma n\| = \sqrt{\lambda_{min}\left((\nabla_v \sigma)^\top (\nabla_v \sigma)\right)} \ .$$

In a similar way we obtain for the extrema of $\|\nabla_v \sigma n\|$ with respect to v, which are given by

$$\max_{v \in \mathbb{S}^2} \|\nabla_v \sigma n\| = \sqrt{\lambda_{max}\left((\nabla \sigma n)^\top (\nabla \sigma n)\right)} \ ,$$

$$\min_{v \in \mathbb{S}^2} \|\nabla_v \sigma n\| = \sqrt{\lambda_{min}\left((\nabla \sigma n)^\top (\nabla \sigma n)\right)} \ .$$

Note that for any matrix A the largest singular value is $\sqrt{\lambda_{max}(A^\top A)}$, and the smallest singular value is $\sqrt{\lambda_{min}(A^\top A)}$, so the extrema of $\|\nabla_v \sigma n\|$ over n or v are given by the largest and smallest singular values of $\nabla_v \sigma$ and $\nabla \sigma n$, respectively. Since $\nabla_v \sigma$ is symmetric, the largest and smallest singular values of $\nabla_v \sigma$ are also largest and smallest absolute eigenvalues of $\nabla_v \sigma$, respectively.

To sum up, for a fixed direction of differentiation $v \in \mathbb{S}^2$ we can compute the average/minimal/maximal change of the stress vectors for all $n \in \mathbb{S}^2$ from the Frobenius norm or the singular values of $\nabla_v \sigma$. And for a fixed normal vector $n \in \mathbb{S}^2$ we can compute the average/minimum/maximum over all directions of differentiation $v \in \mathbb{S}^2$ from the Frobenius norm or the singular values of $\nabla \sigma n$.

As an example we consider the average of $\|\nabla_v \sigma n\|$ for all normal vectors $n \in \mathbb{S}^2$, i.e. we want to visualize $\sqrt{4\pi/3}\|\nabla_v \sigma\|_F$ for each direction v. The values of $\|\nabla_v \sigma\|_F$ can be computed for each v by a quadratic form which is defined by a matrix Q. If we denote the partial derivatives of σ by $\partial_1 \sigma, \partial_2 \sigma, \partial_3 \sigma$ we have

$$
\begin{aligned}
\|\nabla_v \sigma\|_F &= \|v_1 \partial_1 \sigma + v_2 \partial_2 \sigma + v_3 \partial_3 \sigma\|_F \\
&= \left(\operatorname{tr}\left((v_1 \partial_1 \sigma + v_2 \partial_2 \sigma + v_3 \partial_3 \sigma)^T (v_1 \partial_1 \sigma + v_2 \partial_2 \sigma + v_3 \partial_3 \sigma)\right)\right)^{\frac{1}{2}} \\
&= \big(v_1^2 \operatorname{tr}(\partial_1 \sigma^2) + v_2^2 \operatorname{tr}(\partial_2 \sigma^2) + v_3^2 \operatorname{tr}(\partial_3 \sigma^2) \\
&\qquad + 2 v_1 v_2 \operatorname{tr}(\partial_1 \sigma \partial_2 \sigma) + 2 v_1 v_3 \operatorname{tr}(\partial_1 \sigma \partial_3 \sigma) + 2 v_2 v_3 \operatorname{tr}(\partial_2 \sigma \partial_3 \sigma)\big)^{\frac{1}{2}} \\
&= \left(v^T \begin{pmatrix} \operatorname{tr}(\partial_1 \sigma^2) & \operatorname{tr}(\partial_1 \sigma \partial_2 \sigma) & \operatorname{tr}(\partial_1 \sigma \partial_3 \sigma) \\ \operatorname{tr}(\partial_1 \sigma \partial_2 \sigma) & \operatorname{tr}(\partial_2 \sigma^2) & \operatorname{tr}(\partial_2 \sigma \partial_3 \sigma) \\ \operatorname{tr}(\partial_1 \sigma \partial_3 \sigma) & \operatorname{tr}(\partial_2 \sigma \partial_3 \sigma) & \operatorname{tr}(\partial_3 \sigma^2) \end{pmatrix} v \right)^{\frac{1}{2}} =: \left(v^T Q v\right)^{\frac{1}{2}} .
\end{aligned}
$$

The matrix Q is symmetric and positive semidefinite, this allows for a simple visualization of Q by an ellipsoid. We use the eigenvectors v_1, v_2, v_3 of Q as the principal axes and the square roots of the eigenvalues as the corresponding radii. If we scale Q by $4\pi\epsilon^2/3$, then the radii are equal to $\sqrt{4\pi/3}\|\epsilon \nabla_{v_i} \sigma\|_F$ for $i = 1, 2, 3$. This is the average variation of all stress vectors σn for a displacement by ϵ in direction v_i, measured by $\|\nabla_{v_i} \sigma n\|$ as described above. The information shown by such an ellipsoid can be summarized as

$$
\left\{ \left(v, \sqrt{\frac{4\pi}{3}} \|\epsilon \nabla_v \sigma\|_F \right) \mid v \in \mathbb{S}^2 \right\} .
$$

The resulting ellipsoid is shown in Fig. 4 in purple, it shows the same gradient as in the previous Figs. 2 and 3. Additionally the stress tensor is shown as an ellipsoid in yellow, while an exponential mapping has been applied on the stress tensor in order to obtain positive eigenvalues. Of course we are losing a lot of information by showing the gradient in this way. However, the overall change of the stress vectors in a certain direction can be seen much easier than from the visualization in Sect. 4.1.1.

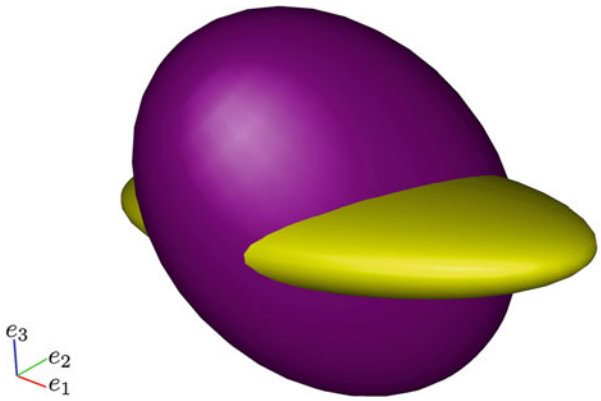


Fig. 4 The *purple ellipsoid* visualizes how much the stress tensor changes in certain directions. This is measured by the Frobenius norm of the directional derivative, which is equivalent to an averaged change of all stress vectors. The *yellow ellipsoid* shows the stress tensor itself

4.2.2 Directional Derivatives of Normal Stress

Now we derive the average/minimum/maximum of the directional derivative of the normal stress $n^\top \nabla_v \sigma n$ for all n while v is fixed or for all v while n is fixed. If we integrate $n^\top \nabla_v \sigma n$ for all $n \in \mathbb{S}^2$ to compute the average, then we obtain by Eq. (4)

$$\int_{\mathbb{S}^2} n^\top \nabla_v \sigma n \, dn = \frac{4\pi}{3} \operatorname{tr}(\nabla_v \sigma) \quad .$$

An average with respect to v makes no sense here, since $n^\top \nabla_v \sigma n$ is linear in v, thus the integral would be zero. Using Eq. (5) the extrema of $n^\top \nabla_v \sigma n$ for $n \in \mathbb{S}^2$ are given by

$$\max_{n \in \mathbb{S}^2} n^\top \nabla_v \sigma n = \lambda_{max}(\nabla_v \sigma) \quad , \quad \min_{n \in \mathbb{S}^2} n^\top \nabla_v \sigma n = \lambda_{min}(\nabla_v \sigma) \quad .$$

For the extrema with respect to $v \in \mathbb{S}^2$ we obtain

$$\max_{v \in \mathbb{S}^2} n^\top \nabla_v \sigma n = \|n^\top \nabla \sigma n\| \quad , \quad \min_{v \in \mathbb{S}^2} n^\top \nabla_v \sigma n = -\|n^\top \nabla \sigma n\| \quad .$$

due to the linearity of $n^\top \nabla_v \sigma n$ in v. Note that $n^\top \nabla \sigma n = \nabla(n^\top \sigma n) \in \mathbb{R}^3$ is the gradient of the normal stress with respect to n.

As an example we consider $n^\top \epsilon \nabla_v \sigma n$ for fixed n and consider the minimum and the maximum change of the normal stress in all directions $v \in \mathbb{S}^2$, i. e. the minimal or maximal normal stress in a linearized neighbourhood of radius ϵ. If we add this minimum or maximum to the actual normal stress $n^\top \sigma n$ we obtain the minimal/maximal normal stress of $\mathbf{t}(v, n)$ in a linearized neighborhood, which is

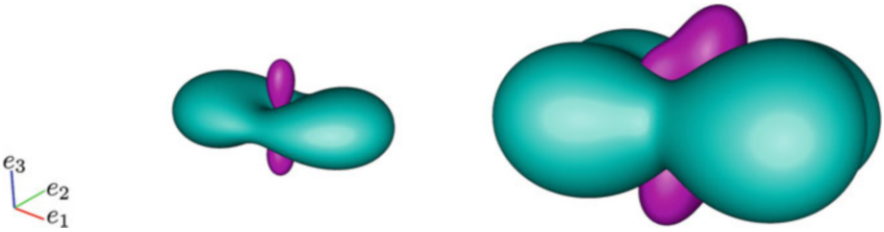


Fig. 5 The *left image* shows a Reynolds glyph, the distance from the origin depicts normal stress in this direction. *Turquoise* corresponds to positive values, *pink* corresponds to negative values. The *right image* shows the envelope of all Reynolds glyphs in a linearized neighborhood

given by

$$\max_{v\in\mathbb{S}^2} n^\top \mathbf{t}(v,n) = n^\top \sigma n + \max_{v\in\mathbb{S}^2} n^\top \epsilon \nabla_v \sigma n = n^\top \sigma n + \epsilon \|n^\top \nabla \sigma n\| \ ,$$

and for the minimum

$$\min_{v\in\mathbb{S}^2} n^\top \mathbf{t}(v,n) = n^\top \sigma n - \epsilon \|n^\top \nabla \sigma n\| \ .$$

We visualize the computed extrema for each n by a surface, while the distance of the surface from the origin is given by the respective extrema. If the normal stress $n^\top \sigma n$ is positive we show the maximum in turquoise, if it is negative we show the minimum in pink. Consequently, the visualized information is given by

$$\left\{ \left(n, \max_{v\in\mathbb{S}^2} |n^\top \mathbf{t}(v,n)| \right) \,\middle|\, n \in \mathbb{S}^2 \right\} \ .$$

Note that the surface with distance $n^\top \sigma n$ from the origin forms a Reynolds glyph, i. e. the glyph is obtained by scaling all normal vectors n by the normal stress of σ with respect to n, see for example [3]. Thus the shown surface can be considered as the envelope of all Reynold glyphs in a linearized neighborhood of radius ϵ, an example is shown in Fig. 5.

5 Application to Tensile Bars

As a simple practical example we consider simulated stress tensor fields for two different tensile bars. Tensile bars are used to check the behavior of a material under load. Here we consider two tensile bars with differently curved notches, which result in different stressing conditions. The stress becomes maximal at the notch and the load was adjusted in order to obtain similar stress tensors at this point. For the stronger curved notch a lower nominal tensile load is necessary to obtain a similar

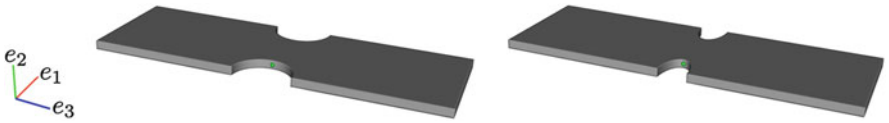


Fig. 6 Geometry of the two tensile bars, one with a less curved notch on the *left* and one with a stronger curved notch on the *right*. The *green dot* indicates the point where we analyze the gradient

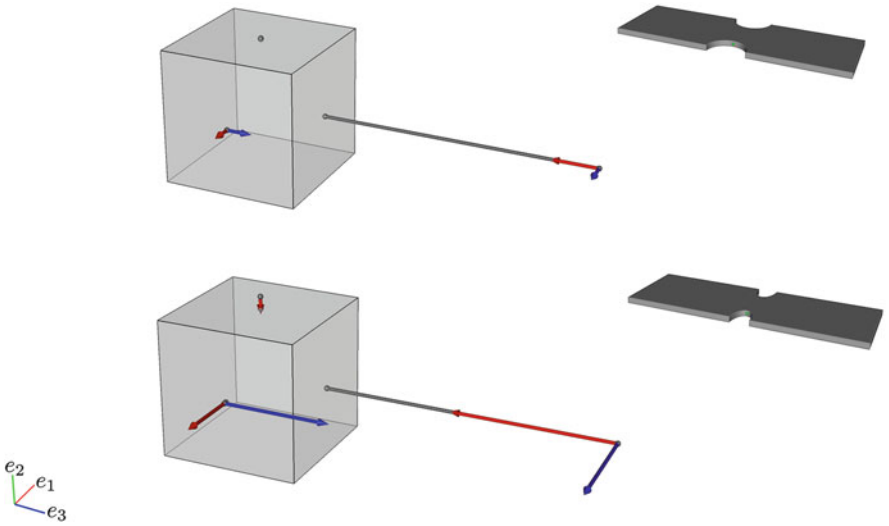


Fig. 7 Variation of the stress vectors for a displacement in the coordinate directions. The *upper image* corresponds to the less curved tensile bar, the *lower image* to the stronger curved tensile bar

stress tensor. The simulations have been performed with the commercial software package ABAQUS.

The geometry of the tensile bars is shown in Fig. 6, the green dots at the center of the notches indicate the points where we want to analyze the gradients. In Fig. 7 we show the stress vectors and their gradients as described in Sect. 4.1.1, the upper image belongs to the tensile bar with less curved notch, the lower image to the tensile bar with stronger curved notch. The gray tubes show that there is almost only tension in e_3-direction, this is the direction in which the bar is stretched. As intended, the stress is very similar for both tensile bars. The red arrows indicate the variation of the stress vectors for a displacement in e_1-direction, i. e. to the middle of the tensile bar. Here we can see for both tensile bars that the normal stress in e_3 direction is decreasing, while some additional normal stress in e_1-direction arises. This effect is more distinct with the stronger curved notch. Thus, while we obtain a similar stress for a lower load in case of the stronger curved notch, the stress decreases also much faster, suggesting a smaller region of high stress. When moving in e_3-direction we

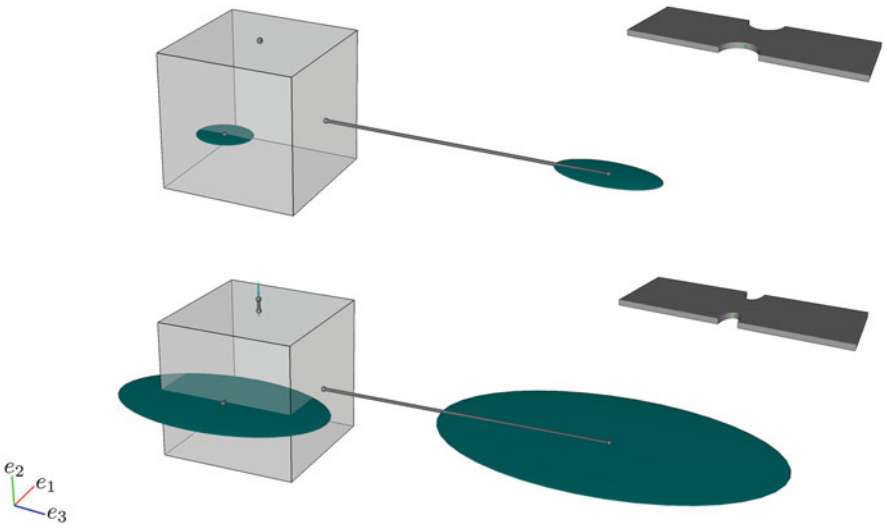


Fig. 8 Envelope of the stress vectors in a linearized neighborhood. The *upper image* corresponds to the less curved tensile bar, the *lower image* to the stronger curved tensile bar

Fig. 9 The *purple ellipsoid* shows the overall change of the stress tensor in all spatial directions, the *yellow ellipsoid* visualizes the stress tensor. The *left image* corresponds to the less curved tensile bar, the *right image* to the stronger curved tensile bar

obtain additional shear stresses for both tensile bars, as indicated by the blue arrows. This indicates a variation of the principal stress directions, which is caused by the curvature of the notches, consequently this effect is also stronger for the tensile bar with the stronger curved notch.

Figure 8 shows the envelopes of the stress vectors introduced in Sect. 4.1.2. Of course, this visualization does not reveal any new information, but it is easy to see which stress vectors occur in a linearized neighborhood, and that the variation for the second tensile bar is much stronger. Note that the considered point is on the boundary of the tensile bar, thus the linearized neighbourhood exceeds the actual data. Consequently, the ellipsoids show the envelope of the stress vectors for a linear extrapolation of the data.

In Fig. 9 the Frobenius norm of the directional derivative is visualized by an ellipsoid, alongside with an ellipsoid visualizing the stress tensor as suggested in Sect. 4.2. With this visualization it is easy to see that the stress tensor changes rapidly in x direction, while there is almost no change in y-direction. Again, it is easy to see that the stress tensor changes much faster for the tensile bar with the stronger curved notch.

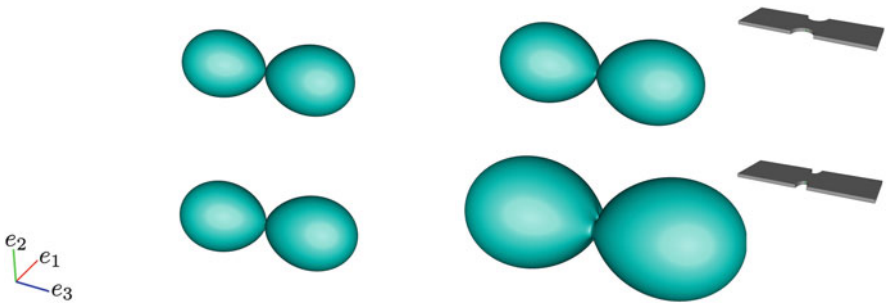


Fig. 10 From *left to right*, the images show the Reynolds glyph for the less curved tensile bar and its envelope, and the Reynolds glyph for the stronger curved tensile bar and its envelope

The Reynolds glyph and the envelope of the Reynolds glyph for a linearized neighborhood is shown in Fig. 10. These glyphs offer a overview over the absolute variation of the normal stresses, which is, of course, also much bigger for the tensile bar with stronger curved notch.

6 Conclusion and Future Work

In this chapter we have proposed several first ideas for the visualization of the gradient of the stress tensor. The basic idea was the visualization of the stress vectors and their variation in a linearized neighborhood. We proposed a glyph showing the complete information contained in the stress tensor and its gradient, which is based on commonly used depictions in engineering textbooks. Due to the complexity of this information, we proposed several simplifications based on mathematical considerations. We selected two of them and visualize them using glyphs. To make the proposed visualizations comprehensible, we used two tensile bars as simple examples.

Our proposed visualization showing the stress vectors and its partial derivatives contains all information on the stress tensor and the stress gradient. But it might need some familiarization for the user to interpret the shown information. Since the stress tensor and its gradient consist of 24 independent components, this might be similar for any glyph which tries to visualize the complete information of these tensors. Thus it seems less promising to focus on glyphs that provide all the information at the same time, instead we want to focus on interaction in the future. This includes comfortable ways to select the normal direction as well as the direction of differentiation. Together with the proposed simplifications of the gradient the user should be able to investigate different properties of the gradient in an interactive way. With such a tool we hope to come closer to our overarching objective: We want to find out which information of the stress gradient is relevant for our application, that is, how must the stress change such that it influences the stability limit of a technical component.

References

1. Borgo, R., Kehrer, J., Chung, D.H., Maguire, E., Laramee, R.S., Hauser, H., Ward, M., Chen, M.: Glyph-based visualization: foundations, design guidelines, techniques and applications. In: Eurographics State of the Art Reports, pp. 39–63 (2013)
2. Dick, C., Georgii, J., Burgkart, R., Westermann, R.: Stress tensor field visualization for implant planning in orthopedics. IEEE Trans. Vis. Comput. Graph. **15**(6), 1399–1406 (2009)
3. Hashash, Y., Yao, J.I., Wotring, D.C., et al.: Glyph and hyperstreamline representation of stress and strain tensors and material constitutive response. Int. J. Numer. Anal. Methods Geomech. **27**(7), 603–626 (2003)
4. Holzapfel, G.A.: Nonlinear Solid Mechanics, vol. 24. Wiley, Chichester (2000)
5. Jeremić, B., Scheuermann, G., Frey, J., Yang, Z., Hamann, B., Joy, K.I., Hagen, H.: Tensor visualizations in computational geomechanics. Int. J. Numer. Anal. Methods Geomech. **26**(10), 925–944 (2002)
6. Kindlmann, G.: Superquadric tensor glyphs. In: Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization, pp. 147–154. Eurographics Association (2004)
7. Kindlmann, G., Ennis, D.B., Whitaker, R.T., Westin, C.F.: Diffusion tensor analysis with invariant gradients and rotation tangents. IEEE Trans. Med. Imaging **26**(11), 1483–1499 (2007)
8. Kratz, A., Auer, C., Stommel, M., Hotz, I.: Visualization and analysis of second-order tensors: moving beyond the symmetric positive-definite case. Comput. Graph. Forum–State Art Rep. **32**(1), 49–74 (2013)
9. Kratz, A., Schoeneich, M., Zobel, V., Hotz, I., Burgeth, B., Scheuermann, G., Stommel, M.: Tensor visualization driven mechanical component design. In: IEEE Proceedings of PacificVis, 2014. IEEE (2014)
10. Kriz, R., Yaman, M., Harting, M., Ray, A.: Visualization of zeroth, second, fourth, higher order tensors, and invariance of tensor equations. Comput. Graph. **21**(6), 1–13 (2005)
11. Milne, I., Ritchie, R.O., Karihaloo, B.L.: Comprehensive Structural Integrity: Cyclic Loading and Fatigue, vol. 4. Elsevier, Amsterdam (2003)
12. Radaj, D.: Ermüdungsfestigkeit: Grundlagen für Leichtbau, Maschinen- und Stahlbau. Springer, Berlin (2013)
13. Schultz, T., Kindlmann, G.L.: Superquadric glyphs for symmetric second-order tensors. IEEE Trans. Vis. Comput. Graph. **16**(6), 1595–1604 (2010)
14. Zobel, V., Stommel, M., Scheuermann, G.: Feature-based tensor field visualization for fiber reinforced polymers. In: 2015 IEEE Scientific Visualization Conference (SciVis), pp. 49–56 (2015). doi:10.1109/SciVis.2015.7429491

Part II
Image Processing and Analysis

Geometries and Interpolations for Symmetric Positive Definite Matrices

Aasa Feragen and Andrea Fuster

Abstract In this survey we review classical and recently proposed Riemannian metrics and interpolation schemes on the space of symmetric positive definite (SPD) matrices. We perform simulations that illustrate the problem of tensor fattening not only in the usually avoided Frobenius metric, but also in other classical metrics on SPD matrices such as the Wasserstein metric, the affine invariant/Fisher Rao metric, and the log Euclidean metric. For comparison, we perform the same simulations on several recently proposed frameworks for SPD matrices that decompose tensors into shape and orientation. In light of the simulation results, we discuss the mathematical and qualitative properties of these new metrics in comparison with the classical ones. Finally, we explore the nonlinear variation of properties such as shape and scale throughout principal geodesics in different metrics, which affects the visualization of scale and shape variation in tensorial data. With the paper, we will release a software package with Matlab scripts for computing the interpolations and statistics used for the experiments in the paper (Code is available at https://sites.google.com/site/aasaferagen/home/software.)

1 Introduction

The space of symmetric positive definite (SPD) matrices, denoted $\mathrm{Sym}^+(n)$, is defined as

$$\mathrm{Sym}^+(n) = \left\{T \in \mathbb{R}^{n\times n} | T_{ij} = T_{ji}\ \forall\ i,j = 1,\ldots,n,\ x^T T x > 0\ \forall\ x \in \mathbb{R}^n \setminus 0\right\},$$

A. Feragen (✉)
Department of Computer Science, University of Copenhagen, Universitetsparken 5, 2100 Copenhagen, Denmark
e-mail: aasa@di.ku.dk

A. Fuster
Department of Mathematics and Computer Science, Eindhoven Technical University, 5600 Einchoven, MB, The Netherlands
e-mail: a.fuster@tue.nl

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_5

where the first property enforces symmetry and the second property enforces positive definiteness. An equivalent definition of positive definiteness is that all eigenvalues of T are strictly positive.

SPD matrices play an important role in many data science applications. They coincide with covariance matrices of multivariate normal distributions, and therefore appear both in information geometry [2] and through covariance descriptors in computer vision [36]. They also represent second order tensors which e.g. model diffusion in diffusion tensor imaging (DTI) [4]. These applications have led to a rich theory of statistics and geometry in *spaces of SPD matrices*, which has interest both from a practical and theoretical point of view. Throughout the paper, we will use the terms "SPD matrix" and "tensor" interchangingly.

In this paper we survey classical and modern geometries defined on the space $\mathrm{Sym}^+(3)$ of SPD 3×3 matrices. All of these geometries can be extended to $\mathrm{Sym}^+(n)$, most of them directly, but for some this is technically more challenging. We perform simulations that illustrate the properties of the different approaches, in particular concerning the preservation of information in tensor interpolation and statistics. Finally, we discuss properties of the different metrics regarding the visualization of statistical properties of datasets in $\mathrm{Sym}^+(3)$, in the context of recent work in visualization [40].

2 The Positive Definite Cone and the Frobenius Metric

It is easy to see that the space $\mathrm{Sym}^+(3)$ is a convex subset of the Euclidean space $\mathbb{R}^{3\times 3}$ of 3×3 matrices. Let $T_1, T_2 \in \mathrm{Sym}^+(3)$ and $t \in [0, 1]$; now $T_1 + t(T_2 - T_1) \in \mathrm{Sym}^+(3)$ because it is clearly still symmetric, and

$$x^T(T_1 + t(T_2 - T_1))x = \underbrace{(1-t)}_{\geq 0}\underbrace{x^T T_1 x}_{>0} + \underbrace{t}_{\geq 0}\underbrace{x^T T_2 x}_{>0} > 0 \quad \text{for all } x \in \mathbb{R}^3 \setminus 0,$$

where the inequality holds because only one of the factors $(1 - t)$ and t can be 0. More precisely, the elements of $\mathrm{Sym}^+(3)$ constitute the convex *positive definite cone* in $\mathbb{R}^{3\times 3}$, which is the interior of the more frequently used positive semidefinite cone defined by $x^T T x \geq 0$.

Since $\mathrm{Sym}^+(3)$ is convex, we can perform linear interpolation between elements of $\mathrm{Sym}^+(3)$ while remaining within the set. These straight lines are geodesics on $\mathrm{Sym}^+(3)$ with respect to the so-called *Frobenius* metric on $\mathrm{Sym}^+(3)$, which is just the inherited Euclidean metric from $\mathbb{R}^{3\times 3}$. In particular, distances with respect to the Frobenius metric are given by the ambient space Euclidean distances:

$$d_F(T_1, T_2) = \|T_1 - T_2\|_2 = \sqrt{\sum_{i=1}^{3}\sum_{j=1}^{3}((T_1)_{i,j} - (T_2)_{i,j})^2}.$$

The Frobenius geodesic $\gamma_F : [0, 1] \rightarrow \mathrm{Sym}^+(3)$ from T_1 to T_2 is trivially computed through Euclidean coordinate-wise interpolation:

$$\gamma_F(t) = T_1 + t(T_2 - T_1).$$

While the Frobenius metric is simple and efficient to work with, it comes with a significant cost when used for statistics on $\mathrm{Sym}^+(3)$. Frobenius interpolation between tensors in $\mathrm{Sym}^+(3)$ leads to a significant *swelling effect*, illustrated in Fig. 1. Here, two tensors are interpolated which have identical, ellipsoidal shape but which have 85° difference in orientation. We observe that the tensors in the middle of the geodesic are much rounder than the endpoint tensors. The swelling effect is problematic for a number of applications in DTI. For example, when tensor interpolation is used for upsampling, the swelling effect leads to smoothing in the upsampled tensor field. The problem becomes more serious when statistics in $\mathrm{Sym}^+(3)$ are used e.g. for voxel-based morphometry. The mean of two tensors is the midpoint of the geodesic connecting them, which carries less shape information than the original two tensors. The mean of multiple tensors will typically exhibit even stronger smoothing of information, leading to reduced predictive or discriminative power.

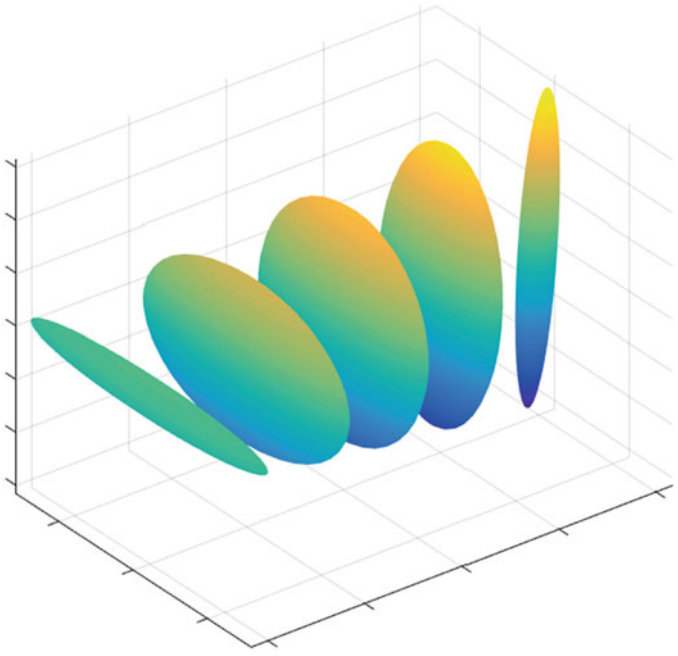


Fig. 1 Samples from a geodesic interpolation between two identical line-like tensors at an angle of 85° in the Frobenius metric. Note that the samples at the middle of the geodesic are very disc-like, thus exhibiting a very different shape from the two endpoints, and thus containing very little orientation information. This is called the swelling effect

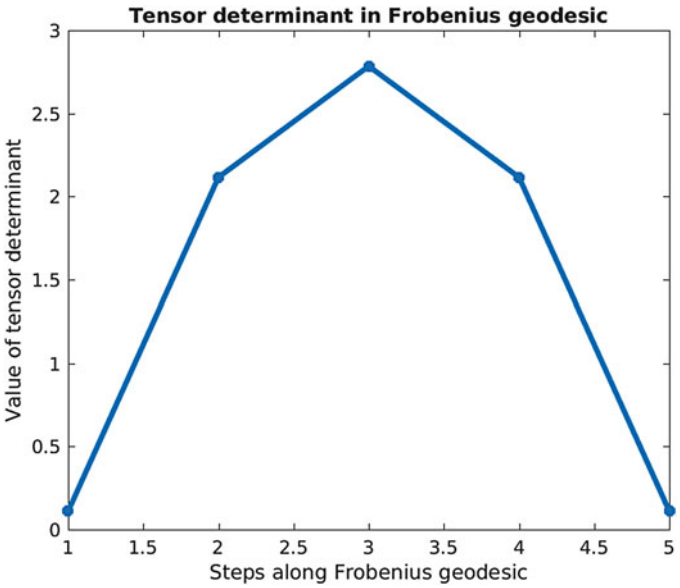


Fig. 2 Values of the tensor determinant along the geodesic illustrated in Fig. 1. The non-constant behavior of the determinant is what, in the literature, is referred to as the *swelling* effect of the Frobenius metric

The swelling effect is defined analytically as an increase in the determinant of the elements in $\mathrm{Sym}^+(3)$ as one interpolates two identically shaped but differently oriented ellipsoidal tensors. Figure 2 shows a plot of the determinant of each tensor depicted along the geodesic in Fig. 1. As seen from the plot, the determinant is non-constant, and this is directly related to the tensor swelling throughout the geodesic connecting the two identically shaped, but differently oriented tensors in Fig. 1.

In this paper, we will first survey classical Riemannian metrics on $\mathrm{Sym}^+(3)$ and illustrate that while most of these avoid the swelling effect, they still exhibit a strong and unwanted *fattening* effect, as is also remarked in [23]. Next, we proceed to reviewing and exploring some more recently proposed metrics and interpolation schemes that aim to avoid the fattening effect by decoupling tensor shape from tensor orientation. While these have modelling advantages, they do come at a price: losing the statistics that come with a well-defined and computationally efficient geometric framework.

2.1 Acknowledgement

This paper was largely motivated by discussions of whether nonlinear geometries on $\mathrm{Sym}^+(3)$ might improve statistics and visualization for populations of tensors at the Dagstuhl seminar "Multidisciplinary Approaches to Multivalued Data: Modeling, Visualization, Analysis" (16142). We return to remark on this question in Sect. 5.5.

3 Classical Riemannian Metrics on $\mathrm{Sym}^+(3)$

Riemannian metrics and other interpolation schemes for SPD matrices became an active area of research with the advent of diffusion tensor imaging (DTI), starting a quest for metrics that avoid the swelling effect while being computationally efficient. We start out by surveying the classics: The *Wasserstein* metric, the *affine-invariant* metric known in other contexts as the Fisher-Rao metric, and the *Log-Euclidean* metric.

3.1 The Wasserstein Metric

The *Wasserstein metric*, also known as the *earth mover's distance*, defines a general distance metric between arbitrary probability distributions on general metric spaces, which intuitively measures the amount of mass needed to transport one distribution into the other. Given a metric space (X, d) which is also a Radon space [3], and given $p \geq 1$, denote by $P_p(X)$ the set of probability measures μ on X such that

$$\int_X d^p(x, x_0) d\mu(x) < \infty.$$

The p^{th} Wasserstein distance between two probability measures $\mu, \eta \in P_p(X)$ is given by

$$W_p(\mu, \eta) = \left(\inf_{\gamma \in \Gamma(\mu,\eta)} \int_{X \times X} d^p(x, y) d\gamma(x, y) \right)^{\frac{1}{p}}, \quad (x, y) \in X \times X,$$

where $\Gamma(\mu, \eta)$ is the set of measures on $X \times X$ whose marginals are μ and η, respectively. We shall focus on the case $p = 2$ for multivariate normal distributions $\mathcal{N}(\mathbf{0}, \Sigma)$ centered at the origin in the metric space $X = \mathbb{R}^3$ [32]. Any such multivariate normal distribution is described entirely by its covariance matrix Σ, and the set of such covariance matrices Σ is exactly the space of SPD matrices $\mathrm{Sym}^+(3)$. For general metric spaces and distributions, the Wasserstein distance is difficult to compute, but for centered normal distributions in $\mathbb{R}^n$, it has an analytical expression. We therefore obtain a Riemannian metric on $\mathrm{Sym}^+(3)$ by representing any SPD matrix $T \in \mathrm{Sym}^+(3)$ as a multivariate normal distribution with zero mean and covariance $\Sigma = T$. This defines the Riemannian Wasserstein metric on $\mathrm{Sym}^+(3)$ as a pull-back from the corresponding metric on the space of normal distributions.

Given this Riemannian metric, we can compute geodesics and geodesic distances, where the geodesic distance $d_W(T_1, T_2)$ between tensors $T_1, T_2 \in \mathrm{Sym}^+(3)$ is the length of the (shortest) geodesic $\gamma_W : [0, 1] \to \mathrm{Sym}^+(3)$ joining T_1 and T_2. In the space of normal distributions, both geodesics and geodesic distances

have analytical solutions [32]. The Wasserstein geodesic for centered normal distributions $\gamma_W : [0, 1] \to \mathrm{Sym}^+(3)$ connecting $T_1, T_2 \in \mathrm{Sym}^+(3)$ is given by

$$\gamma_W(t) = \left((1-t)I + tT_2^{\frac{1}{2}}(T_2^{\frac{1}{2}}T_1T_2^{\frac{1}{2}})^{-\frac{1}{2}}T_2^{\frac{1}{2}}\right) T_1 \left((1-t)I + tT_2^{\frac{1}{2}}(T_2^{\frac{1}{2}}T_1T_2^{\frac{1}{2}})^{-\frac{1}{2}}T_2^{\frac{1}{2}}\right),$$

and the geodesic distance from T_1 to T_2 is given by

$$d_W(T_1, T_2) = W_2(\mathcal{N}(\mathbf{0}, T_1), \mathcal{N}(\mathbf{0}, T_2)) = \mathrm{tr}(T_1) + \mathrm{tr}(T_2) - 2\mathrm{tr}\sqrt{T_2^{\frac{1}{2}}T_1T_2^{\frac{1}{2}}}.$$

Note that for tensors, powers such as $T_i^{-\frac{1}{2}}$ are easily computed through their eigendecomposition. As with any Riemannian metric, this defines a symmetric distance which satisfies the triangle inequality and is nonzero between distinct tensors. In the more general case of normal distributions with positive *semi*definite covariance matrices, the Wasserstein distance induces a stratified space geometry, which is analyzed in detail in [32].

The Wasserstein distance has a nice intuitive interpretation, it is well-understood mathematically, and it is immensely popular. However, looking at Figs. 3 and 4, we observe that in practice, when interpolating two thin ellipsoidal tensors with identical shape but different orientation, the tensors go through significant swelling.

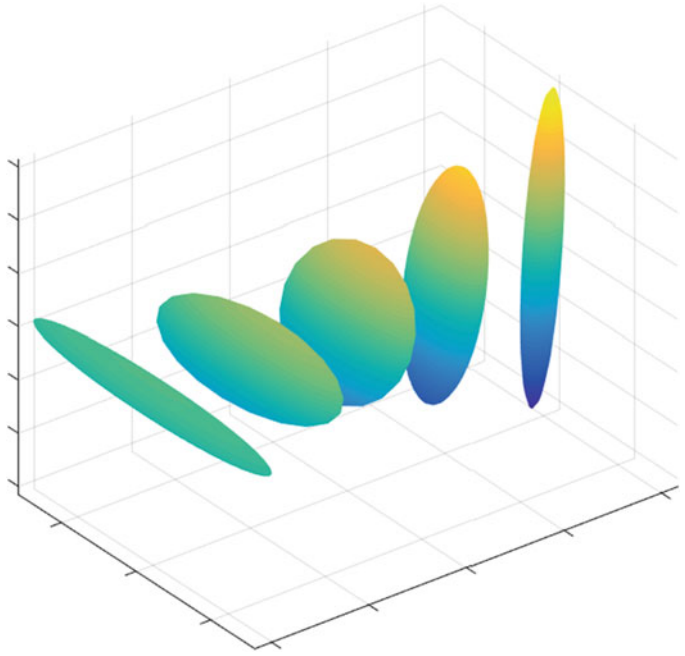


Fig. 3 We observe a swelling of the tensor as we move throughout the geodesic between equally shaped, ellipsoidal tensors

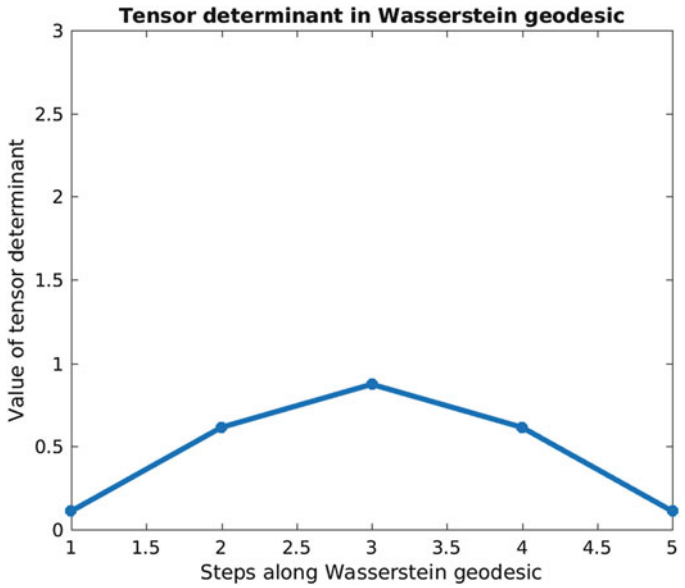


Fig. 4 The tensor determinant increases as we move through the geodesic in Fig. 3

3.2 The Affine-Invariant Metric

The classical affine-invariant metric [7, 14] was introduced to avoid the swelling effect on $\mathrm{Sym}^+(3)$. It is a Riemannian metric, and the geodesic between tensors $T_1, T_2 \in \mathrm{Sym}^+(3)$ is given analytically by

$$\gamma_{AI}(t) = T_1^{\frac{1}{2}} \exp\left(t \log(T_1^{-\frac{1}{2}} T_2 T_1^{-\frac{1}{2}})\right) T_1^{\frac{1}{2}}, \quad t \in [0, 1],$$

and the geodesic distance from T_1 to T_2 is

$$d_{AI}(T_1, T_2) = \|\log\left(T_1^{-\frac{1}{2}} T_2 T_1^{-\frac{1}{2}}\right)\|_F.$$

The affine-invariant metric is a special case of the Fisher-Rao metric [2] on probability distributions, restricted to zero-mean multivariate Gaussian distributions with covariance in $\mathrm{Sym}^+(3)$, as in the previous section.

In Fig. 5 we see an example of an affine invariant geodesic on the same example as above, and in Fig. 6 we track the determinant of the interpolated tensors throughout the geodesic. We see that there is no swelling effect, in the sense that the determinant remains constant throughout the geodesic. However, there is a noticeable *fattening* effect leading to rounder tensors in the middle of the geodesic than at the endpoints.

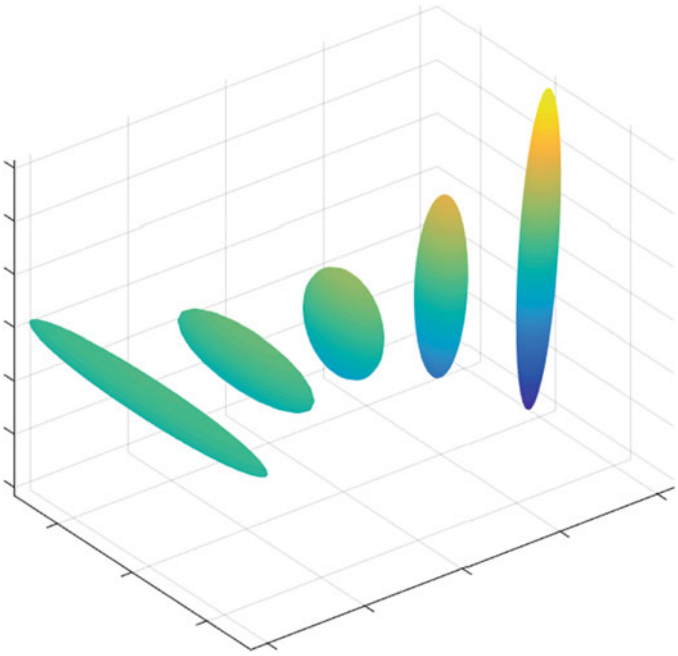


Fig. 5 Samples from a geodesic interpolation between two identical line-like tensors at an angle of 85° in the affine-invariant metric. The samples at the middle of the geodesic are less disc-like than in the Frobenius metric. Note, however, that they still have different (fatter) shape than the endpoint tensors

It is clear from the above equations that the affine-invariant metric is relatively computationally expensive, as it relies on a number of evaluations of matrix exponentials, logs, square roots and inverses, and for this reason, the more tractable *Log-Euclidean* metric was proposed in 2006 [4].

3.3 The Log-Euclidean Metric

The Log-Euclidean metric on $\mathrm{Sym}^+(3)$ [4] utilizes the observation that the matrix exponential defines the one-to-one mapping $\exp\colon \mathrm{Sym}(n) \to \mathrm{Sym}^+(3)$ from the vector space of symmetric 3×3 matrices into the manifold $\mathrm{Sym}^+(3)$ of SPD 3×3 matrices, whose inverse is the matrix logarithm (which is well defined on $\mathrm{Sym}^+(3)$). This means that the differential structure of $\mathrm{Sym}(n) \cong \mathbb{R}^{(n^2+n)/2}$ can be pulled back to $\mathrm{Sym}^+(3)$ via the matrix logarithm, where $\cong$ denotes the natural isometry between the coordinates in $\mathbb{R}^{(n^2+n)/2}$ and the upper triangular entries of a symmetric matrix. Equivalently, the logarithm provides a nonlinear transformation of $\mathrm{Sym}^+(3)$ into the Euclidean space $\mathbb{R}^{(n^2+n)/2}$, where analysis can take place. In this way, the Log-

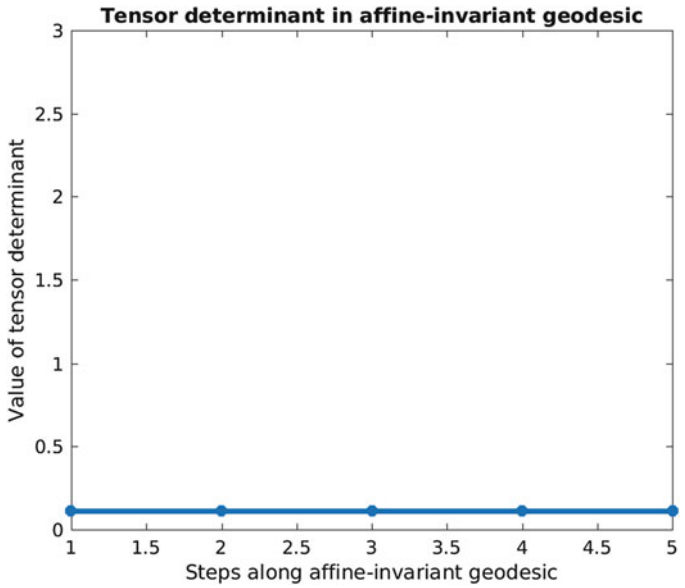


Fig. 6 The tensor determinant is constant along the affine invariant geodesic illustrated in Fig. 5. The affine-invariant metric thus avoids the swelling effect has a fattening effect throughout the geodesic

Euclidean metric defines a *Euclidean* structure on $\mathrm{Sym}^+(3)$. In this geometry, the distance between two tensors T_1 and T_2 is given by

$$d_{LE}(T_1, T_2) = \| \log(T_1) - \log(T_2) \|_F.$$

More importantly, the geodesic between T_1 and T_2 under the Log-Euclidean metric is given by the straight line between $\log(T_1)$ and $\log(T_2)$. However, this straight line resides in $\mathbb{R}^{(n^2+n)/2}$, not in $\mathrm{Sym}^+(3)$. It can be pulled back to $\mathrm{Sym}^+(3)$ using the matrix exponential, giving rise to the following analytical expression for the geodesic $\gamma_{LE}\colon [0, 1] \to \mathrm{Sym}^+(3)$:

$$\gamma_{LE}(t) = \exp\Big(t \log(T_1) + (1 - t) \log(T_2) \Big), \quad t \in [0, 1].$$

The Log-Euclidean distances and geodesics thus avoid many of the involved matrix power computations needed in the affine-invariant framework, although they still require matrix exponentials and logarithms. Moreover, empirically, the Log-Euclidean geodesics are often found to be extremely similar to the affine invariant ones. This is illustrated in our geodesic interpolation example, shown in Fig. 7. As we can see in Fig. 8, the Log-Euclidean geodesic keeps the tensor determinant constant when interpolating identically shaped tensors, and it therefore does not

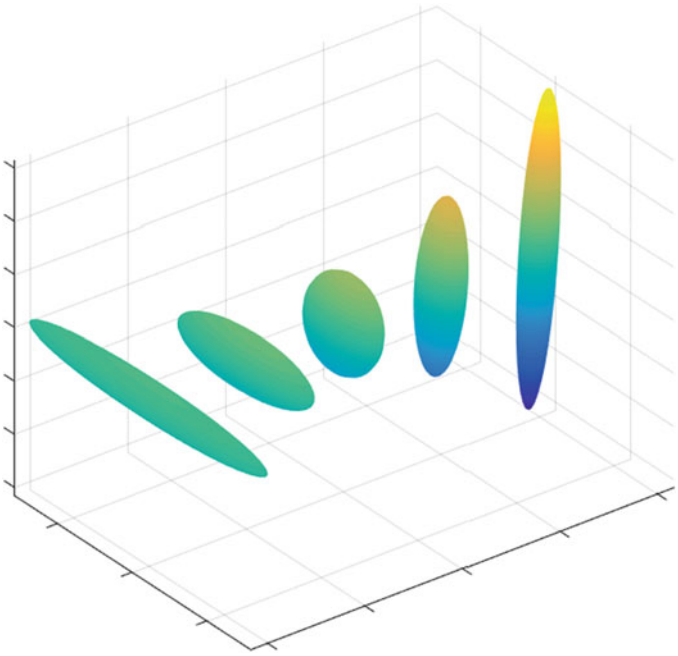


Fig. 7 Samples from a geodesic interpolation between two identical line-like tensors at an angle of 85° in the Log-Euclidean metric

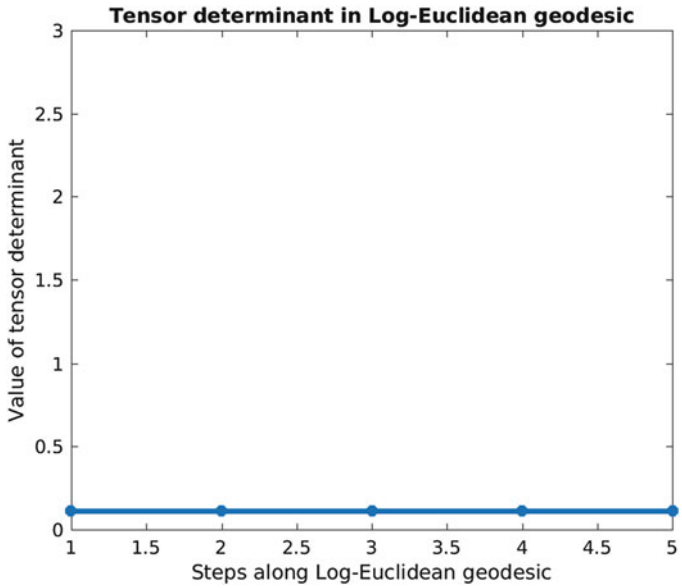


Fig. 8 The tensor determinant is constant along the Log-Euclidean geodesic illustrated in Fig. 7. It therefore also avoids the swelling effect, but exhibits a fattening effect as seen in Fig. 7

exhibit the swelling effect—but we see in Fig. 7 that it does, just like the affine-invariant metric, still lead to a fattening effect in the tensors at the middle of the geodesic. Note also the qualitative similarity between the affine-invariant geodesics in Fig. 5 and the Log-Euclidean geodesics in Fig. 7.

4 Avoiding the Swelling, But Not the Fattening, Effect

Figures 5 and 7 illustrate that while the affine-invariant and Log-Euclidean metrics do keep the tensor determinant fixed and therefore avoid the previously defined swelling effect, they do not preserve tensor shape when interpolating between identically shaped, but differently oriented tensors. Let us quantify this effect further by considering four different shape indices: The fractional anisotropy (FA) of a tensor T is given by [6]

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

where $\lambda_1 \geq \lambda_2 \geq \lambda_3$ are the eigenvalues of T. In the same notation, the three Westin tensor shape indices [38] are given by

$$c_l = \frac{\lambda_1 - \lambda_2}{\lambda_1}, \quad c_p = \frac{\lambda_2 - \lambda_3}{\lambda_1}, \quad c_s = \frac{\lambda_3}{\lambda_1},$$

and quantify the tensor's resemblance to a line, a plane or a sphere, respectively. For the same two endpoint tensors used previously, we plot the four different shape measures throughout the geodesic for the affine invariant and Log-Euclidean metrics in Fig. 9.

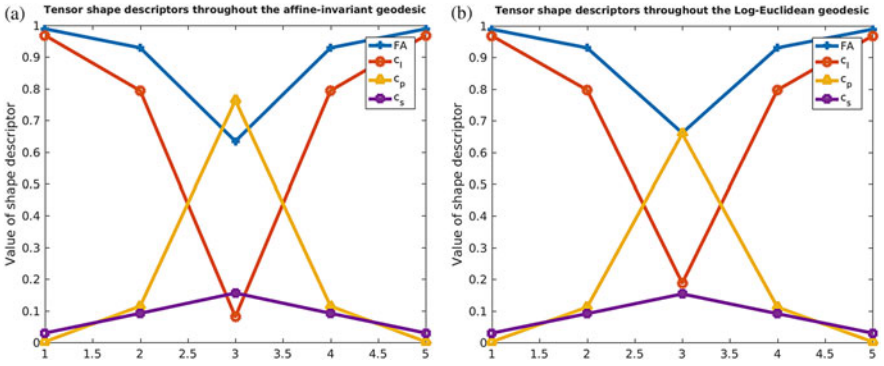


Fig. 9 We quantitatively confirm the fattening effect in the affine invariant and Log-Euclidean metrics: The shape is variable and the tensors become fatter towards the middle of the geodesic. (**a**) Affine-invariant. (**b**) Log-Euclidean

The plots in Fig. 9 clearly confirm the fattening effect: The tensors are less line-like and more plane- and sphere-like towards the middle of the geodesic, despite the fact that the endpoint tensors have identical ellipsoidal shape. To avoid this effect, several approaches have appeared that aim to decouple tensor shape and tensor rotation.

5 Decoupling Shape and Rotation

The eigenvalue decomposition of an SPD matrix

$$T = Q\Lambda Q^{-1} \tag{1}$$

into a rotation matrix $Q \in SO(3)$ given by column-wise eigenvectors and a diagonal matrix Λ containing the eigenvalues, provides a natural way of splitting T into its shape- and orientation properties. This presents an attractive opportunity to avoid the fattening effect, and it is also useful for interpretation. An abundance of approaches to decouple shape and rotation have appeared [9, 16, 18, 23, 24, 30, 37], seemingly independent of each other, and we shall review some of the most important ones below, in order of increasing complexity.

5.1 The Shape-and-Orientation Rotation Metric

In the *shape-and-orientation* metric [37] the tensor $T = Q\Lambda Q^{-1}$ is considered to reside on the Riemannian product manifold $SO(3) \times \mathbb{R}^3_+$, where $SO(3)$ is given the angular geodesic metric and $\mathbb{R}^3_+$ is given the bi-invariant metric. A geodesic γ between two SPD matrices T_1 and T_2 in this metric is a product geodesic $\gamma = \gamma_r \times \gamma_s$, where γ_r is a geodesic between Q_1 and Q_2 in $SO(3)$, and γ_s is a geodesic between Λ_1 and Λ_2, where $T_1 = Q_1\Lambda_1 Q_1^{-1}$ and $T_2 = Q_2\Lambda_2 Q_2^{-1}$ as in (1).

Such geodesics are given analytically by the formulas

$$\gamma_r(t) = Q_1 \exp(t \cdot \log_{SO(3)}(Q_1^{-1}Q_2)),$$
$$\gamma_s(t) = \Lambda_1 \exp(t \cdot \log(\Lambda_1^{-1}\Lambda_2)),$$

where $\log_{SO(3)}$ denotes the log map on $SO(3)$, and the corresponding geodesic $\Gamma\colon [0,1] \to \mathrm{Sym}^+(3)$ from T_1 to T_2 is given by

$$\Gamma(t) = \gamma_r(t) \cdot \gamma_s(t) \cdot \gamma_r(t)^{-1}.$$

An example of a shape-and-orientation geodesic is shown in Figs. 10 and 11, and the determinant and shape indices are shown in Figs. 12 and 13. The corresponding

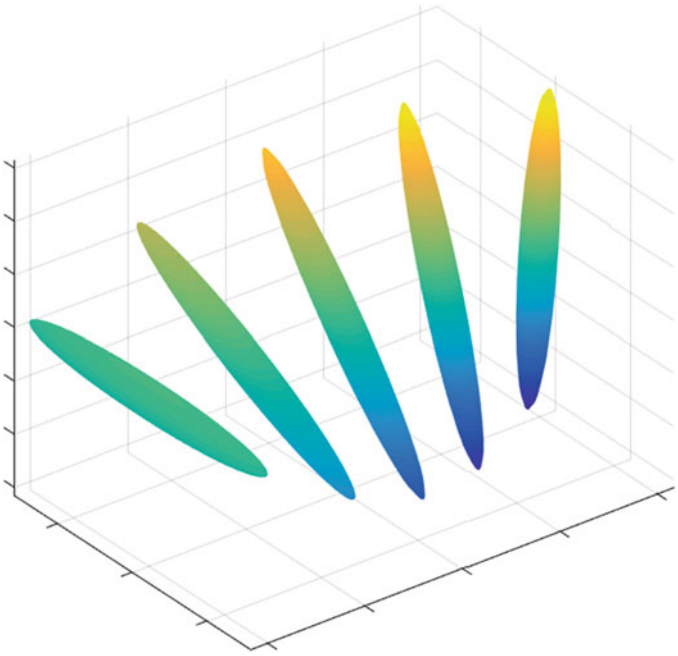


Fig. 10 Samples from a geodesic interpolation between two identical line-like tensors at an angle of 85° in the shape-and-orientation metric

geodesic distance between two tensors T_1 and T_2 is given by

$$d^2_{SAO}\left((Q_1, \Lambda_1), (Q_2, \Lambda_2)\right) = d^2_{SO(3)}(Q_1, Q_2) + d^2_{\mathbb{R}^3_+}(\Lambda_1, \Lambda_2)$$

where

$$d_{SO(3)}(Q_1, Q_2) = \frac{1}{\sqrt{2}} \| \log_{SO(3)}(Q_2 Q_1^T) \|_F, \quad d_{\mathbb{R}^3_+}(\Lambda_1, \Lambda_2) = \| \log(\Lambda_2 \Lambda_1^{-1}) \|_F.$$

As we see from the simulations in Figs. 10 and 11, where we compute the geodesic between identically shaped but differently oriented tensors, the tensor shape is kept constant throughout the deformation—as intended. This is confirmed by the constant determinant and shape indices shown in Figs. 12 and 13. Moreover, the construction is fast and apparently easy to work with as all the formulas are analytical, and the metric is intuitive and preserves tensor shape properties well. However, the construction ignores a fundamental problem: There exist multiple decompositions $T_i = Q_i \Lambda_i Q_i^{-1}$ of the endpoint tensors T_i, $i = 1, 2$, and different choices of Q_i and Λ_i generate different interpolations $\Gamma : [0, 1] \to \mathrm{Sym}^+(3)$. We see this effect very clearly in the example geodesic in Figs. 10 and 11, where the path chosen turns 95° and is—while locally a geodesic—*not* the shortest path between the two tensors (the shortest path turns 85°).

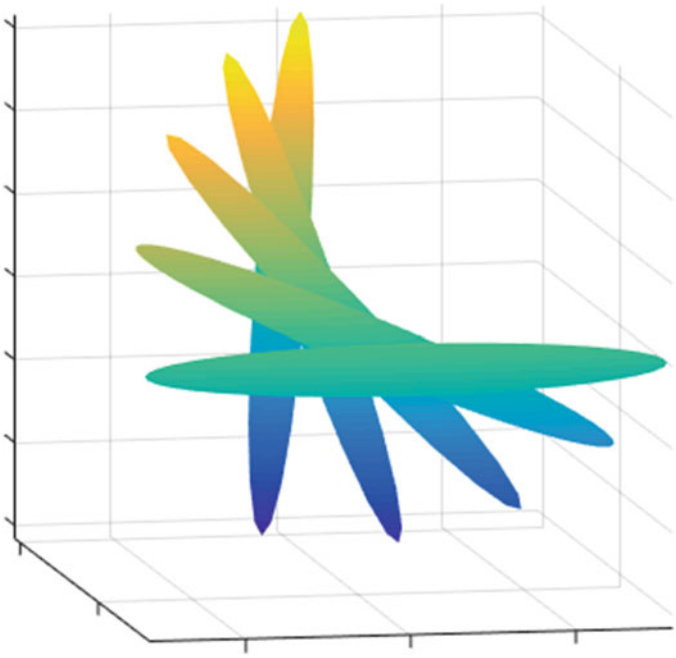


Fig. 11 The same geodesic interpolation as in Fig. 10, from a different viewpoint. It is clear that the interpolation is not the shortest possible 85° interpolation, but a longer 95° interpolation

The underlying problem is that while any point on $SO(3) \times \mathbb{R}^3_+$ does, indeed, correspond to a tensor, the map $F: SO(3) \times \mathbb{R}^3_+ \to Sym^+(3)$ defined by $F(Q, \Lambda) = Q\Lambda Q^{-1}$ is *not* injective, because the eigenvalue decomposition is not unique. The map F is the map that takes an eigenvalue decomposition to its corresponding tensor.

There are several ways in which the eigenvalue decomposition of a tensor, as in Eq. (1), is not unique. First, given any eigenvector e_i of a tensor T, its antipode $-e_i$ is also an eigenvector. The orientation part Q of the tensor decomposition consists of eigenvectors of T, but not all combinations of eigenvectors lead to a matrix $Q \in SO(3)$. In particular, multiplying an eigenvector by -1 gives another eigenvector, but flips the sign of Q, and only those eigenvector matrices that give $\det(Q) = 1$ actually reside in $SO(3)$. Moreover, even those eigenvector sets that do define a $Q \in SO(3)$, are not unique. This is what happens in Figs. 10 and 11. Another source of non-uniqueness is the order of the eigenvalues and eigenvectors in the decomposition. This issue is usually avoided in practice by requiring $\lambda_1 \geq \lambda_2 \geq \lambda_3$ in $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)$. However, this is also problematic, in particular when two eigenvalues approach each other. The problem with multiple representations of the same tensor becomes particularly complicated when the endpoint tensors have two or more identical eigenvalues [18]. To avoid the resulting technical complications, our experiments and code will assume that endpoint tensors have three distinct eigenvalues.

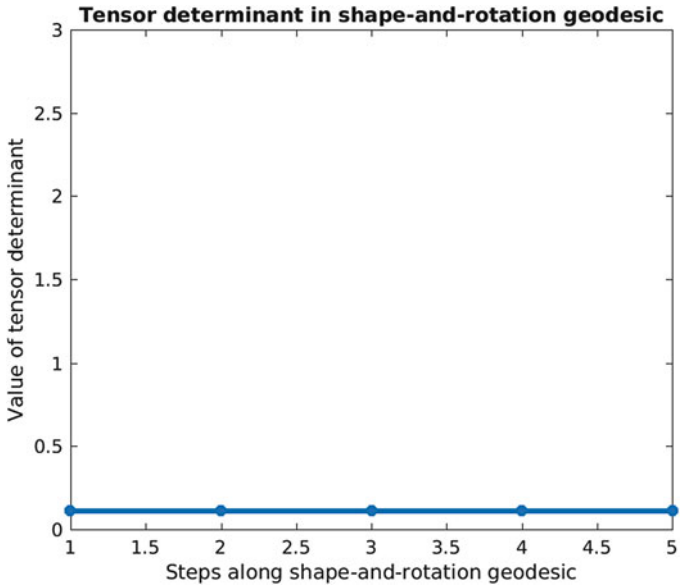


Fig. 12 The tensor determinant is constant along the shape-and-orientation geodesic illustrated in Fig. 10

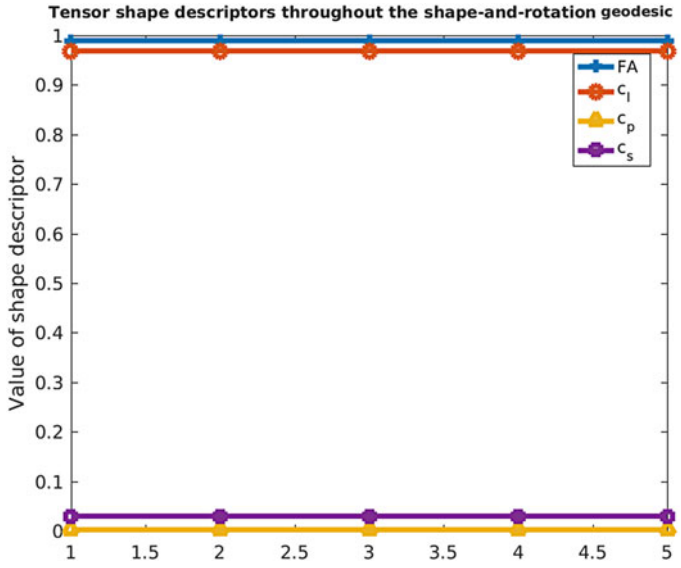


Fig. 13 The FA and Westin shape descriptors are also constant along the shape-and-orientation geodesic illustrated in Fig. 10

5.2 Scaling-Rotation Curves

While the shape-and-orientation metric as defined in [37] is a very simple way of defining a metric using eigenvalue decompositions, it is not the first appearance of the manifold $SO(3) \times \mathbb{R}^3_+$ in an attempt to generate a geometric framework for $\mathrm{Sym}^+(3)$. In a series of papers [18, 23, 30], A. Schwartzman and collaborators define *scaling-rotation* curves and a corresponding geometry, as follows.

A scaling-rotation curve [30] between $T_1, T_2 \in \mathrm{Sym}^+(3)$ is a geodesic in $SO(3) \times \mathbb{R}^3_+$ between representatives (Q_1, Λ_1) and (Q_2, Λ_2) of T_1 and T_2, where the metrics on $SO(3)$ and $\mathbb{R}^3_+$ are the same bi-invariant metrics as in Sect. 5.1. Scaling-rotation curves suffer from the same non-uniqueness problems as the shape-and-orientation metric from Sect. 5.1. A more recent series of papers [18, 23] attempt to handle this by factoring out multiple representations of the same tensor as follows. Given the mapping

$$F: SO(3) \times \mathbb{R}^3_+ \to \mathrm{Sym}^+(3), \quad (Q, \Lambda) \mapsto Q\Lambda Q^{-1}, \tag{2}$$

one can define an equivalence relation $\sim$ on $SO(3) \times \mathbb{R}^3_+$ by setting $(Q, \Lambda) \sim (Q'\Lambda')$ whenever $F(Q, \Lambda) = F(Q'\Lambda')$. This gives rise to a *quotient space*

$$(SO(3) \times \mathbb{R}^3_+)/\sim = (SO(3) \times \mathbb{R}^3_+)/F, \tag{3}$$

whose elements are equivalence classes, denoted $\overline{(Q, \Lambda)} = \{(Q', \Lambda') \in SO(3) \times \mathbb{R}^3_+ : F(Q, \Lambda) = F(Q', \Lambda')\} = F^{-1}(Q\Lambda Q^{-1})$. In the quotient space, each tensor is represented exactly once.

The quotient space $(SO(3) \times \mathbb{R}^3_+)/F$ can be identified with $\mathrm{Sym}^+(3)$, since the map F descends to a $1-1$ mapping $\bar{F}: (SO(3) \times \mathbb{R}^3_+)/F \to \mathrm{Sym}^+(3)$. This quotient space is not a smooth manifold, but it is a *stratified space*, meaning that it is a union of smooth manifolds which are adjacent to each other in a "well-behaved" way (see [28] for details on stratified spaces). The *strata*, or manifold components, are given by elements of $\mathrm{Sym}^+(3)$ with a fixed number of eigenvalue degeneracies, meaning the top stratum consists of tensors with three distinct eigenvalues, the next stratum consists of elements with two identical eigenvalues, etc.

Groisser et al. [18] note that tensors in $\mathrm{Sym}^+(3)$, represented as equivalence classes $F^{-1}(T) = \overline{(Q, \Lambda)}$ in $SO(3) \times \mathbb{R}^3_+$, can be interpolated by *minimal* scaling-rotation curves between equivalence classes $\overline{(Q, \Lambda)}, \overline{(Q', \Lambda')}$. This gives rise to the *scaling-rotation distance*

$$\begin{aligned} &d_{\mathscr{SR}}(T_1, T_2) \\ &= d_{SO(3)\times\mathbb{R}^3_+}\left(\overline{(Q_1, \Lambda_1)}, \overline{(Q_2, \Lambda_2)}\right) \\ &= \min\{d_{SO(3)\times\mathbb{R}^3_+}((Q_1, \Lambda_1), (Q_2, \Lambda_2)) \,|\, T_1 = Q_1\Lambda_1 Q_1^{-1}, T_2 = Q_2\Lambda_2 Q_2^{-1}\}. \end{aligned} \tag{4}$$

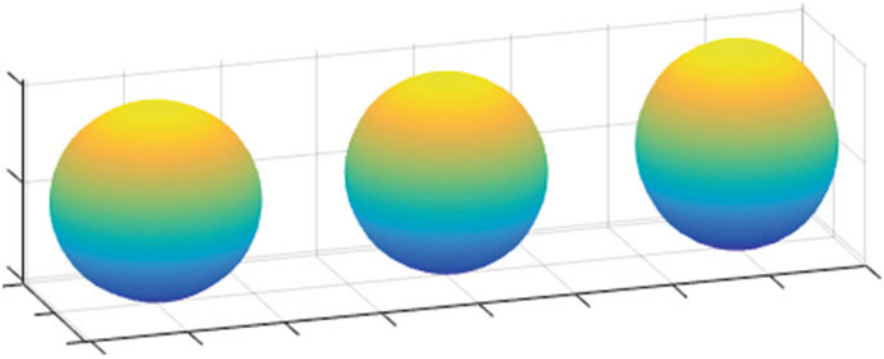


Fig. 14 For sufficiently spherical tensors, the scaling-rotation distance does not satisfy the triangle inequality. The left- and rightmost tensors have identical slightly ellipsoidal shape, but are rotated at an angle of 90°. The tensor in the middle is spherical

5.2.1 The Scaling-Rotation Distance is Not a Metric

As remarked in [18, 23], the scaling-rotation distance is *not* a metric (see Fig. 14), and in particular it is *not* the quotient (distance) metric on $(SO(3) \times \mathbb{R}^3_+)/F$. The quotient metric [8, p. 65] defined on a quotient

$$X/\sim = \{\bar{x} | x \in X, \bar{x} = \bar{y} \text{ if } x \sim y\}$$

is given by

$$\bar{d}(\bar{x}, \bar{y}) = \inf \left\{ \sum_{i=1}^{n-1} d(\bar{z}_i, \bar{z}_{i+1}) | z_1, \ldots, z_n \in X, z_1 \in \bar{x}, z_n \in \bar{y} \right\}.$$

Here, the equivalence classes $\bar{x}, \bar{y}$ are viewed as subsets of X and $d(\bar{x}, \bar{y}) = \inf\{d(x, y) | x \in \bar{x}, y \in \bar{y}\}$ is the set distance between subsets of X. Meanwhile, the scaling-rotation distance is a special case of

$$\tilde{d}(\bar{x}, \bar{y}) = \inf\{d(x, y) | x \in \bar{x}, y \in \bar{y}\}.$$

The quotient metric is the minimal total cost of hopping from $\bar{x}$ to $\bar{y}$ with stopovers at a finite set of equivalence classes $\bar{z}_i$, while the scaling-rotation distance does not allow stopovers. The lack of stopovers causes the scaling-rotation distance to *not* satisfy the triangle inequality, which is why it does not satisfy the criteria for being a metric.

This is easy to see via an example. Consider the two tensors

$$T_1 = \begin{pmatrix} 1+\epsilon & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad T_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1+\epsilon & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

which have identical shape and have orthogonal principal directions. The scaling-rotation distance between these two tensors is independent of ϵ, because the shape component is 0 and the rotation component is fixed, given by a 90° rotation, as long as $\epsilon > 0$.

However, $\lim_{\epsilon \to 0} d(T_i, I) = 0$, for the identity matrix I. These geodesics consist of a very small shape change with no rotational component. This means that for sufficiently small ϵ, we have $d(T_1, I) + d(I, T_2) < d(T_1, T_2)$—which violates the triangle inequality.

Nevertheless, the scaling-rotation distance *does* define "minimal" interpolations that solve some of the problems of the shape-and-orientation metric from [30, 37]. The quotient metric on $(SO(3) \times \mathbb{R}^3_+)/F$ remains unexplored to the best of our knowledge. This is likely due to the computational complexity and non-Riemannian structure of the space.

5.3 Linear Invariant Tensor Interpolation

Based on a set of tensor invariants proposed by Ennis et al. [11], Kindlmann et al. [24] proposed the *geodesic loxodromes* framework in order to avoid fattening when interpolating tensors. A *tensor invariant* is a scalar which depends only on the shape of the tensor—or mathematically speaking, on its eigenvalues. A classical example is the fractional anisotropy (FA). A geodesic loxodrome between two tensors T_1 and T_2 is the shortest path from T_1 to T_2 for which certain tensor shape parameters (either K_1-K_3 or R_1-R_3, as reviewed below) are linearly interpolated. In the original geodesic loxodromes framework, the interpolations were found by optimization, which is inexact and sometimes computationally expensive.

Gahm et al. [16] utilize the following combination of R- and K-invariants from [24], which allows an analytical reconstruction of the eigenvalues along the interpolated path[1]:

$$K_1(T) = \mathrm{tr}(T); \qquad R_2(T) = FA(T) = \sqrt{\frac{3}{2}} \frac{\|\tilde{T}\|_F}{\|T\|_F}; \qquad R_3(T) = 3\sqrt{6} \det\left(\frac{\tilde{T}}{\|\tilde{T}\|_F}\right), \tag{5}$$

where $\|T\|_F$ is the Frobenius norm of T, and

$$\tilde{T} = T - \frac{\mathrm{tr}(T)}{3} I_3 \tag{6}$$

is the anisotropic part of T, the so-called deviatoric tensor. Note that R_2 is the tensor FA used previously, while K_1 measures tensor scale. Now, the path through SPD

[1] Note that there are some typos in the definitions of R_2 and R_3 in [16].

matrices in $\mathrm{Sym}^+(3)$ with eigenvalues $\lambda_1, \lambda_2, \lambda_3$ given by

$$\lambda_i(t) = \frac{1}{3}\left(K_1(t)\right) + \frac{2K_1R_2(t)}{3\sqrt{3-2(R_2(t))^2}}\cos\left(\frac{\arccos(R_3(t)) + P_i}{3}\right), \tag{7}$$

where $P_i = 0, -2\pi, 2\pi$ for $i = 1, 2, 3$, results in a linear interpolation

$$\begin{aligned} K_1(t) &= (1-t)K_1(T_1) + tK_1(T_2), \\ R_2(t) &= (1-t)R_2(T_1) + tR_2(T_2), \\ R_3(t) &= (1-t)R_3(T_1) + tR_3(T_2), \end{aligned} \tag{8}$$

in the corresponding three tensor invariants. This tells us how to interpolate the shape of the tensors while linearly, and in particular, monotonically, interpolating the corresponding tensor invariants.

However, we do not yet know how to interpolate the orientation of the tensors. In [16] this is handled by using the orientation component of a Frobenius geodesic. Let $\gamma_F(t)$ be the tensor at time t in the Frobenius geodesic $\gamma_F\colon [0,1] \to \mathrm{Sym}^+(3)$ from T_1 to T_2, and let

$$Q(t)\tilde{\Lambda}(t)Q(t)^{-1} = \gamma_F(t) \tag{9}$$

be an eigenvalue decomposition of the time t tensor $\gamma_F(t)$. Let the diagonal matrix

$$\Lambda(t) = \begin{pmatrix} \lambda_1(t) & 0 & 0 \\ 0 & \lambda_2(t) & 0 \\ 0 & 0 & \lambda_3(t) \end{pmatrix}, \tag{10}$$

consist of the eigenvalues obtained in Eq. (7). This gives the so-called *linear invariant tensor* (LIT) interpolation $\gamma_{LIT}\colon [0,1] \to \mathrm{Sym}^+(3)$ from T_1 to T_2, defined as

$$\gamma_{LIT}(t) = Q(t)\Lambda(t)Q(t)^{-1}. \tag{11}$$

That is, the orientation component of the interpolation from T_1 to T_2 coincides with that of a Frobenius interpolation from T_1 to T_2, and the shape interpolation component is the one which linearly interpolates K_1, R_2 and R_3, given in Eq. (7).

We see that, at least in our running example, the fattening effect is resolved using the LIT interpolation. Tensor invariant interpolation methods have the advantage that shape is interpolated monotonically, and even linearly, with respect to the prespecified shape invariants. For the combination of invariants used in γ_{LIT} [16], there is an analytical solution, making the interpolation scheme computationally feasible. These are attractive properties.

On the negative side, the particular choice of angular interpolation is not ideal when the difference in orientation is large—this is clear from Fig. 15. This could be resolved by choosing the angular component differently, e.g. from one of the

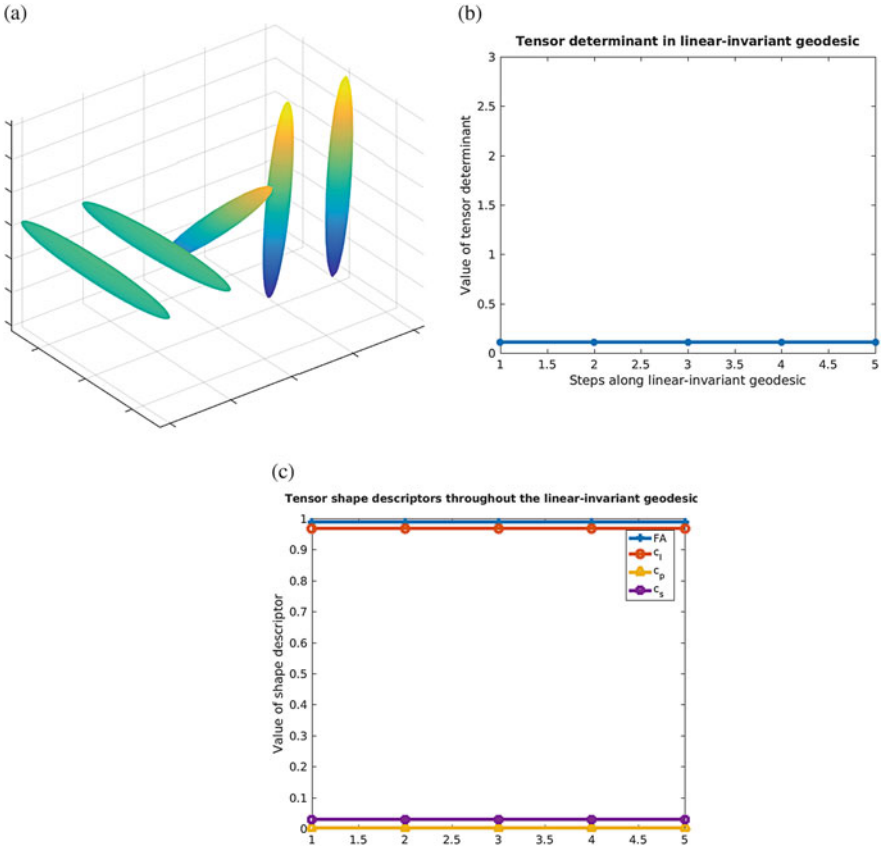


Fig. 15 The LIT interpolation between the same two thin, ellipsoidal tensors as before. We see that the determinant and shape of the tensor is unchanged throughout the geodesic, but we also see that the orientation is not interpolated at a constant rate. To see the reason for this effect, we refer back to the Frobenius interpolation in Fig. 1, where the tensor orientation also does not change at a constant rate. (**a**) The LIT interpolation. (**b**) Tensor determinant. (**c**) Shape descriptors

alternative tensor metrics. Moreover, to the best of our knowledge, the interpolations do not correspond to geodesics in a given geometric space. This makes it unclear to what extent the derived interpolations can be used in a geometric statistics framework to obtain Fréchet means, principal components, regression etc.

5.4 Further Simulations

In the above, we have used as a running example an interpolation between two identically shaped ellipsoidal tensors at an 85° angle of each other. Figures 16 and 17 illustrate additional simulations, illustrating how the different metrics and

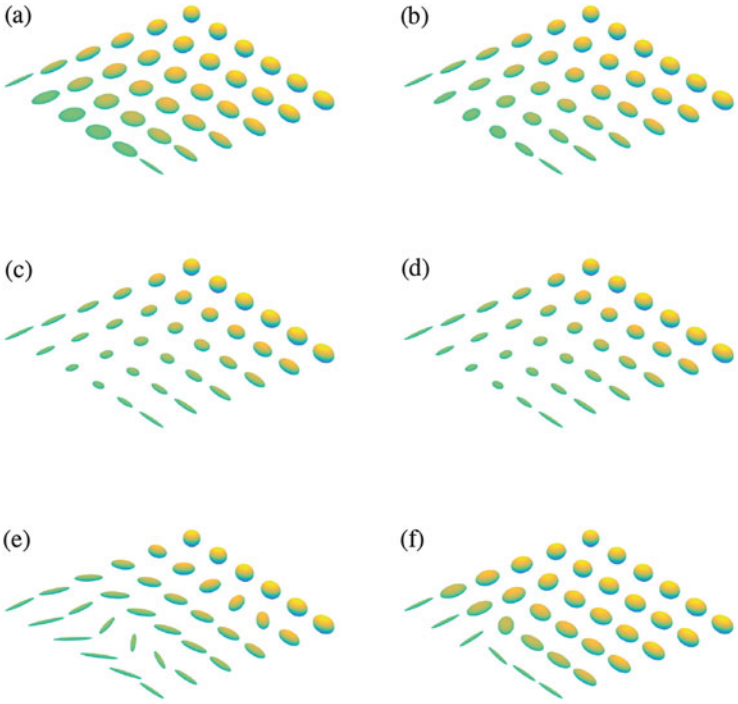


Fig. 16 Interpolation between four tensors located at the corners of the square-like arrangement of tensors, where the corner tensors all have three distinct eigenvalues but different shapes and orientations. (**a**) Frobenius. (**b**) Wasserstein. (**c**) Affine-invariant. (**d**) Log-Euclidean. (**e**) Shape-and rotation. (**f**) Linear invariant tensor interpolation

interpolation schemes handle interpolation between tensors of different shape and less extreme orientation differences.

In Figs. 16 and 17, all four tensors located at the corners of the square-like arrangement of tensors have three distinct eigenvalues to give optimal conditions for the shape-and-orientation metric. Two of the tensors are thin and ellipsoidal; in Fig. 16 one of them is rotated 85°. The two remaining tensors have the same orientation as the first, but different shape: one is a bit fatter, and one is almost spherical. In Fig. 17 the experiment is repeated with a 30° rotation instead of 85°.

Note that for the square geodesic interpolations the optimal interpolation would have been made as a weighted Fréchet mean; however, this strategy does not apply to the linear invariant tensor interpolation. Therefore, all interpolations were made by first interpolating pairs of corners to obtain two "opposite side" interpolations and then interpolating the elements of the sides to obtain the remainder of the square. The choice of which corners to interpolate first will influence the result, and this becomes particularly apparent in the shape-and-rotation and linear tensor invariant interpolations.

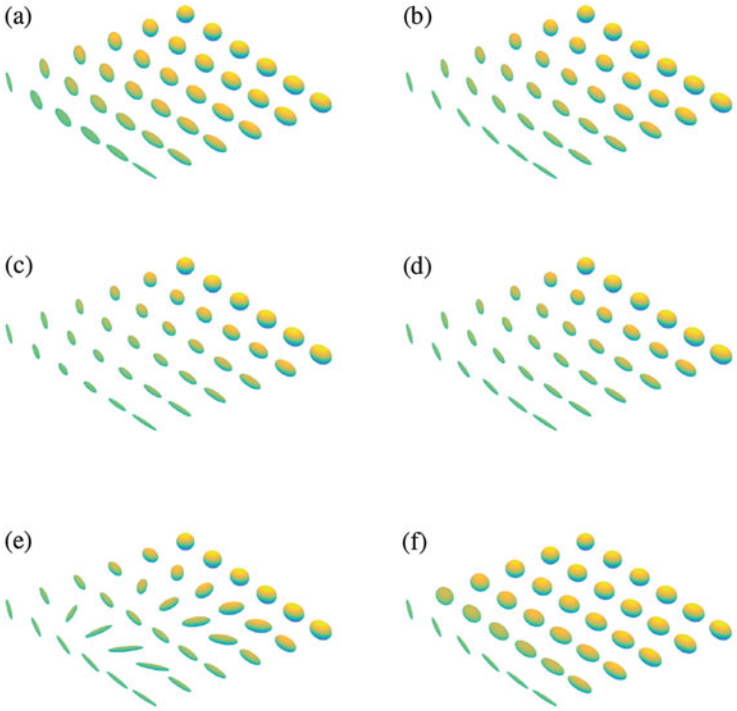


Fig. 17 Interpolation between four tensors located at the corners of the square-like arrangement of tensors, where the corner tensors all have three distinct eigenvalues but different shapes and orientations. Two tensors are thin and ellipsoidal; one of them is rotated 30°. The two remaining tensors have the same orientation as the first, but different shape: One is less thin and one is almost spherical. (**a**) Frobenius. (**b**) Wasserstein. (**c**) Affine-invariant (**d**) Log-Euclidean. (**e**) Shape-and rotation (**f**) Linear invariant tensor interpolation

Code for the experiment will be made available online at https://sites.google.com/site/aasaferagen/home/software.

Note the difference between the first and second columns of the interpolation results for the shape-and-orientation metric in Fig. 16e: The inconsistencies with respect to choice of representation of tensors results in consecutive pairwise interpolations being radically different geodesics in $\mathrm{Sym}^+(3)$, some of them not being shortest paths. This corresponds to our remarks in Sect. 5.1.

Note the abrupt change in orientation in the leftmost column of Fig. 16f. This is an effect of the choice of rotation component in the linear invariant interpolation, which is the rotation found in the corresponding Frobenius geodesic. The connection is visually evident by comparing to the Frobenius geodesic in Fig. 16a.

5.5 *Tensor Statistics and Tensor Decomposition for Visualization Purposes*

This paper was largely motivated by a discussion of visualization of tensor populations at the Dagstuhl seminar "Multidisciplinary Approaches to Multivalued Data: Modeling, Visualization, Analysis" (16142). In a recent paper, Zhang et al. [40] visualize tensor population variation by separately visualizing variation in scale, shape and orientation.

This leads to the question of how scale, shape and orientation can be decomposed in different geometric frameworks. In this section we present simulations that aim to make clear that the choice of metric on $\mathrm{Sym}^+(3)$ affects data variation both quantitatively in the notion of variance, and in the extent to which it is possible to decompose the variation into scale, shape and orientation without misrepresenting the metric. We do so by visualizing the first component of Fletcher principal geodesic analysis [13] on four different simulated tensor populations, shown in Fig. 18, for four of the above discussed metrics.

Each dataset consists of 20 tensors. The tensors in dataset 1 have identical shape and rotation, but variable scale (defined as the Frobenius norm of the tensor matrix); the tensors in dataset 2 have identical shape and scale but variable rotation; the tensors in dataset 3 have identical shape but the scale and rotation of the corresponding tensors in datasets 1 and 2, respectively. Dataset 4 has variable shape, scale and orientation.

Figure 19 shows sampled tensors along the first geodesic principal component for the Frobenius, affine-invariant, Log-Euclidean and shape-and-rotation metrics. The middle sample for each dataset and metric is the Fréchet mean. The first conclusions to be made from Fig. 19 is that the Riemannian metric frameworks are rather different, and that neither of the first three metrics capture the dataset variability very

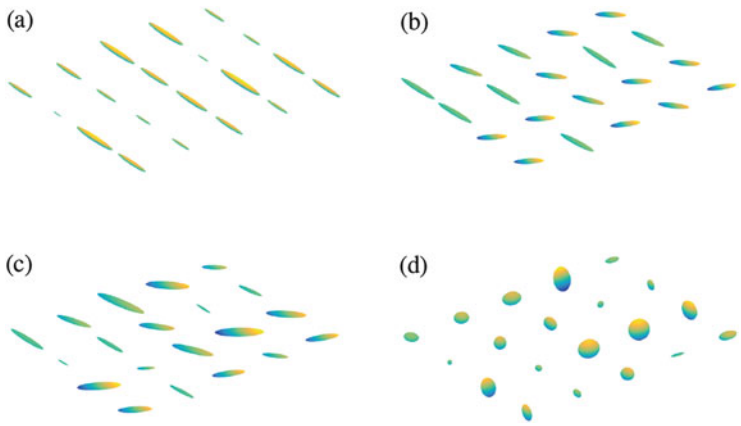


Fig. 18 Four simulated datasets. (**a**) Dataset 1. (**b**) Dataset 2. (**c**) Dataset 3. (**d**) Dataset 4

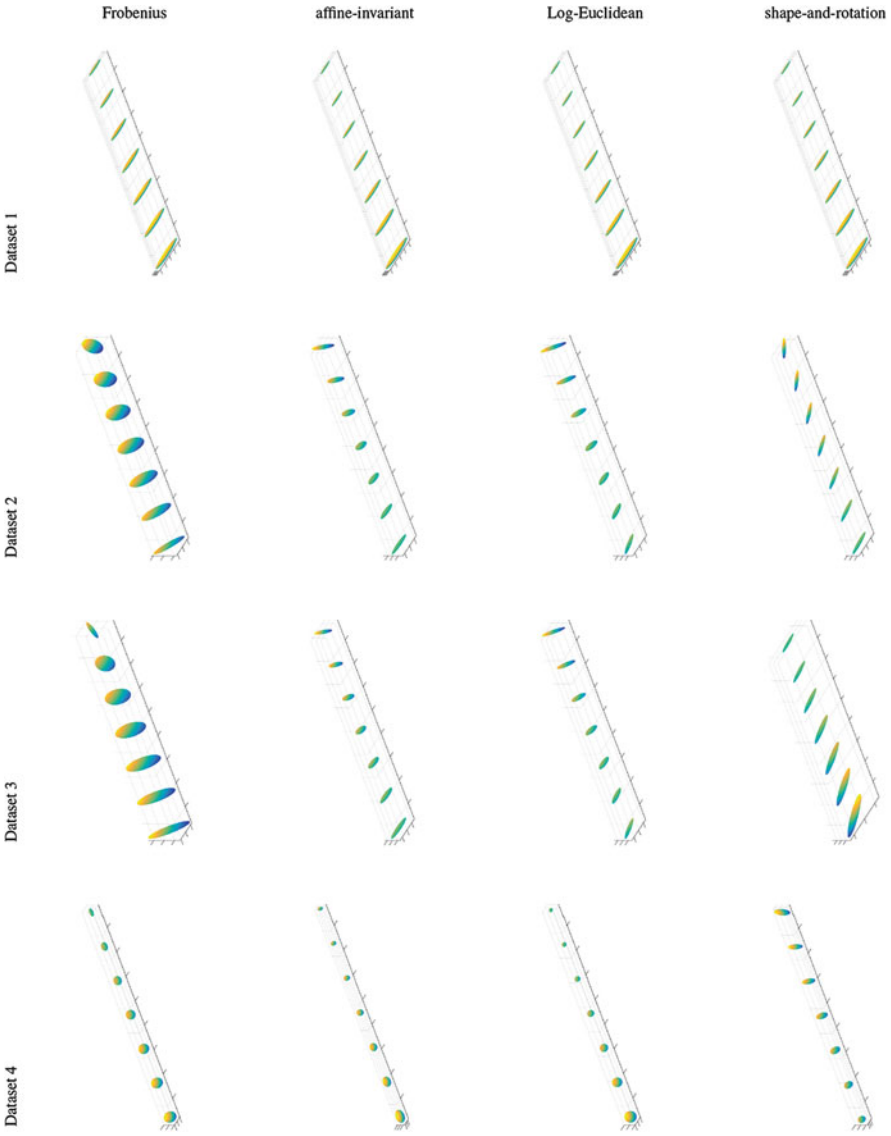


Fig. 19 First geodesic principal components in four Riemannian metrics, for the four simulated datasets

well except for the case where orientation is kept constant. In particular, while every tensor in the datasets 1–3 has identical shape, the geodesic principal components of these datasets for the first three Riemannian metrics indicate shape variation. In dataset 4, however, there *is* shape variation in the dataset, but the tensors sampled along the geodesic principal component do not exhibit much shape variation. The

tendency holds for all of the first three metrics. This indicates that the Riemannian metrics do not capture tensor shape very well in the presence of high orientation variation. This is not surprising.

The shape-and-rotation metric is better at capturing the shape variation (or lack of it), as expected. However, we already know that due to its multiple representation of single tensors, it will overestimate variance and we know the results to be incorrect.

Note that while the shape and scale of the tensors in datasets 1 and 3 are identical, the variation of the shape and scale along the first geodesic principal component is quite different both between the metrics and the datasets. This indicates that one should think carefully about how to separate scale, orientation and shape for visualization of variance when using a geometric framework. For instance, factoring out orientation before applying the geodesic principal component analysis as in dataset 1 changes the captured shape variance dramatically. This might be surprising.

6 Discussion and Conclusion

6.1 Further Related Work

Tensor interpolation and geometric frameworks for analysis of tensors has been an active field of research for a number of years, and in the above we have only touched upon some of the most classical and most recent approaches to tensor interpolation. As an extension of the geometric framework, several approaches have appeared that utilize *divergences* [10, 31]. Divergences are not generally symmetric and therefore do not lead to a geodesic space in an obvious way. They can, however, be closely linked to Riemannian metrics, as in the case of the KL divergence, which infinitesimally coincides with the Fisher-Rao metric [2]. In the context of machine learning, kernel methods have also been proposed for SPD matrices [22]; however, these have been shown to just consist of Gaussian kernels on Euclidean features extracted from the SPD matrices [12]. Aside from tensors, approaches that try to separate rotation from other properties have also appeared [9, 25].

Several surveys and comparisons exist for geometries and interpolations for $\mathrm{Sym}^+(3)$. Moakher et al. [26] compare the affine-invariant metric to the closely related Kullback-Leibler divergence in both geometric properties and in the context of statistics and visualization. Both Peeters et al. [27] and Zhou et al. [41] compare different distance/similarity measures for DTI, including several simple measures along with the affine-invariant and Log-Euclidean metrics representing the Riemannian approaches; the latter has a focus on regularization. What our survey has to offer in comparison is an extensive discussion of nonlinear geometries on $\mathrm{Sym}^+(3)$, a comparison with non-geometric approaches such as LIT [16], a thorough mathematical discussion of the current status of approaches that decompose shape and orientation—and a mathematical explanation why this is not trivial.

Moreover, we offer publically available software[2] online to enable any reader to start working with geometries on $\mathrm{Sym}^+(3)$.

6.2 Geometry Versus Shape Preservation

We have investigated a number of different geometric frameworks for tensor computation: The Frobenius, Wasserstein, affine-invariant and log-Euclidean metrics are all Riemannian metrics on $\mathrm{Sym}^+(3)$. However, they all exhibit the fattening effect, which can lead to unwanted smoothing effects when used for tensor interpolation or statistics. The shape-and-orientation metric is also a Riemannian metric. However, it comes with multiple representations of the same tensor, which leads to inconsistencies and unreliable statistics in practice, as illustrated by our experiments. This can be handled by factoring out the multiple representations in a quotient space. This quotient space is still a geodesic metric space, but it is no longer a Riemannian manifold—it has singularities and is a *stratified space* [18, 28]. The geodesics in this space could be used for interpolation, but a remaining problem for a practical investigation of its usefulness is to obtain an algorithm for computing the quotient metric geodesics. We expect this to be computationally demanding.

We have also reviewed frameworks that let go of geometry and simply ask for tensor interpolations that preserve shape well. These interpolations would very likely be good at preserving signal, but do not give the geometric framework of a geodesic metric space. In particular, this means that we do not have access to statistical approaches such as Fréchet means, hypothesis tests, geodesic principal component analysis or regression, as we have with the more geometric approaches. This also makes interpolation of multiple tensors less well-defined, as we have already observed in Figs. 16 and 17.

6.3 Why Are Second Order Tensors Still Interesting?

Second order tensors were key objects in DTI, but with the advent of HARDI imaging [34], they are often considered "too simple" to warrant further study. We argue the opposite. If you want to build geometries or tools that can handle the challenges of higher order tensors in diffusion-weighted imaging, these tools have better also perform well on second order tensors in DTI. Some of the most natural choices of metrics on the distributions returned by fODF model estimators such as constrained spherical deconvolution [33] or Q-ball [35], are the Wasserstein and Fisher-Rao metrics [17], as these are defined and theoretically well understood for

[2]Upon publication, software will be available at https://sites.google.com/site/aasaferagen/home/software.

general probability distributions. However, as we have seen in this survey, these metrics have unwanted smoothing effects when used for interpolation or statistics on second order tensors, and should not be expected to behave better on higher order tensors [9].

Another route to comparison of higher-order tensors comes through representations of higher order tensors based on second order tensors. Such representations are given by multi-compartment models [1, 21] as well as through Finsler geometry, where any higher order ODF can be represented through a Finsler norm, and any Finsler norm can be represented as an orientation-dependent second order tensor [5, 15]. In both of these representations, a well-chosen metric for second order tensors may be extended to higher order tensors by integrating over orientation. This is an interesting direction for future work.

Finally, second order tensors have other applications. They are used in the recent tensor confinement formalism [39], where the second order tensor is used to represent constraints on the diffusion process. Moreover, second order tensors are in 1-1 correspondence with covariance matrices, and any metric on second order tensors therefore also defines a metric on centered multivariate normal distributions. Statistics on probability distributions have many possible applications, from population statistics on uncertain tractography results represented as Gaussian Processes [20, 29] via evolutionary algorithms for optimization [19], to information geometry [2].

The quest for a descriptive geometric framework for $\mathrm{Sym}^+(3)$ therefore continues.

References

1. Alexander, D., Barker, G., Arridge, S.: Detection and modeling of non-Gaussian apparent diffusion coefficient profiles in human brain data. Magn. Reson. Med. **48**(2), 331–340 (2002)
2. Amari, S.-I., Nagaoka, H.: Methods of Information Geometry. Translations of Mathematical Monographs. American Mathematical Society, Providence (2000)
3. Ambrosio, L., Gigli, N., Savaré, G.: Gradient Flows: In Metric Spaces and in the Space of Probability Measures. Springer Science & Business Media, Berlin (2008)
4. Arsigny, V., Commowick, O., Pennec, X., Ayache, N.: A Log-Euclidean framework for statistics on diffeomorphisms. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI (2006)
5. Astola, L., Florack, L.: Finsler geometry on higher order tensor fields and applications to high angular resolution diffusion imaging. Int. J. Comput. Vis. **92**(3), 325–336 (2011)
6. Basser, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. J. Magn. Reson. Ser. B **111**(3), 209–219 (1996)
7. Batchelor, P.G., Moakher, M., Atkinson, D., Calamante, F., Connelly, A.: A rigorous framework for diffusion tensor calculus. Magn. Reson. Med. **53**(1), 221–225 (2005)
8. Bridson, M., Haefliger, A.: Metric Spaces of Non-Positive Curvature. Springer, Berlin (1999)
9. Çetingül, H., Afsari, B., Wright, M., Thompson, P., Vidal, R.: Group action induced averaging for HARDI processing. In: 9th IEEE International Symposium on Biomedical Imaging, ISBI (2012)

10. Cherian, A., Sra, S., Banerjee, A., Papanikolopoulos, N.: Jensen-bregman logdet divergence with application to efficient similarity search for covariance matrices. IEEE Trans. Pattern Anal. Mach. Intell. **35**(9), 2161–2174 (2013)
11. Ennis, D.B., Kindlmann, G.: Orthogonal tensor invariants and the analysis of diffusion tensor magnetic resonance images. Magn. Reson. Med. **55**(1), 136–146 (2006)
12. Feragen, A., Lauze, F., Hauberg, S.: Geodesic exponential kernels: when curvature and linearity conflict. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2015)
13. Fletcher, P., Lu, C., Pizer, S., Joshi, S.: Principal geodesic analysis for the study of nonlinear statistics of shape. IEEE Trans. Med. Imaging **23**, 995–1005 (2004)
14. Fletcher, P.T., Joshi, S.: Principal geodesic analysis on symmetric spaces: statistics of diffusion tensors. In: Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis, pp. 87–98. Springer, Berlin (2004)
15. Florack, L., Fuster, A.: Riemann-Finsler Geometry for Diffusion Weighted Magnetic Resonance Imaging, pp. 189–208. Springer, Berlin (2014)
16. Gahm, J.K., Wisniewski, N., Kindlmann, G., Kung, G.L., Klug, W.S., Garfinkel, A., Ennis, D.B.: Linear Invariant Tensor Interpolation Applied to Cardiac Diffusion Tensor MRI. Med. Image Comput. Comput. Assist. Interv. **15**(2), 494–501 (2012)
17. Goh, A., Lenglet, C., Thompson, P., Vidal, R.: A nonparametric Riemannian framework for processing high angular resolution diffusion images and its applications to ODF-based morphometry. NeuroImage **56**(3), 1181–1201 (2011)
18. Groisser, D., Jung, S., Schwartzman, A.: Eigenvalue stratification and minimal smooth scaling-rotation curves in the space of symmetric positive-definite matrices. Electron. J. Stat. **11**(1), 1092–1159 (2017)
19. Hansen, N.: The CMA evolution strategy: a comparing review. In: Towards a New Evolutionary Computation, pp. 75–102. Springer, Berlin (2006)
20. Hauberg, S., Schober, M., Liptrot, M., Hennig, P., Feragen, A.: A random riemannian metric for probabilistic shortest-path tractography. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI), Munich, Germany (2015)
21. Inglis, B.A., Bossart, E.L., Buckley, D.L., Wirth, E.D., Mareci, T.H.: Visualization of neural tissue water compartments using biexponential diffusion tensor MRI. Magn. Reson. Med. **45**(4):580–587 (2001)
22. Jayasumana, S., Hartley, R., Salzmann, M., Li, H., Harandi, M.: Kernel methods on the riemannian manifold of symmetric positive definite matrices. In: 2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 73–80. IEEE (2013)
23. Jung, S., Schwartzman, A., Groisser, D.: Scaling-rotation distance and interpolation of symmetric positive-definite matrices. SIAM SIAM J. Matrix Anal. Appl. **36**(3), 1180–1201 (2015)
24. Kindlmann, G.L., Estépar, R.S.J., Niethammer, M., Haker, S., Westin, C.-F.: Geodesic-loxodromes for diffusion tensor interpolation and difference measurement. In: Ayache, N., Ourselin, S., Maeder, A. (eds.) International Conference on Medical Image Computing and Computer-Assisted Intervention MICCAI (1). Lecture Notes in Computer Science, vol. 4791, pp. 1–9. Springer, Berlin (2007)
25. Levy, J.H., Foskey, M., Pizer, S. M.: Rotational flows for interpolation between sampled surfaces. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2008, Anchorage, AK, USA, 23–28 June, 2008, pp. 1–8 (2008)
26. Moakher, M., Batchelor, P.G.: Symmetric Positive-Definite Matrices: From Geometry to Applications and Visualization, pp. 285–298. Springer, Berlin (2006)
27. Peeters, T., Rodrigues, P., Vilanova, A., ter Haar Romeny, B.: Analysis of distance/similarity measures for diffusion tensor imaging. In: Visualization and Processing of Tensor Fields, pp. 113–136. Springer, Berlin (2009)
28. Pflaum, M.: Analytic and Geometric Study of Stratified Spaces. Springer Lecture Notes in Mathematics. Springer, Berlin (2001)

29. Schober, M., Kasenburg, N., Feragen, A., Hennig, P., Hauberg, S.: Probabilistic shortest path tractography in dti using Gaussian process ode solvers. In: Medical Image Computing and Computer-Assisted Intervention (MICCAI), Boston (2014)
30. Schwartzman, A., Dougherty, R.F., Taylor, J.E.: False discovery rate analysis of brain diffusion direction maps. Ann. Appl. Stat. **2**(1), 153–175 (2008)
31. Sra, S.: Positive definite matrices and the s-divergence. Proc. Am. Math. Soc. **144**(7), 2787–2797 (2016)
32. Takatsu, A.: Wasserstein geometry of Gaussian measures. Osaka J. Math. **48**(4), 1005–1026 (2011)
33. Tournier, J., Calamante, F., Gadian, D., Connelly, A.: Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. NeuroImage **23**(3), 1176–1185 (2004)
34. Tuch, D., Reese, T.G., Wiegell, M., Makris, N., Belliveau, J., Wedeen, V.: High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. Magn. Res.Med. **48**(4), 577–582 (2002)
35. Tuch, D.S.: Q-ball imaging. Magn. Reson. Med. **52**(6), 1358–1372 (2004)
36. Tuzel, O., Porikli, F., Meer, P.: Region covariance: a fast descriptor for detection and classification. In: European Conference on Computer Vision (ECCV), pp. 589–600 (2006)
37. Wang, Y., Salehian, H., Cheng, G., Vemuri, B.C.: Tracking on the product manifold of shape and orientation for tractography from diffusion MRI. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23–28, 2014, pp. 3051–3056 (2014)
38. Westin, C.-F., Peled, S., Gudbjartsson, H., Kikinis, R., Jolesz, F.A.: Geometrical diffusion measures for MRI from tensor basis analysis. In: Proceedings of the 5th Scientific Meeting and Exhibition of ISMRM '97, p. 1742, Vancouver Canada (1997)
39. Yolcu, C., Memiç, M., Şimşek, K., Westin, C.-F., Özarslan, E.: NMR signal for particles diffusing under potentials: from path integrals and numerical methods to a model of diffusion anisotropy. Phys. Rev. E **93** (2016)
40. Zhang, C., Schultz, T., Lawonn, K., Eisemann, E., Vilanova, A.: Glyph-based comparative visualization for diffusion tensor fields. IEEE Trans. Vis. Comput. Graph. **22**(1), 797–806 (2016)
41. Zhou, D., Dryden, I.L., Koloydenko, A.A., Audenaert, K.M., Bai, L.: Regularisation, interpolation and visualisation of diffusion tensor images using non-euclidean statistics. J. Appl. Stat. **43**(5), 943–978 (2016)

Towards Processing Fields of General Real-Valued Square Matrices

Bernhard Burgeth and Andreas Kleefeld

Abstract In this paper, a general framework is presented that allows for the fundamental morphological operations such as dilation and erosion for real-valued square matrix fields. Hence, it is also possible to process any field consisting of a subgroup of general matrices with examples like the general linear, symmetric, skew-symmetric, Hermitian, and orthonormal group. Therefore, from the theoretical point of view it is possible to process any field with entries consisting of the aforementioned groups. Extended examples illustrated the different conversion processes and the definition of corresponding pseudo-suprema and pseudo-infima. Furthermore, some possible applications are illustrated.

1 Introduction

In the phrase "image processing for tensor fields" the notion tensor fields in general refers to fields of symmetric (even positive definite) matrices with real entries. This work is devoted to the development of morphological image processing tools for general 3D matrix fields, however, with a certain emphasis on real orthogonal matrices with determinant one. As a general 3D matrix field we denote any mapping

$$f : \Omega \longmapsto \mathrm{M}_{\mathbb{R}}(\mathrm{n})$$

from a two- or three-dimensional image domain Ω into the set $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ of $n \times n$-real matrices M. We will refer to them as $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$-valued images, or even shorter, as $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$-fields. A similar terminology is used when we are dealing with the vector spaces $\mathrm{SYM}(n)$, the space of symmetric $n \times n$-real matrices, $\mathrm{SKEW}(n)$, the space of

B. Burgeth (✉)
Faculty of Mathematics and Computer Science, Saarland University, 66041 Saarbrücken, Germany
e-mail: burgeth@math.uni-sb.de

A. Kleefeld
Forschungszentrum Jülich GmbH, Institute for Advanced Simulation, Jülich Supercomputing Centre, Wilhelm-Johnen-Straße, 52425 Jülich, Germany
e-mail: a.kleefeld@fz-juelich.de

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_6

skew-symmetric $n \times n$-real matrices, and H(n), the space of Hermitian $n \times n$-complex matrices. And it will be these Hermitian matrices that will play a key role in all that follows. We will also be interested in subsets of matrices of $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ that do not form a vector space, for instance, Gl(n), the set of $n \times n$ real matrices with non-vanishing determinant, $\det(G) \neq 0$, forming a group with respect to matrix multiplication. Or SO(3), the set of rotations, a group with respect to matrix multiplication. In the sequel, we will restrict our attention to the cases $n = 3$ for the sake of potential applications and for explanatory purposes to $n = 2$.

SYM(3)-fields are still playing an important role, for example, in DT-MRI, a technique in medical imaging capable to capture the nerve fiber structure in human tissue, see for instance [28, 46], and [41]. Three-dimensional symmetric tensor fields find wide spread applications in areas of civil engineering such as solid and fluid mechanics as well (see [1, 34, 36] and the literature cited therein). And symmetric matrices make an appearance in image processing itself through the structure tensor concept as early as in [32] and [33]; see also, for example, [4, 6] and the references therein. Even for the processing of color images symmetric matrices seem to be useful, [5, 29, 44]. The processing, the analysis, and the visualization of such fields have been at the center of countless research efforts as the references in the literature such as [35] and [28] indicate.

Due to the orthonormal diagonalizability of symmetric matrices, SYM(3)-fields have a rich functional-algebraic calculus. Since $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ does not posses such a rich algebraic structure it is not immediately clear how to establish image processing methods for this very general type of data.

The key idea is complexification, that is, embedding $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ into H(n) in a one-to-one manner. In this way, we will be able to take advantage of the elementary fact that each Hermitian matrix is unitary similar to a diagonal matrix with real entries. In other words, H(n) is just as convenient as a space as SYM(3) is, and many techniques developed for SYM(3) will carry over directly to H(n). Since subsets of $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ are mapped to subsets of H(n) such an embedding opens up paths for the treatment of such subsets otherwise not accessible to image processing techniques.

Three types of operations on data types are the cornerstones of a typical image processing algorithm: linear combination (addition and multiplication with a scalar), multiplication, and finding the sup/inf of a set of data points. For multivariate data very often neither all of these operations are defined nor a substitute is close at hand. It is the aim of this work to provide suitable notions of linear combination, multiplication, and sup/inf for data types that do not lend themselves to these operations: matrix-valued images or matrix fields where the range are real $n \times n$-matrices (no notion of sup/inf), or orthogonal matrices (no notion of sup/inf or linear combination). The above problem has been solved to some extent for real symmetric matrices, since they have a rather rich algebraic analytic structure. The latter approach will serve us as a guide line to achieve similar results in the aforementioned cases of $\mathrm{M}_{\mathbb{R}}(n)$ and $\mathrm{SO}(n)$.

Since even a 2×2-Hermitian matrix has 4 degrees of freedom our possibilities for a visualization of such matrix fields are very limited, at best. The same holds true for the aforementioned classes of matrices. Therefore, we will focus more on

the theoretical aspects providing a proof-of-concept rather than the evaluation of experiments. The visualization of multivariate data is a very active research area in its own right (see for example [19, 40, 42]).

Non-linear continuum mechanics employs a multitude of notions of tensors. In elasticity theory for materials the displacement field for particles is fundamental; the Jacobian of this mapping is then, for the sake of simplification, transformed into the symmetric strain tensor. An equally important quantity in kinematics is the deformation gradient, again the Jacobian of a deformation mapping, and we obtain the non-symmetric stress tensor. In general for modeling and numerical reasons, it is split via polar decomposition into a symmetric and orthogonal matrix (see [36, 43]). From these symmetric matrices mainly scalar quantities are derived to predict material behavior or potential failure in numerical simulations. In order to take advantage of the information captured in the associated field of special orthogonal matrices one has to find sustainable means to process these types of data (possibly via image processing techniques). This chapter is a first step into that direction.

Another field, where the framework of square, non-symmetric matrices might be useful is the processing of image data stemming from different imaging modalities. One might expect that this information and the dependencies between the channels can be translated into the language of (square) matrices. Then these matrix fields may be transformed and analyzed, provided sufficiently powerful processing techniques are at our disposal.

Of even more speculative nature (and as such subject to future research): for the processing of social-economic data (e.g. living conditions, spreading of diseases, etc.) or mathematical biology (e.g. chemotaxis) [25] it might be possible and useful to cast this information into matrix form, and again we are confronted with the task of "handling" matrix fields.

Hence, it appears useful to the authors to struggle for a unified approach to calculus tools for fields of square matrices. This will enable us to design advanced techniques (possible related to image processing) to filter and analyze these fields and the information therein.

This work has the following structure: In Sect. 2, the basics for the $\mathbb{R}$-vector space of Hermitian matrices is explained. The next section gives the basic calculus for fields of Hermitian matrices. Section 4 deals with an isomorphism between $\mathrm{M}_{\mathbb{R}}(n)$ and $\mathrm{H}(n)$ that is constructed to process fields in this space and in various subgroups of $\mathrm{M}_{\mathbb{R}}(n)$. Section 5 introduces the reader to the approximation of the pseudo supremum and infimum as well as averaging. While in Sects. 6 and 7 two applications are discussed in detail. Concluding remarks and an outlook are given in the final Sect. 8.

2 The $\mathbb{R}$-Vector Space of Hermitian Matrices

A Hermitian matrix H is a complex square matrix which equals its own conjugate transpose:

$$H = \overline{H}^{\top} =: H^*$$

The real part, $\Re(H)$, of an Hermitian Matrix H is symmetric, $\Re(H) \in \mathrm{SYM}(n)$, while the imaginary part, $\Im(H)$ is skew-symmetric, $\Im(H) \in \mathrm{SKEW}(n)$. Clearly, one has $H = \Re(H) + \mathrm{i}\Im(H)$. Any Hermitian matrix H can be diagonalized by means of a suitable unitary matrix and, furthermore, all the eigenvalues are real:

$$H = UDU^* .$$

U is unitary, that is, $U^*U = UU^* = I$, and $D = \mathrm{diag}(d_1, \ldots, d_n)$ is a diagonal matrix with real entries in decreasing order, $d_1 \geq \ldots \geq d_n$. H is called positive semi-definite if all of its eigenvalues are positive: $d_1 \geq \ldots \geq d_n \geq 0$. This is equivalent to the positivity of the quadratic form $x^*Hx \geq 0$, where x is a complex vector, $x \in \mathbb{C}^n$. If the inequalities are strict, then the matrix is called positive definite. A matrix H is negative (semi-)definite if $-H$ is positive (semi-)definite. If the matrix H is none of the above, then the Hermitian matrix is called indefinite. This gives rise to a partial order "$\geq$" on the set of Hermitian matrices:

$$A \geq B \text{ if and only if } A - B \text{ is positive semi-definite.}$$

In the case of real symmetric matrices this order is often referred to as Loewner order (see [3]). Note that the Hermitian matrices equipped with the Loewner order do not form a lattice, since the upper bounds of a set $\{H_1, H_2, \ldots, H_n\}$ with $n \geq 2$ is in general not totally ordered, hence supremum as the least upper bound may not exist. The same holds for the infimum. However, the Loewner order allows for a so-called pseudo-supremum as the upper bound with the smallest trace; likewise for a pseudo-infimum, see the definitions in the next section. It is worth noting that the definiteness of a Hermitian matrix is solely determined by the definiteness of its real part, since $x^*\Im(H)x = 0$ for any $x \in \mathbb{C}^n$, as for all skew-symmetric matrices. As a consequence, the order between two Hermitian matrices is given by the order of their symmetric real parts. As it is pointed out in [13], and in more detail in [12, 15], a rich functional algebraic calculus can be set up for symmetric matrices (see [24]). This allows to establish numerous filtering and analysis methods for such fields of symmetric matrices in a rather straightforward way from their scalar counterparts [16]. Hermitian matrices as immediate generalizations of symmetric matrices possess an equally rich and easily manageable calculus.

Processing a matrix field amounts to applying an operator $\mathscr{A}$ to the field. This operation might be a simple concatenation with a (scalar) function f, exploiting the unitary diagonalizability of a Hermitian matrix H, or it might be an application of a function of several variables, a matrix valued differential operator, or even a step

in a numerical algorithm, such as a time step in an explicit scheme to solve partial differential equations. This requires that a certain amount of operations of a matrix calculus is at our disposal. In order to make this work as self-contained as possible we present some very basic notions from calculus for fields of Hermitian matrices in the next section.

3 Rudimentary Calculus for Fields of Hermitian Matrices

In this section, we provide briefly the *basic definitions* for the formulation of a very basic calculus for Hermitian matrix fields.

3.1 Basic Functions for Hermitian Matrices

In this subsection, we explain what a function of a Hermitian matrix is, a reasonable product of two Hermitian matrices, and what fruitful notions of supremum and infimum of two Hermitian matrices are. In part, this is a generalization of some of the material in [12] and [14] to Hermitian matrices.

1. **Functions of Hermitian matrices.** The standard definition of a function f on H(n) is given by [26]:

$$f(H) := U \cdot \mathrm{diag}(f(d_1), \ldots, f(d_n)) \cdot U^* \tag{1}$$

   provided $H = U\mathrm{diag}(\lambda_1, \ldots, \lambda_n)U^*$ is the spectral decomposition of the Hermitian matrix H and $\lambda_1, \ldots, \lambda_n$ lie in the domain of definition of f.
2. **Symmetrized product of Hermitian matrices.** The product of two Hermitian matrices $A, B \in \mathrm{H(n)}$ is not Hermitian unless the matrices commute. However, it is vital to our interests to have a symmetrized matrix product at our disposal. There are numerous options to define a symmetrized matrix product, however, we concentrate on a specific one known from algebra called Jordan product:

$$A \bullet_J B = \frac{1}{2}(AB + BA) \quad \text{for } A, B \in \mathrm{H(n)}\,.$$

   For commuting A and B we have $A \bullet_J B = A \cdot B$. This product is commutative and distributive but not associative. It is one half of the anti-commutator of A and B, but due to its additive structure no determinant product rule holds. Most important, it does not preserve the positive semi-definiteness of its arguments as the following simple example shows:

$$\begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix} \bullet_J \begin{pmatrix} 1 & \mathrm{i} \\ -\mathrm{i} & 1 \end{pmatrix} = \frac{1}{2}\left(\begin{pmatrix} 2 & 2\mathrm{i} \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ -2\mathrm{i} & 0 \end{pmatrix}\right) = \begin{pmatrix} 2 & \mathrm{i} \\ -\mathrm{i} & 0 \end{pmatrix}, \det \begin{pmatrix} 2 & \mathrm{i} \\ -\mathrm{i} & 0 \end{pmatrix} = -1\,.$$

3. **Pseudo-supremum/infimum of two Hermitian matrices.** Inspired by the equations $2\max(a,b) = a + b + |a-b|$ and $2\min(a,b) = a + b - |a-b|$ valid for $a, b \in \mathbb{R}$, we set in the case of Hermitian matrices H_1 and H_2

$$\operatorname{psup}(H_1, H_2) := \frac{1}{2}\left(H_1 + H_2 + |H_1 - H_2|\right) \tag{2}$$

and

$$\operatorname{pinf}(H_1, H_2) := \frac{1}{2}\left(H_1 + H_2 - |H_1 - H_2|\right) \tag{3}$$

to be the pseudo-supremum and the pseudo-infimum of H_1 and H_2. Here $|H_1 - H_2|$ is understood as the absolute value of $H_1 - H_2$ in the sense of the above extension of the absolute value function to Hermitian matrices (refer to (1)).

As previously pointed out, a certain justification for the use of the notation psup respectively pinf is provided by the following

Proposition 1 *If $H_1, H_2 \in H(n)$, then* $\sup(H_1, H_2)$ *is the least upper bound for both H_1 and H_2 in the sense that the smallest eigenvalue of both* $\sup(H_1, H_2) - H_1$ *and* $\sup(H_1, H_2) - H_2$ *is zero. A corresponding statement holds true for* $\inf(H_1, H_2)$ *as the greatest lower bound of H_1 and H_2.*

Proof We can assume that H_1 and H_2 are not comparable in the Loewner order. Then $H_1 - H_2$ is indefinite, hence for its smallest eigenvalue $d_3 < 0$ holds. We have

$$\begin{aligned}
\sup(H_1, H_2) - H_2 &= 1/2(H_1 + H_2 + |H_1 - H_2|) - H_2 \\
&= 1/2(H_1 - H_2 + |H_1 - H_2|) \\
&= 1/2(U \operatorname{diag}(d_1, d_2, d_3)\, U^* + U \operatorname{diag}(|d_1|, |d_2|, |d_3|)\, U^*) \\
&= 1/2(U \operatorname{diag}(d_1 + |d_1|, d_2 + |d_2|, d_3 + |d_3|)\, U^*) \\
&= 1/2(U \operatorname{diag}(d_1 + |d_1|, d_2 + |d_2|, 0)\, U^*)\,.
\end{aligned}$$

A corresponding reasoning with H_1 and $H_2 - H_1$ completes the proof concerning the supremum. The proof of the part for the infimum is analog or can be deduced from the fact that $\inf(H_1, H_2) = -\sup(-H_1, -H_2)$. The extensions of elements of scalar-valued calculus to the matrix-valued setting is given in Table 1. With these basic algebraic-analytic operations at our disposal it is now possible, in principle, to design image processing algorithms for fields of Hermitian matrices.

Table 1 Extensions of elements of scalar-valued calculus (middle) to the matrix-valued setting (right)

Setting	Scalar valued	Matrix-valued
Function	$f : \begin{cases} \mathbb{R} \longrightarrow \mathbb{R} \\ x \mapsto f(x) \end{cases}$	$F : \begin{cases} H(n) \longrightarrow H(n) \\ H \mapsto U\mathrm{diag}(f(\lambda_1), \ldots, f(\lambda_n))\, U^* \end{cases}$
Product	$a \cdot b$	$A \bullet B = \frac{1}{2}(AB + BA)$
Supremum	$\sup(a, b)$	$\mathrm{psup}(A, B) = \frac{1}{2}(A + B + \lvert A - B\rvert)$
Infimum	$\inf(a, b)$	$\mathrm{pinf}(A, B) = \frac{1}{2}(A + B - \lvert A - B\rvert)$

3.2 Examples for Basic Hermitian Matrix Calculus

The following examples may illustrate the above definitions.

Example 1 As a first important example of a function application, we consider the absolute value $|H|$ of an Hermitian matrix H with $n = 3$: it is the positive semi-definite Hermitian matrix obtained by applying the absolute value function to the real eigenvalues of H; that is

$$|H| = U \cdot \mathrm{diag}(|d_1|, |d_2|, |d_3|) \cdot U^*$$

With this at our disposal, we calculate the absolute value and the (pseudo-)infimum and (pseudo-)supremum of the two Hermitian matrices

$$A = \begin{pmatrix} 3 & 1+\mathrm{i} & \mathrm{i} \\ 1-\mathrm{i} & 1 & 0 \\ -\mathrm{i} & 0 & 1 \end{pmatrix}, \qquad B = \begin{pmatrix} 1 & 3-\mathrm{i} & 4 \\ 3+\mathrm{i} & -2 & -6+\mathrm{i} \\ 4 & -6-\mathrm{i} & 5 \end{pmatrix}.$$

For A and B, we obtain the three eigenvalues $[0, 1, 4]$ and $[-7.9622, 2.7363, 9.2259]$, respectively. Hence, $|A| = A$ and $|B|$ is given by

$$|B| = \begin{pmatrix} 4.4884 & -2.4459 - 0.2239\mathrm{i} & 0.4625 + 0.7794\mathrm{i} \\ -2.4459 + 0.2239\mathrm{i} & 6.6745 & -0.3041 + 0.5703\mathrm{i} \\ 0.4625 - 0.7794\mathrm{i} & -0.3041 - 0.5703\mathrm{i} & 8.7613 \end{pmatrix}.$$

The (pseudo-)supremum and (pseudo-)infimum of the two Hermitian matrices are given by

$$\operatorname{psup}(A,B) = \begin{pmatrix} 3.6489 & 0.0738+0.6282\mathrm{i} & 1.9852+1.1527\mathrm{i} \\ 0.0738-0.6282\mathrm{i} & 2.5353 & -2.9214+0.9196\mathrm{i} \\ 1.9852-1.1527\mathrm{i} & -2.9214-0.9196\mathrm{i} & 7.1099 \end{pmatrix},$$

$$\operatorname{pinf}(A,B) = \begin{pmatrix} 0.3511 & 3.9262-0.6282\mathrm{i} & 2.0148-0.1527\mathrm{i} \\ 3.9262+0.6282\mathrm{i} & -3.5353 & -3.0786+0.0804\mathrm{i} \\ 2.0148+0.1527\mathrm{i} & -3.0786-0.0804\mathrm{i} & -1.1099 \end{pmatrix},$$

respectively.

However, we will not process fields of Hermitian matrices for their own sake. Instead, various classes K of matrices with real entries are going to be embedded into H(n) with suitable $n = 2, 3, \ldots$ and processed in this "detour space" before being projected back. The classes K considered in this work are $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$, and its subsets Gl(n), O(n), and SO(n). In the next section, we describe the mapping that embeds sets of real $n \times n$-matrices into H(n). We define the mapping on the largest set of real $n \times n$-matrices, the set $K = \mathrm{M}_{\mathbb{R}}(\mathrm{n})$ itself, which amounts to the aforementioned complexification.

4 An Isomorphism Between $\mathrm{M}_{\mathbb{R}}$(n) and H(n)

Despite its simplicity the following construction is decisive for all that follows. Exploiting the decomposition of $M \in \mathrm{M}_{\mathbb{R}}(\mathrm{n})$ into a symmetric and a skew-symmetric part, $M = S + A$, we define a mapping $\Psi : \mathrm{M}_{\mathbb{R}}(\mathrm{n}) \to \mathrm{H(n)}$ simply by

$$\Psi : M \longmapsto S + \mathrm{i}A = \frac{1}{2}(M + M^{\top}) + \frac{\mathrm{i}}{2}(M - M^{\top}). \tag{4}$$

This mapping is linear and invertible

$$\Psi^{-1} : H \longmapsto \frac{1}{2}(H + H^{\top}) - \frac{\mathrm{i}}{2}(H - H^{\top}) \tag{5}$$

for any $H \in \mathrm{H}(3)$.

Example 2 The so called Pauli matrices

$$\left\{ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\}$$

form a basis of H(2). The corresponding basis of $\mathrm{M}_{\mathbb{R}}(2)$ via Ψ^{-1} reads, in that order:

$$\left\{ \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}, \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right\} .$$

The Pauli matrices are playing an important role in mathematical physics, especially in quantum mechanics, but they are of interest in the theory of Lie groups, quaternions, and Clifford algebras as well (see [2, 18]).

It is slightly more tedious to find the basis of $\mathrm{M}_{\mathbb{R}}(3)$ corresponding to the Gell-Mann matrices (see [2]) the analog to the Pauli matrices, as a basis for H(3). However, we utilize H(n) and the maps Ψ, Ψ^{-1} to tackle image processing tasks for fields of matrices from $\mathrm{M}_{\mathbb{R}}(n)$ by adhering to the following **strategy:** instead of addressing a problem in $\mathrm{M}_{\mathbb{R}}(n)$ with its relatively poor functional-algebraic structure we transfer the problem to H(n), take advantage of the rich calculus there when applying an operation, say $\mathscr{A}$, and finally transform the (pre-) solution back to $\mathrm{M}_{\mathbb{R}}(n)$. In other words, we take advantage of the subsequent commuting diagram (6) (displayed with a slight abuse of notation):

$$\begin{array}{ccc} \mathrm{H(n)} & \xrightarrow{\quad\mathscr{A}\quad} & \mathrm{H(n)} \\ \Psi \uparrow & & \downarrow \Psi^{-1} \\ \mathrm{M}_{\mathbb{R}}(n) & \xrightarrow[\Psi \circ \mathscr{A} \circ \Psi^{-1}]{} & \mathrm{M}_{\mathbb{R}}(n) \end{array} \tag{6}$$

The operation $\mathscr{A}$ is formulated in the language of the matrix-calculus of Table 1. It is the concatenated operation $\Psi \circ \mathscr{A} \circ \Psi^{-1}$ that processes matrices from $\mathrm{M}_{\mathbb{R}}(n)$, or, even more interestingly, from subsets (classes) K of $\mathrm{M}_{\mathbb{R}}(n)$, such as $K \in \{\mathrm{Gl(n)}, \mathrm{O(n)}, \mathrm{SO(n)}\}$. Note that $\Psi(QMQ^{\top}) = Q\Psi(M)Q^{\top}$ for any orthogonal matrix Q, and likewise for Ψ^{-1}. Hence, $\Psi \circ \mathscr{A} \circ \Psi^{-1}$ is rotational invariant whenever $\mathscr{A}$ is.

Remark 1 The classes K we are considering in this section are stable under matrix multiplication. In view of the reasoning above it is close at hand to define a multiplication $\cdot_K$ for Hermitian matrices induced by a class K as follows: For two Hermitian matrices H_1, H_2 we define

$$H_1 \cdot_K H_2 := \Psi\left(\Psi^{-1}(H_1) \cdot \Psi^{-1}(H_2)\right) .$$

These type of products are distributive but not commutative. But what is worse, in general, $H \cdot_K H \neq H^2$. Hence, we refrain from considering these products.

Now, we turn to a rather specific class of matrices, the group O(n) of orthogonal matrices, which forms a subset of measure zero in $\mathrm{M}_{\mathbb{R}}(n)$.

4.1 Embedding SO(n) in H(n)

As an example, we consider the group SO(n), the subgroup of orthonormal $n \times n$-matrices with determinant ± 1. Clearly, the mapping Ψ provides a one-to-one correspondence between the sets SO(n) and $\Psi(\mathrm{SO(n)})$, hence a reasonable processing strategy is as follows: First we embed the subset SO(n) of $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ into H(n) by means of Ψ. Then the (image-)processing is performed (that is, applying $\mathscr{A}$) exploiting the rich algebraic structure of H(n), followed by the canonical mapping Ψ^{-1} back to $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ as indicated in diagram (6). However, if we set $\tilde{\mathscr{A}} := \Psi^{-1} \circ \mathscr{A} \circ \Psi$ and consider $R \in \mathrm{SO(n)}$ then it might very well be that $\tilde{\mathscr{A}}(R) \notin K$. Hence, we need to find the $\|\cdot\|_{\mathrm{F}}$-best approximation $\tilde{R} \in K$ to $\tilde{\mathscr{A}}(R)$, that is

$$\|\tilde{\mathscr{A}}(R) - \tilde{R}\|_{\mathrm{F}}^2 \longrightarrow \min\,, \tag{7}$$

where $\|\cdot\|_{\mathrm{F}}$ stands for the Frobenius norm for matrices. Due to the invariance properties of the Frobenius norm $\|\cdot\|_{\mathrm{F}}$ this establishes a rotational invariant projection Pr_K from $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ back to K, see diagram (8).

$$\begin{array}{ccccc} \mathrm{H(n)} & \xRightarrow{\quad\mathscr{A}\quad} & \mathrm{H(n)} & & \\ \Psi \uparrow & & \downarrow \Psi^{-1} & & \\ \mathrm{M}_{\mathbb{R}}(\mathrm{n}) & \xRightarrow[\Psi \circ \mathscr{A} \circ \Psi^{-1}]{} & \mathrm{M}_{\mathbb{R}}(\mathrm{n}) & \xRightarrow[\mathrm{Pr}_K]{} & K \\ \cup & & & & \\ K & & & & \end{array} \tag{8}$$

In the literature minimization problems such as (7) are referred to as Procrustes problems (see [22, 23]). In a slightly more general case $K = \mathrm{O(n)}$ we have the so-called orthogonal Procrustes problem of finding the orthogonal matrix $\tilde{R}$ closest to the given matrix $\tilde{\mathscr{A}}(R)$. Fortunately, a solution is the orthogonal factor in the polar decomposition of $\tilde{\mathscr{A}}(R)$. The polar decomposition as we need it in this context states that every real square invertible matrix A can be uniquely written as

$$A = O \cdot P$$

with positive definite $P = \sqrt{A^{\top}A}$ and orthogonal $O = A\,P^{-1}$ (refer to [26]). Therefore, we are in the position to give an explicit closed form of this solution for invertible $\tilde{\mathscr{A}}(R)$:

$$\tilde{R} = Pr_{O(n)}(\tilde{\mathscr{A}}(R)) = \tilde{\mathscr{A}}(R)\,\left(\tilde{\mathscr{A}}(R)^{\top}\tilde{\mathscr{A}}(R)\right)^{-1/2}$$

This indeed establishes a projection $Pr_{O(n)}$ from $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ back to O(n). Furthermore, it follows that $\tilde{R} = Pr_{O(n)}(\tilde{\mathscr{A}}(R)) \in \mathrm{SO(n)}$ if $\det(\tilde{\mathscr{A}}(R)) > 0$.

Remark 2 It is not too daring to assume that other sets of matrices, that are interesting subsets of $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ and might possess some suitable group and/or topological structure can be embedded into H(n) without major difficulties. However, to find an efficient back-projection by formula or algorithm is a severe hindrance to the application of the above mentioned processing principle. This topic is the subject of ongoing research.

4.2 Embedding Gl(n) in H(n)

Gl(n) is a proper subset of $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$, $\mathrm{Gl}(\mathrm{n}) \subset \mathrm{M}_{\mathbb{R}}(\mathrm{n})$, and it can very well be that $\Psi^{-1}(H) \notin \mathrm{Gl}(\mathrm{n})$. But this poses no problem from the numerical point of view: Gl(n) is topologically dense in $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ and the set $\mathrm{M}_{\mathbb{R}}(\mathrm{n}) \setminus \mathrm{Gl}(\mathrm{n})$ of non-invertible matrices has zero (Lebesgue-)measure in $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ since $\mathrm{M}_{\mathbb{R}}(\mathrm{n}) \setminus \mathrm{Gl}(\mathrm{n}) = \{M \in \mathrm{M}_{\mathbb{R}}(\mathrm{n}) \mid \det(M) = 0\}$ together with the analytic properties of the determinant. This means that the probability of obtaining numerically a non-invertible matrix after applying Ψ^{-1} to the processed Hermitian matrix is zero. Hence, in the case of $K = \mathrm{Gl}(3)$, a back-projection is numerically in principle not necessary. However, the zero-probability-event has to be taken into account algorithmically (for example, by means of a very small stochastic perturbation of the processed data).

5 Approximate psup, pinf and Averaging

The appearance of a projection Pr_K in general makes any operation, be that averaging, taking the supremum/infimum, or a step in a numerical scheme, an approximate operation. But this is the price one has to pay for relying on the very efficient linear mappings Ψ and Ψ^{-1}. It would not come as a surprise if in the case of a specific class of matrices a more sophisticated, non-linear mapping would make a final back-projection obsolete. Therefore, for instance, taking the p-supremum $\mathrm{psup}^{\mathrm{O(n)}}(M_1, M_2)$ of two matrices $M_1, M_2 \in \mathrm{O(n)}$ amounts to calculate

$$\mathrm{psup}^{\mathrm{O(n)}}(M_1, M_2) = Pr_{\mathrm{O(n)}}\left(\Psi^{-1}\left(\mathrm{psup}^{\mathrm{H(n)}}(\Psi(M_1), \Psi(M_2))\right)\right) .$$

Here, we used the notation psup^K with $K = \mathrm{O(n)}$ to indicate that the supremum is an element of the set K due to the applied back projection Pr_K. The notion pinf^K will be used likewise. For the sake of brevity, we will use exchangeably $\mathrm{psup} = \mathrm{psup}^{\mathrm{H(n)}}$ respectively $\mathrm{pinf} = \mathrm{pinf}^{\mathrm{H(n)}}$. Before we turn to further generalizations and concrete examples, let us state some properties of the psup- and pinf-operations.

Proposition 2 *Let $A, B \in K$ with $K \in \{H(n), M_{\mathbb{R}}(n), Gl(n)\}$. Then the following equalities hold:*

1. $\mathrm{psup}^K(tA, tB) = t\mathrm{psup}^K(A, B)$ *if* $t \geq 0$;
2. $\mathrm{pinf}^K(tA, tB) = t\mathrm{pinf}^K(A, B)$ *if* $t \geq 0$;
3. $\mathrm{psup}^K(A, tA) = \begin{cases} A & \text{if} \quad 0 \leq t \leq 1; \\ tA & \text{if} \quad t > 1; \end{cases}$
4. $\mathrm{pinf}^K(A, tA) = \begin{cases} tA & \text{if} \quad 0 \leq t \leq 1; \\ A & \text{if} \quad t > 1; \end{cases}$
5. $\mathrm{pinf}^K(A, B) + \mathrm{psup}^K(A, B) = A + B$.

Proof This follows directly from the corresponding properties of $\mathrm{psup}^{\mathrm{H(n)}}$ respectively $\mathrm{pinf}^{\mathrm{H(n)}}$ and the linearity of Ψ and Ψ^{-1}. Note that an analog statement for $K = \mathrm{O(n)}, \mathrm{SO(n)}$ would not make sense. However, we may remark that

$$Pr_{\mathrm{O(n)}}(tA) = Pr_{\mathrm{O(n)}}(A) \qquad \text{for every } t \in \mathbb{R}, \text{ and } A \in \mathrm{M}_{\mathbb{R}}(\mathrm{n}).$$

Computationally very useful is the fact that the p-supremum and the p-infimum are related in the expected way

$$\mathrm{pinf}^K(A, B) = -\mathrm{psup}^K(-A, -B).$$

This idea of an approximate operation extends naturally to more elaborate operations that might even depend on several matrix arguments. We will demonstrate this by calculating an approximate p-supremum of more than two orthogonal or simply invertible matrices. To this end we propose an algorithm to obtain an approximate p-supremum/p-infimum of finitely many Hermitian matrices.

5.1 The p-Supremum/p-Infimum of Finitely Many Hermitian Matrices

The Loewner order is only a partial order (see [27]) and hence

$$\mathrm{psup}(H_1, \mathrm{psup}(H_2, H_3)) \neq \mathrm{psup}(H_2, \mathrm{psup}(H_3, H_1)) \neq \mathrm{psup}(H_3, \mathrm{psup}(H_2, H_1))$$

or with the appropriate definition of $A \vee B := \mathrm{psup}(A, B)$ yields

$$(H_1 \vee (H_2 \vee H_3)) \neq (H_2 \vee (H_3 \vee H_1)) \neq (H_3 \vee (H_2 \vee H_1)).$$

Finding even an approximation for $(H_1 \vee H_2 \vee H_3)$ is not an easy task (see [11] for a possible approach). The method in [11] and [13] relies on finding the smallest hyperball enclosing a cloud of points in a relatively high-dimensional space $\mathbb{R}^d$, which entails approximations as well in addition to high computational costs.

Therefore, we propose another iterative procedure to find an approximate supremum of a finite number of Hermitian matrices $H_1, H_2, \ldots, H_m$ of the same size. The "approximate supremum" will not be in general a least upper bound, but an upper bound with a trace as small as possible. The details for the iterative step are as follows.

Iteration

Let us denote by $A_m := \frac{1}{m}\sum_{j=1}^{m} H_i$ the arithmetic mean of the set of matrices $\{H_1, \ldots, H_m\} \subset \mathrm{H(n)}$, and by I_n the $n \times n$-unit matrix.

1. Due to its spectral properties the matrix $A_m - t \cdot I_n = \frac{1}{m}\sum_{j=1}^{m} H_i - t \cdot I_n$ will become an upper bound of $\{H_1, \ldots, H_m\}$ for sufficiently small, potentially negative $t \in \mathbb{R}$. Now, we seek the largest $t \in \mathbb{R}$ of this kind.
2. To this end we calculate for any $i = 1, \ldots, m$ the solutions of the equation

$$\det(A_m - H_i - tI_n) = 0\,,$$

   that is, the eigenvalues $t_{i,j}$ of $A_m - H_i$ with $i = 1, \ldots, m$ and, counting them according to their multiplicity, $j = 1, \ldots, n$.
3. Arranging the eigenvalues $t_{i,1}, \ldots, t_{i,n}$ in descending order for fixed i one can assume $\min_{j=1,\ldots,n}(t_{i,j}) = t_{i,n}$. Next we set

$$i^{(1)} := \operatorname*{argmin}_{i=1,\ldots,m} (t_{1,n}, t_{2,n}, \ldots, t_{i,n}, \ldots, t_{m,n}),$$

   store the associated matrix $H_{i^{(1)}}$, and form the "first generation subset"

$$\{H_1^{(1)}, \ldots, H_{m-1}^{(1)}\} := \{H_1, \ldots, H_m\} \setminus \{H_{i^{(1)}}\}$$

   with $m-1$ elements. Roughly speaking, $H_{i^{(1)}}$ is an outlier in the sense that is the first matrix to fail being dominated by $A_m - H_i - tI_n$ when t increases from $-\infty$ to ∞.
4. Now, we apply the above reckoning to the set $\{H_1^{(1)}, \ldots, H_{m-1}^{(1)}\}$,
   a. single out an outlier $H_i^{(2)}$,
   b. and reduce the set $\{H_1^{(1)}, \ldots, H_{m-1}^{(1)}\}$ to the second generation set given by $\{H_1^{(2)}, \ldots, H_{m-2}^{(2)}\} \subset \{H_1, \ldots, H_m\}$ with only $m-2$ elements.
5. The steps 1 to 3 are repeated until one arrives at a two-element subset given by $\{H_1^{(m-2)}, H_2^{(m-2)}\}$ and a sequence of outliers $H_{i^{(1)}}, H_{i^{(2)}}, \ldots, H_{i^{(m-2)}}$. Combining those results in the sequence $H_{i^{(1)}}, H_{i^{(2)}}, \ldots, H_{i^{(m-2)}}, H_1^{(m-2)}, H_2^{(m-2)}$, gives rise to the following construction of an approximate supremum $\widehat{\mathrm{psup}}$ of $\{H_1, \ldots, H_m\}$.

6. The approximate supremum $\widehat{\text{psup}}$ is defined by

$$\widehat{\text{psup}}(H_1,\ldots,H_m) := H_{i^{(1)}} \vee \left(H_{i^{(2)}} \vee \ldots \vee \left(H_{i^{(m-2)}} \vee (H_1^{(m-2)} \vee H_2^{(m-2)})\right)\ldots\right).$$

For the sake of simplicity the notion $\text{psup}(H_1,\ldots,H_n)$ is used throughout this section instead of $\widehat{\text{psup}}(H_1,\ldots,H_n)$ while always being aware of the approximate character of the later.

Remark 3 The choice of the arithmetic mean for A_m as the matrix to be translated by $-t\cdot I_n$ is motivated by its simple computability and its appearance in the formulas (2) and (3) for the p-supremum/p-infimum of two Hermitian matrices.

Remark 4 An analog construction is feasible for an approximate $\text{pinf}(H_1,\ldots,H_m)$ or by setting $\text{pinf}(H_1,\ldots,H_m) := -\text{psup}(-H_1,\ldots,-H_m)$ (in abbreviated notation).

Example 3 Let us consider the Hermitian matrices $H_1 = \begin{pmatrix} 3 & 1+\mathrm{i} & \mathrm{i} \\ 1-\mathrm{i} & 1 & 0 \\ -\mathrm{i} & 0 & 1 \end{pmatrix}$, the matrix $H_2 = \begin{pmatrix} 1 & 3-\mathrm{i} & 4 \\ 3+\mathrm{i} & -2 & -6+\mathrm{i} \\ 4 & -6-\mathrm{i} & 5 \end{pmatrix}$, $H_3 = \begin{pmatrix} 3 & 2-\mathrm{i} & 4+\mathrm{i} \\ 2+\mathrm{i} & -1 & 2\,\mathrm{i} \\ 4-\mathrm{i} & -2\,\mathrm{i} & 6 \end{pmatrix}$. The above procedure yields

$$\begin{aligned} &\text{psup}(H_1,H_2,H_3) \\ &= \begin{pmatrix} 4.8127 & 1.1557+0.0228\,\mathrm{i} & 2.8865+0.9776\,\mathrm{i} \\ 1.1557-0.0228\,\mathrm{i} & 2.7750 & -1.7909+1.5305\,\mathrm{i} \\ 2.8865-0.9776\,\mathrm{i} & -1.7909-1.5305\,\mathrm{i} & 9.0033 \end{pmatrix}, \end{aligned}$$

and

$$\begin{aligned} &\text{pinf}(H_1,H_2,H_3) \\ &= \begin{pmatrix} -0.2850 & 3.6102-0.7254\,\mathrm{i} & 2.1422+0.1922\,\mathrm{i} \\ 3.6102+0.7254\,\mathrm{i} & -4.4186 & -2.4794+0.3596\,\mathrm{i} \\ 2.1422-0.1922\,\mathrm{i} & -2.4794-0.3596\,\mathrm{i} & -1.1939 \end{pmatrix}. \end{aligned}$$

An elementary calculation gives for the eigenvalues

$$\lambda_i(\text{psup}(H_1,H_2,H_3)-H_k) \overset{\approx}{\geq} 0 \text{ and } \lambda_i(H_k-\text{pinf}(H_1,H_2,H_3)) \overset{\geq}{\approx} 0$$

for $i,k = 1\ldots3$ within a 4-digit numerical accuracy. This indicates that the calculated quantities are optimal trace upper respectively lower bounds of H_1, H_2, and H_3.

Since even a 2×2-Hermitian matrix has 4 degrees of freedom our possibilities for a visualization of such matrix fields are very limited, at best. The same holds true for the aforementioned classes of matrices. Therefore, we will focus more on the theoretical aspects providing a proof-of-concept rather than the evaluation of experiments. Clearly, the visualization of multivariate data is a very active research field in its own right. As future work, we will investigate whether glyphs might be appropriate for the visualization (see for example [19, 40, 42]).

6 Application 1: Elementary Image Processing for Invertible Real Matrices

In this section, we prepare the conceptual basis for elementary image processing for fields of $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$ matrices. For the sake of simplicity, we restrict ourselves to real 2×2-matrices, the extension to higher dimensions is straightforward.

6.1 p-Supremum and p-Infimum in Gl(2)

For the sake of simplicity, we restrict our attention to matrices $A_i \in \mathrm{Gl}(2)$ $(i = 1, 2, 3)$, which we are going to visualize as a pair of their column vectors each. Even more, since a multiplication is available in Gl(2) and the mapping Ψ is linear, we focus solely on the calculation of $\mathrm{psup}(A_1, A_2)$, $\mathrm{psup}(A_1, A_2, A_3)$ and the corresponding p-infima. We begin with an example where our intuition suggests and Proposition 2 provides a certain answer.

Example 4 Let $A_1 = \begin{pmatrix} 1 & -1 \\ 1 & 2 \end{pmatrix}$ and $A_2 = 2 \cdot A_1 = \begin{pmatrix} 2 & -2 \\ 2 & 4 \end{pmatrix}$. We obtain

$$\mathrm{pinf}(A_1, A_2) = \begin{pmatrix} 1 & -1 \\ 1 & 2 \end{pmatrix} \text{ and } \mathrm{psup}(A_1, A_2) = 2 \cdot A_1 = \begin{pmatrix} 2 & -2 \\ 2 & 4 \end{pmatrix} \text{ as expected.}$$

In the sequel, the matrices are visualized by means of their associated column vectors viewed as a pair of coordinate axes or dyads.

In this example, the two 2×2-matrices have one column in common, that is, the corresponding coordinate systems share one axis.

Example 5 Let $A_1 = \begin{pmatrix} 1 & 1 \\ -1 & 2 \end{pmatrix}$ and $A_2 = \begin{pmatrix} 1 & 3 \\ -1 & -2 \end{pmatrix}$ then

$$\mathrm{pinf}(A_1, A_2) = \begin{pmatrix} 1 - 1/6\sqrt{6} & 2 + 1/3\sqrt{6} \\ -1 & -5/6\sqrt{6} \end{pmatrix},$$

$$\mathrm{psup}(A_1, A_2) = \begin{pmatrix} 1 + 1/6\sqrt{6} & 2 - 1/3\sqrt{6} \\ -1 & 5/6\sqrt{6} \end{pmatrix}.$$

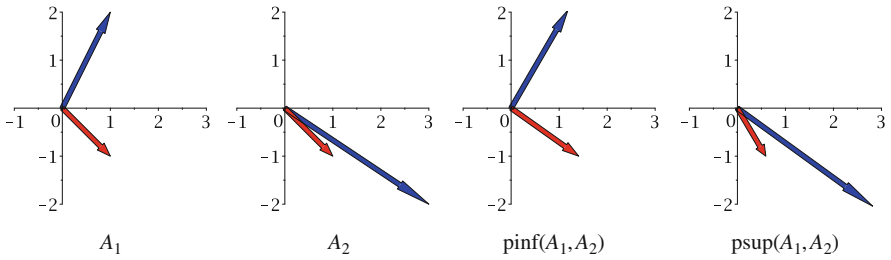


Fig. 1 The matrices A_1, A_2, psup(A_1, A_2), and pinf(A_1, A_2) depicted as dyads of their corresponding column vectors. A certain similarity between A_1 and psup(A_1, A_2) on the one hand side, and A_2 and pinf(A_1, A_2) on the other is noticeable

In Fig. 1 these matrices are displayed as dyads of their column vectors. The column vectors of the matrices are colored in red (1st column) and in blue (2nd column), and the matrices differ only in the latter one. We expect that the first column of both pinf- and psup-matrix are, if not identical, at least very similar. As one can see, this can be observed in Fig. 1.

In the next example, we report on the results with two randomly chosen invertible matrices.

Example 6 For the randomly chosen matrices
$A_1 = \begin{pmatrix} 1.15732069 & -0.289221478 \\ 3.47018765 & -1.07445407 \end{pmatrix}$ and $A_2 = \begin{pmatrix} 3.36404711 & -2.75914955 \\ -0.257407447 & -4.21393882 \end{pmatrix}$
a short calculation gives

$$\text{pinf}(A_1, A_2) = \begin{pmatrix} 0.341012885 & -1.66329185 \\ 1.39645198 & -4.864966040 \end{pmatrix},$$

$$\text{psup}(A_1, A_2) = \begin{pmatrix} 4.18035492 & -1.38507918 \\ 1.81632823 & -0.423426855 \end{pmatrix}.$$

They are depicted in Fig. 2 employing the same coloring as before. We note that the last column of the matrices (accidently) have the same (negative) sign pattern. As one might expect the negativity of these components is transferred to the second columns of the corresponding pinf- and psup-matrices.

Next, an example where we decompose a matrix A into its positive A^+ and negative part A^- concludes this section.

Example 7 Let $A = A_1$ from the previous example,

$$A = \begin{pmatrix} 1.15732069 & -0.289221478 \\ 3.47018765 & -1.07445407 \end{pmatrix},$$

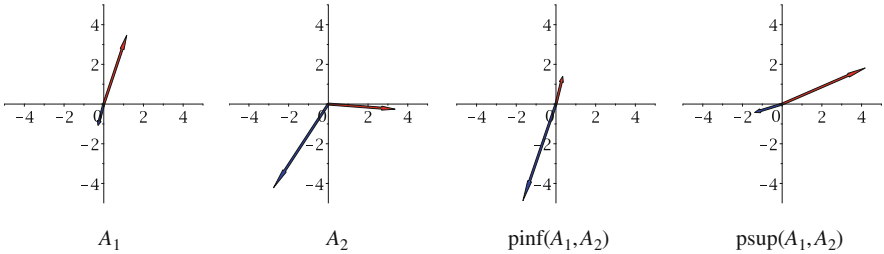


Fig. 2 Dyads representing the column vectors of randomly chosen matrices A_1, A_2, psup(A_1, A_2), and pinf(A_1, A_2). The psup(A_1, A_2) seems to favor the "positive" column vectors of A_1, A_2, while pinf(A_1, A_2) tends more towards the "negative" column vectors of A_1, A_2

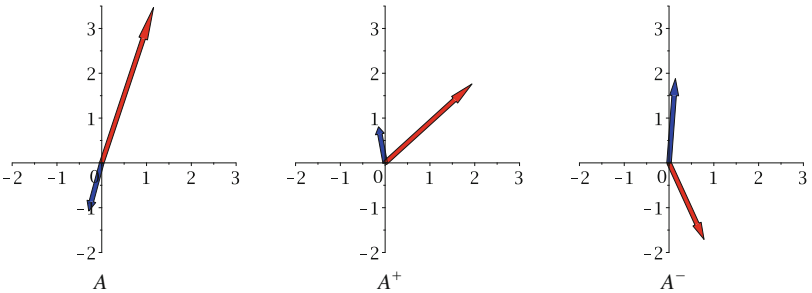


Fig. 3 Matrices A, A^+, and A^- depicted as dyads of their corresponding column vectors

then one obtains

$$A^+ := \text{psup}(A, 0) = \begin{pmatrix} 1.93888944 & -0.146827133 \\ 1.76168695 & 0.805899265 \end{pmatrix},$$

and

$$A^- := -\text{pinf}(A, 0) = \begin{pmatrix} 0.781568745 & 0.142394345 \\ -1.7085007 & 1.88035334 \end{pmatrix}.$$

Note that indeed $A = A^+ - A^-$ holds, as expected. In A^+ the column vector of A with positive components is dominant. In A^- the column vector of A with negative components is emphasized (note the minus sign in the definition of A^-) (Fig. 3). This is in agreement with the findings of the previous examples, refer to Figs. 1 and 2.

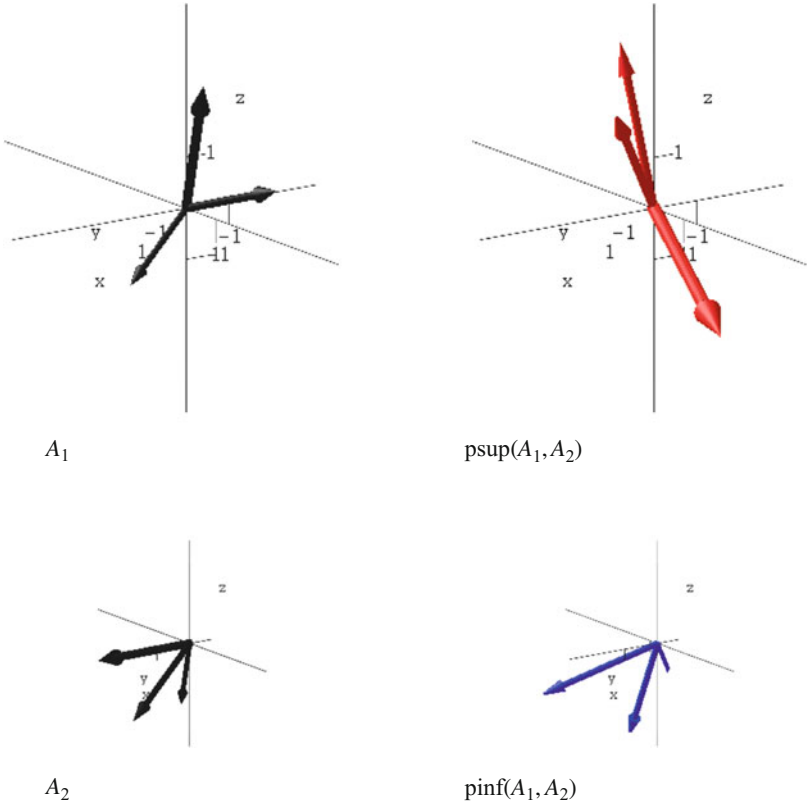
We end this section with an example of p-infimum and p-supremum of two matrices from Gl(3).

Example 8 With the vectors $\mathbf{a} = (1,2,3)^{\mathrm{T}}$, $\mathbf{b} = (2,1,-1)^{\mathrm{T}}$, and $\mathbf{c} = (0,3,1)^{\mathrm{T}}$, we construct the two matrices $A_1 = [\mathbf{a},\mathbf{b},\mathbf{c}]$ and $A_2 = [-1\mathbf{a}, 2\mathbf{b}, -2\mathbf{c}]$ which are in Gl(3), since $\det(A_1) = 18$ and $\det(A_2) = 72$. We obtain the p-supremum and p-infimum of A_1 and A_2 as

$$\mathrm{psup}(A_1,A_2) = \begin{pmatrix} 1.843872858 & 1.827596576 & 0.328372983 \\ 1.359688108 & 4.814268107 & -0.6572189155 \\ 2.422633919 & -1.187442382 & 3.188837471 \end{pmatrix},$$

$$\mathrm{pinf}(A_1,A_2) = \begin{pmatrix} -1.843872858 & 4.172403458 & -0.328372869 \\ -1.359688142 & -1.814268107 & -2.342781089 \\ -2.422634033 & -1.812557613 & -4.188837471 \end{pmatrix},$$

and no back projection is necessary. The matrices are depicted in Fig. 4 as triads spanned by each of their column vectors. The example does not lend itself to



A_1 $\mathrm{psup}(A_1,A_2)$

A_2 $\mathrm{pinf}(A_1,A_2)$

Fig. 4 Matrices A_1, A_2, $\mathrm{psup}(A_1,A_2)$, and $\mathrm{pinf}(A_1,A_2)$ depicted as triads of their corresponding column vectors. In order to account for perspective distortions, the thickness of the arrows mirrors their length as well

an immediate geometric interpretation. However, $A_1 + A_2 = \mathrm{psup}(A_1, A_2) + \mathrm{pinf}(A_1, A_2)$ still holds.

The next section deals with a subset of Gl(n) that is not a vector space but a multiplicative group and as such offers some topological intricacies: O(n) respectively SO(n).

7 Application 2 : Elementary Image Processing for Orthogonal Matrices

Now, we turn to a rather specific class of matrices, the group O(n) of orthogonal matrices, and among those we focus on the (sub-)group SO(n) of special orthogonal matrices, which form subsets of measure zero in $\mathrm{M}_{\mathbb{R}}(\mathrm{n})$. Therefore, a back-projection onto O(n) respectively SO(n) will be in order. The group SO(3) is by far the most important one for applications. Therefore, we restrict ourselves to this case and deal with other instances of O(n) and SO(n) where $n = 2$ and $n > 3$ only in passing, although the processing principle carries over to the higher dimensional setting. Averaging and inf-/sup-operations lie in one form or the other at the heart of most numerical algorithms not only for image processing. In the design of algorithms for SO(n) even linear averaging poses a difficulty since SO(n) is a multiplicative group but not a vector space.

7.1 A Glance at SO(3) and the Cayley Transform

SO(3) is the group of rotations of the three dimensional space $\mathbb{R}^3$ and as such an integral part of mathematics and its applications in engineering, computer science, and physics. A typical element of SO(3) is the matrix

$$R(u,v,w,\theta) := \begin{pmatrix} u^2(1-c)+c & uv(1-c)-ws & uw(1-c)+vs \\ uv(1-c)+ws & v^2(1-c)+c & vw(1-c)-us \\ uw(1-c)-vs & vw(1-c)+us & w^2(1-c)+c \end{pmatrix}.$$

with $c = \cos(\theta)$ and $s = \sin(\theta)$. Here, θ stands for the angle of rotation, $(u,v,w)^\top$ denotes an unit vector determining the oriented axis of rotation. Observing

$$R(u,v,w,\theta) = R(-u,-v,-w,-\theta)$$

we can characterize/parametrize SO(3) as follows:

$$\mathrm{SO}(3) = \{R(u,v,w,\theta) \mid (u,v,w)^\top \in S^2_+,\ \theta \in [-\pi,\pi)\},$$

where the "hemisphere" S^2_+ may be given by the disjoint union

$$S^2_+ := \{(x,y,z)^\top \in S^2 \,|\, z > 0\} \cup \{(x,y,0)^\top \in S^2 \,|\, y > 0\} \cup \{(1,0,0)^\top\}\,.$$

The Cayley transform

$$\mathrm{cay}(z) = \frac{1-z}{1+z}$$

with $z \in \mathbb{C} \setminus \{-1\}$ is a special Moebius transform well-known from the theory of conformal mappings and hyperbolic geometry (see, for example, [39]). It is its own inverse and its domain of definition can be extended to diagonalizable matrices A whose spectrum $\mathrm{SPEC(A)}$ do not contain -1. Now, both (special) orthogonal and skew-symmetric matrices are normal and hence diagonalizable over $\mathbb{C}$ (see [20, 47]). It is not difficult to see that cay is a one-to-one mapping between the set $\mathrm{SKEW}(n) = \{M \in \mathrm{M}_{\mathbb{R}}(\mathrm{n}) \,|\, M^\top = -M\}$ of skew-symmetric matrices and almost all of SO(n):

$$\mathrm{cay} : \{M \in \mathrm{SO(n)} \,|\, -1 \notin \mathrm{SPEC}(n)\} \longrightarrow \mathrm{SKEW}(n)\,.$$

Precisely, it is given by

$$\mathrm{cay}(M) = (I-M)\,(I+M)^{-1}\,,$$

where I denotes the identity and assuming -1 is not an eigenvalue of M. Especially for $n = 3$ we have

$$\mathrm{SKEW}(3) = \left\{ \begin{pmatrix} 0 & x & -y \\ -x & 0 & z \\ y & -z & 0 \end{pmatrix} \middle| \, x,y,z \in \mathbb{R} \right\}$$

and an elementary but tedious calculation gives

$$\mathrm{cay}(R(u,v,w,\theta)) = \frac{\sin(\theta)}{1+\cos\theta} \cdot \begin{pmatrix} 0 & w & -v \\ -w & 0 & u \\ v & -u & 0 \end{pmatrix} =: \tan\left(\frac{\theta}{2}\right) N(u,v,w)\,, \qquad (9)$$

with $\theta \in (-\pi, \pi)$. This mirrors the well-known connection between the Lie group SO(n) and the corresponding Lie algebra $\mathrm{SKEW}(n)$ for $n = 3$, although usually established by means of the exponential map (see [2]). To the best of our knowledge, the Caley transform has been employed in image processing to code an SO(3) field as a color image for the first time in [31]. There, this color image has been subjected to mathematical morphological techniques for color images as in [8–10] and has

been pursued in [30, 44, 45]. Now, we observe that the mapping

$$\begin{array}{ccc} (\mathrm{SKEW}(3), \|\cdot\|_{\mathrm{F}}) & \longleftrightarrow & (\mathbb{R}^3, \|\cdot\|_2) \\ \begin{pmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{pmatrix} & \longmapsto & \sqrt{2} \cdot \begin{pmatrix} x \\ y \\ z \end{pmatrix} \end{array}$$

is an isometry and hence $\|N(x,y,z)\|_{\mathrm{F}} = \sqrt{2}\|(x,y,z)\|_2$. In view of these facts, it becomes apparent how to extract from a given skew-symmetric matrix $N(x,y,z)$ both axis vector and angle of rotation: First normalize

$$N(\tilde{x},\tilde{y},\tilde{z}) = \frac{\sqrt{2}}{\|N(x,y,z)\|_{\mathrm{F}}} \cdot N(x,y,z)$$

such that in addition $(\tilde{x},\tilde{y},\tilde{z}) \in S^2_+$, if needed might be, by introducing $1 = (-1)(-1)$ to ensure $\tilde{z} > 0$.

Then, we have

$$N(x,y,z) = \frac{\|N(x,y,z)\|_{\mathrm{F}}}{\sqrt{2}} N(\tilde{x},\tilde{y},\tilde{z})\,.$$

Considering (9) we conclude that

$$\phi = \pm 2 \arctan\left(\frac{\|N(x,y,z)\|_{\mathrm{F}}}{\sqrt{2}}\right)$$

with the sign $\pm$ depending on $\mathrm{sign}(z)$.

This gives rise to an invertible mapping

$$\begin{aligned} \kappa : \mathrm{SKEW}(3) &\longrightarrow \left\{ \phi \cdot \begin{pmatrix} 0 & \tilde{z} & -\tilde{y} \\ -\tilde{z} & 0 & \tilde{x} \\ \tilde{y} & -\tilde{x} & 0 \end{pmatrix} \,\middle|\, \phi \in (-\pi,\pi)\,,\ (\tilde{x},\tilde{y},\tilde{z}) \in S^2_+ \right\} \\ &= \left\{ \begin{pmatrix} 0 & z & -y \\ -z & 0 & x \\ y & -x & 0 \end{pmatrix} \,\middle|\, \begin{array}{l} (x,y,z) = \phi \cdot (\tilde{x},\tilde{y},\tilde{z})\,, \\ (\phi,(\tilde{x},\tilde{y},\tilde{z})) \in (-\pi,\pi) \times S^2_+ \end{array} \right\}. \end{aligned}$$

However, one readily establishes a one-to-one geometric equivalence $(-\pi,\pi) \times S^2_+ \equiv B^3(0;\pi)$, where $B^3(0;\pi)$ denotes the open ball in $\mathbb{R}^3$ with center 0 and radius $r = \pi$. $B^3(0;\pi)$ is a convex and balanced set (B balanced if and only if $v \in B \Leftrightarrow -v \in B$), and so is $\kappa\big(\mathrm{SKEW}(3)\big)$. Within this set linear averaging and interpolation is feasible. For the sake of brevity, we introduce the notion of a modified Cayley transform $\mathrm{CAY} = \mathrm{cay} \circ \kappa$ with inverse $\mathrm{CAY}^{-1} = \kappa^{-1} \circ \mathrm{cay}^{-1} = \kappa^{-1} \circ \mathrm{cay}$.

Remark 5 It is worth mentioning that the reasoning above can be extended to higher dimensions without major difficulties. At this point the advantage of the Cayley transform over the exponential/logarithmic connection starts to show: the later can be numerically challenging for $n \gg 3$, [38], while the former one requires an inversion only and it is its own inverse.
In the next section, we will employ the (modified) Cayley transform in the design of averaging processes for SO(3).

7.2 Linear Averaging in SO(3)

The problem of averaging rotations has received some attention in the literature, see, for instance, [17, 21]. There specific algebraic properties of SO(3) are exploited, and the technique are geared to the application at hand. A more general approach is considered in [7]. Here, we proceed along a different path.

Averaging via H(3) Regardless of the algebraic structure of SO(3) the mapping Ψ provides an embedding into the real vector space H(3) where linear averaging is feasible. Let $\mathscr{A}_{H(3)}$ denote an averaging operation in H(3), for instance, the convex combination $\mathscr{A}_{H(3)}(H_1,\dots,H_k) = \sum_{\nu=1}^{k} t_\nu H_\nu$ with $t_\nu \geq 0$ for $\nu = 1,\dots,k$ and $\sum_\nu t_\nu = 1$. Then, in view of diagram (8), we can define an operator for elements $A_1,\dots,A_k \in \mathrm{SO}(3)$ by

$$\begin{aligned}\mathscr{A}_{SO(3)}(A_1,\dots,A_k) &= Pr_{SO(3)}\left(\Psi^{-1}\left(\mathscr{A}_{H(3)}(\Psi(A_1,\dots,\Psi(A_k))\right)\right) \\ &= Pr_{SO(3)}\left(\Psi^{-1}\left(\sum_{\nu=1}^{k} t_\nu \Psi(A_\nu),\right)\right).\end{aligned} \tag{10}$$

However, Ψ is $\mathbb{R}$-linear and as such commutes with any averaging operator. Hence,

$$\mathscr{A}_{SO(3)}(A_1,\dots,A_k) = Pr_{SO(3)}\left(\sum_{\nu=1}^{k} t_\nu A_\nu\right)$$

which exactly boils down to the approach proposed by Moakher [37] for SO(3). It provides a rather efficient and theoretically justifiable way to average linearly in the multiplicative Lie group SO(3). An example will demonstrate the effect of the method.

Example 9 Let $A_1 = R((1,2,3);\pi/7)$ and $A_2 = R((3,1,-2);6\pi/7)$. We consider the convex combination $\mathscr{A}_{SO(3)}(A_1,A_2) = (1-\lambda)A_1 + \lambda A_2$ with averaging based on H(3) like it is done in Moakher's approach [37]. The sequence of images below depicts the intermediate states of the interpolation when λ varies from 0 to 1 in steps of 0.125. The visualization with gyros indicates axis and angle of rotation for each of the interpolation matrices. The interpolation does not seem to be as uniform as

one might expect. The transition from the angle $1/7\pi$ to $6/7\pi$ is rather abrupt and the last three steps of the interpolation do result in already quite similar, although non identical, matrices.

Averaging via Cayley Transform Nevertheless, we propose an alternative linear averaging that takes advantage of the Cayley transform. In the previous section the mapping CAY with

$$\mathrm{CAY}(R(u,v,w,\theta)) = \theta \cdot \begin{pmatrix} 0 & w & -v \\ -w & 0 & u \\ v & -u & 0 \end{pmatrix}$$

has been introduced. Let $\mathscr{A}_{Skews(3)}$ denote an averaging operation in $\mathrm{SKEW}(3)$, that is, $\mathscr{A}_{\mathrm{SKEW}(3)}(S_1,\ldots,S_k) = \sum_{v=1}^{k} t_v S_v$. We set

$$\mathscr{A}^{c}_{SO(3)}(R_1,\ldots,R_k) = \mathrm{CAY}^{-1}\left(\mathscr{A}_{\mathrm{SKEW}(3)}\big(\mathrm{CAY}(R_1),\ldots,\mathrm{CAY}(R_k)\big)\right) . \quad (11)$$

Here, the superscript c in $\mathscr{A}^{c}_{SO(3)}$ indicates the dependence of the averaging on the Cayley transform rather than on the embedding into H(3) as in (10). We demonstrate the feasibility of the approach (see formula (11)) in the next

Example 10 Let $A_1 = R((1,2,3);\pi/7)$ and $A_2 = R((3,1,-2);6\pi/7)$, we consider a convex combination $\mathscr{C}_{SO(3)}(A_1,A_2) = (1-\lambda)A_1 + \lambda A_2$, but now employing the Cayley transform based approach in the averaging operator $\mathscr{C}_{SO(3)}$. The sequence of images below again illustrates the intermediate states of the Cayley-interpolation when λ varies from 0 to 1 in steps of 0.125 as shown in Fig. 6. Note that the halo around the arrow is the possible angle and the dark-blue area is the actual given angle. The interpolation seems to be slightly more uniform as the H(3)-based one from the previous example, see Fig. 5. The transition from the angle $1/7\pi$ to $6/7\pi$ resembles more a linear interpolation (Fig. 6). However, the changes in the axes are slightly more sudden than in the previous case of Fig. 5.

7.3 p-Supremum and p-Infimum in SO(3)

Now, we assume the operator $\mathscr{A}$ in the diagrams (6) and (8) represent, with a slight abuse of notation, the operations psup and pinf and proceed in principle as before. The subsequent two examples illustrate the results of the proposed method of calculating the pseudo-infimum pinf and pseudo-supremum psup of two rotations in R^3 represented by two rotation matrices. In the first case, the two rotations have the same axis but different rotation angles.

Example 11 Let $A_1 = R((1,2,3);2/3\pi)$ and $A_2 = R((1,2,3);1/6\pi)$. Surprisingly, we obtain the matrices $\mathrm{pinf}(A_1,A_2) = R((1,2,3);2/3\pi) = A_1$ and $\mathrm{psup}(A_1,A_2) = R((1,2,3);1/6\pi) = A_2$. All four rotations share the same axis, and

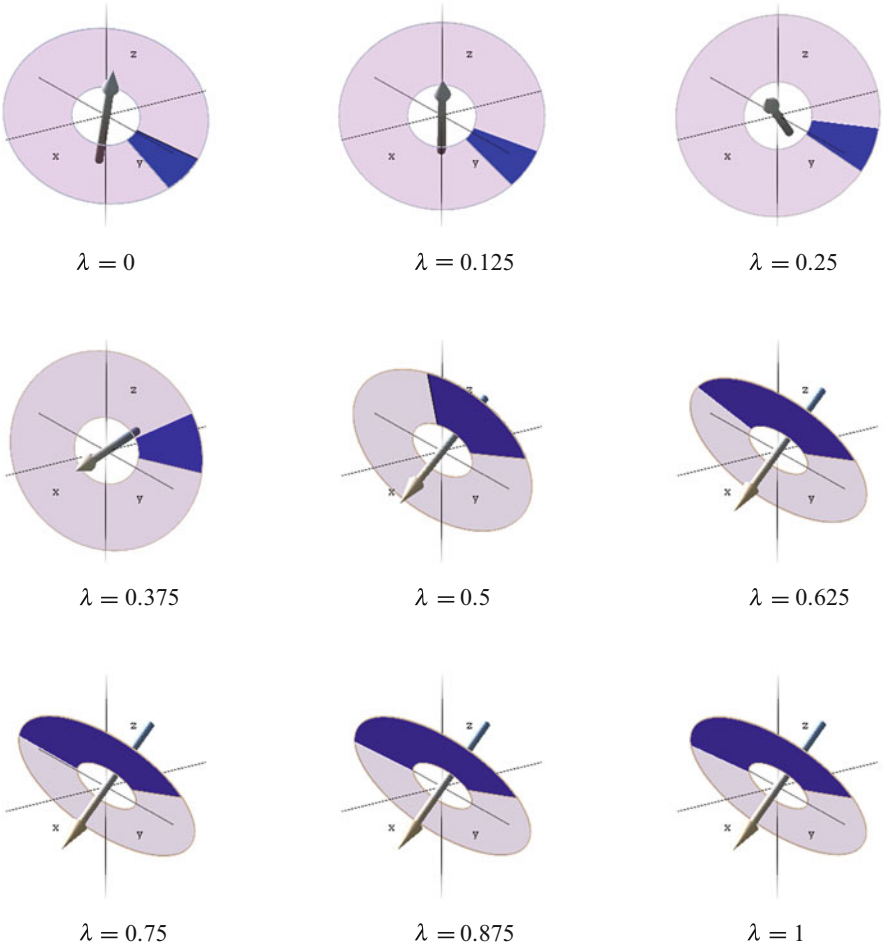


Fig. 5 Interpolation via embedding into H(3). Intermediate steps in the linear interpolation of the matrices A_1 and A_2 visualized as gyros for the interpolation parameter values $\lambda \in \{0, 1/8, \ldots, 7/8, 1\}$

both $\mathrm{psup}(A_1, A_2)$ and $\mathrm{pinf}(A_1, A_2)$ coincide with one of the A_i, but in reversed order with respect to the angle of rotation. The "larger rotation" belongs to the smaller rotation angle, somewhat contrary to our intuition. The matrices are visualized in the following Fig. 7 as gyros as before.

Remark 6 The "reversed-angle phenomenon" might appear less mysterious if one considers the identity matrix I and the fact that $\mathrm{pinf}(R, I) = R$ and $\mathrm{psup}(R, I) = I$, reasonably making the identity the "largest" rotation matrix, although its rotation angle is zero. Nevertheless, in view of $\mathrm{SO}(2) \equiv \{R((0, 0, 1); \phi) \in \mathrm{SO}(3) \,|\, \phi \in [-\pi, \pi)\}$ we can already conclude from this experiment that morphology on the

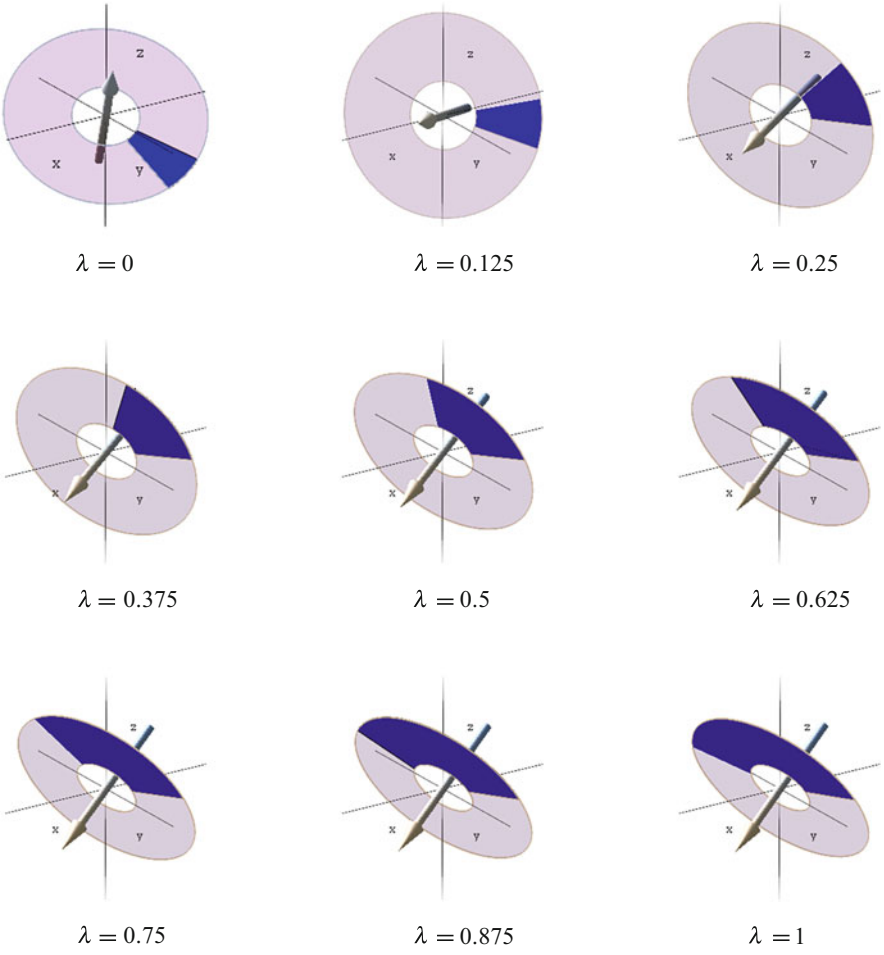


Fig. 6 Interpolation via Cayley transform. Intermediate steps in the linear interpolation of the matrices A_1 and A_2 visualized as gyros for the interpolation parameter values $\lambda \in \{0, 1/8, \ldots, 7/8, 1\}$

group of rotations SO(2) boils down to morphology for ϕ-valued data, that is, gray-scale morphology (although with a reversed total order). This is in agreement with our expectations and the approach in [37].

Now, we turn to the most interesting case: the two rotations differ in the axis as well in the angle of rotation.

Example 12 For the matrices $A_1 = R((1,2,3); 2/3\pi)$ and $A_2 = R((-2,4,2); 3/4\pi)$ the algorithm gives the pseudo-infimum

$$\text{pinf}(A_1, A_2) = R((-0.3525355818, 0.8140679142, 0.461532333); 0.751273688\pi)$$

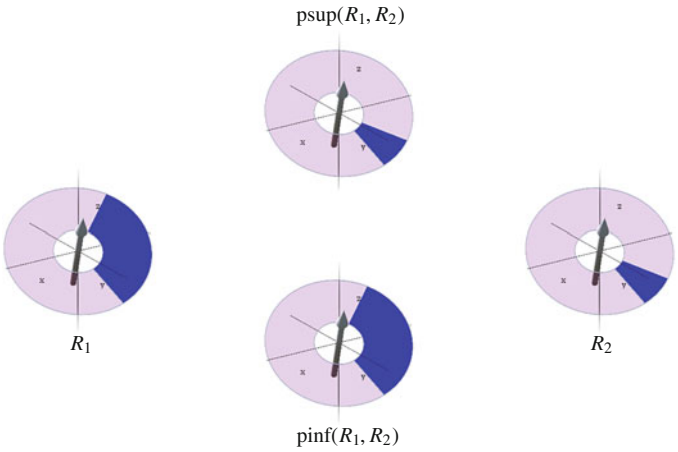


Fig. 7 The matrices A_1 and A_2 represent rotations around the same axis unit vector $1/\sqrt{14}(1,2,3)^\top$ but with $2/3\pi$ and $1/6\pi$. The visualization with gyros indicates axis and angle of rotation for each of the matrices. As expected the axis unit vector of the rotations $\mathrm{psup}(R_1, R_2)$ and $\mathrm{pinf}(R_1, R_2)$ coincides with the one of A_1 and A_2. Surprisingly however, the rotation is the larger, the smaller the angle of rotation is

and

$$\mathrm{psup}(A_1, A_2) = R((0.090789087, 0.657327318, 0.7481163935); 0.26731243\pi)\,.$$

All the matrices are depicted below in Fig. 8. Neither $\mathrm{pinf}(A_1, A_2)$ nor $\mathrm{psup}(A_1, A_2)$ coincides with any of A_1 or A_2. However, A_1 seems to be similar to $\mathrm{psup}(A_1, A_2)$ while A_2 is closer to $\mathrm{pinf}(A_1, A_2)$ and again we observe the reversed order phenomenon concerning the angle.

Our last example is concerned with pinf, psup of three matrices employing the approximate algorithm in Sect. 5.1.

Example 13 Given are the matrices $A_1 = R((0.5, 0, \sqrt{3}/2); 1/3\pi)$, $A_2 = R((\sqrt{3}/3, \sqrt{3}/3, \sqrt{3}/3); 1/3\pi)$, and $A_3 = R((0.5, 0, \sqrt{3}/2); 1/4\pi)$. Then the algorithm in Sect. 5.1 yields

$$\mathrm{pinf}(A_1, A_2, A_3) = R((0.56960815, 0.30525209, 0.76313021); 0.38336817\pi)$$

and

$$\mathrm{psup}(A_1, A_2, A_3) = R((0.56960815, 0.30525209, 0.76313021); 0.249374892\pi)\,.$$

The corresponding gyros are displayed in Fig. 9. Clearly, we have $\mathrm{pinf}(A_1, A_2, A_3)$, $\mathrm{psup}(A_1, A_2, A_3) \notin \{A_1, A_2, A_3\}$, as expected. The axes vectors of rotations for pinf and psup are the same. The angles of rotations differ slightly from those of A_1, and A_3 on the one hand side, and on the other side from A_2. Again we observe the "reversed angle phenomenon".

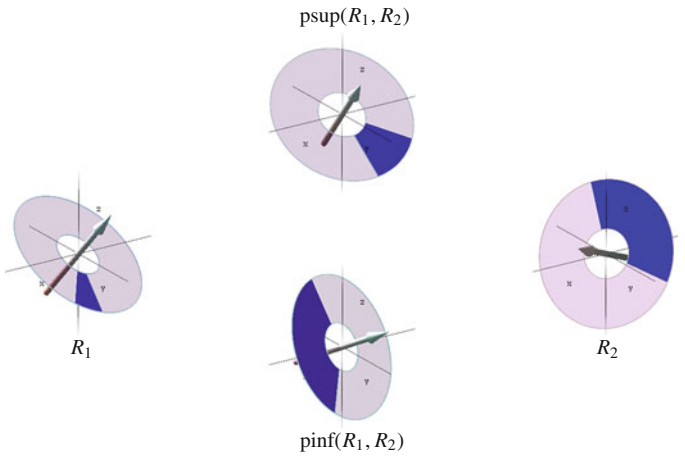


Fig. 8 The matrices A_1 and A_2 represent rotations around the axis unit vectors $1/\sqrt{14}(1,2,3)^\top$, and $1/\sqrt{24}(-2,4,2)^\top$ and rotation angles $2/3\pi$, and $3/4\pi$

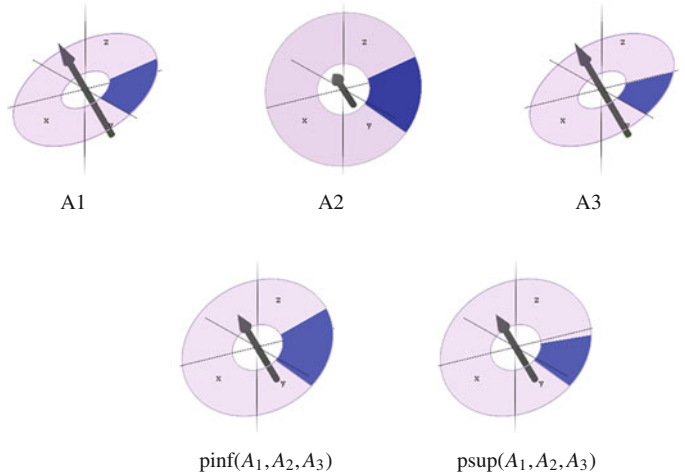


Fig. 9 Approximations of pinf(A_1, A_2, A_3) and psup(A_1, A_2, A_3) of three rotation matrices A_1, A_2, and A_3 calculated with the iterative procedure described in Sect. 5.1

8 Concluding Remarks and Outlook

This work is more of theoretical nature as it tries to pave the way towards image processing of rather general matrix fields. The real matrices of such fields that (or that not yet) appear in applications posses specific properties: they are just square, or invertible, symmetric or skew-symmetric, orthogonal, special orthogonal or other attributes that determine their algebraic, geometric or topological class properties.

Symmetry of matrices warrants a rather rich functional calculus which in turn allows for notions of linear combination, functions of, and operators for, symmetric matrices, and, last but not least, reasonable notions of order, infimum, and supremum. These terms are the building blocks of any numerical algorithm tackling image processing tasks. The properties of the other classes of matrices just mentioned do not lend themselves to a sufficient functional calculus. This shortcoming severely hampers any attempt to develop said key notions. The method of complexification circumvents this difficulty. A linear and invertible transformation, easy to calculate, is used to map the class of real square matrices onto the set H(n) of Hermitian matrices. Via this mapping the other classes of matrices are embedded into H(n) and their images form subsets of H(n) with certain topological properties. H(n) has an amenable functional calculus just as $\mathrm{SYM}(n)$, and in this way the aforementioned key notions can be established, at least in an approximate form. However, in general this requires a back projection onto the embedded set, because these are not stable under the necessary operations. In the case of orthogonal matrices the matrix polar decomposition provides a solution.

Our current research includes the exploitation of the proposed general methodology to develop higher order tools for the processing of the above matrix fields, as well as to open up new classes of matrices for processing and analysis.

Acknowledgements The authors would like to thank the two anonymous referees for their useful comments and valuable insights that helped to improve this work.

References

1. Aris, R.: Vectors, Tensors, and the Basic Equations of Fluid Mechanics. Dover Publications, New York (1989)
2. Berndt, R.: Representations of Linear Groups: An Introduction Based on Examples From Physics and Number Theory. Vieweg+Teubner, Wiesbaden (2007)
3. Bhatia, R.: Matrix Analysis. Springer, New York (1996)
4. Bigun, J.: Vision with Direction. Springer, Berlin (2006)
5. Boroujerdi, A.S., Breuß, M., Burgeth, B., Kleefeld, A.: PDE-based color morphology using matrix fields. In: Aujol, J.F., Nikolova, M., Papadakis, N. (eds.) Scale Space and Variational Methods in Computer Vision: 5th International Conference, SSVM 2015, Lège-Cap Ferret, France, 31 May–4 June 2015, Proceedings, pp. 461–473. Springer, Berlin (2015)
6. Brox, T., Weickert, J., Burgeth, B., Mrázek, P.: Nonlinear structure tensors. Image Vis. Comput. **24**(1), 41–55 (2006)
7. Buchholz, S., Sommer, G.: On averaging in Clifford groups. In: Li, H., Olver, P.J., Sommer, G. (eds.) Computer Algebra and Geometric Algebra with Applications: 6th International Workshop, IWMM 2004, Shanghai, China, 19–21 May 2004 and International Workshop, GIAE 2004, Xian, China, 24–28 May 2004, Revised Selected Papers, pp. 229–238. Springer, Berlin/Heidelberg (2005)
8. Burgeth, B., Kleefeld, A.: Morphology for color images via Loewner order for matrix fields. In: Luengo Hendriks, C.L., Borgefors, G., Strand, R. (eds.) Mathematical Morphology and Its Applications to Signal and Image Processing (Proceedings of the 11th International Symposium on Mathematical Morphology, Uppsala, Sweden, May 27–29). Lecture Notes in Computer Science, vol. 7883, pp. 243–254. Springer, Berlin (2013)

9. Burgeth, B., Kleefeld, A.: An approach to color-morphology based on Einstein addition and Loewner order. Pattern Recogn. Lett. **47**, 29–39 (2014)
10. Burgeth, B., Kleefeld, A.: Order based morphology for color images via matrix fields. In: Westin, C.-F., Vilanova, A., Burgeth, B. (eds.) Visualization and Processing of Tensor Fields and Higher Order Descriptors for Multi-Valued Data, pp. 75–95. Springer, Berlin (2014)
11. Burgeth, B., Papenberg, N., Bruhn, A., Welk, M., Feddern, C., Weickert, J.: Morphology for higher-dimensional tensor data via Loewner ordering. In: Ronse, C., Najman, L., Decencière, E. (eds.) Mathematical Morphology: 40 Years on Computational Imaging and Vision, vol. 30, pp. 407–418. Springer, Dordrecht (2005)
12. Burgeth, B., Bruhn, A., Didas, S., Weickert, J., Welk, M.: Morphology for tensor data: Ordering versus PDE-based approach. Image Vis. Comput. **25**(4), 496–511 (2007)
13. Burgeth, B., Bruhn, A., Papenberg, N., Welk, M., Weickert, J.: Mathematical morphology for tensor data induced by the Loewner ordering in higher dimensions. Signal Process. **87**(2), 277–290 (2007)
14. Burgeth, B., Didas, S., Florack, L., Weickert, J.: A generic approach to diffusion filtering of matrix-fields. Computing **81**, 179–197 (2007)
15. Burgeth, B., Didas, S., Florack, L., Weickert, J.: A generic approach to the filtering of matrix fields with singular PDEs. In: Sgallari, F., Murli, F., Paragios, N. (eds.) Scale Space and Variational Methods in Computer Vision. Lecture Notes in Computer Science, vol. 4485, pp. 556–567. Springer, Berlin (2007)
16. Burgeth, B., Pizarro, L., Breuß, M., Weickert, J.: Adaptive continuous-scale morphology for matrix fields. Int. J. Comput. Vis. **92**(2), 146–161 (2011)
17. Dai, Y., Trumpf, J., Li, H., Barnes, N., Hartley, R.: Rotation averaging with application to camera-rig calibration. In: Zha, H.H., Taniguchi, R.-I., Maybank, S. (eds.) Computer Vision – ACCV 2009: 9th Asian Conference on Computer Vision, Xi'an, 23–27 September 2009, Revised Selected Papers, Part II, pp. 335–346. Springer, Berlin/Heidelberg (2010)
18. Georgi, H.: Lie Algebras in Particle Physics: From Isospin to Unified Theories. Westview Press, Boulder (1999)
19. Gerrits, T., Rössl, C., Theisel, H.: Glyphs for general second-order 2d and 3d tensors. IEEE Trans. Vis. Comput. Graph. **23**(1), 980–989 (2017)
20. Golub, G.H., van Loan, C.F.: Matrix Computations, 4th edn. JHU Press, Baltimore (2013)
21. Gramkow, C.: On averaging rotations. Int. J. Comput. Vis. **42**(1), 7–16 (2001)
22. Higham, N.J.: Matrix Nearness Problems and Applications. Oxford University Press, Oxford (1989)
23. Higham, N.: Matrix Procrustes problems. PhD thesis, The University of Manchester, 1993
24. Higham, N.: Functions of Matrices. Society for Industrial and Applied Mathematics, Philadelphia (2008)
25. Hillen, T., Painter, K.J.: A user's guide to PDE models for chemotaxis. J. Math. Biol. **58**(1), 183–217 (2008)
26. Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press, Cambridge (1990)
27. Horn, R.A., Johnson, C.R.: Topics in Matrix Analysis. Cambridge University Press, Cambridge (1994)
28. Hotz, I., Schultz, T. (eds.): Visualization and Processing of Higher Order Descriptors for Multi-Valued Data. Springer, Berlin (2015)
29. Kleefeld, A., Burgeth, B.: Processing multispectral images via mathematical morphology. In: Hotz, I., Schultz, T. (eds.) Visualization and Processing of Higher Order Descriptors for Multi-Valued Data, pp. 129–148. Springer, Berlin (2015)
30. Kleefeld, A., Breuß, M., Welk, M., Burgeth, B.: Adaptive filters for color images: median filtering and its extensions. In: Trémeau, A., Schettini, R., Tominaga, S. (eds.) Computational Color Imaging: 5th International Workshop, CCIW 2015, Saint Etienne, France, 24–26 March 2015, Proceedings, pp. 149–158. Springer, Berlin (2015)
31. Kleefeld, A., Meyer-Baese, A., Burgeth, B.: Elementary morphology for SO(2)- and SO(3)-orientation fields. In: Benediktsson, A.J., Chanussot, J., Najman, L., Talbot, H. (eds.) Mathematical Morphology and Its Applications to Signal and Image Processing: 12th

International Symposium, ISMM 2015, Reykjavik, Iceland, 27–29 May 2015. Proceedings, pp.458–469. Springer, Berlin (2015)
32. Knutsson, H.: A tensor representation of 3-D structures. In: 5th IEEE-ASSP and EURASIP Workshop on Multidimensional Signal Processing, Noordwijkerhout, The Netherlands, September 1987. Poster presentation.
33. Knutsson, H.: Representing local structure using tensors. In: The 6th Scandinavian Conference on Image Analysis, pp. 244–251, Oulu, Finland, June 1989. Report LiTH–ISY–I–1019, Computer Vision Laboratory, Linköping University, Sweden (1989)
34. Kratz, A., Schöneich, M., Zobel, V., Burgeth, B., Scheuermann, G., Hotz, I., Stommel, M.: Tensor visualization driven mechanical component design. In: Proceedings of the IEEE Pacific Visualization Symposium 2014, pp. 145–152 (2014)
35. Laidlaw, D., Weickert, J. (eds.): Visualization and Processing of Tensor Fields. Springer, Berlin (2009)
36. Marsden, J.E., Hughes, T.J.R.: Mathematical Foundations of Elasticity. Dover Publications, New York (1994)
37. Moakher, M.: Means and averaging in the group of rotations. SIAM J. Matrix Anal. Appl. **24**(1), 1–16 (2002)
38. Moler, C., Van Loan, C.: Nineteen dubious ways to compute the exponential of a matrix. SIAM Rev. **20**, 801–836 (1978)
39. Ramsay, A., Richtmyer, R.D.: Introduction to Hyperbolic Geometry. Universitext Series. Springer, Berlin (1995)
40. Schultz, T., Kindlmann, G.L.: Superquadric glyphs for symmetric second-order tensors. IEEE Trans. Vis. Comput. Graph. **16**(6), 1595–1604 (2010)
41. Schultz, T., Nedjati-Gilani, G., Venkataraman, A., O'Donnell, L., Panagiotaki, E. (eds.): Computational Diffusion MRI and Brain Connectivity. In: Mathematics and Visualization. Springer, Berlin (2014)
42. Seltzer, N., Kindlmann, G.: Glyphs for Asymmetric Second-Order 2D Tensors. Computer Graphics Forum, pp. 141–150 (2016)
43. Simo, J.C., Hughes, T.J.R.: Computational Inelasticity. Interdisciplinary Applied Mathematics. Springer, New York (1998)
44. Welk, M., Kleefeld, A., Breuß, M.: Non-adaptive and amoeba quantile filters for colour images. In: Benediktsson, J.A., Chanussot, J., Najman, L., Talbot, H. (eds.) Mathematical Morphology and Its Applications to Signal and Image Processing: 12th International Symposium, ISMM 2015, Reykjavik, Iceland, 27–29 May 2015. Proceedings, pp. 398–409. Springer, Berlin (2015)
45. Welk, M., Kleefeld, A., Breuß, M.: Quantile filtering of colour images via symmetric matrices. Math. Morphol. Theory Appl. **1**, 136–174 (2016)
46. Westin, C.-F., Vilanova, A., Burgeth, B. (eds.): Visualization and Processing of Tensor Fields and Higher Order Descriptors for Multi-Valued Data. Springer, Berlin (2014)
47. Zhang, F.: Matrix Theory: Basic Results and Techniques. Universitext. Springer, New York (1999)

Towards Grey Scale-Based Tensor Voting for Blood Vessel Analysis

Daniel Jörgens and Rodrigo Moreno

Abstract Tensor Voting is a technique that uses perceptual rules to group points in a set of input data. Its main advantage lies in its ability to robustly extract geometrical shapes like curves and surfaces from point clouds even in noisy scenarios. Following the original formulation this is achieved by exploiting the relative positioning of those points with respect to each other. Having this in mind, it is not a straight forward task to apply original tensor voting to greyscale images. Due to the underlying voxel grid, digital images have all data measurements at regularly sampled positions. For that reason, the pure spatial position of data points relative to each other does not provide useful information unless one considers the measured intensity value in addition to that.

To account for that, previous approaches of employing tensor voting to scalar images have followed mainly two ideas. One is to define a subset of voxels that are likely to resemble a desired structure like curves or surfaces in the original image in a preprocessing step and to use only those points for initialisation in tensor voting. In other methods, the encoding step is modified e.g. by using estimations of local orientations for initialisation.

In contrast to these approaches, another idea is to embed all information given as input, that is position in combination with intensity value, into a 4D space and perform classic tensor voting on that. In doing so, it is neither necessary to rely on a preprocessing step for estimating local orientation features nor is it needed to employ assumptions within the encoding step as all data points are initialised with unit ball tensors. Alternatively, the intensity dimension could be partially included by considering it in the weighting function of tensor voting while still employing 3D tensors for the voting. Considering the advantage of a shorter computation time for the latter approach, it is of interest to investigate the differences between these two approaches.

Although different methods have employed an ND implementation of tensor voting before, the actual interpretation of its output, that is the estimation of a local hyper surface at each point, depends on the actual application at hand. As

D. Jörgens (✉) • R. Moreno
School of Technology and Health, KTH Royal Institute of Technology, Hälsovägen 11c, 14157 Huddinge, Sweden
e-mail: daniel.jorgens@sth.kth.se; rodrigo.moreno@sth.kth.se

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_7

we are especially interested in the analysis of blood vessels in CT angiography data, we study the feasibility of detecting tubular structures and the estimation of their orientation totally within the proposed framework and also compare the two mentioned approaches with a special focus on these aspects.

In this chapter we first provide the formulation of both approaches followed by the application-specific interpretations of the shape of 4D output tensors. Based on that, we compare the information inferred by both methods from both synthetic and medical image data focusing on the application of blood vessel analysis.

1 Introduction

Tensor voting (TV) has been widely used in different tasks of computer vision and especially analysis of images. Its robustness to outliers makes it a valuable tool in applications where noise is an issue. Traditionally, it has been applied to point cloud data in two- or three-dimensional space with spatial locality [21]. Based on rules modeling human perceptual principles the (only) feature of such measured data points—namely their position in $\mathbb{R}^3$—is exploited during the voting step in order to estimate the spatial relationship among points in a local neighbourhood.

Since its invention tensor voting has been applied in various applications such as motion estimation [5, 9, 28, 36], inpainting/image repairing [10, 31], image registration [37], segmentation [16, 20, 24, 27, 33, 36], (sharp) feature extraction [11, 29], centreline extraction [4, 14, 22], curvature estimation [17, 35], clustering [13, 18] as well as manifold learning [23]. Regardless of the specific application, the majority of these approaches interprets tensor voting as a tool for inference of geometrical structures in 3D. Usually, this demands for a reformulation of the particular problem at hand in terms of the original formulation which usually includes a reformulation of the input data to an equivalent point cloud.

In case of extracting structures from image data in a medical context (e.g. for segmentation), a common strategy towards applying tensor voting is to employ another method to extract structure candidates from the input intensity data (e.g. edge detection [7], centreline extraction [14] or direction sensitive filtering [32]) providing input positions and possibly initial estimates of orientation at those points. This information is then refined in a subsequent tensor voting step in order to reject outliers as well as to close gaps [27, 30]. In this case, the outcome of tensor voting is heavily relying on the performance of the preprocessing method which further needs to be tuned separately. In this way, the grey scale values of the image are not available within the core of tensor voting which is the cause of two main shortcomings. First, errors from the preprocessing might be propagated instead of eliminated by tensor voting. Second, the intensity information is completely disregarded by tensor voting. We argue that it might be of advantage to use the raw data as the input of tensor voting in order to achieve data communication during the voting step in a more complete way.

Different approaches that directly include the intensity information into the method have been published. In [19], all pixel positions are considered as input to tensor voting. Initialisation is done as in the first sparse voting stage with ball tensors (isotropic, unoriented structure estimates) which are scaled by the particular intensity value in the original image at each point. By this strategy, bright structures are given a higher importance in the first voting step. The authors also provide suggestions to transform the intensities prior to voting in order to make the data conform with the assumption of providing the structure of interest at the brightest grey values. A problem that we foresee with including grey value information in this way is the fact that the relation between high intensity and probability to belong to the structure of interest is usually not proportional. Under the assumption of noise with zero mean, intensity values of voxels in homogeneous regions are scattered both above and below the mean. Unfortunately, the approach by Loss et al. in [19] creates a bias by giving more importance to voxels with intensities above the mean. To achieve the correct behaviour when aiming at extracting structures of similar intensity (homogeneous regions), we propose to regard the relative intensity *difference* instead of the absolute value. By this, variation of intensities in either direction of the mean are treated equally.

A general strategy to employ tensor voting for image structure estimation is presented in [26]. Initialisation prior to the first voting step is done by deriving stick tensor estimates at all image positions from the image gradient (basically, magnitude and direction are used). Similarly to the previously mentioned approach by Loss et al. [19], no thresholding is applied such that the whole image data is considered as input to tensor voting. Nevertheless, intensity information is only regarded in the initialisation step to arrive at orientation estimates and only spatial information is included in the core of the TV algorithm which might lead to undesired structure propagation into unstructured regions.

Considering intensity differences for characterising structure properties with tensor voting is not new. For example, the method in [11] performs tensor voting in 4D for grey value images and 6D for colour ones, by including the intensity as additional dimensions in the voting space. Their ultimate analysis of the tensor voting results extracts any features in the particular N-dimensional space that can be classified as a homogeneous region, a sharp edge or as a corner whereby these terms are understood as such in terms of either spatial or "colorimetrical" properties. That means, the authors do not aim at distinguishing between a geometrical corner structure of homogeneous colour and a geometrical flat surface exhibiting a corner pattern in terms of colour. Instead, in our application—which is medical images containing blood vessels—we are interested in the geometrical structure of blood vessels which we assume to be of tubular shape. For that reason it is important to investigate the orientation of the resulting voting tensors in order to analyse only the spatial-geometrical structure it describes. In that way we extend the interpretation of 4D tensors in [11] for our purposes.

The main application in this paper is the analysis of blood vessels in 3D CT angiography (CTA) images. The segmentation of this kind of vessels is an important step for visualisation and, finally, diagnosis and quantification of vascular pathologies [15]. Proposing effective methods for solving this problem is challenging due to the high level of noise and low resolution of the images with respect to the diameter of the vessels. Thus, despite the efforts made in the last two decades [15], this research field is still very active. Our work is especially making use of the concept of "height ridges" which was employed for tubular object segmentation [1]. Instead of extracting ridge points by analysing the Hessian [1, 6], we make use of tensor voting to detect tubular structures in the data. Similar to the medialness used in [8], a specific saliency map derived from the resulting tensors indicates the probability for a specific point of being located at the centre of a tubular structure. Further, the tube orientation is encoded in the same tensor as well which makes the algorithm suitable for tracking purposes. Local analysis of orientations in a neighbourhood of a point could be the basis for detecting branches as well as stenoses.

The goal in this chapter is to formulate the problem of extracting tubular structures of homogeneous intensity values from grey value images within the framework of tensor voting. Thereby, we formulate the problem in a way such that the method (TV) regards all input information (i.e. position in space + intensity value) simultaneously in the voting step. Geometrically, this strategy can be interpreted as finding curves in an artificial four-dimensional space in which the original image data represents a three-dimensional surface. As we are only interested in the spatial structure we have to regard the tensor orientation in the final analysis of the tensor features, which gives the opportunity to (a) detect tubular structures and (b) estimate their orientation at the same time. Finally, we compare the 4D results with those obtained from a simplified 3D-version of the algorithm which only includes the intensity information in the weighting function in the voting step. By relating its formulation to the 4D approach, we aim at giving it well-founded justification, in particular it can be thought of as an approximation of the full 4D approach which exploits the smaller computational efforts needed for the lower dimensional version of tensor voting.

In general, the idea of using tensors for analysing blood vessels is not novel (e.g. in [2, 3]) and in that context it has been proposed to augment the image dimensions with a fourth dimension originating from the image which was employed for tensor computation. However, to the best of our knowledge, this is the first attempt to use tensor voting directly on grey scale data in this specific application.

This chapter is structured as follows. First, we give the interpretation of a 4D tensor in terms of shape and orientation in a space of 3D spatial dimensions augmented with a 4th intensity dimension. Second, we briefly state the essential formulations of the employed 4D tensor voting approach from which we subsequently derive the modifications for the 3D simplification. Finally, we test both formulations with respect to their ability to detect tubular structures in different noise scenarios for synthetic intensity profiles and ultimately apply the detection algorithm in real data. Further, we indicate the performance for orientation estimation at the detected tube positions.

2 Adapting Tensor Voting to Grey Scale Image Processing

Traditionally, tensor voting has been proposed for point clouds. As mentioned in [21], acceptable input sites consist of a point which might be augmented with additional initial estimates of local structure. By looking in detail into how these input data are used, it becomes clear that the local structure estimates serve as predefined initialisations in the TV algorithm and are not regarded as actual input features. The only information the tensor voting algorithm treats as actual input is the location of the input sites. Observing that this information has three degrees of freedom (spatial position), one can conclude that the actual features regarded by tensor voting are the spatial relationships between the input points. While their spatial position does not change during the voting, the initial structure estimates are re-estimated and updated after each pass of tensor voting.

Based on that view of the original tensor voting formulation, we argument that all information which should be completely regarded in all stages of the voting algorithm should be encoded in the input space. That means in particular, that additional features would be regarded as extra dimensions as in [11]. In doing so, it is ensured that all information is actually consistently regarded in the algorithm throughout several TV passes and the perception-based rules of the voting step are applied to all input information. The input point cloud in the original tensor voting formulation derived for an input image I can be represented as

$$\Omega_I^3 = \{(x, y, z) \in \mathbb{R}^3 \mid g(I)(x, y, z) = 1\},$$

where g is an indicator function that returns 1 for points where a specific feature was detected. Instead, we want to use an input point cloud that considers the intensity:

$$\Omega_I^4 = \{(x, y, z, t) \in \mathbb{R}^4 \mid I(x, y, z) = t\}.$$

As already mentioned, the main advantage of this formulation is that it does not depend on the function g.

In this chapter we propose to employ tensor voting on this four-dimensional data derived from the 3D input in order to analyse contained structures. Our specific goal here is to apply it in a medical context which implies both the existence of noisy conditions and the goal of detecting vessels (i.e. tubular structures). In the following, we present the prerequisites of our method.

2.1 Interpretation of Tensor Shape and Orientation

The basis for detecting a certain structure in tensor voting applications is the interpretation of the shape of the resulting tensors. Based on its eigendecomposition,

the maps $s_1, \ldots s_n$, defined as

$$s_i = \begin{cases} \frac{\lambda_i - \lambda_{i+1}}{\lambda_1}, & \text{if } i < n \\ \frac{\lambda_n}{\lambda_1}, & \text{if } i = n \end{cases}, \tag{1}$$

where λ_i denotes the i-th largest eigenvalue of a particular tensor T in an n-dimensional voting space, indicate the saliency of structure of corresponding dimension at the specific point of investigation, whereby the dimensionality depends on the chosen mode of structure encoding. In case, the tensors encode the tangent space of a structure, s_i denotes the saliency of an i-dimensional structure, whereas in case of normal space encoding, s_i corresponds to a structure of dimension $(n-i)$. Normally, when the dimensionality of the desired structure is known, it suffices to investigate the particular saliency maps in order to determine the appearance of a structure of the respective dimensionality at a point. This strategy works in case all dimensions of the voting space are representing the same entity (e.g. a spatial direction).

In our setting, we still want to detect structures of a certain dimensionality, but only in spatial terms. Therefore, the orientation of a voting tensor has an influence on the spatial dimensionality of the encoded feature. For the following descriptions, assume an orthonormal basis of $\mathbb{R}^4$ is given by $\{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y, \hat{\mathbf{e}}_z, \hat{\mathbf{e}}_I\}$, where $\hat{\mathbf{e}}_x$, $\hat{\mathbf{e}}_y$ and $\hat{\mathbf{e}}_z$ denote three spatial axes and $\hat{\mathbf{e}}_I$ describes the intensity axis. Further, let $\mathsf{T} \in \mathrm{PSD}(4)$ be the resulting tensor after tensor voting at a particular point, where $\mathrm{PSD}(n)$ is the set of positive semi-definite $n \times n$ matrices over $\mathbb{R}$, and $\hat{\mathbf{e}}_1, \ldots, \hat{\mathbf{e}}_4$ are its eigenvectors ordered by the descending magnitude of the corresponding eigenvalues. Then, the angle of the f-dimensional subspace spanned by $\{\hat{\mathbf{e}}_1, \ldots, \hat{\mathbf{e}}_f\}$ with respect to $\hat{\mathbf{e}}_I$, denoted by $\theta_I^{(f)}$, is given as

$$\theta_I^{(f)} = \operatorname{acos}\left(\left| < \hat{\mathbf{p}}_I^{(f)}, \hat{\mathbf{e}}_I > \right| \right), \tag{2}$$

where $\hat{\mathbf{p}}_I^{(f)}$ is the normalised projection of $\hat{\mathbf{e}}_I$ onto the subspace defined by

$$\mathbf{p}_I^{(f)} = \sum_{i=1}^{f} < \hat{\mathbf{e}}_I, \hat{\mathbf{e}}_i > \hat{\mathbf{e}}_i \ .$$

Assuming that the tensors in tensor voting encode the normal space of a structure at a point, we analyse the corresponding tangent space to arrive at an interpretation of the spatial feature it actually encodes. Let $\{\hat{\mathbf{e}}_1, \ldots, \hat{\mathbf{e}}_f\}$ span the f-dimensional normal space of a certain structure at a certain point. In case $\theta_I^{(f)}$ is small, $\hat{\mathbf{e}}_I$ is closely aligned to that normal space (i.e. it is contained) and the corresponding

Table 1 Interpretation of a tensor's shape and orientation in 4D space (3D spatial + 1D intensity)

Angle $\theta_I^{(f)}$	1D normal space	2D normal space	3D normal space	4D normal space
Small	Homogeneous volume	Homogeneous surface	Homogeneous curve	Inhomogeneous volume
Large	Inhomogeneous surface	Inhomogeneous curve	Inhomogeneous volume	–

Here, the 'fD normal space' is spanned by the first f eigenvectors of a tensor. A large saliency s_f indicates its importance for the overall tensor shape

$(4-f)$-dimensional tangent space can be interpreted as being 'purely spatial',[1] so the interpretations from classical tensor voting apply for that tangent space (and therefore for the underlying structure). As opposed to that, in the case of a large $\theta_I^{(f)}$ (i.e. close to 90°), $\hat{\mathbf{e}}_I$ is perpendicular to the normal space meaning that the latter can be seen as 'purely spatial' and therefore its interpretation in the classical terms is valid.

Additionally to the spatial interpretation, $\hat{\mathbf{e}}_I$ lying in the normal or tangent space provides information about the distribution of intensity values in a local neighbourhood of a point. In those terms, $\theta_I^{(f)} \approx 0°$ represents homogeneous intensities (i.e. similar range) whereas $\theta_I^{(f)} \approx 90°$ indicates inhomogeneously distributed values.

Ultimately, the 4D normal space captures spatial and intensity differences that appear to be on the same scale and can thus be interpreted as encoding a volume of inhomogeneous intensities regardless of the angle which is always equal to zero in that case.

Table 1 summarises the explained interpretations and Fig. 1 provides further schematic examples.

2.2 *Voting Formulation*

The coupling of spatial dimensions x, y, z and intensity I has implications on the definition of angles and distances in the Euclidean sense. In order to steer the weight of distance in direction of $\hat{\mathbf{e}}_I$ and in spatial terms independently,[2] the weighting of distance and curvature is done separately for those dimensions. In particular, the projection of the distance vector $\mathbf{v}_4$ in 4D onto the three-dimensional spatial subspace is used to determine the weight from the traditional weighting

[1] By the term 'purely spatial' we refer to the fact that the respective subspace is particularly a subspace of the space spanned by $\{\hat{\mathbf{e}}_x, \hat{\mathbf{e}}_y, \hat{\mathbf{e}}_z\}$.

[2] It is not totally independent due to normalisation and rotation in the constructed 4D space. Instead we refer to independence in terms of their influence on the weighting function d in tensor voting.

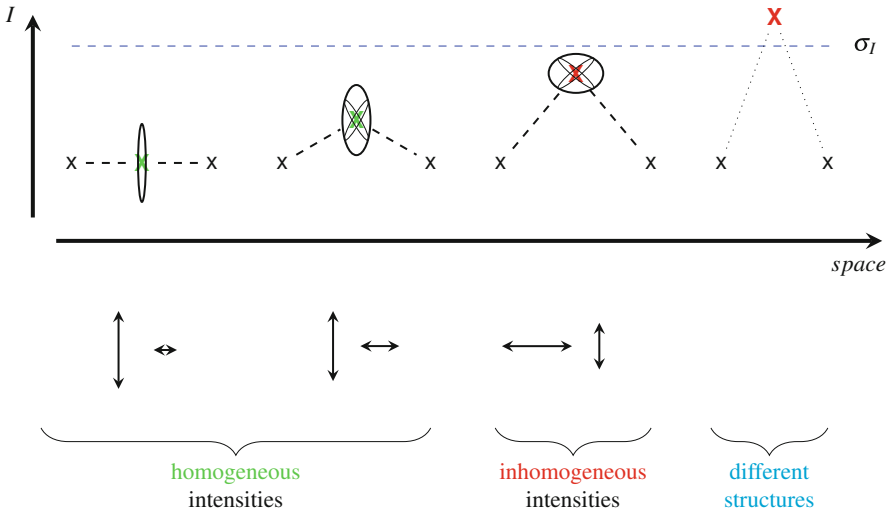


Fig. 1 Example schematically showing the concept of the intensity-augmented voting space in 2D (one spatial dimension + intensity dimension). The approach relies on the assumptions that the target regions to be extracted have a more or less homogeneous intensity range. *First row:* Four cases of ball voting from the outer positions to the middle location are shown where the middle point exhibits different intensity values in each case. The range of weights in intensity direction that are significantly larger than zero (determined by parameter σ_I in the weighting function (cf. Sect. 2.2)) is depicted by the *dashed line*. For example, in the fourth case, voting weights vanish due to large intensity differences and no votes are cast. Note, that the weighting of votes is not shown in this visualisation. *Second row:* Decomposition of the resulting tensor after voting at the particular middle points into first order tensors (sticks) which are ordered by their magnitude. For the third case, the first stick is perpendicular to the I-dimension indicating larger intensity inhomogeneities

function d_{orig}, whereas the intensity difference ΔI determines an additional weight. Following the proposed weighting in [25], this ultimately results in the formulation of the weighting function of the generic voting step as

$$\hat{d}_4(\mathbf{v}_4; \sigma_{space}, b_{space}, \sigma_I, b_I) = d_{\text{orig}}(\mathbf{v}_3; \sigma_{space}, b_{space}) \cdot e^{-\left(\frac{\Delta I}{\sigma_I}\right)^2 - b_I \cdot \kappa_I^2}, \tag{3}$$

where $\mathbf{v}_4 = (\Delta x, \Delta y, \Delta z, \Delta I)^\mathsf{T}$, $\mathbf{v}_3 = (\Delta x, \Delta y, \Delta z)^\mathsf{T}$ and κ_I denotes the curvature of the osculating circle within the plane spanned by the voting direction $\mathbf{v}_4$ and the intensity axis $\hat{\mathbf{e}}_I$. The weighting for the sparse voting step, denoted $\hat{d}_4^{\mathsf{BV}}$, is similarly modified, but omitting the curvature term.

For the implementation of the four-dimensional voting algorithm, we choose the method proposed by Mordohai et al. [23] while replacing the employed weightings with our particular weighting functions. Even though the proposed strategy performs tensor voting in 4D, the data are obviously not dense in all dimensions. As the original input data assigns exactly one intensity value to each voxel, the transformed

point cloud in 4D is actually a 3D surface. Therefore, the computation is performed in 3D while employing 4D tensors.

Considering the fact during all steps of voting in the experiments only the input sites are participating one might call the operations *sparse*. However, as opposed to the intensity dimension in which only a single input site per spatial position exists, the spatial 3D subspace is sampled regularly and in that sense, one can consider voting as *dense* in those dimensions.

2.3 Simplified Approach in 3D

In order to reduce the computational effort, it is worthwhile investigating if a 3D implementation maintaining the basic idea for grey scale images (i.e. to be sufficiently close to the 4D formulation) can still provide the same information in terms of the outcome which we are interested in (i.e. the detection of tubular structures).

Since the basic source of computational overhead stems from the additional dimension of the voting space (i.e. intensity), we omit it for the 3D formulation. Regarding the fact that the actual algorithm is carried out on a 3D surface and only the voting is done in 4D, we could omit the 4th tensor dimension and employ 3D TV with a modified weighting function instead. We choose the latter in relation to Eq. (3) as

$$\hat{d}_3(\mathbf{v}_3, \Delta I; \sigma_{space}, b_{space}, \sigma_I) = d_{\text{orig}}(\mathbf{v}_3; \sigma_{space}, b_{space}) \cdot e^{-\left(\frac{\Delta I}{\sigma_I}\right)^2}, \tag{4}$$

where $\mathbf{v}_3$ is defined as in Sect. 2.2. As the curvature w.r.t. $\hat{\mathbf{e}}_I$ does not make sense here any more, we omit that term. This amounts in the same weighting function as in Eq. (3) with $b_I = 0$.

Regarding distance both in terms of spatial relation and intensity difference in the weighting function is closely related to the idea of bilateral filtering [34]. Making use of the weighting in Eq. (4) basically means to weigh those votes more which originate from positions that exhibit a similar intensity value. Thereby, spatial structure at each point is mainly inferred by considering similar points in the local neighbourhood which makes it assumingly possible to detect the structure of homogeneous intensity regions. The interpretation of tensor shape and orientation is the same as in the traditional applications as now only spatial dimensions (which are interchangeable) are involved in representation of structure.

As in case of the 4D approach, we employ the voting algorithms proposed in [23].

2.4 Differences Between 4D and 3D Approach

As both formulations employ the same weighting function including the intensity information in addition to the spatial location, the main differences can be found in the tensor representation of local structure. While in the 3D case all regarded dimensions have a spatial meaning and thereby tensor interpretation as in classical tensor voting applies, in our 4D formulation the mentioned coupling of spatial dimensions with intensity has implications on distances and angles defined in the resulting space.

Regarding the first voting step (i.e. ball voting), which—following [23]—is given by

$$\mathsf{BV}(\mathbf{v}_4) = \hat{d}_4^{\mathsf{BV}}(\mathbf{v}_4) \left(I - \frac{\mathbf{v}_4 \mathbf{v}_4^{\mathsf{T}}}{||\mathbf{v}_4||^2} \right), \tag{5}$$

voting is induced only by isotropic tensors and the step does not depend on an angle in terms of a rotation. Nevertheless, the deformation of such an isotropic tensor in Eq. (5) tangentially to the voting direction $\mathbf{v}_4$ involves normalisation, which inherently includes the computation of a norm and thereby distance is involved. This introduces an implicit weighting for the spatial dimensions depending on the angle between spatial and intensity dimensions. In case of no intensity difference, the distance would be dominated by the spatial dimensions only. On the other hand, a large intensity difference would dominate the length of $\mathbf{v}_4$ instead, and thereby give a different weighting to the entries of BV related to the spatial dimensions. The same spatial neighbourhood with different intensity differences could exhibit a different spatial structure that would be derived from the voting tensors.

Similarly, the subsequent generic voting steps, which additionally include an angle and rotation steps, are affected by the intensity dimension.

3 Tube Detection and Orientation Estimation

In our current application in the context of this chapter we aim at detecting regions of homogeneous intensity. As such we expect the intensity value in the noise-free case to be approximately constant in a local neighbourhood. The first assumption we employ here is that different regions are well separated in terms of different grey scale values and do therefore not (or just very little[3]) interfere. The second assumption is that noise is expected to be symmetrically distributed around a zero mean, such that the influence over a sufficiently large number of voters results in

[3]In our terms this means either, that the number of voxels belonging to a different structure but having a similar intensity (and therefore a high voting weight) is low, or the intensity difference is sufficiently large, such that their influence is lowered by a low voting weight.

a growing of the voting tensor in intensity dimension but maintains the spatial structure (and we can interpret the features directly by their projection due to an intensity angle θ_I close to 90°). The experiments have shown that the symmetry actually also applies to the tubular structures for which points on the centreline can exhibit inhomogeneous intensity distributions in the noise-free case as well due to the particular intensity profile if the latter is not totally flat.

In many TV applications, feature detection for a specific dimensionality is usually performed by investigating the corresponding saliency map in terms of maxima detection. Unfortunately, finding maxima as proposed in [21] can yield many false positives in noisy images. Instead, we classify the overall shape of a tensor by only looking at the maximal saliency at a point. This has empirically given promising results in terms of detecting the central points of tubular structures in our experiments.

For the purpose of detecting a tubular structure, we search for a tensor to have the curve feature as its dominant structure (i.e. curve saliency is the largest among the saliency maps). According to the interpretations in Table 1, we can derive the set of candidate points as either

$$c_3 = \{x \in \Omega_I^3 \mid \arg\max_i(s_i(x)) = 3 \ \wedge \ \left|\theta_I^{(3)}(x)\right| < \alpha_{threshold}\} \tag{6}$$

for homogeneous intensity interpretation or

$$c_2 = \{x \in \Omega_I^3 \mid \arg\max_i(s_i(x)) = 2 \ \wedge \ \left|\theta_I^{(2)}(x) - 90°\right| < \alpha_{threshold}\} \tag{7}$$

for inhomogeneous intensity interpretation of the output tensors. In both cases s_i denotes the i-th saliency map, $\theta_I^{(i)}$ the angle of the corresponding i-dimensional normal space w.r.t. the intensity axis $\hat{\mathbf{e}}_I$ and Ω_I^3 is the set of voxels at which the input image is defined (i.e. g is the identity in that case).

For points in c_3 the three-dimensional feature is dominant. In that situation, the normal space is three-dimensional and contains $\hat{\mathbf{e}}_I$ (for a reasonable $\alpha_{threshold}$). Therefore, the tangent space is 'purely spatial' and we can retrieve an estimate for the tube orientation by the projection of the 4D tangent onto the three-dimensional spatial subspace.

In case of points in c_2, curve structures are represented by the 2D feature exhibiting a large angle $\theta_I^{(2)}$ of its two-dimensional normal space w.r.t. the intensity dimension, i.e. its dimensions are 'purely spatial'. Here, we project the two spanning eigenvectors $\hat{\mathbf{e}}_1$ and $\hat{\mathbf{e}}_2$ onto the spatial dimensions and compute the estimate of the orientation of the tube structure as the normal of those projections in $\mathbb{R}^3$.

4 Experimental Results

The proposed 4D approach offers basically six different parameters that influence its behaviour. First, the scaling of the input data plays of course an important role, as it defines angles and distances in the created 4D space which are crucial for the perceptual rules enforced in the voting algorithm. In order to be able to choose the parameters consistently for our experiments on real data, the intensity range of the input data was scaled to the interval $[0, 255]$. Further, a parameter w_I was used for an additional weighting of the intensity dimension. Although it gives the opportunity to steer the behaviour of the algorithms, it was set to 0.8 throughout our experiments in order to fix its influence. However, the effect of w_I on the results of both voting approaches remains to be investigated in later studies.

Other parameters are the inherent tensor voting parameters for weighting distance and curvature in the decay function. As mentioned earlier, we do not regard the influence of the curvature in intensity direction at this point and set $b_I = 0$ for all experiments. As in our current focus (blood vessels, i.e. tubes exhibiting relatively low curvature on the scale we look at them) curvature is not of central importance, we further keep the spatial curvature weight $b_{space} = 20$ fixed in the following investigations. Thus, the main parameters of the method are σ_{space} and σ_I.

Finally, the value of $\alpha_{threshold}$ employed for the detection of the tube candidates obviously influences the performance of the detectors c_2 and c_3. However, in combination with the detection approaches introduced in Sect. 3 the choice of $\alpha_{threshold}$ is restricted.[4] Empirically, we have chosen $\alpha_{threshold} = 5°$ for our experiments in this chapter.

In all our experiments we employ two stages of tensor voting. First all input sites are encoded as isotropic tensors and the corresponding initial voting step is referred to as *ball voting*. The subsequent voting step computes votes for each basic tensor[5] and is referred to as *generic voting* [23].

4.1 4D Tensor Voting

4.1.1 Relationship Between Intensity Weight σ_I and Noise Level

In order to investigate the validity of our interpretation of the 4D tensors given in Table 1, we analyse the transition of the tensors' shape in synthetic data of homogeneous intensity under influence of different levels of additive Gaussian noise. In this scenario, the effect of the noise level on the estimation of a homogeneous intensity volume structure can be separately analysed. In particular,

[4] It has to be chosen reasonably small such that the mentioned projections make sense.

[5] $\mathsf{T}_i = \sum_{j=1}^{i} \mathbf{e}_j \mathbf{e}_j^{\mathrm{T}}$ is the i-th basic tensor of T where $\mathbf{e}_j$ are the (ordered) eigenvectors of T.

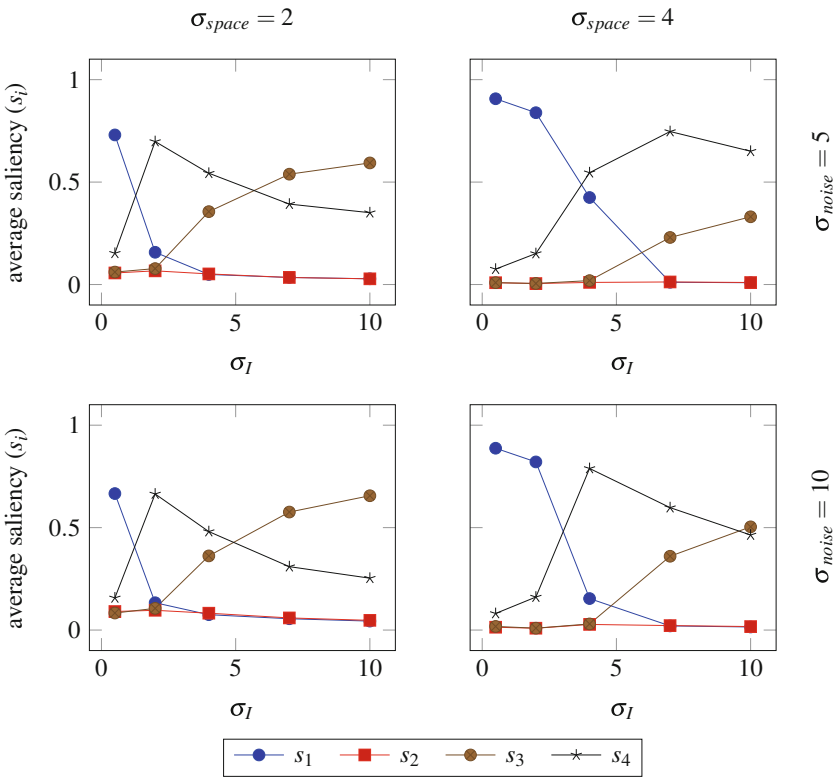


Fig. 2 Effect of noise on tensor voting results in homogeneous intensity regions. Depicted are the average saliencies of the tensors in a $7 \times 7 \times 7$ neighbourhood of constant intensity with additive Gaussian noise of standard deviation $\sigma_{noise} = 5$ (*first row*) and $\sigma_{noise} = 10$ (*second row*) after two 4D voting passes (ball voting, generic voting)

we can see for which settings the interpretation of homogeneous (*first row*) and inhomogeneous (*second row*) intensities is applicable.

Figure 2 shows the transition of the different saliency maps s_i with a varying intensity weight σ_I for specific, constant spatial scale σ_{space}. The saliency curves are computed as the average saliency of the tensors within a $7 \times 7 \times 7$ neighbourhood. The results for small intensity weights correspond to the interpretations for "homogeneous intensities" in the *first row* of Table 1, larger ones agree with the interpretations given in the table's *second row*. Especially for $\sigma_{space} = 2$ a transition from a dominant one-dimensional shape via a four-dimensional to finally a three-dimensional shape can be observed. Further, the graphs' shapes appear to be scaled along σ_I with growing σ_{space}. This suggests that the expected tensor shape might be predicted based on the ratio of σ_I and σ_{space}. The comparison between the results on different noise levels indicates that this transition appears for a smaller range of intensity weights in case of higher noise levels.

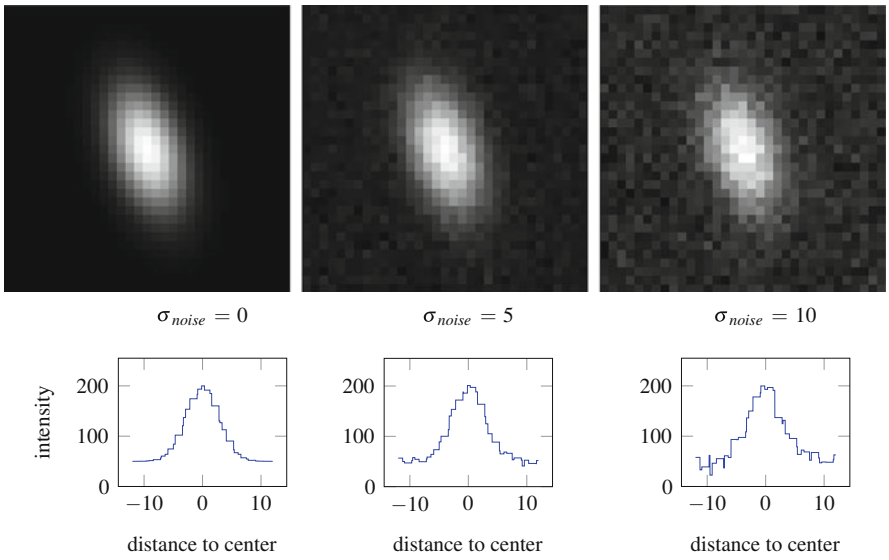


Fig. 3 Synthetic tube data with Gaussian intensity profile ($\sigma_{tube} = 4$) and additive Gaussian noise of different standard deviations (σ_{noise}). *First row*: Exemplary slices in the middle of the volume. Note, that the tube is not oriented perpendicular to the shown slice. *Second row*: Intensity profiles perpendicular to tube direction (interpolated using *nearest neighbours*)

4.1.2 Estimation of Vessel Centre Candidates

In order to quantify the performance of the proposed approaches we apply them to synthetic test data. The latter contains a tube exhibiting a Gaussian intensity profile ($\sigma_{tube} = 4$) perpendicular to its centreline. Additive Gaussian noise of different standard deviations is imposed. The test data is visualised in Fig. 3.

As mentioned earlier we aim at detecting tubular structures as curve features (in the sense of spatial curves, i.e. curves in 3D). For that reason, we have to choose the spatial scale of tensor voting (σ_{space}) larger than the actual tube width in order to arrive at the expected feature at the tube centre. Since the tube's intensity profile is not a step function, the actual width depends on what intensity range (given by σ_I) we consider as tube interior. In order not to invalidate this assumption for some choices of σ_I we choose $\sigma_{space} = 4$ in our further experiments.

Figure 4 shows the resulting saliency maps of tensors along a line across the tube's centre perpendicular to its orientation. Applying the interpretations of Table 1 according to our findings from Sect. 4.1.1, we see that the particular curve saliency (s_3 in case of $\sigma_I = 1$, s_2 otherwise) is the dominating saliency at the tube centres which is the basic requirement for the tube detectors in Eqs. (6) and (7) to work.

In order to compare the results of the proposed tube detectors, c_3 and c_2, we investigate them in terms of their specificity in classifying a point as lying close to the tube centre. For that purpose, we define the interior of the vessel as the area of radius two voxel lengths around the centreline. Detected points in c_2 and c_3 are

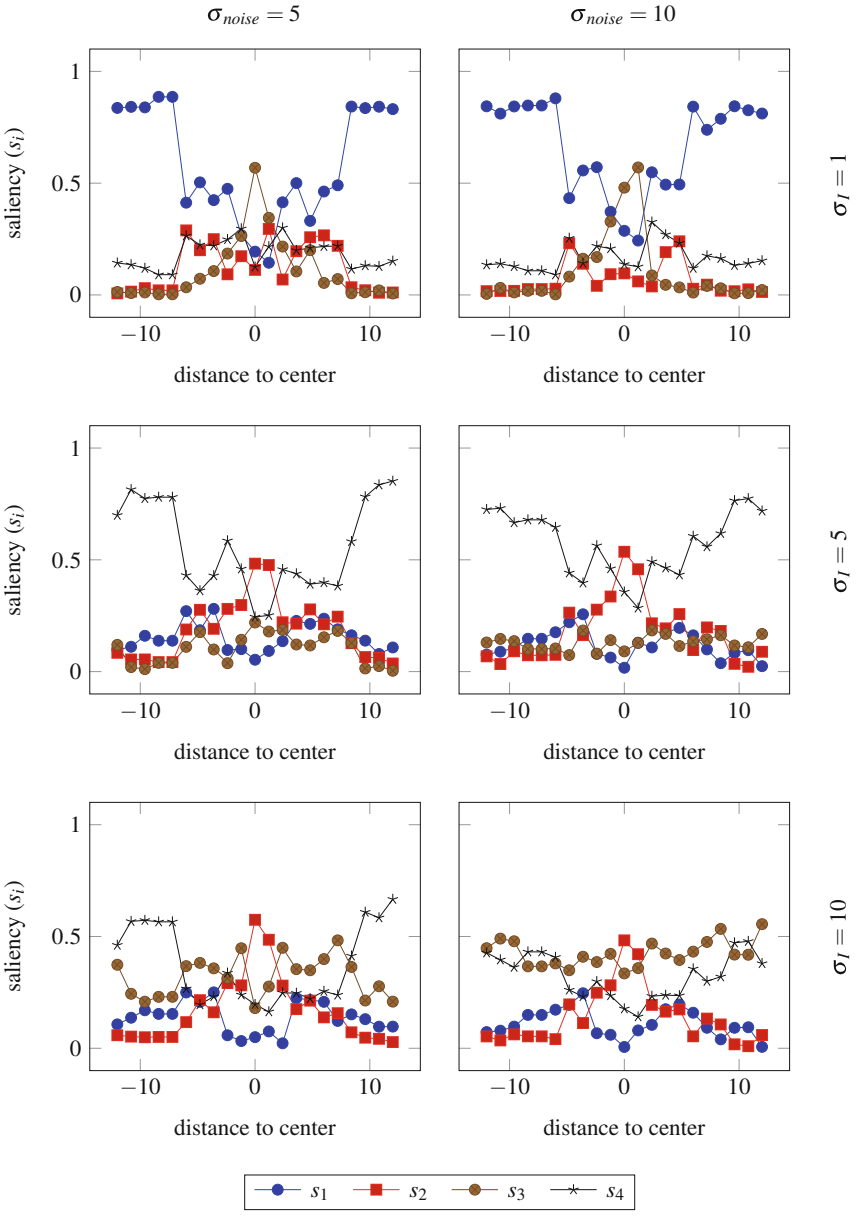


Fig. 4 Saliency profiles s_i perpendicular to tube direction after two passes of 4D tensor voting (ball voting, generic voting) with varying intensity scale σ_I and fixed spatial scale $\sigma_{space} = 4$ on tube data with Gaussian intensity profile ($\sigma_{tube} = 4$). Distance to tube centre is indicated on the horizontal axis

Table 2 Results of tube detection for Gaussian intensity profile ($\sigma_{tube} = 4$) after two passes of 4D tensor voting (ball voting, generic voting, $\sigma_{space} = 4$)

Detector	σ_I	$\sigma_{noise} = 0$			$\sigma_{noise} = 5$			$\sigma_{noise} = 10$		
		inliers	spec	$\overline{\Delta\theta}$	inliers	spec	$\overline{\Delta\theta}$	inliers	spec	$\overline{\Delta\theta}$
c_3	0.5	124	0.19	3.68	136	0.72	14.48	113	0.75	18.45
	1	218	0.52	3.00	149	0.89	9.93	101	0.92	12.73
	1.5	248	0.53	3.71	143	0.89	9.61	98	0.94	11.14
	2	200	0.71	3.38	79	0.94	9.08	62	0.96	10.85
	3	4	1.00	0.71	6	1.00	10.83	6	0.86	10.15
	4	0	–	–	0	–	–	1	1.00	16.24
	5	0	–	–	0	–	–	0	–	–
	10	0	–	–	0	–	–	0	–	–
c_2	0.5	0	–	–	0	–	–	0	–	–
	1	0	–	–	0	–	–	0	–	–
	1.5	0	–	–	0	–	–	0	–	–
	2	4	1.00	0.82	0	–	–	0	–	–
	3	62	1.00	1.84	8	1.00	3.35	2	1.00	5.25
	4	141	0.72	2.66	50	0.98	4.63	25	0.84	6.62
	5	177	0.62	3.09	123	0.9	5.44	84	0.93	7.06
	10	231	1.00	3.06	220	0.99	4.41	156	1.00	5.79

Points are classified based on detectors c_3 (hom. intensity, Eq. (6)) and c_2 (inhom. intensity, Eq. (7)) with $\alpha_{threshold} = 5°$. Detected points exhibiting a distance to the tube centre of less then 2 voxel lengths are treated as *inliers*. The ratio between inliers and tube candidates is denoted *spec*, the mean angular error between the estimated tube orientation for inliers and ground truth is $\overline{\Delta\theta}$ (in degrees)

called *tube candidates* of which those lying within the defined range are defined to be *inliers*. Based on that, the specificity (*spec*) is computed as the ratio between number of *inliers* and number of *tube candidates*.

Table 2 shows the results of both detectors ($\alpha_{threshold} = 5°$) for different intensity weights in the common setting. Based on the number of inliers in all noise cases it can be concluded that the "homogeneous intensity" detector c_3 is applicable up to $\sigma_I = 2$ whereas c_3 is sensitive for $\sigma_I > 3$ and there is no overlap where both detectors are sensitive to tube structures in our test case. This transition at $\sigma_I \approx 3$ corresponds with the point in Fig. 2 where the dominant saliency switches from s_1 to s_4. As expected the number of inliers decreases with higher noise levels. Still the computed specificity is high which shows that the tube candidates are usually concentrated around the tube centre. The low specificity in case of $\sigma_{noise} = 0$ in many cases is remarkable. Visual inspection showed, that for c_3 results, outliers are still concentrated around the centre and considering a larger radius would yield a high specificity as well. Opposed to that, in case of c_2, the low values stem from detection artefacts on the outer border of the tube structure which shows, that for those specific intensity weights ($\sigma_I = 4/5$) in combination with the chosen tube profile, c_2 is not the right detector in the noise-free case.

For analysing the performance of the approaches in estimating the orientation of detected tube structures, we report the mean angle difference between estimated orientations for *inliers* and the known tube orientation in the synthetic data experiments (cf. column $\overline{\Delta\theta}$ in Table 2). As expected, the estimates are more accurate on lower noise levels. At the same time, inliers detected by c_2 (i.e. larger σ_I) exhibit better accuracy then those of c_3. This might be due to the fact that a low value of σ_I is restricting the range of intensity values that are assigned a high weighting in the voting step. Thus, choosing a small intensity weight results in less received votes contributing to the estimation of the spatial orientation and therefore the influence of noise is larger. For all investigated noise levels $\sigma_I = 10$ for detector c_3 is providing the best performance in terms of the investigated parameters.

To confirm the characteristic behaviours found in the previous synthetic data experiments we apply the algorithm to CT angiography data [12]. Figure 5 shows one exemplary slice of the investigated sub volume of data set 00. Intensity profiles of exemplary vessels exhibiting different characteristics in terms of contrast and size in the presented slice are depicted. Compared to the Gaussian profiles in the

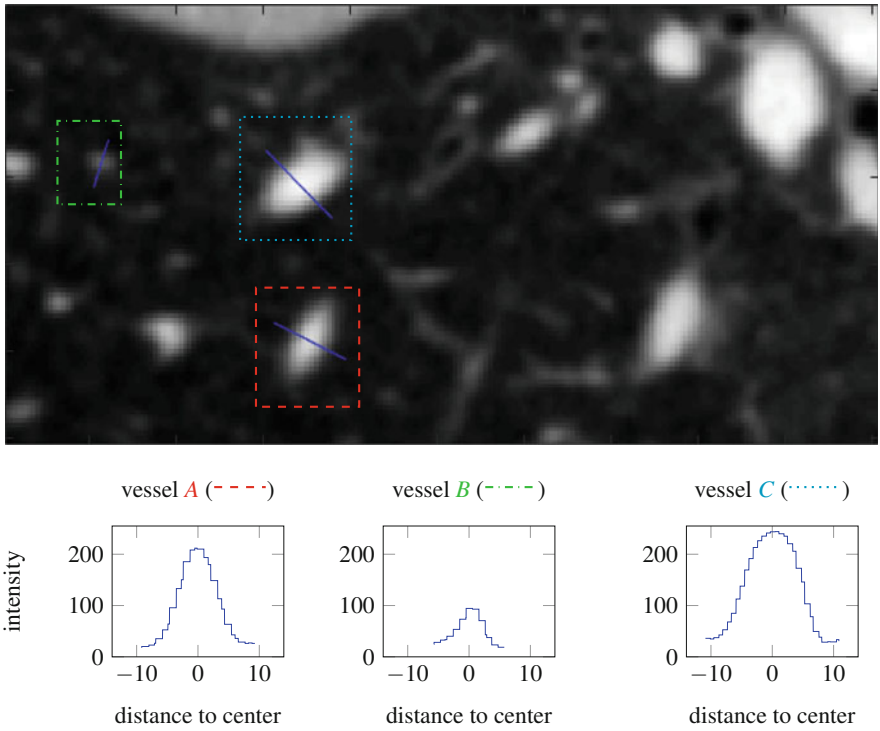


Fig. 5 CTA data set 00 from [12]. One exemplary slice of the investigated sub volume of the chosen data set is shown. The actual intensity profiles (analysed along the *blue lines* within the shown slice using *nearest neighbour* interpolation) for three vessels with different characteristics are plotted below. Note, that the profiles are shown after rescaling (cf. beginning of Sect. 4)

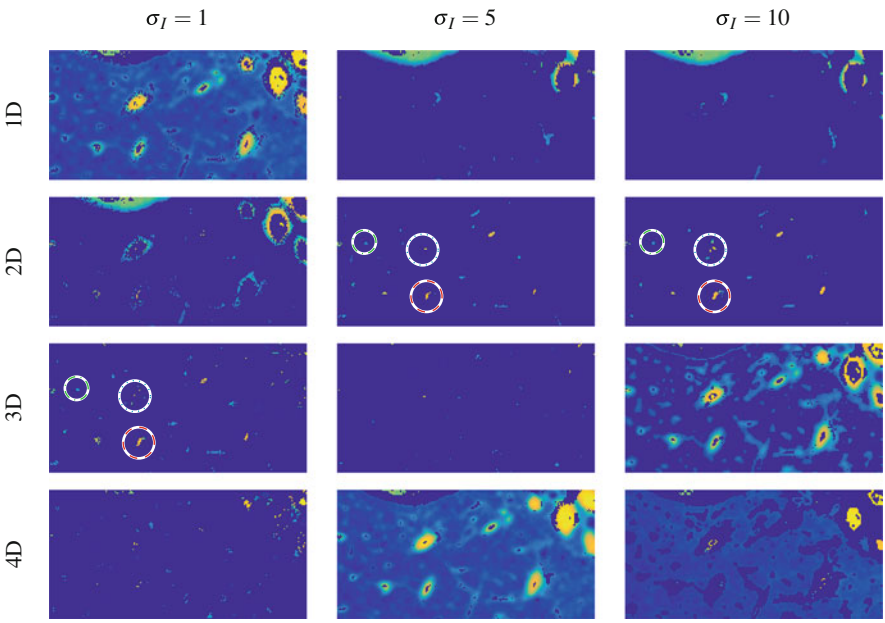


Fig. 6 Classification of salient structures in exemplary slice of CT data set after two passes of 4D tensor voting (ball voting, generic voting) with scale $\sigma_{space} = 4$. In the i-th row only those voxels of the input data for which s_i is the largest saliency map are shown. In case of $\sigma_I = 1$ tube candidates can be detected as 3D structures, whereas for $\sigma_I = 5$ and $\sigma_I = 10$ tube candidates are found as 2D structures (cf. Table 1). The *circles* mark the tube candidates belonging to the three vessels highlighted in Fig. 5. Note, that we use a different colormap in this figure in order to obtain better contrast in dark regions

synthetic data, two (vessels A and C) show a slightly larger, though comparable, difference between background and vessel centre intensity as well as a slightly steeper slope on the borders. Opposed to that, vessel B has clearly less contrast and a smaller width. Visual comparison suggests most similarity with the synthetic profile for the medium noise level of $\sigma_{noise} = 5$.

The results of structure classification after two passes of 4D tensor voting on the presented data set are shown in Fig. 6 for the same exemplary slice as in Fig. 5. The i-th column is depicting only those voxels from the original image data that exhibit the saliency map s_i as the maximal saliency. A similar classification behaviour in relation to the intensity weight σ_I as seen in the previous experiments can be observed. For $\sigma_I = 1$, the background (i.e. volume structure) is detected by a dominant 1D tensor shape, for $\sigma_I = 5$, it is visible in the 4D map and for $\sigma_I = 10$, the larger intensity inhomogeneities (mainly large gradients at structure borders) are captured in the 3D feature map. It can be seen that spatial curve candidates are detected as expected by the 3D (for $\sigma_I = 1$) respectively 2D (for $\sigma_I = 5/10$) feature maps. However, especially for $\sigma_I = 5$, comparison of the holes in the 4D map with the 2D and 3D feature maps indicate that vessels with a low contrast are captured in

the three-dimensional saliency map instead of the 2D one. That means that for those vessels the regarded intensity range determined by the intensity weight is too large to obtain a maximal curve saliency at their centres. This suggest potential to obtain better tube detection in this case by either combining both features or by employing a more advanced detection algorithm.

Focusing on the three vessels marked in Fig. 5, it can be seen that while vessels *A* and *B* show curve structure candidates (in terms of Table 1) close to their centres for all tested values of σ_I in Fig. 6, vessel *C* does not clearly exhibit the expected structure estimate at its centre. This is likely to be due to its larger width such that the spatial scale of σ_{space} is not sufficient to detect the vessel as a curve structure especially for larger σ_I.

Visual inspection of the orientation estimation results at detected tube candidates reveals that the estimates exhibit consistent orientations (cf. Fig. 7b). This supports

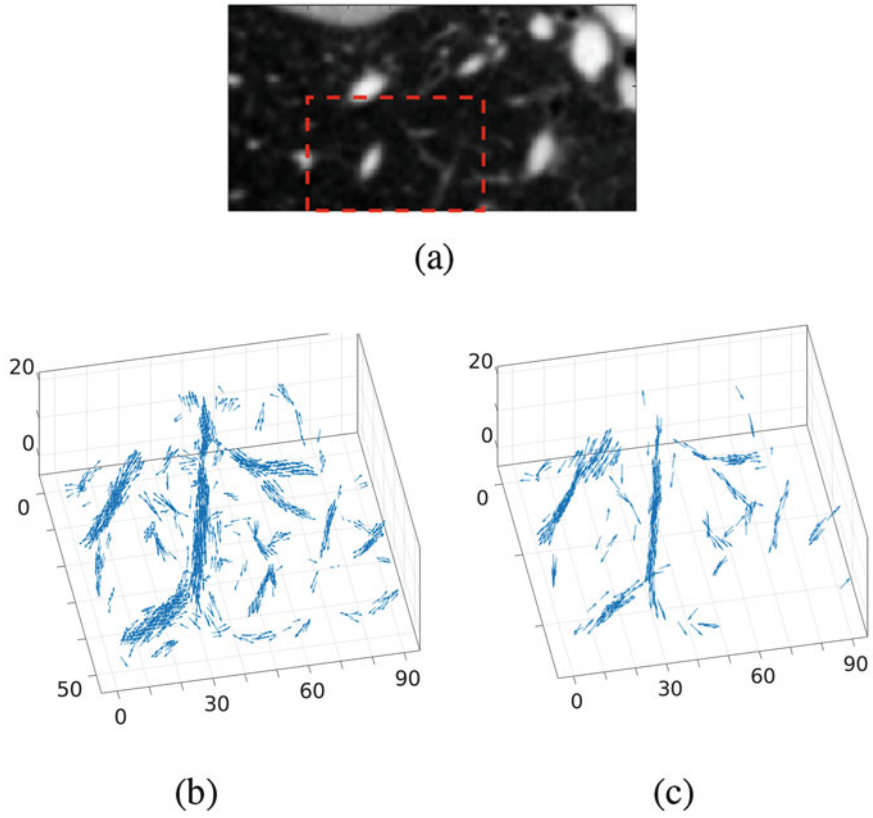


Fig. 7 Estimated orientations for detected tube candidates in CTA data ($\sigma_{space} = 4$, $\sigma_I = 5$). The visualisations on the right show results in 17 subsequent slices of the area indicated on the *left*. (**a**) slice 8 of 17 of CTA input data (**b**) estimated orientations for tube candidates after 4D tensor voting (detected by c_2) (**c**) estimated orientations for tube candidates after 3D tensor voting

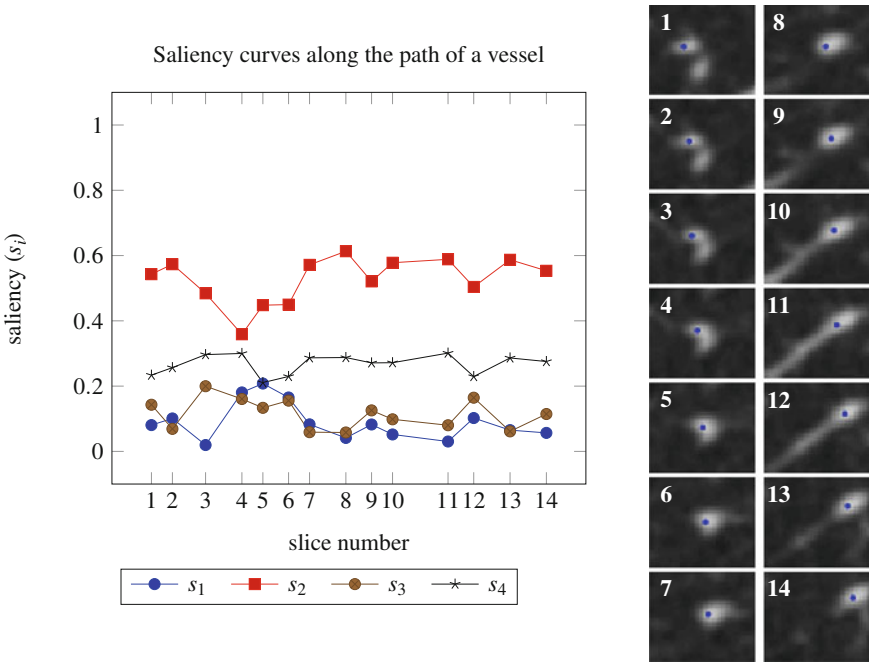


Fig. 8 Saliency curves along the path of a vessel in vicinity of a branching. On the *right*, 14 slices from data set 00 are shown from a region containing the branching of two pairs of vessels, whereby the *blue dots* mark the manually defined centreline points (one per slice). On the *left*, for each of those points the saliencies s_i of the particular tensor at such a point are plotted ($\sigma_{space} = 4, \sigma_I = 5$). Focussing on the detection of curve structures in the given setting, a drop of s_2 can be observed for slices 3–6 (A) and 9, 12 (B) which could be related to the presence of a branching vessel joining the current path. A slight increase of s_1 for those slices could indicate the more flat structure of a branching compared to a single vessel. Regard the different localisations of the two bifurcations in the graph of s_2 (among 4 (A) and 2 (B) slices, respectively). These could possibly be explained by the different angles of the bifurcations and the orientations of the vessels with respect to the image plane. Note that the distance between subsequent labels of the horizontal axis corresponds to the path length along the vessel

the conclusions from synthetic experiments that tube orientations at the detected candidates are reasonably accurate.

For the purpose of analysing blood vessels, the method is not only required to detect the centre of regular, tube-like structures, but also needs to handle situations which deviate from the assumptions introduced in Sect. 3. A possible approach in such cases could be to analyse how the tensor shape changes when following a path close to the centreline of a vessel. In Fig. 8 the evolution of the different saliency curves along a vessel exhibiting two branching situations is shown. Comparing these curves with the image slices in that region (on the right in Fig. 8), one can observe a lowered curve saliency, i.e. s_2, in slices 3–6 as well as in slices 9 and 12 corresponding with the appearance of a branching in the particular slices. Further, the slightly increased surface saliency, s_1, indicates the presence of a more flat

structure in those slices which could be the plane which is locally defined by the direction of the involved vessel branches. Based on those observations, the combination of s_1 and s_2 shows potential for being the basis of a branching detector usable for vessel tracking.

4.2 Comparison of 4D and 3D Tensor Voting Approaches

We compare the 4D approach with the simplified 3D formulation in terms of their particular results in the presented experiments. As can be seen from Fig. 9, the influence of parameter σ_I is not as essential as in the 4D approach. In all shown settings the 3D saliency (indicating the volume structure) is significantly larger than the other saliencies maps. In particular for $\sigma_{space} = 4$ the curves are more or less saturated for $\sigma_I \geq 5$.

Similar stability of the 3D tensor voting results with respect to the intensity weight can be observed for the saliency profiles on the synthetic tube data set (Fig. 10). Focusing on the tube centre candidates which we detect as curves (i.e.

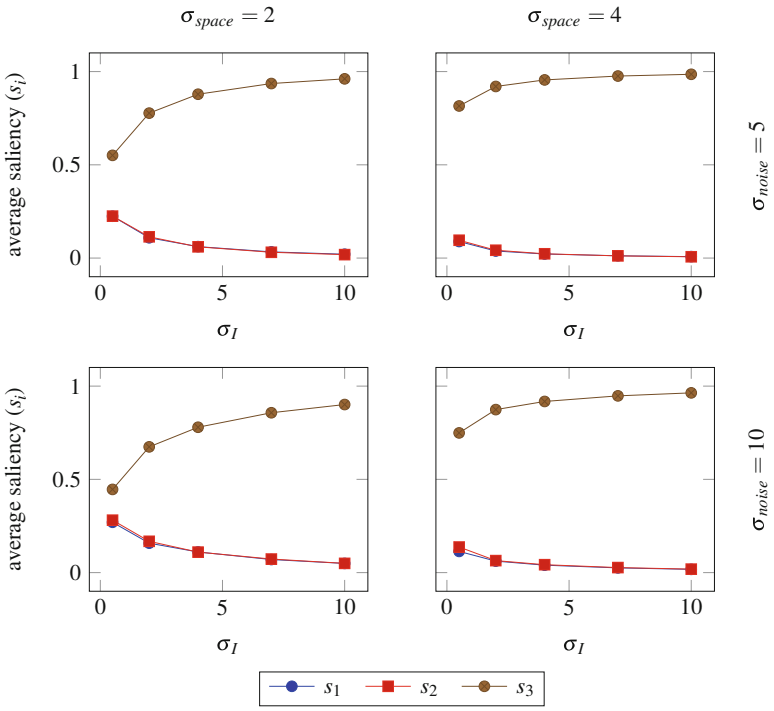


Fig. 9 Effect of noise on tensor voting results in homogeneous intensity regions. Depicted are the average saliencies of the tensors in a $7 \times 7 \times 7$ neighbourhood of constant intensity with additive Gaussian noise of standard deviation $\sigma_{noise} = 5$ (*first row*) and $\sigma_{noise} = 10$ (*second row*) after two 3D voting passes (ball voting, generic voting)

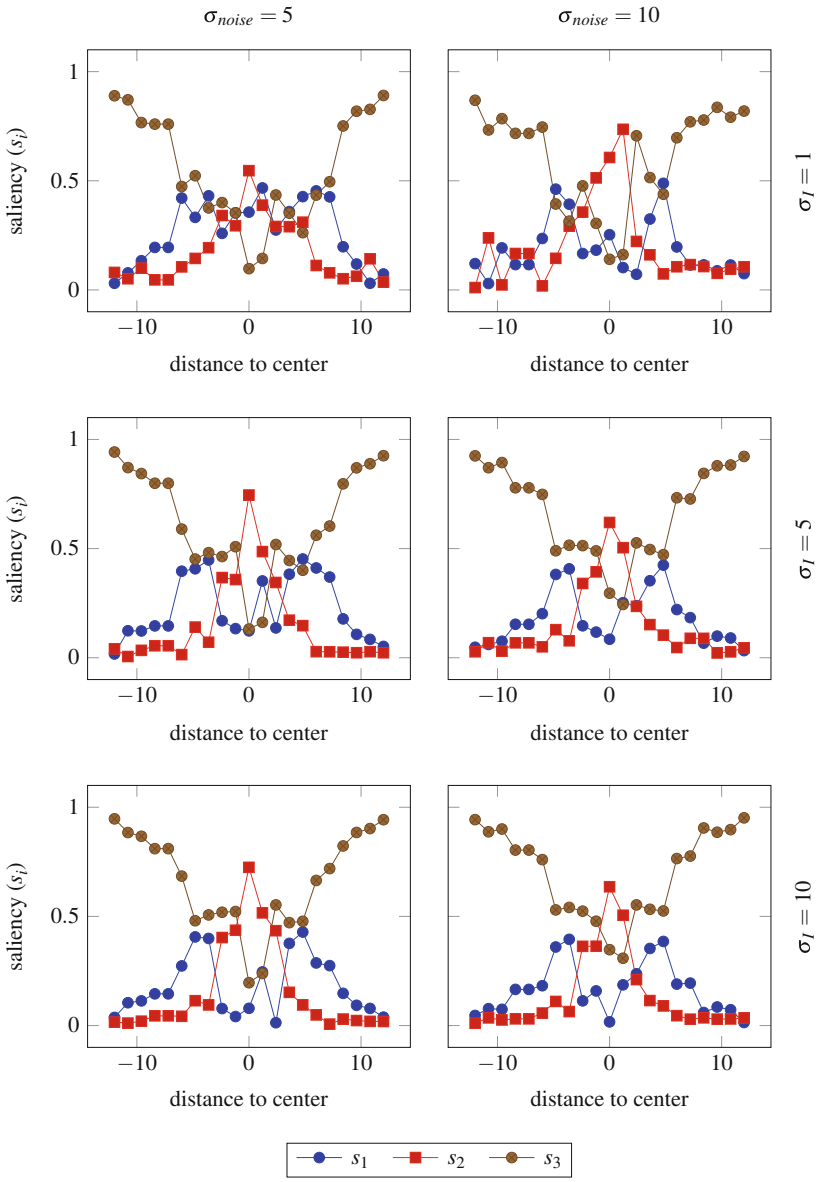


Fig. 10 Saliency profiles s_i perpendicular to tube direction after two passes of 3D tensor voting (ball voting, generic voting) with varying intensity scale σ_I and fixed spatial scale $\sigma_{space} = 4$ on tube data with Gaussian intensity profile ($\sigma_{tube} = 4$). Distance to tube centre is indicated on the horizontal axis

Table 3 Results of tube detection for Gaussian intensity profile ($\sigma_{tube} = 4$) after two passes of 3D tensor voting (ball voting, generic voting, $\sigma_{space} = 4$)

	Noise-free			$\sigma_{noise} = 5$			$\sigma_{noise} = 10$		
σ_I	inliers	spec	$\overline{\Delta\theta}$	inliers	spec	$\overline{\Delta\theta}$	inliers	spec	$\overline{\Delta\theta}$
0.5	85	0.12	6.28	188	0.42	18.51	180	0.29	20.51
1.0	262	0.53	4.45	208	0.79	12.10	162	0.51	14.94
1.5	312	0.80	3.98	220	0.95	9.81	167	0.64	12.67
2.0	335	0.88	3.67	226	0.98	8.65	168	0.74	10.71
3.0	335	0.93	2.94	250	1.00	6.85	176	0.89	8.14
4.0	334	0.99	2.61	254	1.00	5.90	187	0.94	7.01
5.0	323	1.00	2.52	253	1.00	5.02	194	0.98	6.42
10.0	273	1.00	2.66	241	1.00	3.60	202	1.00	4.78

Points at which s_2 (i.e. curve saliency) is largest among the saliency maps are classified as tube candidates. Detected points exhibiting a distance to the tube centre of less then 2 voxel lengths are treated as *inliers*. The ratio between inliers and tube candidates is denoted *spec*, the mean angular error between the estimated tube orientation for inliers and ground truth is $\overline{\Delta\theta}$ (in degrees)

dominant 2D tensor shape) the 2D saliency is the maximal saliency map close to the tube centre in all investigated settings. Compared to the results after 4D tensor voting, the peaks of s_2 at the centre are more distinct (i.e. larger than 0.5) than those of the corresponding saliencies in Fig. 4. Generally, the shown saliency curves tend to be smoother with larger intensity weight which is due to the larger number of votes due to a larger acceptable intensity range (cf. explanation in Sect. 4.1.2).

Table 3 shows the quantitative results of tube candidate detection and orientation estimation for two passes of 3D tensor voting. Note that results for only one detector are available as we base our detection in the 3D case on finding voxels that exhibit a dominant 2D shape. Generally, the number of detected tube candidates is larger for smaller noise levels. While specificity is increasing for all noise levels the number of inliers seems to exhibit a peak. In the noise-free case, this appears to be the case for $\sigma_I \in [2, 4]$ and in case of $\sigma_{noise} = 5$ for $\sigma_I \in [3, 5]$, i.e. slightly shifted. From our experiments this behaviour cannot be observed for $\sigma_{noise} = 10$ in the tested settings. Nevertheless, this might still be observed for experiments with a more dense sampling of the intensity weights for $\sigma_I > 5$.

The accuracy of estimating the tube orientation is (especially in the noise cases) best for $\sigma_I = 10$. Compared to the results of 4D tensor voting, the 3D approach seems to provide slightly better performance in terms of number of inliers and accuracy of the orientation estimation. Nevertheless, both approaches show a satisfying accuracy in estimating the tube's orientation.

In agreement with the synthetic experiments, the classification results for the 3D approach on the real data set (cf. Fig. 11) show an increasing noise robustness with larger σ_I. However, the 2D feature map contains less vessel regions for $\sigma_I = 10$ compared to the smaller intensity weights, i.e. there is a trade-off between robustness and sensitivity along with that parameter. This corresponds to the results from Table 3 which suggested a maximum of detected inliers to appear for $\sigma_I < 10$.

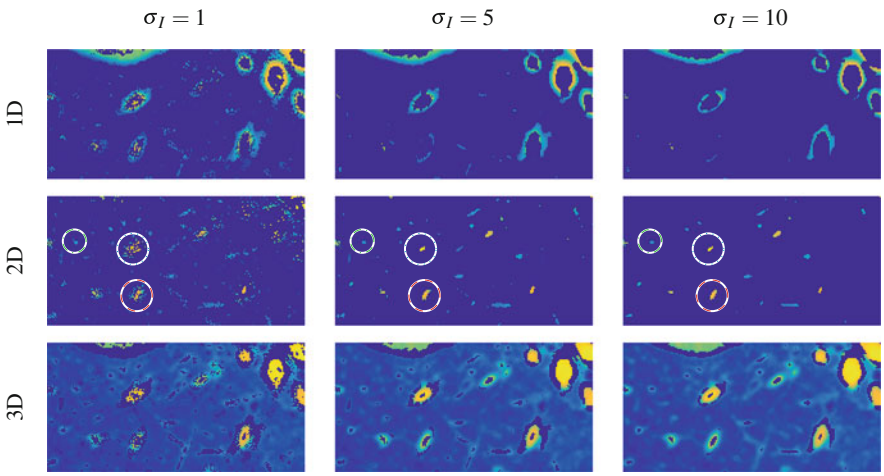


Fig. 11 Classification of salient structures in exemplary slice of CT data set after two passes of 3D tensor voting (ball voting, generic voting) with scale $\sigma_{space} = 4$. In the i-th row only those voxels of the input data for which s_i is the largest saliency map are shown. Tubular structures can be detected as salient 2D structures. The *circles* mark the three vessels highlighted in Fig. 5. Note, that we use a different colormap than in Fig. 5 to obtain better contrast in dark regions

Compared to the 4D voting results, the classification seems to be sensitive to a larger number of vessels than the 4D approach considering only the 2D feature map as suggested by Table 1. Comparing the 3D feature map in Fig. 11 and the 4D feature map in Fig. 6 for $\sigma_I = 5$, they seem to provide similar information, which indicates the potential to obtain better tube detection results if considering both 2D and 3D information for tube detection in 4D or applying a more sophisticated detection algorithm. Another observation from comparing the 3D and 4D feature maps of both approaches is that the 4D detections often seem to be more concentrated around vessel centres. Concerning the orientation estimation shown in Fig. 7c, the estimates show a satisfying consistency similarly to the 4D results.

5 Discussion

In this paper we have proposed a direct way to include intensity information from grey value images in a tensor voting formulation with the aim to extract tubular structures from image volumes. In contrast to previous approaches that focussed on adjusting the input data of tensor voting regarding information from the image intensities, our formulations incorporate intensity information in the core of the voting step and by that regard it as an additional feature within the method. With the intention to stay close to the original formulation of tensor voting we propose a 4D tensor voting approach which includes intensity information in both the tensor

representation as well as the weighting function of the voting step while at the same time suggests a straightforward simplification to a 3D voting formulation. Our experiments show that both algorithms have the ability to detect tubular structures and directly estimate their orientation based on the assumption to find them as spatial curve structures after tensor voting.

Generally, both approaches (3D and 4D) show a similar performance in terms of detection of tube structures and their orientation estimation. Nevertheless, the 3D approach seems to be more robust towards noise effects in terms of both comparison criteria (given the simple detection approach of just finding the dominant tensor shape). In case of the 3D approach, an increased robustness along with increasing intensity weight could be observed which we explain with a larger number of received votes per voxel for larger σ_I. Accordingly, the different noise robustness of the 4D approach might be induced by the additional weighting of the spatial dimensions due the additional intensity dimension in the four-dimensional voting space (cf. Sect. 2.4). This effect is more apparent in case the intensity differences are significantly larger than the corresponding spatial distances. Therefore imposing noise on a region of homogeneous intensities is directly inducing (increasing) this weighting effect even without any influence of the intensity weight σ_I in the actual weighting function. The influence of this effect can be adjusted by the scaling of the input data (and therefore scaling of the relative intensity differences) by parameter w_I which will be subject of future investigations.

Nevertheless, the 4D approach showed—based on the simple detection algorithm—a slightly higher concentration of detections around the tube centres which might be of advantage when aiming at extracting only the centreline of a tubular structure. Still, it has to be analysed further if this selectiveness is due to a higher sensitivity to noise in the vicinity of the centre or if the tensor structures in four dimensions are better suited to detect the centre only. In this context, also the role of parameter $\alpha_{threshold}$ employed in both detectors (c_2 and c_3) in terms of its influence on the detection results might be of interest for further investigation. Possibly, by including the angle $\theta_I^{(f)}$ in the interpretation of the 4D tensors in a more sophisticated manner (instead of considering only those close to 0° and 90°) one could exploit encoded information more efficiently.

Another possible advantage of the 4D approach could be the opportunity to bridge gaps in tube structures caused by abrupt change of intensity in direction of the centreline. Aiming at the detection of stenoses in blood vessels that might cause this effect, a subsequent dense voting step[6] could close gaps of a certain size. Then, by analysing the saliencies along the intensity axis such bridged gaps could be found at spatial positions which exhibit multiple saliency maxima. This would include both detection and handling of such abnormalities in the data within the tensor voting framework.

[6]dense especially along intensity axis

It must be noted that the presented results partly rely on the quality of the employed tube detectors which are of simple type. Future work will focus on exploiting more information provided by the voting results for the structure detection. Considering that the 4D tensors inherently contain more information than those in the 3D formulation,[7] more advanced strategies for detecting tubular structures based on 4D tensors could potentially lead to better results in terms of centre detection and noise robustness in general. The fact that the combined 2D and 3D feature maps of the 4D voting results shown in Fig. 6 for $\sigma_I = 5$ already seem to offer more information for the detection, further supports that hypothesis. An approach relying on saliency curves (cf. Fig. 8) seems promising in an algorithm for both centreline detection as well as handling of bifurcations. A thorough analysis of the methods at stenoses is part of our current research.

An important property of methods for vessel detection is the opportunity for multi scale analysis. Often, Hessian-based approaches require several passes with a different scale parameter in order to be sensitive to vessels of different sizes. In general, the two presented approaches show the potential to be sensitive to tubular structures on several scales during one execution of the algorithm. As mentioned in Sect. 4.1.2, the minimum requirement for a vessel to be detected is the spatial scale σ_{space} to be chosen larger than the actual vessel width. The resulting tensor shapes (cf. Figs. 6 and 11) show that for one pass of the algorithm on a specific scale vessels of several sizes exhibit distinct tensor shapes. However, especially the choice of the spatial curvature weight b_{space} as well as the intensity scale σ_I are likely to affect the algorithms' ability to extract vessels of different size. This interplay between those parameters as well as the usefulness of regarding curvature in intensity direction steered by parameter b_I in the weighting function, will be subject to future investigations. In that context, also the detection of end points of the smallest vessels in the tree is of interest.

In view of the tensor voting adaption proposed in [26], the proposed weighting function in Eq. (4) could be a reasonable extension to that algorithm in order to make the core of it aware of the underlying image structure as well. By that, its weakness of tensor propagation from structured to unstructured regions could possibly be approached.

6 Conclusion

Based on the experiments presented in this chapter, the comparison of the 3D and 4D approaches suggests to favour the 3D approach over the 4D formulation for the purpose of tube detection. Besides the slightly better results in our investigated settings the shorter computation depicts a strong argument for that choice. However,

[7] Even though the 3D approach regards intensity in the voting process, the tensor representation does only provide information about spatial structure.

considering a tracking approach exploiting the sparse nature of blood vessel structures could render the computational effort for both approaches reasonable. In spite of that, the 4D tensor voting approach could be a reasonable alternative in order to take advantage of a potential extension for detection and handling of shape abnormalities.

Acknowledgements We thank Hortense Kirisli, Theo van Walsum and Wiro Niessen for providing the CTA data used in the experiments. This research has been supported by the Swedish Research Council (VR), grants no. 2014-6153 and 2012-3512, and the Swedish Heart-Lung Foundation (HLF), grant no. 2011-0376.

References

1. Aylward, S., Bullitt, E., Pizer, S., Eberly, D.: Intensity ridge and widths for tubular object segmentation and description. In: Proceedings of the Workshop on Mathematical Methods in Biomedical Image Analysis, pp. 131–138. IEEE, Piscataway (1996)
2. Cetin, S., Unal, G.: A higher-order tensor vessel tractography for segmentation of vascular structures. IEEE Trans. Med. Imaging **34**(10), 2172–2185 (2015)
3. Cetin, S., Demir, A., Yezzi, A., Degertekin, M., Unal, G.: Vessel tractography using an intensity based tensor model with branch detection. IEEE Trans. Med. Imaging **32**(2), 348–363 (2013)
4. Cheng, G., Zhu, F., Xiang, S., Pan, C.: Accurate urban road centerline extraction from VHR imagery via multiscale segmentation and tensor voting. Neurocomputing **205**, 407–420 (2016)
5. Dumortier, Y., Herlin, I., Ducrot, A.: 4-d tensor voting motion segmentation for obstacle detection in autonomous guided vehicle. In: 2008 IEEE Intelligent Vehicles Symposium, pp. 379–384. IEEE, Piscataway (2008)
6. Frangi, A.F., Niessen, W.J., Vincken, K.L., Viergever, M.A.: Multiscale vessel enhancement filtering. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 130–137. Springer, Berlin (1998)
7. Franken, E., van Almsick, M., Rongen, P., Florack, L., ter Haar Romeny, B.: An efficient method for tensor voting using steerable filters. In: Computer Vision–ECCV, ECCV'06, pp. 228–240. Springer, Berlin (2006)
8. Fridman, Y., Pizer, S.M., Aylward, S., Bullitt, E.: Segmenting 3D branching tubular structures using cores. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 570–577. Springer, Berlin (2003)
9. Guest, I.W.: Tensor voting on sparse motion vector estimation. In: Proceedings of the Fifteenth Annual Symposium of the Pattern Recognition Association of South Africa (PRASA), pp. 7–11 (2004)
10. Jia, J., Tang, C.K.: Image repairing: Robust image synthesis by adaptive nd tensor voting. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Proceedings, vol. 1, pp. 643–650. IEEE, Piscataway (2003)
11. Kim, H.S., Choi, H.K., Lee, K.H.: Feature detection of triangular meshes based on tensor voting theory. Comput. Aided Des. **41**(1), 47–58 (2009)
12. Kirişli, H., Schaap, M., Metz, C., Dharampal, A., Meijboom, W.B., Papadopoulou, S., Dedic, A., Nieman, K., De Graaf, M., Meijs, M., et al.: Standardized evaluation framework for evaluating coronary artery stenosis detection, stenosis quantification and lumen segmentation algorithms in computed tomography angiography. Med. Image Anal. **17**(8), 859–876 (2013)
13. Kustra, J., Jalba, A., Telea, A.: Robust segmentation of multiple intersecting manifolds from unoriented noisy point clouds. In: Computer Graphics Forum, vol. 33, pp. 73–87. Wiley Online Library (2014)

14. Leng, Z., Korenberg, J., Roysam, B., Tasdizen, T.: A rapid 2-D centerline extraction method based on tensor voting. In: 2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 1000–1003 (2011)
15. Lesage, D., Angelini, E.D., Bloch, I., Funka-Lea, G.: A review of 3D vessel lumen segmentation techniques: Models, features and extraction schemes. Med. Image Anal. **13**(6), 819–845 (2009)
16. Liu, M.: Efficient segmentation and plane modeling of point-cloud for structured environment by normal clustering and tensor voting. In: 2014 IEEE International Conference on Robotics and Biomimetics (ROBIO), pp. 1805–1810 (2014)
17. Lombardi, G., Casiraghi, E., Campadelli, P.: Curvature estimation and curve inference with tensor voting: a new approach. In: Advanced Concepts for Intelligent Vision Systems, pp. 613–624. Springer, Berlin (2008)
18. Lombardi, G., Rozza, A., Casiraghi, E., Campadelli, P.: A novel approach for geometric clustering based on tensor voting framework. In: Proceedings of the 21st Italian Workshop on Neural Nets WIRN11, pp. 129–138 (2011)
19. Loss, L., Bebis, G., Parvin, B.: Iterative tensor voting for perceptual grouping of ill-defined curvilinear structures. IEEE Trans. Med. Imaging **30**(8), 1503–1513 (2011)
20. Loss, L.A., Bebis, G., Chang, H., Auer, M., Sarkar, P., Parvin, B.: Automatic segmentation and quantification of filamentous structures in electron tomography. In: Proceedings of the ACM Conference on Bioinformatics, Computational Biology and Biomedicine, pp. 170–177. ACM, New York (2012)
21. Medioni, G., Tang, C.K., Lee, M.S.: Tensor voting: theory and applications. In: Proceedings of RFIA, Paris, France, vol. 3 (2000)
22. Miao, Z., Wang, B., Shi, W., Wu, H.: A method for accurate road centerline extraction from a classified image. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **7**(12), 4762–4771 (2014)
23. Mordohai, P., Medioni, G.: Dimensionality estimation, manifold learning and function approximation using tensor voting. J. Mach. Learn. Res. **11**, 411–450 (2010)
24. Moreno, R., Garcia, M., Puig, D.: Robust color image segmentation through tensor voting. In: 20th International Conference on Pattern Recognition (ICPR), pp. 3372–3375 (2010)
25. Moreno, R., Garcia, M., Puig, D., Pizarro, L., Burgeth, B., Weickert, J.: On improving the efficiency of tensor voting. IEEE Trans. Pattern Anal. Mach. Intell. **33**(11), 2215–2228 (2011)
26. Moreno, R., Pizarro, L., Burgeth, B., Weickert, J., Garcia, M.A., Puig, D.: Adaptation of tensor voting to image structure estimation. In: New Developments in the Visualization and Processing of Tensor Fields, pp. 29–50. Springer, Berlin (2012)
27. Mottini, A., Descombes, X., Besse, F.: Axon extraction from fluorescent confocal microscopy images. In: 9th IEEE International Symposium on Biomedical Imaging, pp. 764–767 (2012)
28. Nicolescu, M., Medioni, G.: A voting-based computational framework for visual motion analysis and interpretation. IEEE Trans. Pattern Anal. Mach. Intell. **27**(5), 739–752 (2005)
29. Park, M.K., Lee, S.J., Lee, K.H.: Multi-scale tensor voting for feature extraction from unstructured point clouds. Graph. Model. **74**(4), 197–208 (2012)
30. Risser, L., Plouraboué, F., Descombes, X.: Gap filling of 3-D microvascular networks by tensor voting. IEEE Trans. Med. Imaging **27**(5), 674–687 (2008)
31. Schultz, T.: Towards resolving fiber crossings with higher order tensor inpainting. In: New Developments in the Visualization and Processing of Tensor Fields, pp. 253–265. Springer, Berlin (2012)
32. Strokina, N., Kurakina, T., Eerola, T., Lensu, L., Kälviäinen, H.: Detection of curvilinear structures by tensor voting applied to fiber characterization. In: Scandinavian Conference on Image Analysis, pp. 22–33. Springer, Berlin (2013)
33. Tao, L., Murino, V., Medioni, G.: A tensor voting approach for the hierarchical segmentation of 3-D acoustic images. In: First International Symposium on 3D Data Processing Visualization and Transmission, pp. 126–135. IEEE, Piscataway (2002)
34. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: 6th International Conference on Computer Vision, pp. 839–846. IEEE, Piscataway (1998)

35. Tong, W.S., Tang, C.K.: Robust estimation of adaptive tensors of curvature by tensor voting. IEEE Trans. Pattern Anal. Mach. Intell. **27**(3), 434–449 (2005)
36. Wang, J., Lu, H., Liu, Q.: Feature space analysis using low-order tensor voting. In: 2004 International Conference on Image Processing, vol. 4, pp. 2681–2684 (2004)
37. Yigitsoy, M., Navab, N.: Structure propagation for image registration. IEEE Trans. Med. Imaging **32**(9), 1657–1670 (2013)

Local Geometric Descriptors for Multi-Scale Probabilistic Point Classification of Airborne LiDAR Point Clouds

Jaya Sreevalsan-Nair and Beena Kumari

Abstract Point classification is necessary for detection and extraction of geometric feature (folds, creases, junctions, surfaces), and subsequent 3D reconstruction of point-sampled geometry of topographic data captured using airborne LiDAR technology. Geometry-based point classification (line-, surface-, point-type features) is determined using shape of the local neighborhood, given by the local geometric descriptor (LGD) at every point in the point cloud. Covariance matrix of local neighborhoods is the conventionally used LGD in the LiDAR community. However, it is known that covariance analysis has drawbacks in detection of sharp features, which are a subset of the line-type features. Here, we compare the performance of new variants of existing LGDs, such as weighted covariance matrix, and that based on tensor voting concept, in geometric classification with that of covariance matrix. We propose a multi-scale probabilistic saliency map based on eigenvalues of the LGDs for computing the classification. Usually the state-of-the-art performance analyses of LGDs in the classification outcomes are done downstream after feature extraction. We propose that the comparisons may be done upstream at the classification stage itself, which can be achieved by expressing these LGDs as positive semidefinite second-order tensors. We perform qualitative comparisons of the tensor fields based on shape and orientation of the tensors, and the classification outcomes using visualizations. We visualize LGDs using superquadric tensor glyphs and point rendering, using our proposed saliency map as colormap. Our detailed comparative analysis shows that the new variant of LGDs based on tensor voting classify line-type features, especially sharp features, better than covariance-based LGDs. Our proposed LGD based on tensor voting performs better than the covariance matrix, for our goal of detecting sharp features, e.g. gabled roofs in buildings.

J. Sreevalsan-Nair (✉) • B. Kumari
Graphics-Visualization-Computing Lab, International Institute of Information Technology Bangalore, Bangalore, Karnataka, India
e-mail: jnair@iiitb.ac.in; beena.kumari@iiitb.org

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_8

1 Introduction

Geometric feature detection is a key operation in the processing of three-dimensional (3D) point clouds, which includes surface reconstruction, surface matching, shape detection, registration of point clouds, and finding deformations in time-varying point sampled geometry. The definition of feature is however application-specific, which makes the process of feature extraction subjective. Features are generally defined as entities which help the user to gain meaningful insight from the data. In some cases, features may be a subset of either raw or derived data, which persist through multiple scales, time-steps, and/or other attributes, which give multiple series of the same dataset.

We focus on *geometry-based point* or *geometric classification*, which is the first step in *detection and classification of features*, such as folds, creases, junctions, and planar surfaces. In point clouds, the features are determined using local geometric descriptors (LGDs). Such a descriptor at a point is a variable which *describes* the shape of local neighborhood of any point [4]. The shape of the local neighborhood as a key criterion for point classification, which gives the *geometric classes*,[1] namely line-, surface-, and (critical/degenerate) point-type features. The outcomes of *point classification* imply that the point will belong to a feature, namely, the line-, surface-, or point-type feature, respectively. The choice of LGD plays a crucial role in influencing the point classification outcomes. We refer to geometry-based point classification as "classification," hereafter.

The 3D point-sampled geometric datasets generally encountered in graphics and modeling communities and LiDAR (Light Detection and Ranging) datasets are essentially 2.5-dimensional data. Specifically, in the case of airborne LiDAR data, the point cloud is a height map of two-dimensional planar geometry (latitude-longitude coordinate system). Airborne LiDAR point clouds include geometry of arbitrary topology from multiple objects in environmental scans, owing to which several methods in point-sampled geometry cannot be directly extended to LiDAR point clouds. The resolution of the scans are different. In airborne LiDAR points, the resolution is lower, compared to other point-sampled geometry obtained from indoor or other controlled settings.

Covariance analysis of local neighborhood of LiDAR point clouds is used conventionally for geometry-based point classification, which gives satisfactory results [2, 10]. Classification using covariance matrix is followed by a processing step, such as minimum spanning tree [6, 10, 23] to find sparse set of points which reveals the features. Multi-scale extension of the covariance-based methods has improved the identification of various visually significant features [10, 23]. However, pruning of an already sparse sets of points leaves these methods ineffective in detecting sharp features. At the same time, the computer vision community

[1] Here, we use *geometric classes* and *feature classes* interchangeably.

have proposed the use of weighted covariance matrix and tensor voting[2] for unique signatures for local surface description [28] and perceptual organization for feature extraction [22, 30].

To bridge the gap between these communities, we propose improving classification outcomes in LiDAR point clouds, specifically of line-type features which include sharp features, by using variants of weighted covariance matrix and voting tensor as LGDs. Our goal is to compare the performance of the aforementioned LGDs in extracting sharp features, such as gabled roofs in buildings.

Motivating Problem In the LiDAR community, 3D point clouds are beneficial as they do not have as many occlusion problems or shadow casts found in aerial imagery. In practice, fusion of the two types of datasets gives good results for object-based classification (buildings, road, natural ground, and vegetation). Building detection is a key outcome, which is followed by outline delineation and 3D reconstruction of the detected buildings. The outlines of building, which are derived using aerial imagery, are known to be more accurate and of better quality in comparison to those from point clouds [25]. In practice, building outlines are derived using a fusion of imagery and 3D data. Improving 3D building reconstruction implies improving either the fusion of datasets from hybrid sources, or the geometry extraction of building outlines exclusively from LiDAR point clouds; both of which have been identified as open research problems [25]. For the latter, we propose extraction of line-type features and assembling them to extract outlines. In this paper, we address the improvement of line-type feature extraction by proposing a probabilistic method for identifying points which belong to the line-type features. In the scope of this paper, we perform only the geometry-based point classification, along with preliminary results of line-type feature extraction.

Here, we compare the performance of different variants of LGDs based on the weighted covariance matrix and tensor voting, for geometry-based classification. The latter takes into account continuity and proximity principles. Tensor voting methods are known to behave consistently across a variety of datasets, including de-noised LiDAR point clouds. Additionally, we propose application of anisotropic diffusion after applying tensor voting, for enhancing line-type features, as well as giving classification outcomes similar to that of covariance matrix. Owing to the inherent uncertainty in classification, we propose a probabilistic approach for computing saliency map that gives the classification. We additionally use a multi-scale method to compute our novel probabilistic saliency map, where the radius of the local neighborhood is the scale [10, 22, 23]. Since we did not find any existing literature on methods of comparing classification outcomes of two different LGDs, for performance analysis, we perform comparisons by aligning the reference frames of the LGDs. We perform qualitative comparative analysis using visualization. For visualizing the classification outcomes, we take the following approach of: (a)

[2]Here, we disambiguate *tensor voting* as the algorithm, and *voting tensor* as the second-order tensor, which is the outcome of the algorithm.

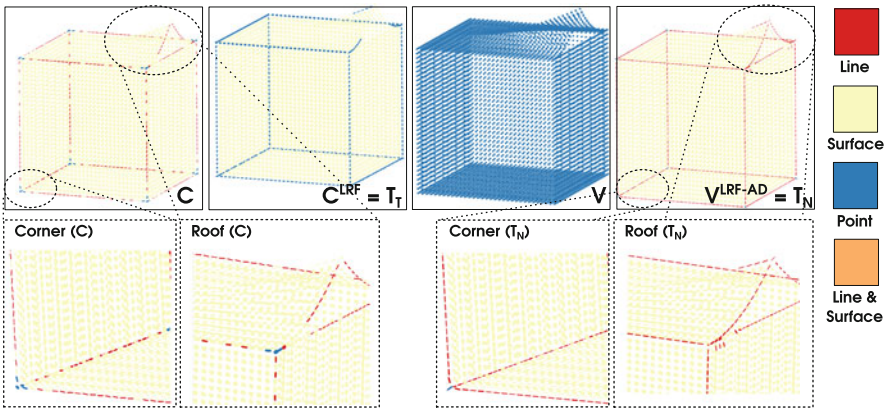


Fig. 1 Superquadric tensor glyphs for visually comparing LGDs: (*Top*) C, C^{LRF}, V, and V^{LRF-AD}, using a geometric class-based colormap. The LGDs computed at scale, $r = r_{min}$ for smooth-feature (6,177 points). The *bottom row* shows zoomed out portions of the corner and roof, where glyphs represent (*left*) C and (*right*) V^{LRF-AD}. Note that, in comparison to C, V^{LRF-AD} shows the line-type features using more cylindrical glyphs with *red color*, indicating higher saliency in line-type features. Thus, our novel saliency map determines the geometric classification

representing LGDs as second-order tensors and using state-of-the-art superquadric tensor glyph visualization [12, 26], and (b) using our novel multi-scale saliency map to guide the colormap for visualization. We propose the use of juxtaposed views and classification matrix visualization, for composite visualizations for comparative analysis of classification outcomes of different LGDs. Our work is unique in comparing point classification outcomes. Existing methods for surface reconstruction and surface fitting use classification as one of the preliminary steps, without any in-depth analysis, which limits the scope of usage of geometry for classification.

Using our proposed methods and additional quantitative analysis, we have shown how different LGDs perform in line-type feature detection (Fig. 1[3]). We observe that more points are detected as line-type features, when using tensor voting in comparison to conventional covariance analysis. The larger number of points allows better extraction of sharp (line-type) features.[4] Our contributions are:

1. multi-scale probabilistic geometric classification of airborne LiDAR point clouds using a novel saliency map;

[3]In Figs. 1, 4, and 7–10, color coding correspond to a geometric class or the combination of classes a point belongs to, which is determined by using the saliency maps. We use the colorblind safe color palette options from ColorBrewer2.0 http://colorbrewer2.org/.

[4]We have demonstrated results on fan-disk and smooth-feature datasets, apart from airborne LiDAR datasets, purely for more comprehensible comparative analysis of LGDs for classification.

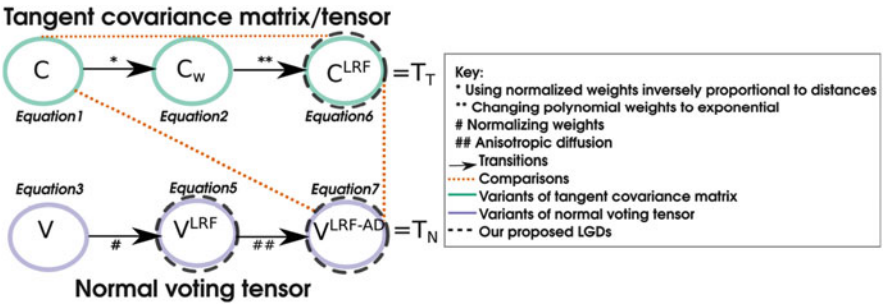


Fig. 2 Summary of the different LGDs we compute and how we compute them. The *dotted lines* indicate the pairwise comparisons of LGDs done in this paper

2. novel approaches for visual comparison of performance of LGDs (Figs. 1 and 2) in classification, such as: (a) expressing LGDs as positive semidefinite second-order tensors, and consequently, using superquadric tensor glyphs for visualization; and (b) visualizing the classification matrix;
3. comparative analysis of variants of two existing LGDs for point classification, computed using weighted covariance matrix and tensor voting, respectively.

Notations We refer to a point in a point cloud, $\mathscr{P}$, as $x \in \mathscr{P}$, and the local neighborhood of x, for a given radius r, is $N(x) = \{x, y \in \mathscr{P} : \|y - x\| < r\}$. Covariance matrix, voting tensor and weighted covariance matrix are annotated as C, V, and C_w. Our proposed modifications to the LGDs are indicated as superscripts, e.g. *LRF* and *AD* refer to the alignment of the local reference frame (LRF) and anisotropic diffusion, respectively. Thus, C^{LRF} and V^{LRF} are the LRF-aligned weighted covariance matrix and voting tensor, respectively. V^{LRF-AD} is the tensorial outcome of applying anisotropic diffusion on V^{LRF}. We use shorthand notations for our proposed LGDs, C^{LRF} and V^{LRF-AD}, as T_T and T_N, to highlight the tensor representation of the LGDs and their relationship to the tangent and normal spaces of the point, respectively.

2 Related Work

We describe relevant work on LGDs, its tensor representations, and the tensor field visualization techniques. We have briefly looked at relevant work on the use of LGDs for the classification in computational geometry and LiDAR communities.

LGDs for Point Clouds Gumhold et al. [6] have proposed feature classification and extraction using correlation matrix and neighborhood graph. The correlation matrix is the total least squares problem for surface fitting, which has been proposed by Hoppe et al. [8]. In [6], tensor voting scheme was explicitly avoided for

feature detection and extraction, in order to avoid volume discretization. However, more recent research [22] has shown that tensor voting can be applied directly on point clouds without volume discretization. Gelfand et al. [4] have made a distinction between high- and low-dimensional LGDs. High-dimensional or richer LGDs (spin images, shape contexts, their proposed integral volume descriptors) are generally used for object recognition and shape retrieval. The low-dimensional LGDs (curvature-related descriptors) are easier to compute, store and use – hence is used with voting schemes or iterative alignment methods for shape matching and point cloud registration. LGDs considered for the point classification are generally low-dimensional ones. Integration based LGDs are recommended for feature classification in [4], however point classification in LiDAR point clouds is conventionally done using differentiation-based low-dimensional descriptors.

Tensor Representation of LGD Knutsson [13] has used tensor definition for structure tensor, which is based on differentiation of functions and is used as LGD for 3D space. Knutsson et al. [14] have enumerated LGDs for images, some of which can be extended to 3D point clouds.

Point/Feature Classification In the LiDAR community, geometric classification is often called "structural classification." Point classification and feature classification are often interchangeably used in literature. Several methods for feature classification of point clouds exist in literature [22, 31]. Attributes from feature/structural classification (e.g. linearity, anisotropy, etc.) are extensively used for semantic or object-based classification in LiDAR point clouds [2, 16, 17], which use covariance analysis. Kim et al. [11] have used 3D tensor voting for wall and roof segmentation and classification in LiDAR point clouds. They have generated surface meshes from the point clouds, where tensor voting is used for fitting a planar surface in local neighborhoods and segmenting the surface mesh [19]. The mesh segmentation is similar to [19, 30]. While our proposed LGD using tensor voting can be used for orientation just as in [11], our goal is to extract line-type features, which can be further used for segmentation.

Covariance analysis of local neighborhoods based on centroid is a robust method for normal estimation [15], but not necessarily for finding the shape of the neighborhoods. Two neighboring points with approximately similar local neighborhoods with the same centroid will have the same LGD, which makes the LGD fail specifically for the classification of sharp features. Moreover, all neighbors in the local neighborhood are considered equal in the covariance matrix, which is not the case. The rationale is that its nearer neighbors have more "influence" in reconstructing the surface at the point than its distant neighbors. Thus, a weighted covariance matrix computed with respect to a point is a more accurate "signature" of a point compared to the conventionally used covariance matrix. Kobbelt and Botsch [15] have discussed that for consistent orientation of normals estimated using covariance analysis, propagation along a minimum spanning tree will be needed. Similarly, Tombari et al. [28] have made the argument of lack of repeatability of sign of LRF when using the covariance matrix, and have proposed a weighted covariance matrix based on the point itself instead of the centroid, for surface matching. They have proposed an

LGD for surface matching, based on signatures of histograms, where local reference frame can be made unique, repeatable, and an invariant for a point. For classification of sharp features, moving least squares method [1, 3] or Gauss map clustering [31] have been used effectively, which are not centroid-based methods. Local tensor-based techniques are a tradeoff between computational complexity and accuracy in feature detection; e.g. use of tensor voting [22, 30] for feature classification.

Multi-Scale Classification Pauly et al. [23] have proposed the use of multi-scale surface variation, estimated using covariance matrix of local neighborhood. There, surface variation at a user-defined scale gives feature weights, which on appropriate thresholding gives features. Keller et al. [10] have used a similar multi-scale approach, for LiDAR point clouds, in determining feature weights from covariance matrix of local neighborhoods. However, the difference between the methods in [23] and [10] is that a single adaptive scale and averages across multiple scales have been used, respectively. Algorithms for finding optimal neighborhood size or scale has been of interest to the LiDAR community [2, 5, 17, 33].

Tensor Voting Guy and Medioni [7] have proposed tensor voting scheme to detect and classify the feature points in structured point cloud. The voting scheme uses the proximity and continuity principles of Gestalt psychology to propagate the votes. Each point is encoded as tensor based on given input data information, and nonlinear tensor voting has been performed using local neighbors to update the tensor information at each point in 3D data. Saliency maps, computed from eigen analysis of the second-order tensor obtained after vote aggregation, give the likelihood of each point belonging to different feature classes, i.e., ball-, stick-, or plate-tensor.

Park et al. [22] have used tensor voting and surface variation to classify and detect line features in point clouds, where the surface variation function is computed using a multi-scale method. Our proposed tensor, obtained after anisotropic diffusion upon tensor voting, V^{LRF-AD}, is different from that from tensor voting, V, in [22] as the classification in the latter does not confirm to the conventional method [10], as point-type features in the former are equivalent to line-type features in the latter, as per the classification given in the latter; while ours matches with the conventional method. Park et al. compute an optimal scale from multiple scales for further analysis, whereas we propose an *aggregated* multi-scale saliency map. In our case, the anisotropic diffusion after tensor voting enhances line-type feature detection as well as modifies the classification outcomes of the new tensor to match with that of the covariance matrix. Our approach based on anisotropic diffusion is inspired from [30], where anisotropic diffusion is performed after tensor voting for feature classification and extraction in polygonal mesh data, and subsequent mesh segmentation.

Tensor Field Visualization We use superquadric tensor glyphs [12, 26] for comparing different (positive-semidefinite) second-order tensor fields of the LGDs. These glyphs are designed using mirror symmetry, symmetry, continuity, and disambiguity as guiding principles. Thus, by design, these glyphs show the shape

and orientation of a positive-semidefinite second-order tensor more effectively than other conventionally used glyphs. Zhang et al. [34] have recently proposed the use of *tender glyphs* for simultaneously comparing two different tensor fields.

3 Local Geometric Descriptors

Conventionally used covariance-based methods have limitations in capturing line-type features [31]. They often detect points on the sharp features contrarily as weak line-type features (Fig. 1). However, the use of an LGD which enhances the sharp features can improve the classification. Hence, we propose two different LGDs, which are variants of the weighted covariance matrix, C_w, and tensor voting, V. We further exploit the tensor representation of the LGDs for its analysis.

In our work, C_w is an improvement over C for two reasons. First, C_w at the point gives a unique LGD, thus giving a "signature" of the point itself. Additionally, in the case of the classification using C_w. the normal estimation is not done using the centroid. Second, the contribution of all neighbors in the computation of $C(x)$ are equal, i.e., a distant neighbor will "influence" the shape of the local neighborhood as much as a neighbor in closer proximity. Contrarily, these contributions are inversely proportional to the distance between the neighbor and the point, x. Hence, a weighted covariance matrix, such as C_w, is ideal as an LGD. Here, we study how the changes proposed by Tombari et al. [28] influence the geometric classification. Similarly, tensor voting can also be considered as an improvement over C. Tensor voting approach follows the Gestalt principles of proximity and continuity, which makes the resultant tensor viable as an LGD.

Comparing the performance of LGDs can be done in two ways, namely, comparing the relative performance of each LGD against the ground truth and comparing the outcomes of the LGDs, taking two at a time. For the latter, the classification outcomes for two LGDs, say C_w and V, are comparable if the comparisons can be done using a common invariant, e.g. local reference frame (Fig. 3). Additionally, one must be aware that there are cases where the outcomes of the classification of the two LGDs are different and hence, not comparable, e.g. C (or C_w) and V (Fig. 1). However, certain modifications to the LGDs can resolve these differences and facilitate the comparisons subsequent to modifications, e.g., anisotropic diffusion after tensor voting makes its classification outcomes comparable to those of C (Fig. 1).

Local Geometric Descriptors LGDs at a point, in $\mathscr{P}$, encodes the geometry of the local neighborhood of the point. LGDs are important, as the local geometric analysis leads to the global description for the entire point cloud. Here, we focus on two LGDs, computed using the conventionally used covariance matrix and tensor voting, respectively. Here, we discuss the construction of these variants of the LGDs.

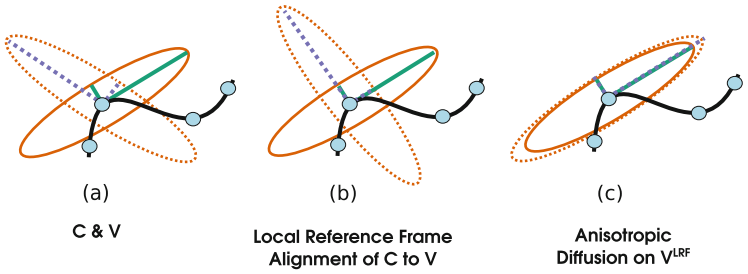


Fig. 3 A schematic showing modifications to LGDs, C and V, to compare their point classification outcomes effectively. For points/samples (*circle*) on a curve (in $\mathbb{R}^2$), the LRF and shape of local neighborhood obtained from the spectral decomposition of the LGD are shown (*solid lines* and *ellipse* for those of C; and *dashed lines* and *ellipse* for those of V). (**a**) C and V at a point; (**b**) The LRF of C has been aligned to that of V; (**c**) Anisotropic diffusion has been applied to V, which gives V^{LRF-AD}. Similar to C, this is applicable to C_w as well

Weighted Covariance Matrix The covariance matrix [8], also called correlation matrix [6], is computed as $C(x) = \sum_{y \in N(x)} (y - \bar{x})(y - \bar{x})^T$, where $\bar{x}$ is the centroid of the local neighborhood $N(x)$. We refer to it as the "tangent covariance matrix," as it is constructed using tangent vectors. C is conventionally used as an LGD in the LiDAR community [2, 5, 17, 33]. However, in computer vision applications, weighted covariance matrix is used for surface description and local sign disambiguation [28]. Since we are performing surface description as opposed to surface fitting in LiDAR point clouds, we propose the use of weighted covariance matrix, C_w (Eq. 1) for the same. The weights have the property of inverse proportionality to distance between the concerned point and its neighbor. *Local reference frames* (LRF) are invariants that define the local neighborhood of a point uniquely. The eigenvectors of the LGD are conventionally used as basis vectors of the LRF. C_w gives repeatable LRFs, as opposed to C. Repeatable LRFs give non-repeatability of sign of principal directions across multiple objects in the scene [28], which are applicable in the case of LiDAR point clouds.

Voting Tensor In tensor voting [19], there is a notion of a receiver and a voter, where the voter casts a vote to the receiver. A vote, defined in the form of a tensor, contains information of the neighbor in the local neighborhood. A predefined voting field is used to aggregate the votes as a tensor, and to provide a saliency map for each feature type. Saliency values of the resultant tensor, which is the sum total of all information collected at the receiver, are scalar values derived from the spectral decomposition of the tensor. The saliency values, "pointness," "curveness," and "surfaceness", correspond to feature (or structural) classes, namely, ball- (point), stick- (line), and plate- (surface) tensors, respectively. Thus, these classes correspond to the feature classes in point sampled geometry. In tensor voting, we initialize the tensor $V(x)$ as a ball tensor, for unoriented points (i.e. points without predetermined normal information) [22], as is our case (Eq. 1). In Eq. (1), I_d is a d-dimensional identity matrix, and scale parameter σ. We use the radius of the neighborhood, r as the scale

parameter, thus, $\sigma = r$. While σ influences the vote propagation range, the Gaussian function, μ_y, is the attenuation factor for the size of the vote collected from y at x. Here, we can see that $V(x)$ is a weighted covariance matrix just as $C_w(x)$, with the difference that they are constructed from normals and tangents, respectively. For $y \in N(x)$, $t(y) = (y - x)$ and $z(y) = \frac{\|t(y)\|_2}{r}$,

$$C_w(x) = \sum_{y \in N(x)} w_y t(y) t(y)^T; \qquad V(x) = \sum_{y \in N(x)} \mu_y . \left(I_d - \frac{t(y) t(y)^T}{t(y)^T t(y)} \right) \quad (1)$$

$$\text{where } w_y = \frac{1 - z(y)}{\sum_{y \in N(x)} 1 - z(y)}; \qquad \text{and } \mu_y = exp\left(- \frac{\|t(y)\|_2^2}{\sigma^2} \right).$$

Alignment of Local Reference Frame for Local Geometric Descriptors An LRF is an invariant of the LGD of a point, and a common LRF for two LGDs implies that the orientations of the local neighborhoods defined by the LGDs are the same, at the point. Thus, comparison of LRF-aligned LGDs reduces to qualitative comparison of the shapes of these local neighborhoods at each point. Hence, we propose variants of C_w and V, whose LRFs are aligned, to facilitate their comparisons. Our construction of these proposed LGDs are:

1. Modification to V, by normalizing the weights for I_d in Eq. (1). Given $\sigma = r$, we get $\mu_y = exp(-z(y)^2)$. Thus, we replace μ_y (Eq. 1) with μ'_y (Eq. 2) to give an *LRF-aligned voting tensor*, denoted as $V^{LRF}(x)$.

$$V^{LRF}(x) = \sum_{y \in N(x)} \mu'_y . \left(I_d - \frac{t(y) t(y)^T}{t(y)^T t(y)} \right), \text{ where } \mu'_y = \frac{\mu_y}{\sum\limits_{y \in N(x)} \mu_y} \quad (2)$$

2. Modification to C_w, by normalizing the tangent vector $t(y)$ and modifying the weights in the computation of C_w from w_y in Eq. (1) to w'_y. This change in weight (Eq. 3) is permissible as $w'y$ has the same property that the weights w_y and μ_y (Eq. 1) have, i.e., they are monotonous functions of dimensionless distance quantity $z(y)$, and are inversely proportional to the distance of the neighbor to the point, i.e. $\|t(y)\|$. Thus, we get the *LRF-aligned tangent covariance matrix*, $C^{LRF}(x)$, as follows:

$$C^{LRF}(x) = \sum_{y \in N(x)} w'_y t(y) t(y)^T, \text{ where } w'_y = \frac{\mu_y}{t(y)^T t(y) \cdot \sum\limits_{y \in N(x)} \mu_y}. \quad (3)$$

However, by aligning LRFs for the LGDs this way, the weights for weighted covariance matrix, C^{LRF}, are not convex, unlike the weights proposed in [28]. Since convexity is not recommended as a requirement in [28], we disregard it for the sake of alignment of LRFs. Even though Eqs. (2) and (3) lead to different computations of the LGDs, they effectively perform rotations of the eigenvectors, as schematically shown in Fig. 3. Hence, we refer to these modifications as LRF "alignment".

Anisotropic Diffusion on Normal Voting Tensor While tensor voting is good for perceptual grouping using its saliency maps, the point classification through its tensor V or its variant (V^{LRF}) is not similar to that of the covariance matrix (C) or its variants (C_w or C^{LRF}), as shown in Fig. 1. Hence, we perform anisotropic diffusion after aggregating votes in the tensor, so that the resulting tensor V^{LRF-AD} will give similar classification outcomes as C. We explain the rationale behind differences and the resolution using anisotropic diffusion in Sect. 6.

Wang et al. have used V^{LRF-AD} as the diffusion tensor in the heat diffusion equation [30], for polygonal mesh segmentation. The rationale is that anisotropically controlling diffusion velocities in the diffusion tensor, would lead to slower diffusion across sharp edges and faster diffusion along sharp edges. Thus, anisotropic diffusion has been proven to enhance line-type features in a surface mesh. Along these lines, we propose applying anisotropic diffusion after tensor voting, so that the resultant tensor can be used as the LGD of the point cloud to classify line-type features more accurately than the tangent covariance matrix. We perform anisotropic diffusion by modifying the eigenvalues of the diffusion tensor as an exponential term and addition of a diffusion parameter, δ, which controls diffusion velocities.

Anisotropic diffusion is applied to V^{LRF} by modifying its eigenvalues $(\lambda_V^{LRF})_i$ (for $i = 0, 1, 2$) using an exponential function, $f((\lambda_V^{LRF})_i) = exp(-(\lambda_V^{LRF})_i/\delta)$. However, the use of an exponential function reverses the order of eigenvalues:

$$(\lambda_V^{LRF})_0 \geq (\lambda_V^{LRF})_1 \geq (\lambda_V^{LRF})_2, \text{ gives}$$
$$f((\lambda_V^{LRF-AD})_0) \leq f((\lambda_V^{LRF-AD})_1) \leq f((\lambda_V^{LRF-AD})_2.$$

Thus, for anisotropically diffused LRF-aligned voting tensor V^{LRF-AD}, with indices $i = \{0, 1, 2\}$ corresponding to the descending order of eigenvalues, we get:

$$(\lambda_V^{LRF-AD})_i = exp(-(\lambda_V^{LRF})_{2-i}/\delta); \text{ and } (e_V^{LRF-AD})_i = (e_V^{LRF})_{2-i} \quad (4)$$

$$\text{Finally, } V^{LRF-AD}(x) = \sum_{i=0}^{2} (\lambda_V^{LRF-AD})_i \cdot (e_V^{LRF-AD})_i (e_V^{LRF-AD})_i^T \quad (5)$$

Impact of Proposed Modifications Here, we mathematically analyze the modifications caused by the alignment of LRFs of C_w and V and anisotropic diffusion of V, shown schematically in Fig. 3. Let us first compare the eigenvalues and eigenvectors from $C^{LRF}(x)$, and $V^{LRF}(x)$. Suppose the set of eigenvectors for $C^{LRF}(x)$ and $V^{LRF}(x)$ are given by $(e_C^{LRF})_i$ and $(e_V^{LRF})_i$, respectively, for $i = 0, 1, 2$. Similarly, let the set of eigenvalues be $(\lambda_C^{LRF})_i$ and $(\lambda_V^{LRF})_i$, respectively.

$$V^{LRF}(x) = I_d - C^{LRF}(x), \text{ from Eqs. (2) and (3).} \quad (6)$$

$$(\lambda_V^{LRF})_i = 1 - (\lambda_C^{LRF})_{(2-i)}, \text{ and hence, } (e_V^{LRF})_i = (e_C^{LRF})_{(2-i)}, \text{ for } i = 0, 1, 2. \quad (7)$$

This implies that the ordering of the eigenvalues are reversed in $V^{LRF}(x)$ compared to those of $C^{LRF}(x)$, thus reversing the order of their eigenvectors (i.e. major eigenvalue of V^{LRF} corresponds to minor eigenvalue of C^{LRF}, and so on). This

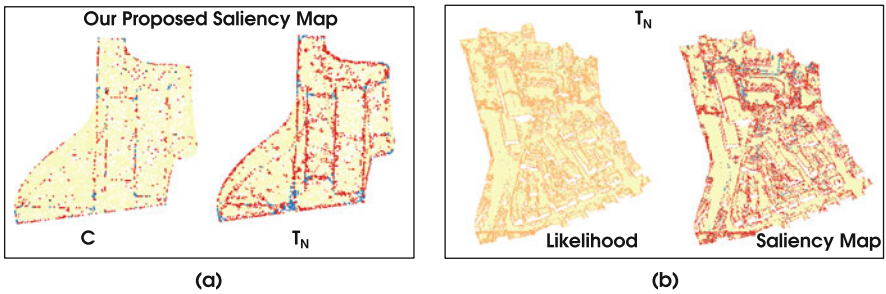


Fig. 4 (**a**) Comparing LGDs for fan-disk (10,921 points), using our proposed saliency map. (*Left*) is the tangent covariance matrix C and (*right*) is our proposed anisotropically diffused voting tensor V^{LRF-AD} (i.e., T_N). (**b**) Comparison of saliency maps computed for our proposed voting tensor V^{LRF-AD}, using likelihood values in shape classes [line,point,surface], given in (*left*) Eq. (8), and (*right*) Eq. (9), for Region-1 in Vaihingen (179,997 points)

explains the difference in the classification outcomes between the descriptors in Fig. 1. However, it is desirable that the ordering of the eigenvalues be the same, so that the eigenvectors at a point can be the same, thus giving the same LRF and similar classification outcomes from both descriptors. A similar transformation is used in [21] to change tangential orientations to normal ones for structure estimation in images.

Thus, from Eqs. (4) and (6), we conclude that $(e_V^{LRF-AD})_i = (e_C^{LRF})_i$. This shows that the LRFs of V^{LRF-AD} and C^{LRF} match (i.e. their major eigenvectors correspond to each other, and similarly, their middle and minor ones). This also implies that the ordering of their eigenvalues also match, thus giving similar classification outcomes for both the LGDs, i.e. variants of the covariance matrix and the anisotropically diffused voting tensor (Figs. 1 and 4a).

Positive Semidefinite Second-Order Tensor Representation We did not find any metrics in current literature for comparing LGDs and their classification outcomes. We propose expressing these LGDs as a specific mathematical quantity, for validating applications of mathematical operations such as, comparisons and replacements. We have found that second-order tensors are the most natural choice of quantity, owing to the inherent construction of the LGDs as tensor products. Local neighborhood can be described using various variables or approaches, such as spin images, covariance analysis, tensor voting, curvature, etc. Here, we revisit the use of tensors in representing LGDs used for structural analysis in image processing and computer vision [14], to apply the same for point-sampled geometry. While some LGDs are explicitly defined as tensors (e.g. in tensor voting [20] and curvature tensor [27, 29]), the others are described using matrix representation (covariance matrix [8, 10]). Nevertheless, in many such cases, the LGD is constructed using an outer product or tensor product of vectors, $T(x) = xx^T$, where $x \in \mathbb{R}^d$ is either the tangent or the normal at the point. From Eqs. (1), (2), (3), and (5), we see that all LGDs discussed in this paper are indeed *positive semidefinite second-order tensors.*

By design, the tensor fields of LGDs correspond to the *tangent* or the *normal spaces to the manifold* defined by the point cloud, respectively, at the concerned point [24]. The tangent and normal spaces form disjoint sets themselves. Here, the covariance matrix and its variants (C, C_w, C^{LRF}) correspond to the tangent space; and the tensor and its variants from tensor voting (V, V^{LRF}, V^{LRF-AD}), to the normal space. Hence, hereafter, we use the shorthand notations for C^{LRF} and V^{LRF-AD} are T_T and T_N, respectively.

4 Multi-Scale Probabilistic Point Classification

For probabilistic point classification, we propose an appropriate saliency map of the feature classes at each point, which is aggregated from LGDs across multiple scales.

Multi-Scale Approach The advantage of using multi-scale approaches in comparison to single scale is that they do not rely on a single scale for procuring the geometric information. The distributed approach improves the richness of the geometric information. Determining the shape of the local neighborhood has high degree of uncertainty, especially in multi-object boundaries in LiDAR/computer vision datasets. When using single scale, obtained optimally or adaptively, the error margin becomes high for such boundary cases. Additionally, multi-scale approach determines persistence from various scales, as opposed to adaptive scale, which depends on a scale where the persistence is high. Thus, multi-scale methods capture persistence of features better. However, the number of scales and bounds of the scales become crucial for the success of the multi-scale method. Currently, these values are user-determined, which is a limitation. Multi-scale approaches also fail for cases where there is a large difference in sampling density in the dataset, as the covariance analysis fails in the case of sparse local neighborhoods. This can be alleviated by using k-nearest neighborhoods instead of spherical neighborhood [15]. One of the the drawbacks of using adaptive scales or optimal scales is that they become computationally intensive for large point clouds. More operations are required, in the former, for computing adaptive scales for each point, and in the latter, for optimizing a single scale for all points.

Despite their drawbacks, multi-scale approaches have worked favorably for point classification in certain applications, as preserving the persistence of features across several scales gives a better measure of the feature strength or saliency map [10, 22, 23]. Keller et al. [10] have proposed probabilistic point classification for LiDAR datasets, where eigenvalues of the covariance matrix is used to determine neighborhood shape and feature strengths at each scale, and finally, the likelihood of the point belonging to the three feature classes are averaged across scales. For each scale, r (i.e., radius of local neighborhood), given that $\lambda_0(x,r) \geq \lambda_1(x,r) \geq \lambda_2(x,r)$, which are eigenvalues of $C(x)$, three subsets corresponding to the classes are defined to bin all the points in the point cloud, using a user-defined threshold, ϵ: $P_l(r) = \{x \in \mathscr{P} | \lambda_1(x,r)/\lambda_0(x,r) < \epsilon\}$; $P_s(r) = \{x \in \mathscr{P} | \lambda_2(x,r)/\lambda_0(x,r) < \epsilon\}$;

and $P_p(r) = \{x \in \mathscr{P} | \lambda_2(x,r)/\lambda_0(x,r) \geq \epsilon\}$. Here, by design, $P_l(r) \subset P_s(r)$. The likelihood of a point belonging to (line, surface, point) feature classes, $\{L_l, L_s, L_p\}$, are computed using a multi-scale approach. While Keller et al. have not used the likelihoods explicitly for classifying the points, they have used these values to compute feature strengths, which is further used for generating feature graphs, and refining the classification. Given N_r scales uniformly sampled in $[r_{min}, r_{max}]$,

$$\mathbf{L_{l,s,p}(x)} = \frac{1}{N_r} \cdot \sum_{\mathbf{i=1}}^{\mathbf{N_r}} \boldsymbol{\phi}_{\mathbf{l,s,p}}(\mathbf{x}, \mathbf{r_i}), \qquad \text{where } \phi_s(x,r) = \begin{cases} 1, \text{ if } \lambda_2(x,r) \geq \epsilon\lambda_0(x,r) \\ 0, \text{ otherwise} \end{cases};$$

$$\phi_l(x,r) = \begin{cases} 1, \text{ if } \lambda_1(x,r) < \epsilon\lambda_0(x,r) \\ 0, \text{ otherwise} \end{cases}; \text{ and } \phi_p(x,r) = \begin{cases} 1, \text{ if } \lambda_2(x,r) < \epsilon\lambda_0(x,r) \\ 0, \text{ otherwise} \end{cases} \tag{8}$$

The averaging operation of the likelihoods across scales, can be seen as a union of N_r mutually exclusive events of the point belonging to the given class for N_r scales, where the events have equal probability, i.e. $1/N_r$. However, $L_{l,s,p}(x)$, as computed in Eq. (8), causes higher incidence of points having "mixed" likelihoods, specifically across line- and surface-type features (visualized as points belonging to "line & surface" class in Fig. 4b). This is due to the subset relationship, $P_l(r) \subset P_s(r)$ and binarization of the scale-wise contribution of each class towards multi-scale aggregation (Eq. 8). The former leads to the probabilities not adding up to one. The latter causes ambiguities in classification of points that lie in boundaries of two different objects or point clusters e.g. a point lying on a line but in the close proximity of a junction. Additionally, the high dependency of the algorithm on a user-defined parameter ϵ leads to erroneous classification. To alleviate the ambiguity, we propose a probabilistic multi-scale saliency map.

Multi-Scale Saliency Map for Probabilistic Point Classification Saliency map of an attribute set is a set of values which maps a value for specific attributes, based on the relative "saliency" or "differentiating quality" of the attribute with respect to the others in the set. For instance, in tensor voting, eigenvalues of the resultant tensor have been used conventionally as saliency map of its eigenvectors [7]. In LiDAR point clouds, saliency maps are derived from confidence index of 1-, 2- and 3-dimensional features (i.e. line, surface, and point, respectively) [2]. We propose to use likelihoods of a point belonging to the feature classes as the saliency map of classification of the point. These likelihoods, which explicitly imply the classification outcomes, are derived from the spectral decomposition of LGD, thus, preserving the properties of the LGD. Thus, our classification is a probabilistic or fuzzy one.

We use the likelihood with which a tensor can belong to each of the three shape classes [12, 32] found in 3D point clouds. Given that $\lambda_0(x) \geq \lambda_1(x) \geq \lambda_2(x)$, $c_{l,s,p}(x,r)$, we compute the likelihood of the point x belonging to line, surface, or point class, at a given scale r, $\{c_l, c_s, c_p\}(x,r)$. At each scale, we ensure the sum

of likelihoods for a point, $(c_l + c_s + c_p)$, is unity. For multi-scale computation, we average the likelihoods of a point belonging to a given class across N_r scales, i.e.

$$\mathbf{L_{l,s,p}(x)} = \frac{1}{N_r} \cdot \sum_{\mathbf{i=1}}^{\mathbf{N_r}} \mathbf{c_{l,s,p}(x, r_i)}; \quad \text{where} \quad c_l(x,r) = \frac{\lambda_0(x,r) - \lambda_1(x,r)}{\lambda_0(x,r) + \lambda_1(x,r) + \lambda_2(x,r)};$$
$$c_s(x,r) = \frac{2 * (\lambda_1(x,r) - \lambda_2(x,r))}{\lambda_0(x,r) + \lambda_1(x,r) + \lambda_2(x,r)}; \text{ and } \quad c_p(x,r) = \frac{3 * \lambda_2(x,r)}{\lambda_0(x,r) + \lambda_1(x,r) + \lambda_2(x,r)}. \tag{9}$$

Since multiple scales can be viewed as a union of mutually exclusive events with equal probability, we can justify averaging of saliency maps across scales for computing the final saliency map. Our proposed saliency map disambiguates classification of points to one of the three classes (Fig. 4b).

A Note on Scale Parameters Scale parameters are used for computation of both the tensor V (Eq. 1), using tensor voting, as well as saliency maps (Eq. 9). However, the parameters are used differently. In the former, during the voting process, as the scale increases, more tokens (or votes) can influence the given point due to increase in size of kernels, and smooth out the noisy data and weak features. Similarly, when the scale is reduced, more details are preserved, which undesirably makes tensor voting more sensitive to noise. In multi-scale classification, scale is used to measure the persistence of the features, thus ensuring a robust feature extraction method. The scale can also be viewed as the regulator for smoothness [23].

5 Comparison of Local Geometry Descriptors

We use tensor field visualization for qualitative comparison of LGDs and their classification outcomes. For comparative visualizations, we use both juxtaposed views and classification matrix visualization. In juxtaposed views, we use point rendering as well as tensor glyph visualization, using a geometric class-based colormap. In both renderings, our proposed saliency map is used to demonstrate the shape and orientation of the LGD and its classification outcomes. Our work is unique in the application of superquadric tensor glyphs for visualizing LGDs in 3D LiDAR point clouds.

5.1 Juxtaposed Views

The visualization tasks we propose for qualitatively comparing the two second-order tensor fields T_T and T_N are to compare the tensor properties (i.e. shape and orientation) at each point in $\mathscr{P}$, and to compare the classification outcomes at each point. Juxtaposed views [9] of corresponding visualization of the fields for each of

the tasks is an effective design pattern as long as we implicitly establish the same orientation and affine transformations for visualizations corresponding to both the fields in a single view. Juxtaposed view is easy to implement and effectively conveys the information on comparing the fields. We use juxtaposed views of LiDAR point clouds using both tensor glyph visualization and point rendering (Figs. 1 and 4, respectively).

Tensor Glyph Visualization We use superquadric tensor glyphs to compare the tensor properties of the LGDs in a single scale. The glyphs are colored using our proposed saliency map. Both saliency map and shape of glyphs encode the information on the likelihood of a point falling into the three feature classes, for instance, red color and cylindrical shape of glyphs indicate line-type features (Fig. 1). Thus, the saliency value and the shape of the local neighborhood are both encoded in the glyph visualization of the tensor field.

In addition to single scale analysis, we visualize glyphs of *a multi-scale aggregated tensor representation of the LGD*, or simply multi-scale aggregated LGD. However, we have used this tensor purely for representation, as we have not explored the multi-scale aggregation of LGDs in detail. The aggregated LGD is generated by averaging the unit vectors of eigenvectors and normalized eigenvalues (as the saliency values, c_l, c_s, c_p) values. By construction, the saliency map of the multi-scale aggregated LGDs is the multi-scale saliency map (Eq. 9).

Choice of Colormaps Visualization of saliency maps is done by mapping the data to the RGB color model and using the colormap to render the points. A trivial mapping of the parameterized values $\{c_l(x), c_s(x), c_p(x)\}$ to (red, blue, green) channels, respectively, enables us to visualize the differences in the classification outcomes, computed from different LGDs. This is applicable for both point and glyph rendering. However, this channel-based colormap results in a color palette with red and green hues, which is not favorable for protanopic vision [18]. To resolve this, we use the likelihoods (Eq. 8) or saliency values (Eq. 9) of a point to determine the class(es) (i.e., line, surface, and point-type features) the point is highly likely to belong to. Both Eqs. (8) and (9) constrain the classes to be either line-, surface-, or point-type features. In the likelihood based model, the points could belong to two classes simultaneously, which specifically occurs in the case of line- and surface-type feature classes. Hence, overall, we use a geometric class-based color palette, for "line", "surface", "point", and "line & surface" classes (Fig. 1).

Geometry-Based Point Classification Algorithm

1. octree construction and outlier removal in $\mathscr{P}$ (as given in [10]);
2. computation of LGD for each point $x \in \mathscr{P}$ ($C(x)$, $T_T(x)$, or $T_N(x)$) (Eqs. 1, 3 or 5) or its variants;
3. probabilistic point classification by spectral decomposition of LGD for each point, and computation of saliency map for each scale, r (Eq. 9);
4. multi-scale point classification by averaging saliency maps across scales (Eqs. 8 or 9)

5. visualization of feature classification using (a) juxtaposed views of point or glyph rendering using saliency map, or (b) classification matrix of a specific feature class.

5.2 *Classification Matrix Visualization*

We use a superimposed visualization of a *classification matrix* of a feature class of two different LGDs. The classification matrix stores four sets of points which have been labeled as the concerned feature class, by both the LGDs, or exclusively by each of the LGDs, or by neither. The matrix rows pertain to the sets of points classified by the first descriptor, and the columns are those by the second one. Numerically, the classification matrix gives the size of these sets in its cells, and here, we extend its use to visualizing the points belonging to each cell. We color each of the four sets using different colors. We show the classification matrix visualizations for line class in Figs. 5 and 6, to compare between C and T_N, as well as, between T_T and T_N, respectively.

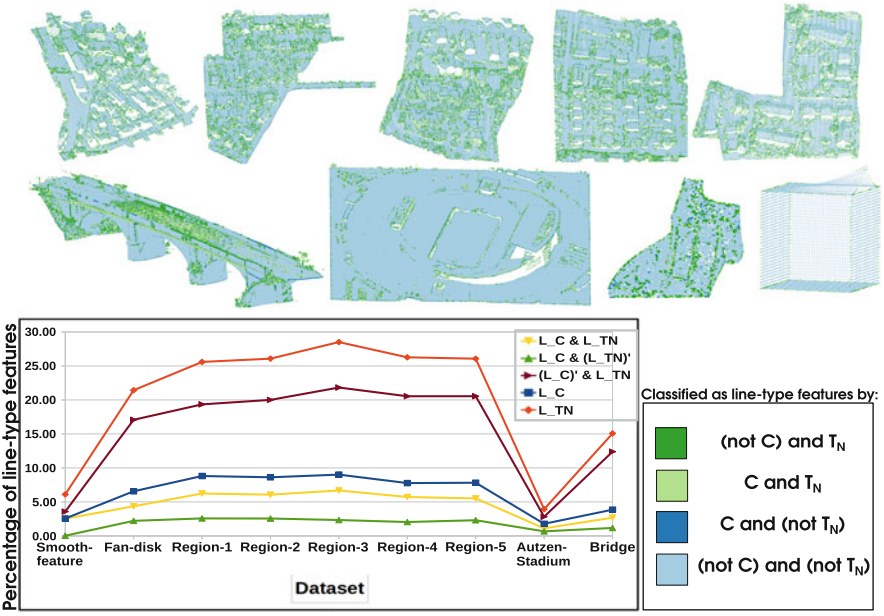


Fig. 5 Classification matrix visualization of line-type features between two LGDs, C and T_N. The *color mapping* shows which LGD caused a point to be labelled as a line-type feature. L_{TN} is the set of points identified as line-type by T_N and L_C by C. The y-axis in the plot shows the percentage of points identified as line-type, given by the sizes of the sets themselves or set intersections

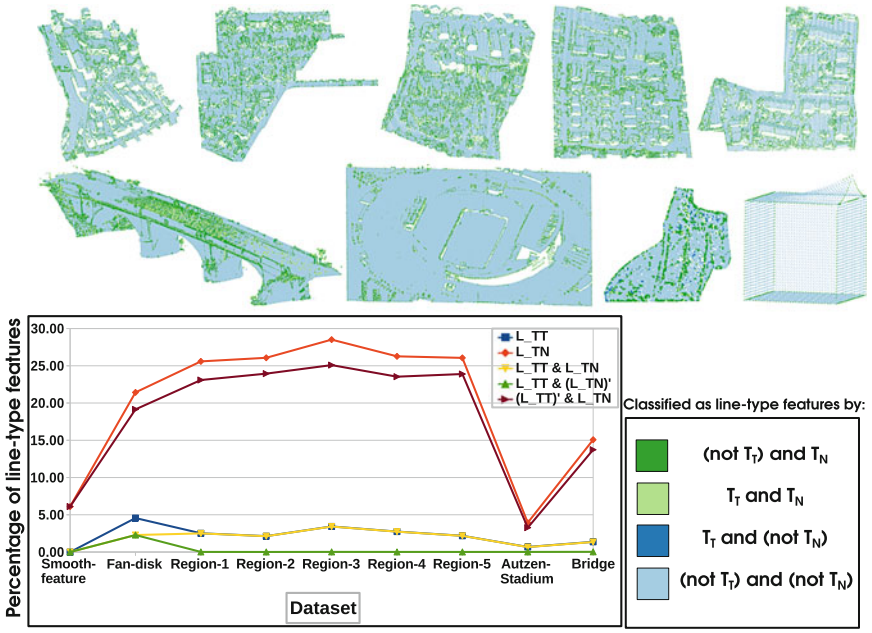


Fig. 6 Classification matrix visualization of line-type features between T_T and T_N. *Color mapping* is similar to that in Fig. 5. L_{TN} is the set of points identified as line-type by T_N and L_{TT} by T_T

6 Experiments and Results

We compare LGDs (C, T_T, and T_N) and their classification outcomes to demonstrate that choice of an appropriate LGD can improve line-type point classification (Fig. 2). Comparing T_T and T_N shows the performance of two different LGDs in classification, with a fixed LRF. Finally, comparing C and T_N displays the improvement of the outcomes in the classification when using the latter, over the former. C is the state-of-the-art LGD used for airborne LiDAR point clouds.

We have used datasets (Table 1) of airborne LiDAR point clouds[a], and point set geometry (for bridge)[b] to showcase the results of our chosen LGDs and their classification outcomes. There are no datasets existent with ground truth for geometry-based point classification. Hence, we have used simple manifold datasets[c] to showcase for preliminary results, only because the geometry in these datasets is more predictable (i.e. visually verifiable) than the LiDAR datasets. Visual comparative study of behavior and classification outcomes of various LGDs can be better done on these datasets, even in the absence of ground truth for classification.

Experiments Apart from qualitative comparisons in juxtaposed views of point rendering (Fig. 7) and glyph visualizations (Fig. 8), we have quantified points classified as line-type features using classification matrices (Figs. 5 and 6). We have also studied the influence of scale in LGD computation using tensor glyph visualization

Table 1 Datasets and the parameters used for computing the local geometric descriptors in our experiments

Dataset	Region-5[a]	Region-1[a]	Region-2[a]	Region-3[a]	Region-4[a]	Autzen Stadium[b]	Bridge[c]	Smooth-feature[c]	Fan-disk[c]
Size (#points)	163,305	179,997	231,725	266,675	323,896	693,895	5,284,131	6177	10,921
$r_{min}(T_T)$	0.009	0.009	0.009	0.009	0.009	0.007	0.004	0.100	0.015
$r_{max}(T_T)$	0.011	0.011	0.011	0.011	0.011	0.009	0.006	0.200	0.020
$r_{min}(T_N)$	0.009	0.009	0.009	0.009	0.009	0.008	0.006	0.100	0.040
$r_{max}(T_T)$	0.011	0.011	0.011	0.011	0.011	0.010	0.008	0.200	0.050

$r_{min}(T_T)$ and $r_{max}(T_T)$ are user-defined minimum and maximum values for scales, i.e. radius of local neighborhood, used when computing C, C_w, and C^{LRF}. Similarly, $r_{min}(T_N)$ and $r_{max}(T_N)$ are user-defined values for V, V^{LRF}, and V^{LRF-AD}

[a] http://www2.isprs.org/commissions/comm3/wg4/3d-semantic-labeling.html

[b] http://www.liblas.org/samples/

[c] http://visionair.ge.imati.cnr.it/ontologies/shapes/viewmodels.jsp

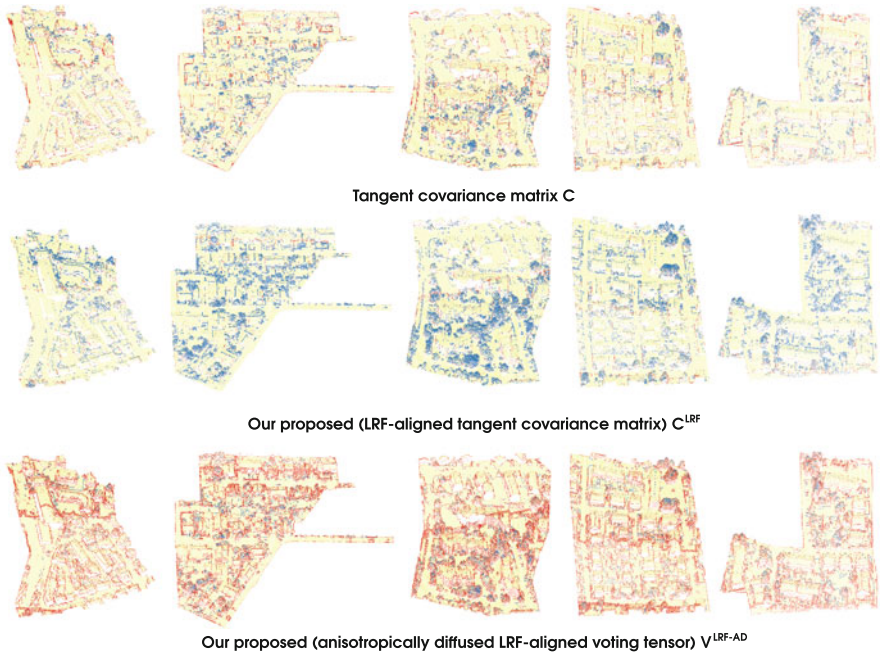


Fig. 7 Point rendering of the classification outcomes of LGDs for Regions 1–5 (*left to right*) of Vaihingen benchmark datasets (airborne LiDAR point clouds). Here, images are juxtaposed column-wise, for comparison

(Fig. 8b), and influence of diffusion velocity in computation of V^{LRF-AD} (Fig. 9). The default value of diffusion parameter δ is 0.16, for computing V, V^{LRF}, V^{LRF-AD}, as has been used in [30]. Dataset-specific parameters are given in Table 1. For point classification using algorithm by Keller et al. [10], we have used threshold parameter, $\epsilon = 0.5$. For multi-scale approach, the radii of local neighborhood, used as scales, are measured with respect to a normalized bounding box of the dataset. The normalization of the bounding box gives a canonical view volume, which is a cube of size 2 units and centered at (0,0,0), in $\mathbb{R}^3$.

Overall Performance of T_N in Point Classification We observe that T_N captures sharp features such as edges on smooth-feature, arches and railings in the bridge, stadium field, etc. better than the covariance matrix (Figs. 5 and 7). However, T_N does not detect point-type features as well as T_T (and C). The computational complexity of T_N is similar to that of C and T_T. The better performance of T_N in line-feature detection can be explained by the tensor voting approach. V is generated in the normal space which encodes local surface geometry better. V (or V^{LRF}) detects surface-type features well, as V is constructed in the normal space, which captures local surface geometry better. {*Note that all classification outcomes considered in this work use the definition as provided in the case of the covariance*

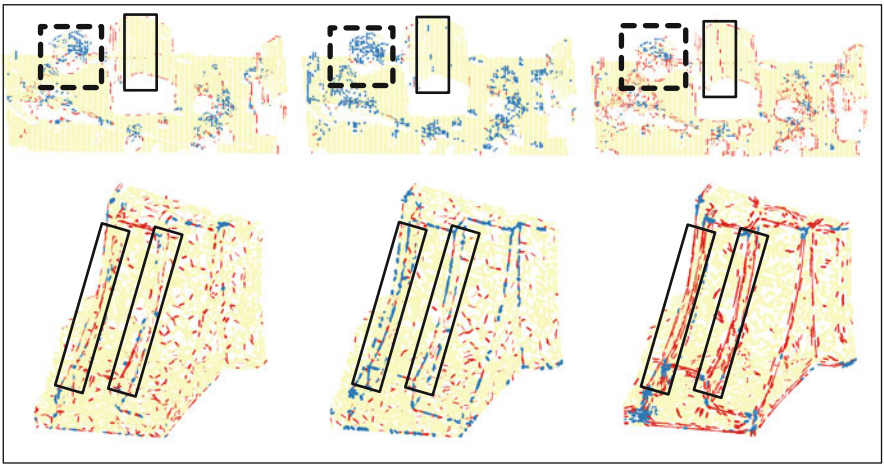
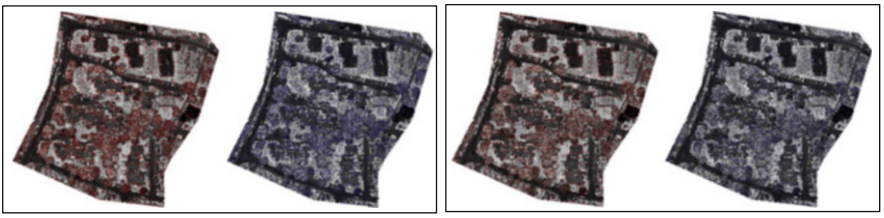


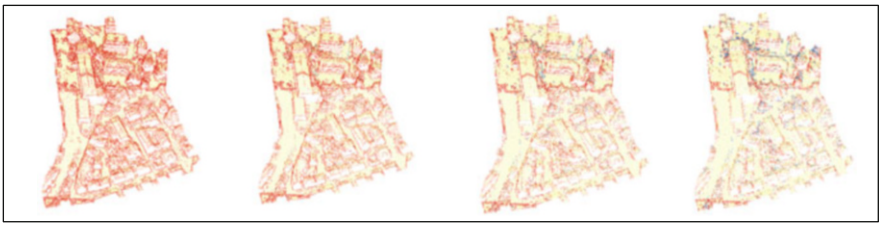
Fig. 8 Superquadric tensor glyph visualization of LGDs of $\mathscr{P}$, for tensor representation at one of the scales. The *solid and dashed boxes* highlight the sharp features (e.g., gabled roof, edges, etc.), and undefined geometry (e.g. foliage), respectively. (**a**) C,T_T, and T_N for Region-3 and fan-disk at $r = r_{min}$. (**b**) Region-1: C computed at r_{min}, r_{max} and multi-scale aggregate

matrix, C.} Upon anisotropic diffusion, these surface-type features are classified as line-type features. Thus, owing to the "persistence" of line-type features in V^{LRF}, the feature detection is more accurate in V^{LRF-AD}, compared to that to C (or C_w or C^{LRF}). However, a noticeable shortcoming of T_N is in detection of degenerate points (junctions/corners), which is not as effective as that by T_T or C_w. Since the point-type features are not preserved during anisotropic diffusion, they tend to be detected as weak line-type features, e.g. the corners (Fig. 1) and foliage (Fig. 8).

The plot comparing T_N with C (Fig. 5) shows that T_N detects more points as line-type features than C. T_N detects a significant percentage of points detected as line-type features exclusively; and a larger percentage of the points detected as line-type by C, is detected by T_N as well. A very small percentage of points are detected as line-type by C exclusively. Except in smooth-feature, T_N detects as many line-type features exclusively, as are detected by both T_N and C. Similar results are observed in the plot comparing T_N with T_T (Fig. 6), except that T_T detects fewer line-type features exclusively, compared to C.



(a) Multi-scale aggregate of local maxima of line- (red) and point- (blue) type features overlaid on LiDAR intensity map of Region 3, computed using C (left) and T_N (right).



(b) Increasing value of diffusion value, at 0.08, 0.12, 0.16, and 0.20 (from left to right)

Fig. 9 (*Top*) Visualization of multi-scale aggregate of local maxima of the saliency map for line-type (red) and point-type (blue) features. The likelihood of a point to be local maxima of line-type saliency is used as saturation of red, and similarly, that of point-type saliency, as saturation of blue. (*Bottom*) Effects of change in diffusion parameter δ when computing T_N for Region-1. Default value for all other experiments is $\delta = 0.16$ [30]. *Boxes* show sharp features, such as gabled roof. (**a**) Multi-scale aggregate of local maxima of line- (red) and point- (blue) type features overlaid on LiDAR intensity map of Region 3, computed using C (*left*) and T_N (*right*). (**b**) Increasing value of diffusion value, at 0.08, 0.12, 0.16, and 0.20 (from *left to right*)

Why Does V^{LRF-AD} Detect Line-Type Features Better? The classification outcomes using V (or V^{LRF}) are different from that of C (or C_w or C^{LRF}) (Fig. 1). However, they have certain *patterns* of correspondences, such as surface-type in former are line-type in latter. These patterns are visible from both tensor shapes and saliency maps (Figs. 1 and 8) and are better understood using the LRF-aligned descriptors. They are attributed to the reversing of ordering of eigenvalues observed when the LRFs of C^{LRF} and V^{LRF} align, i.e. $((\lambda_V^{LRF})_i = 1-(\lambda_C^{LRF})_{(2-i)}$ for $i = 0, 1, 2$ given in Eq. (6)). Thus, a disc-shaped neighborhood (for surface-type features) in V $[(\lambda_V^{LRF})_0 \gg (\lambda_V^{LRF})_1 \approx (\lambda_V^{LRF})_2]$ is equivalent to cylindrical-shaped neighborhood (for line-type features) in C $[(\lambda_C^{LRF})_0 \approx (\lambda_C^{LRF})_1 \gg (\lambda_C^{LRF})_2]$. Similarly, point- and surface-type features in C are captured as point-type features in V. Additionally, the classification outcomes of V "persist" through anisotropic diffusion in our proposed LGD, V^{LRF-AD}.

T_T (and hence, C) and T_N have similar computational complexity as, $T_T + T_N = I_d$ (used in the definition of ball-tensor). The complexity of tensor voting is given as $O(dNl \log N)$ for N points in $\mathscr{P} \in \mathbb{R}^d$ [20]. Space requirements for all second-order LGDs are $O(Nd^2)$. The point-rendering using saliency maps is less computationally expensive than the glyph-based visualization. Hence, we use point-rendering to

analyze the dataset globally, and glyphs for local regions-of-interest. The latter is specifically useful in finding misclassifications in a local region.

Juxtaposed Views Juxtaposed views of the tensor glyph visualizations show the combined differences in tensor shapes and the saliency maps, across the instances; e.g., the line-type features on sharp edges in gabled roof and fan-disk (Fig. 8a) clearly are more cylindrical in T_N compared to that of T_T and C. While comparing C and T_N, we need to consider the orientation of the glyphs, in addition to the color and size, since they are not LRF-aligned. However, for comparison of LRF-aligned C^{LRF} and $V^{LRF} - ad$, we can ignore comparisons of glyph orientation. Glyph rendering across different scales as well as using multi-scale aggregate (by averaging unit eigenvectors and normalized eigenvalues across scales) do not show any perceivable differences in LiDAR datasets (Fig. 8b). Additionally, glyph rendering and saliency maps help the user to decide some of the user-defined parameters. Our method requires several user-defined parameters, namely, range of scale, number of scales, and diffusion parameter, similar to [10, 30].

Multi-Scale Probabilistic Feature Classification Multi-scale aggregation of saliency maps, by averaging, has shown more accurate classification of surface- and line-type features, as the scale-wise differences are implicitly taken into consideration. As can be seen in the plot (Fig. 10), the number of points classified in each feature class does not change considerably across scales, for different LGDs (C, T_T, T_N), We chose the fan-disk dataset here as it has considerable distribution of points across the feature classes. We also observe that the surface-type features

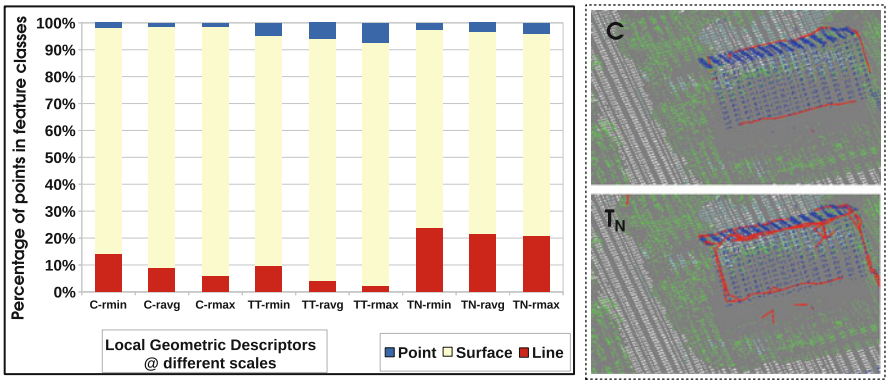


Fig. 10 (*Left*) The plot shows the variation in percentage of points detected in each feature class using each descriptor, in each scale, for fan-disk. From *left-to-right*, three columns represent r_{min}, r_{avg}, r_{max}, in that order; for each LGD. Columns 1–3 correspond to C, 4–6 to T_T, and 7–9 to T_N. With minor variations in classification across scales for each LGD, T_N detects more line-type features (*red*) comparatively. (*Right*) Point rendering of buildings (*blue*), foliage (*green*), road (*white*) of a region of interest of LiDAR point cloud of Region-4 of Vaihingen dataset, and preliminary results of line features (*red*) extracted using feature graph in [10]. Note that sharp features in gabled roofs can be observed in LGD T_N, unlike in C

which are misclassified as line-type features, get correctly classified in higher scales (Fig. 8b). In the case of LiDAR point clouds, multi-scale aggregation of local maxima of the saliency maps for line- and point-type features from T_N do not give useful information (Fig. 9), as is expected in tensor voting.

Anisotropic Diffusion The change in diffusion parameter influences the outcomes of line-type feature classification in T_N, e.g. gabled roofs in the Vaihingen dataset highlighted as the line-type features (black boxes in Fig. 9). At lower values of diffusion velocity parameter, δ, points are over-expressed as line-type features, and increase of δ causes most of those features to increasingly be classified as point-type. Outcomes at higher values of δ appear to be similar to those of T_T.

Weighted Covariance Analysis T_T (or C_w) captures sharp and point-type features better than C in two different instances (Figs. 1, 7, and 8). Firstly, T_T detects sharp features as strong point-type ones; whereas C captures them as weak line-type features, e.g. edges in fan-disk dataset (Fig. 8). Capturing sharp features as point-type features by T_T (or C_w) need not be considered inaccurate as these points can be treated as degenerate points owing to being intersection points of multiple planes. Secondly, the points in the foliage of trees in Vaihingen dataset (Fig. 8) are captured as point-type features, owing to random orientation of normals, by T_T (or C_w), whereas both C and T_N detect them as a mixture of line- and point-type features. Foliage is considered as a crowd of multiple objects, where T_T (or C_w) resolves the shape of local neighborhood at the point as spherical. However, the use of centroid in C gives inaccurate result in resolving the same, due to the random distribution of the points in the neighborhood. These two observations for T_T (or C_w) are attributed to both modifications; the point-based analysis (as opposed to the centroid-based) of the local neighborhood as well as weighted contributions of neighbors (i.e. using weights inversely proportional to the distances of neighbor to the point).

7 Conclusions

Here, we have studied LGDs of a point cloud for LiDAR datasets, for geometry-based point classification. We have compared two LGDs with the conventionally used covariance matrix, namely, an anisotropically diffused voting tensor, T_N, and a weighted covariance matrix, T_T. T_N performs better extraction of line-type features in point clouds than the conventionally used C; e.g. the gabled roofs in the Vaihingen dataset are detected (Fig. 10). T_T, enhances point-type features, however we reserve its in-depth study as future work. For comparing the classification outcomes accurately, we have introduced variants of these LGDs by aligning their LRFs. Expressing the LGDs as positive semidefinite tensors has enabled us to use tensor field techniques on the LGDs, e.g. visualization for qualitative comparison. We have introduced a probabilistic multi-scale saliency map based on its LGDs for geometric classification. We have used superquadric tensor glyphs for visually representing the LGDs of a point cloud. The visualizations have enabled us to

compare and qualitatively analyze the behavior of these LGDs, e.g. shape and orientation.

Our proposed methods have a limitation of user-defined parameters for multi-scale computation and diffusion parameter, as its predecessors [10, 30]. While these parameters give the user the control to bring out the best outcomes upon visual inspection, it would help the user to have a default setting, which can be automated.

Acknowledgements The authors wish to thank Akshay Jindal for running experiments; Srujana Merugu, Ingrid Hotz, T. K. Srikanth, and Vijay Natarajan, as well as several participants of Dagstuhl seminar 16142 for their helpful discussions; and anonymous reviewers for suggestions on improving the manuscript. This work has been partially funded by NRDMS programme of Dept. of Science and Technology, Government of India. The second co-author has been funded by sponsored projects with EMC^2-RSA India Pvt.; and FRHS, Bangalore, during her graduate study.

References

1. Daniels, J., Ha, L.K., Ochotta, T., Silva, C.T.: Robust smooth feature extraction from point clouds. In: IEEE International Conference on Shape Modeling and Applications, 2007. SMI'07, pp. 123–136. IEEE, Piscataway (2007)
2. Demantké, J., Mallet, C., David, N., Vallet, B.: Dimensionality based Scale Selection in 3D LiDAR Point Clouds. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. **38**(Part 5), W12 (2011)
3. Fleishman, S., Cohen-Or, D., Silva, C.T.: Robust moving least-squares fitting with sharp features. In: ACM Transactions on Graphics (TOG), vol. 24, pp. 544–552. ACM, New York (2005)
4. Gelfand, N., Mitra, N.J., Guibas, L.J., Pottmann, H.: Robust global registration. In: Proceedings of the Third Eurographics Symposium on Geometry Processing, SGP '05. Eurographics, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (2005). http://dl.acm.org/citation.cfm?id=1281920.1281953
5. Gressin, A., Mallet, C., Demantké, J., David, N.: Towards 3D LiDAR point cloud registration improvement using optimal neighborhood knowledge. ISPRS J. Photogramm. Remote Sens. **79**, 240–251 (2013)
6. Gumhold, S., Wang, X., MacLeod, R.: Feature extraction from point clouds. In: Proceedings of 10th International Meshing Roundtable, vol. 2001. Citeseer (2001)
7. Guy, G., Medioni, G.: Inference of Surfaces, 3D Curves, and Junctions from Sparse, Noisy, 3D Data. IEEE Trans. Pattern Anal. Mach. Intell. **19**(11), 1265–1277 (1997)
8. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Surface reconstruction from unorganized points. SIGGRAPH Comput. Graph. **26**(2), 71–78 (1992). doi: doi:10.1145/142920.134011. http://doi.acm.org/10.1145/142920.134011
9. Javed, W., Elmqvist, N.: Exploring the design space of composite visualization. In: 2012 IEEE Pacific Visualization Symposium (PacificVis), pp. 1–8. IEEE, Piscataway (2012)
10. Keller, P., Kreylos, O., Vanco, M., Hering-Bertram, M., Cowgill, E.S., Kellogg, L.H., Hamann, B., Hagen, H.: Extracting and visualizing structural features in environmental point cloud LiDaR data sets. In: Topological Methods in Data Analysis and Visualization, pp. 179–192. Springer, Berlin (2011)
11. Kim, E., Medioni, G.: Urban scene understanding from aerial and ground lidar data. Mach. Vis. Appl. **22**(4), 691–703 (2011)
12. Kindlmann, G.: Superquadric tensor glyphs. In: Proceedings of the Sixth Joint Eurographics-IEEE TCVG conference on Visualization, pp. 147–154. Eurographics Association (2004)

13. Knutsson, H.: Representing local structure using tensors. In: 6th Scandinavian Conference on Image Analysis, Oulu, Finland, pp. 244–251. Linköping University Electronic Press (1989)
14. Knutsson, H., Westin, C.F., Andersson, M.: Representing local structure using tensors II. In: Image Analysis, pp. 545–556. Springer, Berlin (2011)
15. Kobbelt, L., Botsch, M.: A survey of point-based techniques in computer graphics. Comput. Graph. **28**(6), 801–814 (2004)
16. Kumari, B., Sreevalsan-Nair, J.: An interactive visual analytic tool for semantic classification of 3d urban lidar point cloud. In: Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, p. 73. ACM, New York (2015)
17. Lalonde, J.F., Unnikrishnan, R., Vandapel, N., Hebert, M.: Scale selection for classification of point-sampled 3d surfaces. In: Fifth International Conference on 3-D Digital Imaging and Modeling, 2005. 3DIM 2005, pp. 285–292. IEEE, Piscataway (2005)
18. Light, A., Bartlein, P.J.: The end of the rainbow? Color schemes for improved data graphics. EOS **85**(40), 385–391 (2004)
19. Medioni, G., Tang, C.K., Lee, M.S.: Tensor voting: theory and applications. Proceedings of RFIA, Paris, France **3** (2000)
20. Mordohai, P., Medioni, G.: Dimensionality estimation, manifold learning and function approximation using tensor voting. J. Mach. Learn. Res. **11**, 411–450 (2010)
21. Moreno, R., Pizarro, L., Burgeth, B., Weickert, J., Garcia, M.A., Puig, D.: Adaptation of tensor voting to image structure estimation. In: New Developments in the Visualization and Processing of Tensor Fields, pp. 29–50. Springer, Berlin (2012)
22. Park, M.K., Lee, S.J., Lee, K.H.: Multi-scale tensor voting for feature extraction from unstructured point clouds. Graph. Model. **74**(4), 197–208 (2012)
23. Pauly, M., Keiser, R., Gross, M.: Multi-scale feature extraction on point-sampled surfaces. Comput. Graphics Forum **22**(3), 281–289 (2003)
24. Renteln, P.: Manifolds, Tensors, and Forms: An Introduction for Mathematicians and Physicists. Cambridge University Press, Cambridge (2013)
25. Rottensteiner, F.: Status and further prospects of object extraction from image and laser data. In: 2009 Joint Urban Remote Sensing Event, pp. 1–10. IEEE, Piscataway (2009)
26. Schultz, T., Kindlmann, G.L.: Superquadric glyphs for symmetric second-order tensors. IEEE Trans. Vis. Comput. Graph. **16**(6), 1595–1604 (2010)
27. Taubin, G.: Estimating the tensor of curvature of a surface from a polyhedral approximation. In: Proceedings of the Fifth International Conference on Computer Vision, 1995, pp. 902–907. IEEE, Piscataway (1995)
28. Tombari, F., Salti, S., Di Stefano, L.: Unique signatures of histograms for local surface description. In: Computer Vision–ECCV 2010, pp. 356–369. Springer, Berlin (2010)
29. Tong, W.S., Tang, C.K.: Robust estimation of adaptive tensors of curvature by tensor voting. IEEE Trans. Pattern Anal. Mach. Intell. **27**(3), 434–449 (2005)
30. Wang, S., Hou, T., Li, S., Su, Z., Qin, H.: Anisotropic elliptic PDEs for feature classification. IEEE Trans. Vis. Comput. Graph. **19**(10), 1606–1618 (2013)
31. Weber, C., Hahmann, S., Hagen, H.: Methods for feature detection in point clouds. In: OASIcs-OpenAccess Series in Informatics, vol. 19. Schlöss Dagstuhl-Leibniz-Zentrum fuer Informatik (2011)
32. Westin, C.F., Peled, S., Gudbjartsson, H., Kikinis, R., Jolesz, F.A.: Geometrical diffusion measures for MRI from tensor basis analysis. In: 5th Scientific Meeting and Exhibition of International Society of Magnetic Resonance in Medicine ISMRM '97. ISMRM (1997)
33. Yang, B., Dong, Z.: A shape-based segmentation method for mobile laser scanning point clouds. ISPRS J. Photogramm. Remote Sens. **81**, 19–30 (2013)
34. Zhang, C., Schultz, T., Lawonn, K., Eisemann, E., Vilanova, A.: Glyph-based comparative visualization for diffusion tensor fields. IEEE Trans. Vis. Comput. Graph. **22**(1), 797–806 (2016)

Part III
Diffusion Modeling and Microstructure

Diffusion MRI Anisotropy: Modeling, Analysis and Interpretation

Rutger H.J. Fick, Marco Pizzolato, Demian Wassermann, and Rachid Deriche

Abstract The micro-architecture of brain tissue obstructs the movement of diffusing water molecules, causing tissue-dependent, often anisotropic diffusion profiles. In diffusion MRI (dMRI), the relation between brain tissue structure and diffusion anisotropy is studied using oriented diffusion gradients, resulting in tissue- and orientation-dependent diffusion-weighted images (DWIs). Over time, various methods have been proposed that summarize these DWIs, that can be measured at different orientations, gradient strengths and diffusion times into one "diffusion anisotropy" measure. This book chapter is dedicated to understanding the similarities and differences between the diffusion anisotropy metrics that different methods estimate. We first discuss the physical interpretation of diffusion anisotropy in terms of the diffusion properties around nervous tissue. We then explain how DWIs are influenced by diffusion anisotropy and the parameters of the dMRI acquisition itself. We then go through the state-of-the-art of signal-based and multi-compartment-based dMRI methods that estimate diffusion anisotropy-related methods, focusing on their limitations and applications. We finally discuss confounding factors in the estimation of diffusion anisotropy and current challenges.

1 Introduction

In brain imaging, diffusion anisotropy is a manifestation of tissues obstructing the otherwise free diffusion of water molecules. Brain tissues with different structural make-ups, e.g. healthy or diseased, influence the diffusion differently [15, 51]. Relating the observed diffusion with the underlying tissue structure has been one of diffusion MRI's (dMRI's) main challenges. This challenge can be seen as a variant of the work *Can One Hear The Shape of a Drum* by Kac [40]. Basser et al. [14] were the first to determine the voxel-wise orientational dependence of diffusion

R.H.J. Fick (✉) • M. Pizzolato • D. Wassermann • R. Deriche
Université Côte d'Azur, Inria, Athena Project Team, Sophia Antipolis, France
e-mail: rutger.fick@inria.fr; marco.pizzolato@inria.fr; demian.wassermann@inria.fr; rachid.deriche@inria.fr

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_9

in biological tissue by fitting a tensor to the signals of non-collinearly oriented diffusion gradients [63]. For the first time, this representation made it possible to describe both the tissue orientation and the "coherence" of the underlying tissue by means of rotationally-invariant indices such as the Fractional Anisotropy (FA) [12]. Since then, a plethora of dMRI models have been proposed to more accurately relate tissue properties to the measured signal by using less or different assumptions on the tissue configuration or increasing the requirements of the signal acquisition [see e.g. 4, 8, 9, 16, 42, 43, 55, 68, 72].

This chapter is meant as a review of current methods that either directly estimate diffusion anisotropy measures or anisotropy-related tissue properties. We restrict ourselves to diffusion anisotropy as a property of the overall diffusion signal (such as FA); anisotropy as a property of the dispersion of diffusion micro-environments; and anisotropy as a property of the micro-environment itself. In the latter two cases, we view a micro-environment as the diffusion profile of a single axon and its immediate surroundings. Using this definition, the signal measured from a bundle of axons can be seen as an ensemble of micro-environments, each having their own orientation and signal contribution to the overall signal. As an example of a method that estimates this micro-environment dispersion, we include the Neurite Orientation Dispersion and Density Imaging (NODDI) [73]. To describe anisotropy as a property of one micro-environment, we include the Spherical Mean Technique [42]. Finally, as diffusion anisotropy is a consequence of diffusion restriction and hindrance, we will put special emphasis on the influence and modeling of the *diffusion time*. This angle of approach is timely, as STimulated Echo Acquisition Mode (STEAM) pulse sequences have recently allowed the in vivo exploration of long diffusion times [25, 32], verifying the existence of time-dependent diffusion even at low b-values [32].

The structure of this chapter is as follows: In Sect. 2, we first provide a physical interpretation on what diffusion anisotropy is. In Sect. 3 we clarify the mechanics of a standard Pulsed Gradient Spin Echo (PGSE) sequence and how the measured signal is influenced by the tissue. In Sect. 4, we describe the most relevant techniques that have been proposed to estimate and interpret diffusion anisotropy. We describe the differences between different signal-based anisotropy measures and relate them to axon dispersion-related metrics. We then discuss the time-dependence of anisotropy measures in Sect. 5. In this last section, we also go into the diffusion-time-dependence of diffusion restriction in the extra-axonal space [17, 52]. Finally, we discuss challenges and confounding issues that these methods face in Sect. 6.

2 Diffusion Anisotropy: The Phenomenon

The characteristics of diffusion anisotropy in the brain depend on how the diffusion process is restricted or hindered by the boundaries of the nervous tissue. To get an idea of this relationship, we first discuss the general concept of individual

spin movement and the Ensemble Average Propagator (EAP) in the presence of restricting boundaries in Sect. 2.1. We then discuss to a greater extent the variety and complexity of the nervous tissue in Sect. 2.2.

2.1 Diffusion and the Ensemble Average Propagator

In a fluid, water particles follow random paths according to Brownian motion [28]. When we consider an *ensemble* of these particles in a volume, we can describe their average probability density $P(\mathbf{r};\tau)$ that a particle will undergo a displacement $\mathbf{r} \in \mathbb{R}^3$ during diffusion time $\tau \in \mathbb{R}^+$. This quantity is often referred to as the diffusion propagator or the ensemble average propagator (EAP) [44]. In a free solution, the EAP can be described by a Gaussian distribution as

$$P(\mathbf{r};\tau) = \frac{1}{\sqrt{4\pi D\tau}} e^{-\frac{\|\mathbf{r}\|^2}{4D\tau}} \tag{1}$$

where D is the diffusion coefficient. Equation (1) shows that the likelihood that particles travel further increases when either D or τ increases. While keeping D constant, this concept can be made clear using isocontours such that $P(\mathbf{r};\tau) = c$ with $c > 0$. Figure 1 shows the same isocontour for diffusion times $\tau_1 < \tau_2 < \tau_3$ in four schematic representations of different tissue types. As can be seen by the growth of the isocontours, using longer τ increases the likelihood that particles travel further. The shape of the isocontour depends on the structure of the surrounding tissue. From left to right, in free water, where Eq. (1) is a good approximation, particles are unrestricted and travel furthest with isotropic, Gaussian probability. Next, at a coarse diffusion scale, gray matter tissue can often be seen as generally unorganized and hinders diffusion equally in all directions. For this reason, these tissues also produce isotropic contours, but smaller than those in free water. In axon bundles, here illustrated as gray lines, axons are mostly aligned with the bundle axis. Particle movement is restricted perpendicular to this direction and is relatively free along it, causing *anisotropic* isocontours [46, 50, 64]. Finally, in areas where two bundles cross there is a mix between the isocontours of each bundle.

Note that we intentionally drew the isocontours for τ_1 more isotropic than those of τ_3 in the right two white matter tissues. For shorter τ, particles have not had much time to interact with surrounding tissue, resulting in a similar probability that a particle travels in any direction. The isocontours for very short τ will therefore always be isotropic. For longer τ, particles have had more time to interact with the tissue, either traveling far along a relatively unrestricted direction, or staying close to its origin along a restricted direction, resulting in more anisotropic profiles [62]. When the tissue can be seen as axially symmetric (i.e. in a single bundle), this means that the perpendicular diffusivity $D_\perp$ becomes τ-dependent and decreases

as τ increases [22]. Different tissue types will induce different τ-dependence of the EAP [53, 54].

2.2 *Microstructure of the Brain: The Complicated Reality*

Images such as those in Fig. 1 are useful to illustrate general properties of different brain tissues. However, it is important to realize that these are extreme simplifications. Strictly speaking, the terms gray and white matter are only valid in the context of gross anatomy. Gray matter is only distinguished from white matter, in that gray matter contains numerous cell bodies and relatively few myelinated axons, while white matter is composed chiefly of long-range myelinated axon tracts and contains relatively very few cell bodies. The brain also contains glial cells of various kinds who support the functioning of neurons.

In white matter bundles, while axons are often near-tubular, their diameter, amount of myelination and the space between them varies significantly. Water particles diffusing in different parts of this tissue, e.g. the intra- or extra-axonal space, are restricted in ways that are characteristic of that tissue type. However, the diffusion process can only be probed over large ensembles of tissues. This is illustrated by the fact that axon diameters in humans range between 0.2 and 2 μm [1, 49], while the imaging resolution of diffusion MRI is around a millimeter. Appropriately dealing with the heterogeneity of the tissue and its complex influence on water diffusion is one of the big challenges of diffusion MRI. In the next section, we will go into more detail on how diffusion MRI can be used to measure a signal that is related to the EAP.

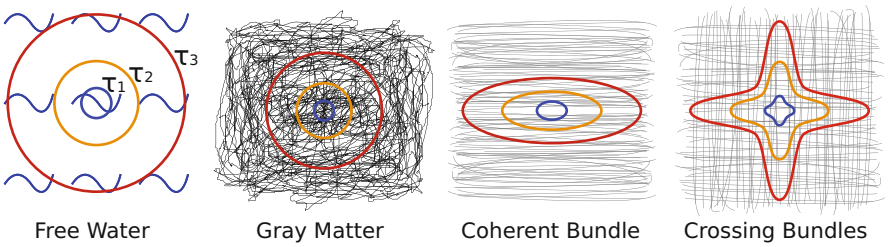


Fig. 1 Schematic representations of different tissue types with their corresponding $P(\mathbf{r}, \tau)$ isocontours for different diffusion times $\tau_1 < \tau_2 < \tau_3$. In the "Free Water" image the *blue curves* just indicate the presence of water, while the *grey lines* in the right three images represent tissue boundaries. Longer τ lets particles travel further, indicated by the smaller *blue isocontour* for τ_1 to the largest *red isocontour* for τ_3. The shape of the isocontour depends on the structure of the surrounding tissue. Image inspired by Alexander [3]

3 Measurements of Diffusion with Diffusion-Weighted MRI

The estimation of diffusion anisotropy can be thought, in first approximation, as the assessment of the amount of preference that the diffusion process has for a specific spatial direction, compared to the others, in terms of diffusivity. Therefore, this assessment requires sensing the diffusion signal along multiple spatial directions, regardless of the representation adopted to describe the signal itself. In MRI, this is typically done by acquiring a collection of images of the target object, e.g. the brain. Each image is acquired when the experimental conditions within the magnet's bore determine a specific *diffusion-weighting* along the selected spatial direction: this is a Diffusion-Weighted Image (DWI). The diffusion-weighting is globally encoded by the *b-value* [46], measured in $\mathrm{s/mm^2}$, a quantity that is the reciprocal of the diffusivity, D ($\mathrm{mm^2/s}$). The intensity of the diffusion-weighting, i.e. the b-value, is determined by the acquisition setup.

The most common type of acquisition is the Pulsed Gradient Spin-Echo sequence (PGSE) [60], where a DWI is obtained by applying two diffusion gradients with intensity $G = \|\mathbf{G}\|$ (T/m) and duration δ (s) to the tissue, separated by the separation time Δ (s). We illustrate this sequence in Fig. 2. The resulting signal is 'weighted', along the applied gradient direction, with b-value [60]

$$b = \gamma^2 G^2 \delta^2 \left(\Delta - \frac{\delta}{3} \right) \tag{2}$$

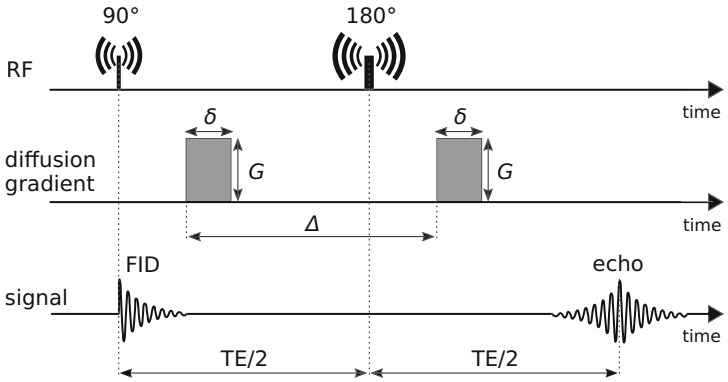


Fig. 2 Schematic illustration of the pulsed gradient spin echo sequence (PGSE). The sequence is represented as the time evolution, i.e. occurrence and duration, of radio-frequency pulses (RF) in the *first line*, diffusion gradient pulses in the *second line*, and measured signal in the *third line*. The illustration reports the 90° and 180° RF pulses separated by half the echo-time TE, two diffusion gradient pulses of strength G and duration δ, separated by a time Δ, the free induction decay (FID) and echo of the measured signal

where γ is, when measuring water diffusion, the nuclear gyromagnetic ratio of the water proton ^{1}H. The measurement of the diffusion signal is directly related to the concept of *attenuation*. Indeed, in the presence of diffusion, the signal intensities $S(b)$ of the voxels of a DWI are lower than the corresponding intensities when the image is acquired without diffusion-weighting $S_0 = S(0)$. Along the selected gradient direction, the quantity $E(b) = S(b)/S_0$ expresses, for each voxel, the attenuation of the diffusion-weighted signal. In the absence of restrictions to the diffusion process, the attenuation is [60]

$$E(b) = \frac{S(b)}{S_0} = e^{-bD}, \tag{3}$$

which expresses an exponential attenuation profile, as is illustrated in Fig. 3a.

In the case of the PGSE sequence, this attenuation phenomenon can be interpreted as the result of a differential mechanism. The sequence, shown in Fig. 2, starts with a 90° radio-frequency pulse after which it is possible to measure a signal, namely Free Induction Decay (FID), that is related to the macroscopic spins' net magnetization. After a time TE/2, with TE being the echo-time, a second 180° radio-frequency pulse has the effect of generating an echo of the signal whose peak is at time TE, corresponding to the end of the sequence [34]. The first diffusion gradient pulse is applied between the two radio-frequency pulses. Here, we assume

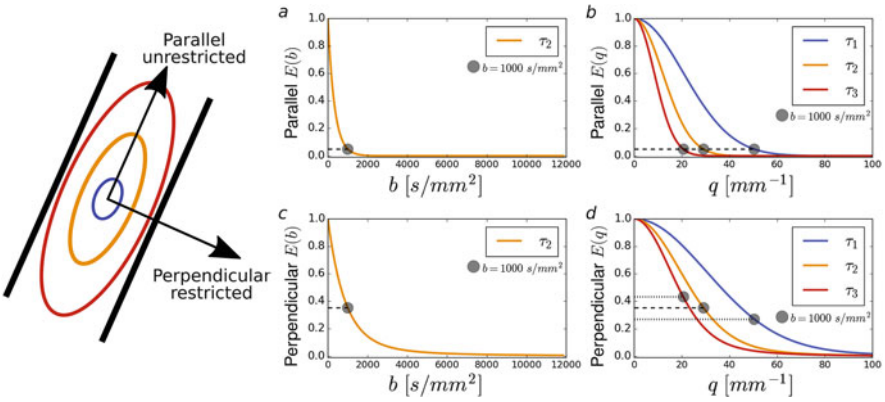


Fig. 3 The effect of b-value, q-value and diffusion time on the signal attenuation when diffusion is unrestricted or restricted by the presence of a barrier (a cylinder). The graphs report the signal attenuations along a direction parallel to the restriction (**a,b**)—represented by *two black barriers* in the schematic image—and along the perpendicular direction (**c,d**), where the diffusion process is restricted. The attenuations are reported as function of the b-value (**a,c**) with diffusion time τ_2, and q-value (**b,d**) for increasing diffusion times $\tau_1 < \tau_2 < \tau_3$. Dots indicate the attenuation measured at $b = 1000\,\mathrm{s/mm^2}$. Note that if we were to plot the single cylinder signal attenuation in *d* in log-scale we would find diffraction patterns [19], but in practice these are never visible because nervous tissues contain distributions of axon diameters, whose diffraction patterns average out to a *smooth line*

the *narrow gradient pulse* condition $\delta \ll \Delta$, which implies that spins are static during the application of the gradient pulses [63]. Under this assumption, after the first gradient pulse, a spin located at position $\mathbf{r}_1$ is subject to a phase accumulation $\phi_1 = \gamma\delta\mathbf{G} \cdot \mathbf{r}_1$. After a time Δ from the start of the first gradient pulse, and after the 180° radio-frequency pulse, a second gradient pulse of equal magnitude and duration to the first is applied. If the spin has moved to a position $\mathbf{r}_2$ the phase accumulation during the second pulse is $\phi_2 = \gamma\delta\mathbf{G} \cdot \mathbf{r}_2$. However, the 180° radio-frequency pulse has the effect of changing the sign of the second gradient pulse. Therefore, at the end of the sequence, i.e. at the echo-time TE, the spin has acquired a net phase shift

$$\phi = \phi_2 - \phi_1 = \gamma\delta\mathbf{G} \cdot (\mathbf{r}_2 - \mathbf{r}_1) = \gamma\delta\mathbf{G} \cdot \mathbf{r} \tag{4}$$

which is null in the case the spin remained static, i.e. $\mathbf{r} = \mathbf{r}_2 - \mathbf{r}_1 = 0$.

The signal attenuation takes into account an ensemble of spins and can be related to the ensemble average propagator (EAP), $P(\mathbf{r}, \tau)$, via a Fourier relationship under the q-space formalism [18, 63]

$$E(\mathbf{q}, \tau) = \int_{\mathbb{R}^3} P(\mathbf{r}, \tau) e^{j2\pi\mathbf{q}\cdot\mathbf{r}} d\mathbf{r} \tag{5}$$

where $\mathbf{q}$ is the wave vector and τ is diffusion time, which for the PGSE sequence are expressed as

$$\mathbf{q} = \frac{\gamma\delta\mathbf{G}}{2\pi} \text{ and } \tau = \Delta - \delta/3. \tag{6}$$

These quantities influence differently the diffusion-weighting, i.e. the b-value. Indeed, the wave frequency $q = \|\mathbf{q}\|$ is expressed in mm^{-1} and is the reciprocal of the spin displacement $r = \|\mathbf{r}\|$ expressed in mm. As such, by increasing the measured spatial frequency q, we can obtain a higher resolution of its inverse Fourier transform, the diffusion propagator $P(\mathbf{r}, \tau)$ described by r. In addition, the diffusion time τ expresses the time interval during which spins are allowed to diffuse before measurement. A longer diffusion time allows the spins to move a longer distance causing, in the absence of restrictions to the diffusion process, a larger net phase shift, i.e. a stronger attenuation of the signal. Therefore, expressing the diffusion-weighting in terms of $\mathbf{q}$ and τ can provide useful insights on the signal nature. In the absence of restrictions to the diffusion process, Eq. (5) has a closed form. This is obtained by substituting q and τ of Eq. (6) into the formulation of the b-value expressed in Eq. (2), such that $b = 4\pi^2 q^2 \tau$ and Eq. (3) becomes

$$E(q, \tau) = e^{-4\pi^2 q^2 \tau D} \tag{7}$$

which expresses a Gaussian attenuation profile as function of q. However, Eqs. (3) and (7) are valid when the diffusion process can be considered unrestricted, e.g. when the movement of spins is not obstructed by the presence of a barrier. In the case of restricted diffusion, for instance when the signal is measured along a direction perpendicular to a barrier, these equations are no longer valid.

Unrestricted and restricted scenarios are depicted by the schematic representation in the left side of Fig. 3, where the diffusion process occurs between two parallel barriers, i.e. the restriction, and where two arrows represent the measurement directions parallel and perpendicular to the restriction. The figure also illustrates the signal attenuation in the case of parallel, unrestricted diffusion (Fig. 3a,b), and in the case of perpendicular, restricted diffusion (Fig. 3c,d). The unrestricted attenuations are obtained with Eqs. (3) and (7), whereas the restricted ones are simulated as the diffusion signal attenuation generated within an ensemble of cylinders along the direction perpendicular to the cylinders' axes [19]. Moreover, the curves in Fig. 3a and c are reported as function of the b-value with diffusion time τ_2, whereas the curves in Fig. 3b and d are functions of the q-value and are reported for increasing diffusion times $\tau_1 < \tau_2 < \tau_3$. In the graphs, points of each curve, corresponding to $b = 1000\ s/mm^2$, are highlighted with a dot.

Along the unrestricted direction (Fig. 3a,b) the attenuation values, indicated by dots, are lower than the corresponding ones along the restricted direction (Fig. 3c,d). Indeed, when diffusion is restricted by the presence of the barrier, the spins are subject to a smaller net displacement and the signal attenuates less.

The choice of q and τ to obtain a certain diffusion-weighting, i.e. a specific b-value, assumes different relevance in terms of signal attenuation depending on whether diffusion is restricted or not. In the absence of restrictions (Fig. 3a,b), an increase of q-value or diffusion time always attenuates the signal, and points with different q and τ, but with same b-value, render the same amount of attenuation (Fig. 3b). However, when diffusion is restricted (Fig. 3c,d), an increase of the diffusion time τ implies letting the spins diffuse a longer distance with the consequence of experiencing more restriction. In this case, the Gaussian attenuation expressed by Eq. (7) is not longer valid. Indeed, different combinations of q and τ render different non-Gaussian profiles of signal attenuation, and points with same b-value—the dots of Fig. 3d—correspond to different attenuations.

The estimation of diffusion anisotropy, based on the diffusion signal attenuation along different gradient directions, depends on the chosen experimental parameters, especially q and τ. Indeed, different sets of parameters lead to different signal attenuations—depending on the underlying diffusion process—and consequently to different measurements of anisotropy. Ideally, a complete characterization of anisotropy would require the measurement of the diffusion signal attenuation for many gradient directions, q-values, and diffusion times τ. However, the optimal sampling is still under debate. In practice, with reference to diffusion anisotropy, the choice of how to sample the diffusion signal depends on the application and on the chosen signal representation. For instance, in DTI it is common practice to measure only one sample, i.e. one b-value, per gradient direction [47] as illustrated in Fig. 4.

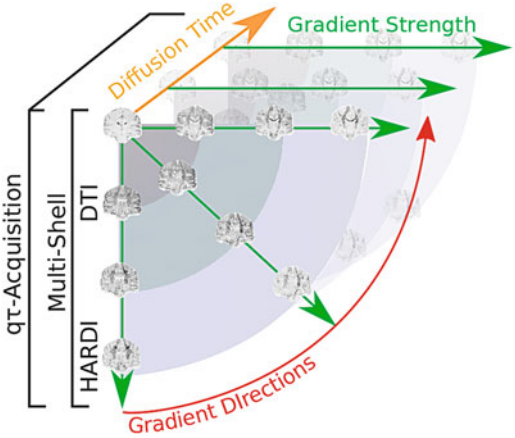


Fig. 4 Schematic for different types of acquisition schemes. DTI schemes are typically acquired at a low gradient strength, typically $b < 1000\text{s}/\text{mm}^2$, for a minimum of 6 gradient directions. HARDI typically uses higher gradient strengths, typically $b > 2000\text{s}/\text{mm}^2$ with over 40 gradient directions. Multi-shell uses DTI and HARDI shells for a given diffusion time, and finally qτ-acquisitions are basically multi-shell acquisitions for a range of diffusion times

However, with High Angular Resolution Diffusion Imaging (HARDI) the number of directions—typically above 40—and the b-value is considerably increased to obtain a signal representation with a higher angular resolution [69]. Other signal representations require signal acquisition at different diffusion-weightings. A common choice, namely *multi-shell*, consists in acquiring different q-shells while fixing the diffusion time. Each shell represents a collection of samples in the three-dimensional space with the same q-value. These samples can be imagined as lying on a sphere, as shown in Fig. 4, and it is convenient to distribute them uniformly on the spherical surface to obtain an optimal spatial coverage. This concept can be expanded among shells such that all of the acquired samples lie on different non-collinear directions [20]. The multi-shell concept can be extended to τ-shells, called a qτ-acquisition [30], since nowadays there exist signal representations that exploit different values for both q and τ. In this case, a complete q-shell scheme—with samples distributed along different gradient directions and with different diffusion-weightings—is acquired for each desired diffusion time.

4 The Inter-Model Variability of Diffusion Anisotropy

We now return to the metric that is most commonly used as a marker for *changes* in tissue microstructure: diffusion anisotropy. Simply meaning "deviation from diffusion isotropy", different interpretations of diffusion anisotropy have been

proposed using different acquisition requirements and mathematical underpinnings [12, 42, 55, 68]. Out of these, the clinical applications of Fractional Anisotropy (FA) [12] of the Diffusion Tensor Imaging (DTI) model [14] has been most widely explored. Changes in FA have been related to brain diseases such as ischemia, multiple sclerosis, trauma, brain tumors and many more [see e.g. reviews by 7, 59]. For this reason, FA is seen as a "potential biomarker" for these disease patterns, where biomarker is a portmanteau of "biological marker" [61]. However, the fact that FA is sensitive to all these processes also means that it is specific to none of them.

Diffusion anisotropy measures, as a rule of thumb, always have the following three properties:

- They are rotationally invariant, i.e., insensitive to rotations.
- They are normalized, with zero being the lowest measure for diffusion anisotropy and one being the highest.
- They somehow describe "deviation from diffusion isotropy".

The last point is intentionally left open to interpretation, which is exactly the point we are making in this section. To illustrate this, we discuss seven different anisotropy measures; Fraction Anisotropy (FA) [12], Relative Anisotropy (RA), Kurtosis Fractional Anisotropy (KFA), Generalized Fraction Anisotropy (GFA) [68]; Propagator Anisotropy (PA) [55]; Orientation Dispersion Index (ODI) [73]; and microscopic Fractional Anisotropy (μFA) [42]. All of these except ODI and μFA are signal-based metrics for diffusion anisotropy. We added these two to illustrate that the concept of diffusion anisotropy transcends signal-based metrics.

We start this section by first detailing the data of the Human Connectome Project [58] that we use to illustrate different diffusion anisotropy measures in Sect. 4.1. We then describe the inter-model variability of the mathematical definition and estimation of signal-based diffusion anisotropy in Sect. 4.2. We then detail anisotropy as a property of axon dispersion of micro-environments (ODI) in Sect. 4.3, and as a property of one micro-environment (μFA) in Sect. 4.4. The anisotropy measures of all presented techniques are qualitatively and quantitative compared in Figs. 5, 6 and 7.

4.1 Data Set Description and Adopted Notation

We use the MGH Adult Diffusion Data of the Human Connectome Project to illustrate different measures of diffusion anisotropy [5, 33, 45, 58]. This data set was acquired at particularly high b-values $\{0, 1000, 3000, 5000, 10000\}$ s/mm^2 with $\{40, 64, 64, 128, 256\}$ directions, respectively. The diffusion time and pulse separation time in this data are $\delta/\Delta = 12.9/21.8$ ms with $1.5 \times 1.5 \times 1.5$ mm^3 resolution and $TE/TR = 57/8800$ ms.

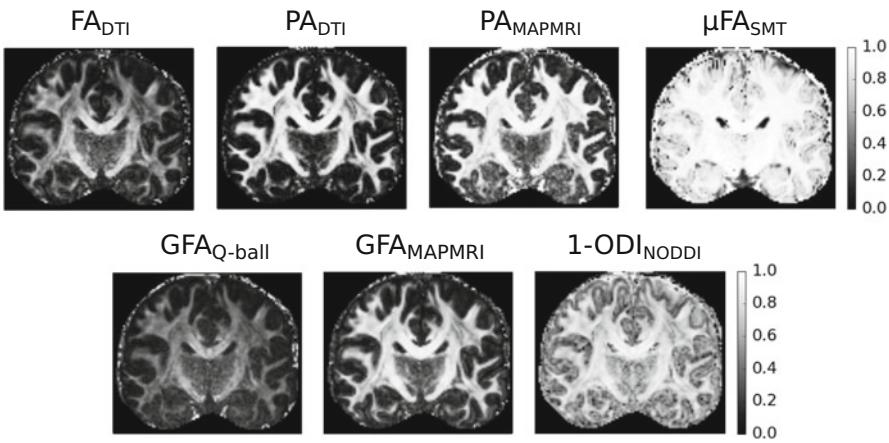


Fig. 5 Visualization of various normalized measures describing diffusion anisotropy, where the corresponding model is given in the subscript. In the top row, we show Fractional Anisotropy (FA) and Propagator Anisotropy (PA) of Diffusion Tensor Imaging (DTI), PA by Mean Apparent Propagator (MAP)-MRI and micro-FA by Spherical Mean Technique (SMT). In the bottom row, we show Generalized Fractional Anisotropy (GFA) by Q-ball Imaging and by MAP-MRI and finally one minus the Orientation Dispersion Index (ODI) by Neurite Orientation Dispersion and Density Imaging (NODDI). The complement of ODI is shown for overall coherence, since high ODI normally indicates low anisotropy

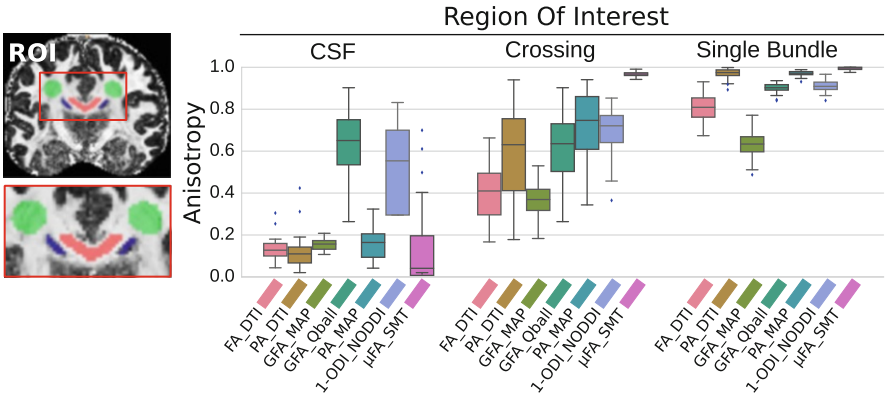


Fig. 6 Box-plots of anisotropy measures in estimated in different regions of interest, indicated as the different colors in the coronal brain slice in the right corner. We illustrate anisotropy in free water (CSF), a single bundle (Corpus Callosum) and a crossing area (Centrum Semiovale). It can be seen that the same metric for different techniques, or different metrics for the same technique can result in different estimates of anisotropy. Note that the estimates of μFA (the most-right metric per group in pink) in the crossing and single bundles are so consistent that the boxplots appear almost flat

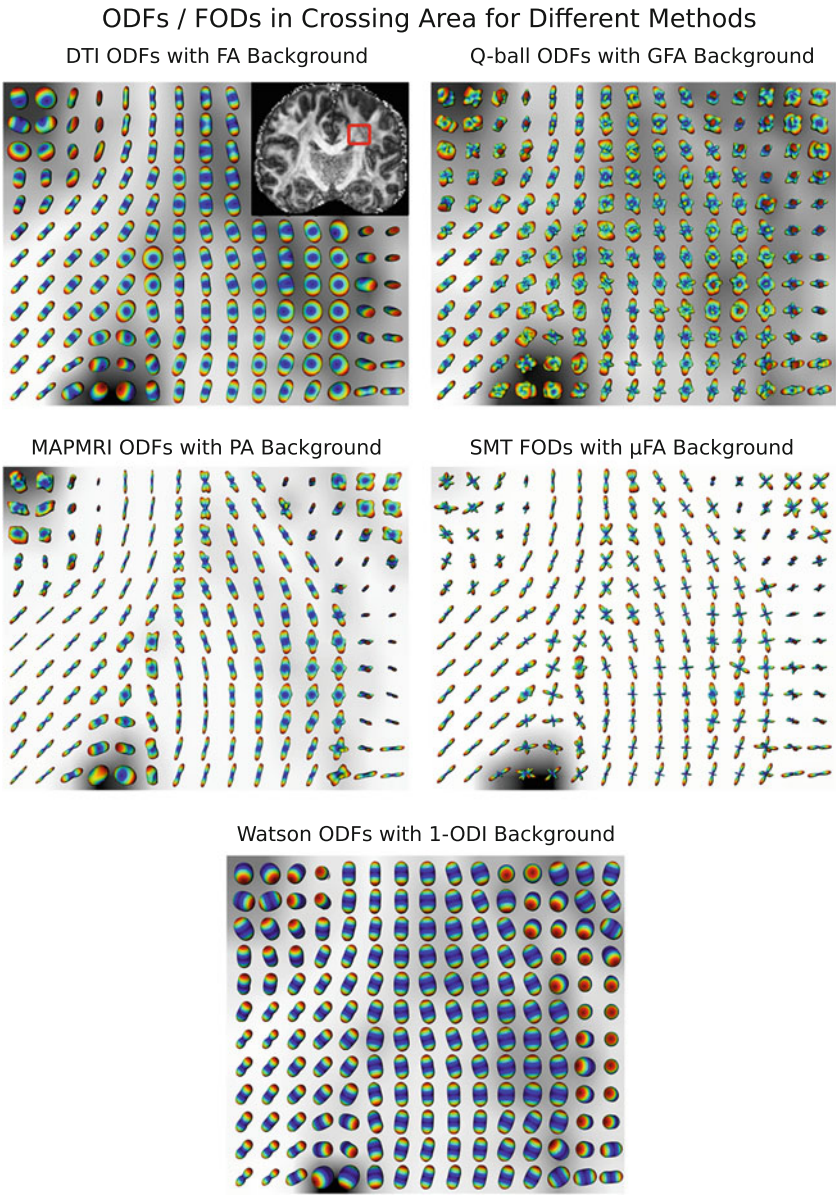


Fig. 7 Orientation Distribution Functions (ODFs) for DTI, Q-ball, MAPMRI, the Watson distributions of NODDI and Fiber Orientation Distributions (FODs) of SMT in an area where it is known there are crossing bundles. Each method has its corresponding anisotropy measure as background texture. It can be seen that DTI finds an average orientation, where Q-Ball, MAPMRI and SMT find crossing structures. The FODs, being the result of a deconvolution, show sharper peaks, and more consistent crossings than the ODF techniques. It is important to realize that while NODDI separates the signal contributions of intra- and extra-axonal diffusion, its dispersion index jointly describes the spread of both of these compartments, and produces very similar ODFs as DTI

4.2 Diffusion Anisotropy as a Signal Property

In this section, we discuss five different anisotropy measures known as Fraction Anisotropy (FA) [12], Relative Anisotropy (RA), Kurtosis Fractional Anisotropy (KFA) [39], Generalized Fraction Anisotropy (GFA) [68] and Propagator Anisotropy (PA) [55].

Signal-based models directly estimate the EAP $P(\mathbf{r}|\tau)$ from the measured signal attenuation $E(\mathbf{q}, \tau)$, using the Fourier relationship in Eq. (5). Notice that $P(\mathbf{r}|\tau)$ is a *conditional* probability density for diffusion time τ, as the Fourier transform is only over the $\mathbf{q}, \mathbf{r}$ space. We will interchangeably use real displacement vector $\mathbf{r} = r\mathbf{u}$ with its distance and direction $r \in \mathbb{R}^+, \mathbf{u} \in \mathbb{S}^2$ and q-space vector $\mathbf{q} = q\mathbf{g}$ with its q-space distance and gradient orientation $q \in \mathbb{R}^+, \mathbf{g} \in \mathbb{S}^2$. The following anisotropy measures are then defined as some difference or ratio between the isotropic and anisotropic parts of $P(\mathbf{r}|\tau)$.

Relative and Fractional Anisotropy Starting with one of the oldest measures for diffusion anisotropy, Relative Anisotropy (RA) and Fractional Anisotropy (FA) [12] are specific for the DTI model [14]. DTI solves the Fourier transform by generalizing the Stejskal-Tanner equation for unbounded media [63] to three dimensions:

$$E(b) = \exp(-b\mathbf{g}^{\mathrm{T}}\mathbf{D}\mathbf{g}) \quad \text{or} \quad E(\mathbf{q}, \tau) = \exp(-4\pi^2 q^2 \tau \mathbf{g}^{\mathrm{T}}\mathbf{D}\mathbf{g}) \tag{8}$$

with $\mathbf{D}$ a 3×3 symmetric positive-definite diffusion tensor. Notice that Eq. (8) is Gaussian over q and exponential over τ, which will be important in studying time-dependence in Sect. 5. FA describes fraction of the "magnitude" of $\mathbf{D}$ that we can ascribe to anisotropic diffusion in terms of its eigenvalues $\{\lambda_1, \lambda_2, \lambda_3\}$, whereas RA divides the magnitude of the anisotropic part by that of the isotropic part as

$$FA = \frac{\mathrm{std}(\lambda)}{\mathrm{rms}(\lambda)} = \sqrt{\frac{1}{2}} \frac{\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_3 - \lambda_1)^2}}{\sqrt{\lambda_1^2 + \lambda_2^2 + \lambda_3^2}} \tag{9}$$

$$RA = \frac{\mathrm{std}(\lambda)}{\mathrm{mean}(\lambda)} = \frac{3\sqrt{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_3 - \lambda_1)^2}}{\lambda_1 + \lambda_2 + \lambda_3}. \tag{10}$$

Both measures are zero when the medium is isotropic, but only FA is normalized between zero and one, which likely led to the prevalence of FA over RA in the community. The interpretation of FA has known limitations as DTI cannot represent crossing tissue configurations, but finds some average, Gaussian approximation that best fits the overall signal [14]. We illustrate this by estimating DTI's Orientation Distribution Function $ODF(\mathbf{u}|\tau)$, representing the probability density that a

diffusing particle will travel along direction $\mathbf{u}$ by marginalizing r as

$$ODF(\mathbf{u}|\tau) = \int_0^\infty P(r\mathbf{u}|\tau)r^2 dr \tag{11}$$

where r^2 is the Jacobian of the radial integration to ensure that the integral of the ODF is unity [2, 66]. Notice that Eq. (11) is general, and can be used for any method that estimates $P(r\mathbf{u}|\tau)$. For instance, Eq. (11) can be given analytically for DTI as

$$ODF_{\text{DTI}}(\mathbf{u}) = \frac{1}{4\pi|\mathbf{D}|^{\frac{1}{2}}(\mathbf{u}^{\text{T}}\mathbf{D}^{-1}\mathbf{u})^{\frac{3}{2}}}. \tag{12}$$

We show DTI's ODFs in a crossing area in Fig. 7, where it can be seen that round profiles with low FA are found where other methods detect crossings.

Kurtosis Fractional Anisotropy In Diffusion Kurtosis Imaging (DKI) [39] the non-Gaussian aspects of the signal attenuation are represented using the Taylor expansion. Using a summation instead a matrix product like in Eq. (8), DKI describes the signal attenuation as

$$E(b) = \exp(-b\sum_{i,j}^{3} g_i g_j D_{ij} + \frac{b^2}{6}\overline{D}^2 \sum_{i,j,k,l}^{3} g_i g_j g_k g_l W_{ijkl} + O(b^3)) \tag{13}$$

where $\overline{D}$ is the mean diffusivity with Diffusion Tensor $\mathbf{D}$ and non-Gaussian Kurtosis Tensor $\mathbf{W}$. In complete analogy to FA, the Kurtosis Fractional anisotropy is defined as

$$KFA = \frac{\text{std}(\mathbf{W})}{\text{rms}(\mathbf{W})}. \tag{14}$$

KFA therefore represents the anisotropy of the non-Gaussian aspects of the signal. DKI-based metrics have been shown to be more sensitive to pathology that DTI-based ones [21] and variations of its definition in terms of directional variation have been explored [35].

Generalized Fractional Anisotropy GFA [68] was proposed for High Angular Resolution Diffusion Imaging (HARDI) techniques [69], that use the Funk-Radon Transform (FRT) to estimate ODFs capable of describing multiple axon directions [70]. It is noteworthy that the only difference between DTI and HARDI is that in HARDI more gradients $\mathbf{g}$ are measured at a higher b-values to gain a better angular resolution of the ODF, see Fig. 4. This means that still no *radial* information is known of $P(r\mathbf{u}|\tau)$, and Gaussian decay over r is assumed to estimate the ODF in

Eq. (11). For any ODF, the GFA is given as

$$GFA = \frac{\text{std}(ODF)}{\text{rms}(ODF)} = \sqrt{\frac{n\sum_{i=1}^{n}(ODF(\mathbf{u}_i) - \langle ODF(\mathbf{u})\rangle)^2}{(n-1)\sum_{i=1}^{n} ODF(\mathbf{u}_i)^2}} \tag{15}$$

where $ODF(\mathbf{u})$ is the value of the ODF in direction $\mathbf{u}$, n is the number of evaluated ODF directions and $\langle ODF(\mathbf{u})\rangle$ is the mean ODF intensity. We show Q-ball Imaging ODFs [2, 26, 68] in the top-right of Fig. 7, where now crossing structures can be seen. Though, it must be said that FRT has been applied to a variety of HARDI methodologies, a selection of which is summarized in Tristan-Vega et al. [67]. Moreover, notice that GFA intensities in the crossing and single bundles areas are more similar than seen with DTI's FA in the top-left, but the overall intensities of GFA and FA are different. We illustrate this by comparing estimated anisotropy intensities in different ROIs, for different measures and techniques in Fig. 6. It can be seen that the same metric for different techniques, or different metrics for the same technique can result in different estimates of anisotropy.

Propagator Anisotropy PA was proposed for the multi-shell, Mean Apparent Propagator (MAP)-MRI technique [55]. MAP-MRI can be seen as a generalization of DTI, and allows for the estimation of three-dimensional $P(r\mathbf{u}|\tau)$, where now both restricted (non-Gaussian) diffusion over r and crossing axons can be represented. MAP-MRI is not unique in this respect, as a plethora of multi-shell techniques have been proposed for this purpose [See e.g. 10, 27, 37, 56], but MAP-MRI's formulation allows for easy estimation of a large variety of q-space properties using efficient regularization [31]. PA is defined as a measure of dissimilarity between the reconstructed $P(\mathbf{r}|\tau)$ and its closest isotropic approximation $P_{\text{iso}}(\mathbf{r}|\tau)$. First, the inner product between two EAPs is defined as

$$\langle P(\mathbf{r}|\tau)P_{\text{iso}}(\mathbf{r}|\tau)\rangle = \int_{\mathbb{R}^3} P(\mathbf{r}|\tau)P_{\text{iso}}(\mathbf{r}|\tau)d\mathbf{r}. \tag{16}$$

The similarity between two propagators is measured as an angular measure of covariance in analogy with the vector product [11]:

$$\cos\theta_{\text{PA}} = \sqrt{\frac{\langle P(\mathbf{r}|\tau)P_{\text{iso}}(\mathbf{r}|\tau)\rangle}{\langle P(\mathbf{r}|\tau)P(\mathbf{r}|\tau)\rangle\langle P_{\text{iso}}(\mathbf{r}|\tau)P_{\text{iso}}(\mathbf{r}|\tau)\rangle}}. \tag{17}$$

PA is then defined using the angular *dissimilarity* measure $\sin\theta_{\text{PA}} = \sqrt{1-\cos\theta_{\text{PA}}^2}$ and scaling function $\sigma(t,\epsilon)$ as

$$PA = \sigma(\sin\theta_{\text{PA}}, 0.4) \quad \text{with} \quad \sigma(t,\epsilon) = \frac{t^{3\epsilon}}{1 - 3t^{\epsilon} + 3t^{2\epsilon}}. \tag{18}$$

where we note that scaling parameter $\epsilon = 0.4$ was chosen by Özarslan et al. [55] to yield "the desired level of contrast in real images". PA can be estimated for any method that reconstructs $P(\mathbf{r}|\tau)$. For instance, for DTI it is given as

$$\cos\theta^2_{\text{PA-DTI}} = \frac{8u_0^3 u_x u_y u_z}{(u_x^2+u_0^2)(u_y^2+u_0^2)(u_z^2+u_0^2)} \tag{19}$$

where the displacement is given as a function of DTI's eigenvalues as$\{u_x, u_y, u_z\} = \sqrt{2\tau\{\lambda_1, \lambda_2, \lambda_3\}}$ [13], and u_0 is DTI's nearest isotropic propagator [55]. We show PA_{MAPMRI} and PA_{DTI} in Figs. 5 and 6, where this measure indeed seems to show good contrast between isotropic, crossing and single bundle tissues.

4.3 Anisotropy as Orientation Dispersion of Micro-Environments

As the only multi-compartment model that we consider in this chapter, the neurite orientation dispersion and density imaging (NODDI) model [73] parameterizes diffusion anisotropy as the dispersion of the diffusion signal of individual axon segments around a central bundle axis. In NODDI, axons as are represented as sticks—cylinders with zero radius and parallel diffusivity $\lambda_\parallel$—and are dispersed according to a Watson distribution $W(\kappa, \boldsymbol{\mu})$, where κ is the concentration parameter that is inversely related to axon dispersion, and $\boldsymbol{\mu} \in \mathbb{S}^2$ is the bundle direction. NODDI also separates the signal contribution of the Cerebrospinal Fluid (CSF) as an isotropic Gaussian with diffusivity D_{iso} (i.e. a Ball) and the hindered extra-axonal compartment—the diffusion directly around the axons—as an axially symmetric Tensor (i.e. a Zeppelin) with parallel and perpendicular diffusivity $\lambda_\parallel^{\text{ext}}$ and $\lambda_\perp^{\text{ext}}$. The overall signal representation is then

$$E^{\text{NODDI}}_{\text{Watson}} = \underbrace{f_{\text{CSF}}\overbrace{E_{\text{iso}}(D_{\text{CSF}})}^{\text{Ball}}}_{\text{CSF}} + \overbrace{W(\kappa,\boldsymbol{\mu})}^{\text{Watson}} *_{\mathbb{S}^2} \left[\underbrace{f_h\overbrace{E_{\text{h}}(\lambda_\perp^{\text{ext}}, \lambda_\parallel^{\text{ext}})}^{\text{Zeppelin}}}_{\text{Hindered Extra-Axonal}} + \underbrace{f_r\overbrace{E_r(\lambda_\parallel)}^{\text{Stick}}}_{\text{Intra-Axonal}} \right] \tag{20}$$

where the volume fractions of the CSF, hindered and intra-axonal compartment sum up to unity as $f_{\text{CSF}} + f_h + f_r = 1$, and $*_{\mathbb{S}^2}$ represents the spherical convolution that distributes the per-axon diffusion signal (both the stick and the hindered compartment) according to the Watson distribution [41]. In practice, NODDI fixes all diffusivities of the different model components and focuses on the estimation of $\kappa, \boldsymbol{\mu}$ and the volume fractions, leading to biases when the preset diffusivities don't reflect the true diffusivities. On the other hand, not fixing any diffusivities and simultaneously fitting all parameters leads to multi-modality in the parameters space—different tissue representations can produce the same diffusion signal [38].

NODDI reparameterizes the concentration parameter κ into the normalized Orientation Dispersion Index (ODI)

$$\text{ODI} = \frac{2}{\pi}\arctan(1/\kappa) \tag{21}$$

where ODI $= 0$ now means no dispersion, i.e. parallel axons and therefore high diffusion anisotropy, and ODI $= 1$ represents completely dispersed (isotropic) diffusion. We fitted the NODDI model using the NODDI toolbox[1] and illustrate the complement of ODI in Figs. 5 and 6. It can be seen that the contrast of previously discussed signal-based anisotropy measures and ODI is quite similar. Figure 7 also shows the ODFs of the estimated Watson distributed, which resemble the DTI ODFs in many cases.

4.4 Anisotropy as a Property of Micro-Environments

Similarly as NODDI, the Spherical Mean Technique (SMT) [42] also represents the diffusion signal as a distribution of individual axon segments with a spherical Fiber Orientation Distribution FOD($\mathbf{n}$). However, unlike NODDI, SMT does not do any separately model the hindered and CSF compartments, nor does it assume any parameterization of the FOD, and only assumes that the FOD is a probability density such that $\int_{\mathbb{S}^2} \text{FOD}(\mathbf{n}) = 1$ and the individual axon segment is represented by an axially symmetric tensor with perpendicular and parallel diffusivity $\lambda_\perp$ and $\lambda_\parallel$. It then follows that, for a given b-value, the spherical integral ϵ of the overall diffusion signal $E_b(\mathbf{n})$ and that of the individual axon segment $K_b(\mathbf{n})$ must be equal such that

$$\epsilon_E(b) = \int_{\mathbb{S}^2} E_b(\mathbf{n})d\mathbf{n} = \int_{\mathbb{S}^2} (\text{FOD} *_{\mathbb{S}^2} K_b)(\mathbf{n})d\mathbf{n} = \int_{\mathbb{S}^2} K_b(\mathbf{n})d\mathbf{n} = \epsilon_K(b, \lambda_\perp, \lambda_\parallel). \tag{22}$$

It is possible to solve this equation for $\lambda_\perp$ and $\lambda_\parallel$ using constrained least squares such that $0 < \lambda_\perp < \lambda_\parallel < \lambda_{\text{free}}$ with λ_{free} the free water diffusivity. Once $\lambda_\parallel, \lambda_\perp$ are known, the per-axon fractional anisotropy (μFA) is calculated as in Eq. (9), with $\lambda_1 = \lambda_\parallel$ and $\lambda_2 = \lambda_3 = \lambda_\perp$. As we show in Figs. 5 and 6, the estimation of μFA in the SMT framework is now independent of axon dispersion or crossing tissue configurations, that are very noticeable in for instance FA and ODI. However, SMT cannot distinguish between axon bundles, meaning that when two bundles with different diffusivities exist within one voxel the method can only estimate the average of the two. Lastly, the per-voxel FOD can now be obtained using standard

[1] http://www.nitrc.org/projects/noddi_toolbox

techniques such as Constrained Spherical Deconvolution [65], allowing for the recovery of very sharp orientation profiles as shown in Fig. 7.

5 Sensitivity to Diffusion Time

Recent work has put focus on the diffusion time dependence of the diffusion coefficient [32]. When this effect has an orientational dependence, it also directly affects the estimation of diffusion anisotropy. To illustrate this, we use Camino [24] to simulate the diffusion signal in two substrates consisting of parallel axons—modelled as cylinders—with gamma distributed radii. We show cross-sections of these substrates in Fig. 8. We simulate a multi-shell acquisition with two shells using b-values $\{1000, 3000\}$ s/mm^2 with 30 and 60 gradient directions, respectively, and one b_0 image without diffusion weighting. We set pulse length $\delta = 1$ ms and vary pulse separation Δ from 1 to 40 ms, while scaling the gradient strength to keep the b-values constant. While here we set δ constant to simplify the example, varying δ also influences the diffusion signal and its impact should not be ignored in practice [6].

We show the progress of FA, GFA and PA on the right of Fig. 8 for both distributions. It can be seen that all metrics describe the signal becoming more anisotropic as diffusion time increases, even though different metrics report different levels of anisotropy. In all cases, distribution 1, having more densely packed, smaller axons, produces a more anisotropic signal.

Fitting just DTI to the $b = 1000\,\mathrm{s/mm^2}$ data, we show the progress of the diffusivities parallel ($D_\parallel$) and perpendicular ($D_\perp$) to the cylinder axis on the left side of Fig. 9. It can be seen that time-dependence of diffusion anisotropy is caused by the time-dependence of $D_\perp$; lowering as diffusion time increases, indicating that diffusion is becoming more restricted. $D_\parallel$ remains constant and is equal for the two distributions, indicating free parallel diffusion in this simple substrate. It should be

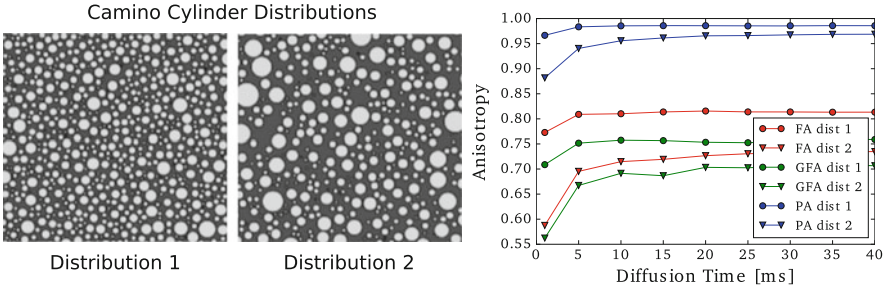


Fig. 8 *left*: Cross-sections of Camino cylinder substrates with gamma-distributed radii. Distribution 1 has smaller radii and less extra-cellular space than distribution 2. *right*: Diffusion time dependence of anisotropy measures from the two left distributions. The estimated anisotropy increases as longer diffusion times are used, until a plateau is reached

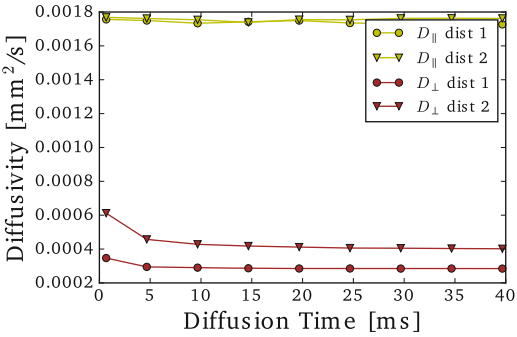


Fig. 9 The perpendicular and parallel diffusion coefficients over diffusion time for the distributions in Fig. 8

noted that Fieremans et al. [32] actually found $D_\parallel$ to be *more* time-dependent than $D_\perp$ in vivo, indicating that parallel diffusion is not completely free in real tissues.

5.1 Anisotropy Due to Axon Packing

It is known that in myelinated axons, nerve conduction velocity is directly proportional to axon diameter [71]. Furthermore, histology studies how that realistic axon diameters are distributed between [0.2–2] μm [1]. In this section, we continue to underline the importance of diffusion time dependence by discussing how it can be used to infer information on the axon diameter distribution and axon packing. The first model to exploit diffusion time dependence in this way was the composite hindered and restricted model of diffusion (CHARMED) model [8]. CHARMED models axons as impermeable, parallel cylinders with fixed diameter distribution, meaning only the intra-axonal volume fraction was estimated, but not the axon diameter distribution itself. CHARMED was later extended as AxCaliber [9] to actually estimate the gamma distribution of axon diameters. To do this, AxCaliber requires measurements exactly perpendicular to the axon direction, for different gradient strengths and diffusion times, and fits an intra-axonal and an extra-axonal tissue compartment

$$E(q_\perp, \tau) = \nu_\mathrm{r} E_\mathrm{r}(q_\perp, \tau|\alpha, \beta) + (1 - \nu_\mathrm{r}) E_\mathrm{h}(q_\perp, \tau) \tag{23}$$

where ν_r is the restricted water volume fraction, $E_\mathrm{r}(q_\perp, \tau|\alpha, \beta)$ is the perpendicular diffusion signal of cylinders [19] with Gamma distribution parameters α, β and $E_\mathrm{h}(q_\perp, \tau|D_\mathrm{h})$ is a Gaussian with diffusivity D_h as in Eq. (7). However, there are two reasons why this model has significant limitations:

- When modeling the time-dependence of the intra-axonal signal for physically feasible ranges, the signal is basically already restricted for the shortest diffusion times, see Fig. 10.

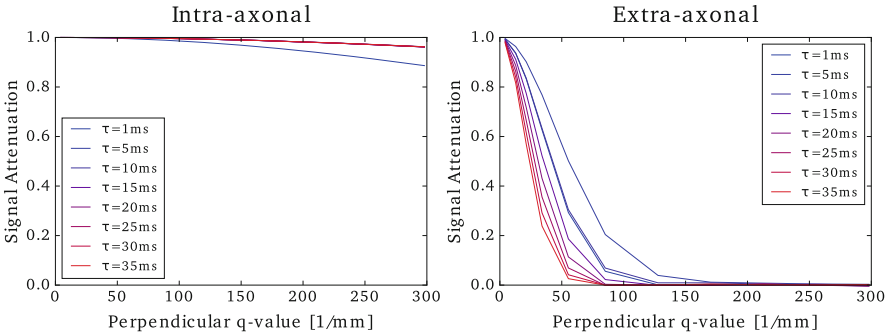


Fig. 10 The perpendicular intra-axonal and extra-axonal signal attenuation for distribution 1 of Fig. 8 for different diffusion times. It can be seen that the intra-axonal signal only shows sensitivity to diffusion time between 1 and 5 ms, which is barely possible to achieve in practice. On the other hand, the extra-axonal signal shows a lot of contrast over this dimension

- The non-negligible axon dispersion present in any bundle prevents the assumption that only the perpendicular direction is measured [48, 57].

This means that most signal variation over τ actually originates from restriction in the extra-axonal space. This behavior has recently been characterized in terms of *structural disorder* of the axon packing [17, 52]. In fact, this diffusion time dependence of the extra-axonal space has recently been implemented into the AxCaliber framework to improve axon diameter estimation [25]. In this work, E_h in Eq. (23) was replaced with

$$E_h(\mathbf{q}, \tau) = e^{-4\pi^2 \tau \|\mathbf{q}\|^2 \mathbf{g}^T \mathbf{D}_h \mathbf{g}} \tag{24}$$

where $\mathbf{D}_h$ is made *axon-packing dependent* as

$$\mathbf{D}_h = \begin{pmatrix} D_{h,\parallel} & 0 & 0 \\ 0 & D_{h,\infty} + A\frac{\ln(\Delta/\delta)+3/2}{\tau} & 0 \\ 0 & 0 & D_{h,\infty} + A\frac{\ln(\Delta/\delta)+3/2}{\tau} \end{pmatrix} \tag{25}$$

where $D_{h,\infty}$ is the bulk diffusion constant and A is a characteristic coefficient that scales approximately as the square of the correlation length, which in turn is proportional to the outer axonal diameter (which includes the myelin sheath) [17]. However, this augmentation does not correct for the dispersion, nor the already restricted intra-axonal diffusion at short diffusion times.

Further evidence of structural disorder was recently found in vivo by investigating stimulated echo diffusion tensor imaging for diffusion times up to 600 ms [32]. Remarkably, both parallel and perpendicular directions show non-Gaussian diffusion for diffusion times between 45 and 600 ms. This result comes after a long time of disagreement on whether or not there was time-dependence present in the

nervous tissue. Hopefully, this finding will cast new insights on the interpretation of DTI studies using clinical diffusion times $\tau > 20\,\text{ms}$. With this in mind, it is highly likely that new scalar indices will soon be proposed to describe the amount of structural disorder as a new type of tissue biomarker.

6 Discussion

In this chapter, we made a specific effort to review a diffusion anisotropy related measures, coming from either signal-based dMRI models that estimate the EAP from the signal as a whole, or models that use a multi-compartment approach to estimate axon packing or axon dispersion. We started by describing the relation between the tissue and the diffusion propagator in Sect. 2 and illustrated this relation schematically in Fig. 1. We then explained the details of the PGSE protocol in Sect. 3, where we provided the sequence explicitly in Fig. 2 and illustrated different acquisition schemes (DTI, HARDI, multi-shell and $q\tau$) in Fig. 4. We also clarified the effect of diffusion restriction on the observed diffusion signal attenuation in the case of cylinders in Fig. 3.

We then reviewed an extensive, but probably still not an exhaustive list of diffusion anisotropy measures in Sect. 4. We first provided the mathematical description of signal-based anisotropy measures such as the DTI-based FA and RA [12]; DKI-based KFA [39]; Q-ball-based GFA [68] and MAP-MRI-based PA [55]. We also described the NODDI-based ODI [73] in Sect. 4.3 and SMT-based μFA [42] in Sect. 4.4. We illustrated these metrics together qualitatively in Fig. 5; quantitatively in CSF, crossing and single bundle areas in Fig. 6; and illustrating the ODFs of these methods in Fig. 7. These comparisons were meant to illustrate the similarities in contrasts that different definitions of diffusion anisotropy provide, although their mathematical underpinnings may be different.

On Signal-Based Anisotropy Within the group of signal-based models, many studies have related changes in FA to a variety of pathologies, see e.g. the review by Assaf and Pasternak [7]. Also characterizing the non-Gaussian parts of the data, DKI-based anisotropy measures have shown to be more sensitive than DTI-based ones[21]. However, the literature shows that the more complex the estimation method and required acquisition scheme becomes, the fewer validation studies there are. For example, only a few studies have shown the potential of using GFA [23] or PA [29], while large-scale comparisons like those for FA are missing.

Moreover, the typical criterium for being a biomarker is that the measure of interest should provide a statistically significant difference between healthy and diseased populations. However, care should be taken in prematurely calling a non-specific marker such as diffusion anisotropy a biomarker. As an illustration, in the particular case of Parkinson's disease, after many studies had claimed that FA could be used as a diagnostic biomarker, a systematic review of these studies actually

showed that on its own, *it cannot* [36]. It is likely that the non-specificity of diffusion anisotropy will continue to confound its interpretation as a biomarker for pathology.

On Multi-Compartment-Based Anisotropy To overcome this lack of specificity, multi-compartment approaches strive to separate the signal contributions of different tissue compartments using biophysical models. However, it is important to realize that these models still describe diffusion anisotropy in some reparameterized way. For example, while NODDI is a multi-compartment model that separates the signal contributions of CSD, intra- and extra-axonal compartments, it can only describe one axon bundle using a Watson distribution with a single ODI, which is a function of the concentration parameter κ. Illustrating the Watson ODFs together with the signal-based ODFs in Fig. 7, we indeed find its similarity to others, in particular DTI. Of course, ODI has a different interpretation than FA, but it is important to see how they are related. Furthermore, μFA describes the per-axon micro-environment and is theoretically insensitive to crossing or dispersed axon configurations.

On Diffusion Time-Dependence Then, in Sect. 5 we analyze the time-dependence of diffusion anisotropy and its origin. We illustrate in Fig. 8 that anisotropy (in a simulated ensemble of cylinders) is a function of diffusion time—longer diffusion time results in more perpendicular restriction, which translates to a higher anisotropy. However, this experiment was still limited by its simplicity, as with this setup we cannot replicate long-distance diffusion time dependence illustrated in vivo by Fieremans et al. [32].

Then, in Sect. 5.1 we go into some detail on the origin of the time-dependence, which is the extra-axonal space. We first show in Fig. 10 that the intra-axonal signal is already restricted before 5 ms, whereas the typical minimum diffusion time in PGSE experiments is around 10 ms. This means that, as was initially shown by Novikov et al. [52], diffusing particles in the extra-axonal space, which were previously assumed to be Gaussian, are in fact still subject to some level of restriction due to axon packing. The differences in diffusion anisotropy over time shown in Fig. 8 must, therefore, be a result of the restriction in the extra-axonal space.

7 Conclusion

In this chapter, we have reviewed the inter-model variability of diffusion anisotropy estimation, both signal- and multi-compartment-based, as well as illustrated its sensitivity to especially short diffusion times. It is clear that there are many ways of defining diffusion anisotropy, depending on the chosen signal representation and acquisition scheme. Depending on the complexity of the devised metric, contrast differences can be observed both qualitative and quantitatively for different tissue types. Nonetheless, also great similarities can be appreciated between the different metrics.

Acknowledgements Data were provided [in part] by the Human Connectome Project, WU-Minn Consortium (Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54MH091657) funded by the 16 NIH Institutes and Centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University. This work was partly supported by ANR "MOSIFAH" under ANR-13-MONU-0009-01 and the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (ERC Advanced Grant agreement No 694665: CoBCoM). The author Marco Pizzolato expresses his thanks to Olea Medical and the Provence-Alpes-Côte d'Azur (PACA) Regional Council for providing grant and support. We thank Mauro Zucchelli for useful discussions.

References

1. Aboitiz, F., Scheibel, A.B., Fisher, R.S., Zaidel, E.: Fiber composition of the human corpus callosum. Brain Res. **598**(1), 143–153 (1992)
2. Aganj, I., Lenglet, C., Sapiro, G., Yacoub, E., Ugurbil, K., Harel, N.: Reconstruction of the orientation distribution function in single-and multiple-shell q-ball imaging within constant solid angle. Magn. Reson. Med. **64**(2), 554–566 (2010)
3. Alexander, D.C.: An introduction to computational diffusion MRI: the diffusion tensor and beyond. In: Visualization and Processing of Tensor Fields, pp. 83–106. Springer, Berlin (2006)
4. Alexander, D.C., Hubbard, P.L., Hall, M.G., Moore, E.A., Ptito, M., Parker, G.J.M., Dyrby, T.B.: Orientationally invariant indices of axon diameter and density from diffusion MRI. NeuroImage **52**(4), 1374–1389 (2010)
5. Andersson, J.L.R., Xu, J., Yacoub, E., Auerbach, E., Moeller, S., Ugurbil, K.: A comprehensive gaussian process framework for correcting distortions and movements in diffusion images. In: Proceedings of the 20th Annual Meeting of ISMRM, p. 2426 (2012)
6. Åslund, I., Topgaard, D.: Determination of the self-diffusion coefficient of intracellular water using PGSE NMR with variable gradient pulse length. J. Magn. Reson. **201**(2), 250–254 (2009)
7. Assaf, Y., Pasternak, O.: Diffusion tensor imaging (DTI)-based white matter mapping in brain research: a review. J. Mol. Neurosci. **34**(1), 51–61 (2008)
8. Assaf, Y., Freidlin, R.Z., Rohde, G.K., Basser, P.J.: New modeling and experimental framework to characterize hindered and restricted water diffusion in brain white matter. Magn. Reson. Med. **52**(5), 965–978 (2004)
9. Assaf, Y., Blumenfeld-Katzir, T., Yovel, Y., Basser, P.J.: AxCaliber: a method for measuring axon diameter distribution from diffusion MRI. Magn. Reson. Med. **59**(6), 1347–1354 (2008)
10. Assemlal, H.-E., Tschumperlé, D., Brun, L.: Efficient and robust computation of PDF features from diffusion MR signal. Med. Image Anal. **13**(5), 715–729 (2009)
11. Avram, A.V., Sarlls, J.E., Barnett, A.S., Özarslan, E., Thomas, C., Irfanoglu, M.O., Hutchinson, E., Pierpaoli, C., Basser, P.J.: Clinical feasibility of using mean apparent propagator (MAP) MRI to characterize brain tissue microstructure. NeuroImage **127**, 422–434 (2016)
12. Basser, P.J.: Inferring microstructural features and the physiological state of tissues from diffusion-weighted images. NMR Biomed. **8**(7), 333–344 (1995)
13. Basser, P.J.: Relationships between diffusion tensor and q-space MRI. Magn. Reson. Med. **47**(2), 392–397 (2002)
14. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophys. J. **66**(1), 259 (1994)
15. Beaulieu, C.: The basis of anisotropic water diffusion in the nervous system – a technical review. NMR Biomed. **15**(7–8), 435–455 (2002)
16. Behrens, T.E.J., Woolrich, M.W., Jenkinson, M., Johansen-Berg, H., Nunes, R.G., Clare, S., Matthews, P.M., Brady, J.M., Smith, S.M.: Characterization and propagation of uncertainty in diffusion-weighted MR imaging. Magn. Reson. Med. **50**(5), 1077–1088 (2003)

17. Burcaw, L.M., Fieremans, E., Novikov, D.S.: Mesoscopic structure of neuronal tracts from time-dependent diffusion. NeuroImage **114**, 18–37 (2015)
18. Callaghan, P.T.: Principles of Nuclear Magnetic Resonance Microscopy, vol. 3. Clarendon Press, Oxford (1991)
19. Callaghan, P.T.: Pulsed-gradient spin-echo NMR for planar, cylindrical, and spherical pores under conditions of wall relaxation. J. Magn. Reson. Ser. A **113**(1), 53–59 (1995)
20. Caruyer, E., Lenglet, C., Sapiro, G., Deriche, R.: Design of multishell sampling schemes with uniform coverage in diffusion MRI. Magn. Reson. Med. **69**(6), 1534–1540 (2013)
21. Cheung, M.M., Hui, E.S., Chan, K.C., Helpern, J.A., Qi, L., Wu, E.X.: Does diffusion kurtosis imaging lead to better neural tissue characterization? A rodent brain maturation study. NeuroImage **45**(2), 386–392 (2009)
22. Cohen, Y., Assaf, Y.: High b-value q-space analyzed diffusion-weighted MRS and MRI in neuronal tissues – a technical review. NMR Biomed. **15**(7–8), 516–542 (2002)
23. Cohen-Adad, J., El Mendili, M.M., Lehéricy, S., Pradat, P.F., Blancho, S., Rossignol, S., Benali, H.: Demyelination and degeneration in the injured human spinal cord detected with diffusion and magnetization transfer MRI. NeuroImage **55**(3), 1024–1033 (2011)
24. Cook, P.A., Bai, Y., Nedjati-Gilani, S.K.K.S., Seunarine, K.K., Hall, M.G., Parker, G.J., Alexander, D.C.: Camino: open-source diffusion-MRI reconstruction and processing. In: 14th Scientific Meeting of the International Society for Magnetic Resonance in Medicine, vol. 2759. Seattle, WA (2006)
25. De Santis, S., Jones, D.K., Roebroeck, A.: Including diffusion time dependence in the extra-axonal space improves in vivo estimates of axonal diameter and density in human white matter. NeuroImage **130**, 91–103 (2016)
26. Descoteaux, M., Angelino, E., Fitzgibbons, S., Deriche, R.: Regularized, fast, and robust analytical q-ball imaging. Magn. Reson. Med. **58**(3), 497–510 (2007)
27. Descoteaux, M., Deriche, R., Le Bihan, D., Mangin, J.-F., Poupon, C.: Multiple q-shell diffusion propagator imaging. Med. Image Anal. **15**(4), 603–621 (2011)
28. Einstein, A.: Investigations on the Theory of the Brownian Movement. Courier Corporation (1956)
29. Fick, R., Daianu, M., Pizzolato, M., Wassermann, D., Jacobs, R.E., Thompson, P.M., Town, T., Deriche, R.: Comparison of biomarkers in transgenic alzheimer rats using multi-shell diffusion MRI. In: Computational Diffusion MRI. Springer, Berlin (2016)
30. Fick, R., Petiet, A., Santin, M., Philippe, A.-C., Lehericy, S., Deriche, R., Wassermann, D.: Multi-spherical diffusion MRI: exploring diffusion time using signal sparsity. In: Computational Diffusion MRI. Springer, Berlin (2016)
31. Fick, R.H.J., Wassermann, D., Caruyer, E., Deriche, R.: MAPL: tissue microstructure estimation using Laplacian-regularized MAP-MRI and its application to HCP data. NeuroImage **134**, 365–385 (2016)
32. Fieremans, E., Burcaw, L.M., Lee, H.-H., Lemberskiy, G., Veraart, J., Novikov, D.S.: In vivo observation and biophysical interpretation of time-dependent diffusion in human white matter. NeuroImage **129**, 414–427 (2016)
33. Greve, D.N., Fischl, B.: Accurate and robust brain image alignment using boundary-based registration. NeuroImage **48**(1), 63–72 (2009)
34. Hahn, E.L.: Spin echoes. Phys. Rev. **80**(4), 580 (1950)
35. Hansen, B., Jespersen, S.N.: Kurtosis fractional anisotropy, its contrast and estimation by proxy. Sci. Rep. **6** (2016)
36. Hirata, F.C.C., Sato, J.R., Vieira, G., Lucato, L.T., Leite, C.C., Bor-Seng-Shu, E., Pastorello, B.F., Otaduy, M.C.G., Chaim, K.T., Campanholo, K.R., et al.: Substantia nigra fractional anisotropy is not a diagnostic biomarker of parkinson's disease: a diagnostic performance study and meta-analysis. Eur. Radiol. **27**(6), 2640–2648 (2017)
37. Hosseinbor, A.P., Chung, M.K., Wu, Y.-C., Alexander, A.L.: Bessel Fourier orientation reconstruction (BFOR): an analytical diffusion propagator reconstruction for hybrid diffusion imaging and computation of q-space indices. NeuroImage **64**, 650–670 (2013)

38. Jelescu, I.O., Veraart, J., Fieremans, E., Novikov, D.S.: Degeneracy in model parameter estimation for multi-compartmental diffusion in neuronal tissue. NMR Biomed. **29**(1), 33–47 (2016)
39. Jensen, J.H., Helpern, J.A., Ramani, A., Lu, H., Kaczynski, K.: Diffusional kurtosis imaging: the quantification of non-Gaussian water diffusion by means of magnetic resonance imaging. Magn. Reson. Med. **53**(6), 1432–1440 (2005)
40. Kac, M.: Can one hear the shape of a drum? Am. Math. Mon. **73**(4), 1–23 (1966)
41. Kaden, E., Knösche, T.R., Anwander, A.: Parametric spherical deconvolution: inferring anatomical connectivity using diffusion MR imaging. NeuroImage **37**(2), 474–488 (2007)
42. Kaden, E., Kruggel, F., Alexander, D.C.: Quantitative mapping of the per-axon diffusion coefficients in brain white matter. Magn. Reson. Med. **75**(4), 1752–1763 (2016)
43. Kaden, E., Kelm, N.D., Carson, R.P., Does, M.D., Alexander, D.C.: Multi-compartment microscopic diffusion imaging. NeuroImage **139**, 346–359 (2016)
44. Kärger, J., Heink, W.: The propagator representation of molecular transport in microporous crystallites. J. Magn. Reson. (1969) **51**(1), 1–7 (1983)
45. Keil, B., Blau, J.N., Biber, S., Hoecht, P., Tountcheva, V., Setsompop, K., Triantafyllou, C., Wald, L.L.: A 64-channel 3t array coil for accelerated brain MRI. Magn. Reson. Med. **70**(1), 248–258 (2013)
46. Le Bihan, D., Breton, E.: Imagerie de diffusion in-vivo par résonance magnétique nucléaire. C. R. Acad. Sci. **93**(5), 27–34 (1985)
47. Le Bihan, D., Mangin, J.-F., Poupon, C., Clark, C.A., Pappata, S., Molko, N., Chabriat, H.: Diffusion tensor imaging: concepts and applications. J. Magn. Reson. Imaging **13**(4), 534–546 (2001)
48. Leergaard, T.B., White, N.S., De Crespigny, A., Bolstad, I., D'Arceuil, H., Bjaalie, J.G., Dale, A.M.: Quantitative histological validation of diffusion MRI fiber orientation distributions in the rat brain. PLoS One **5**(1), e8595 (2010)
49. Liewald, D., Miller, R., Logothetis, N., Wagner, H.-J., Schüz, A.: Distribution of axon diameters in cortical white matter: an electron-microscopic study on three human brains and a macaque. Biol. Cybern. **108**(5), 541–557 (2014)
50. Merboldt, K.-D., Hanicke, W., Frahm, J.: Self-diffusion NMR imaging using stimulated echoes. J. Magn. Reson. (1969) **64**(3), 479–486 (1985)
51. Moseley, M.E., Cohen, Y., Kucharczyk, J., Mintorovitch, J., Asgari, H.S., Wendland, M.F., Tsuruda, J., Norman, D.: Diffusion-weighted MR imaging of anisotropic water diffusion in cat central nervous system. Radiology **176**(2), 439–445 (1990)
52. Novikov, D.S., Jensen, J.H., Helpern, J.A., Fieremans, E.: Revealing mesoscopic structural universality with diffusion. Proc. Natl. Acad. Sci. **111**(14), 5088–5093 (2014)
53. Özarslan, E., Basser, P.J., Shepherd, T.M., Thelwall, P.E., Vemuri, B.C., Blackband, S.J.: Observation of anomalous diffusion in excised tissue by characterizing the diffusion-time dependence of the MR signal. J. Magn. Reson. **183**(2), 315–323 (2006)
54. Özarslan, E., Shepherd, T.M., Koay, C.G., Blackband, S.J., Basser, P.J.: Temporal scaling characteristics of diffusion as a new MRI contrast: findings in rat hippocampus. NeuroImage **60**(2), 1380–1393 (2012)
55. Özarslan, E., Koay, C.G., Shepherd, T.M., Komlosh, M.E., İrfanoğlu, M.O., Pierpaoli, C., Basser, P.J.: Mean apparent propagator (MAP) MRI: a novel diffusion imaging method for mapping tissue microstructure. NeuroImage **78**, 16–32 (2013)
56. Rathi, Y., Michailovich, O., Laun, F., Setsompop, K., Grant, P.E., Westin, C.-F.: Multi-shell diffusion signal recovery from sparse measurements. Med. Image Anal. **18**(7), 1143–1156 (2014)
57. Ronen, I., Budde, M., Ercan, E., Annese, J., Techawiboonwong, A., Webb, A.: Microstructural organization of axons in the human corpus callosum quantified by diffusion-weighted magnetic resonance spectroscopy of n-acetylaspartate and post-mortem histology. Brain Struct. Funct. **219**(5), 1773–1785 (2014)

58. Setsompop, K., Kimmlingen, R., Eberlein, E., Witzel, T., Cohen-Adad, J., McNab, J.A., Keil, B., Tisdall, M.D., Hoecht, P., Dietz, P., et al.: Pushing the limits of in vivo diffusion MRI for the human connectome project. NeuroImage **80**, 220–233 (2013)
59. Soares, J., Marques, P., Alves, V., Sousa, N.: A hitchhiker's guide to diffusion tensor imaging. Front. Neurosci. **7**, 31 (2013)
60. Stejskal, E.O., Tanner, J.E.: Spin diffusion measurements: spin echoes in the presence of a time-dependent field gradient. J. Chem. Phys. **42**(1), 288–292 (1965)
61. Strimbu, K., Tavel, J.A.: What are biomarkers? Curr. Opin. HIV AIDS **5**(6), 463 (2010)
62. Tanner, J.E.: Transient diffusion in a system partitioned by permeable barriers. application to NMR measurements with a pulsed field gradient. J. Chem. Phys. **69**(4), 1748–1754 (1978)
63. Tanner, J.E., Stejskal, E.O.: Restricted self-diffusion of protons in colloidal systems by the pulsed-gradient, spin-echo method. J. Chem. Phys. **49**(4), 1768–1777 (1968)
64. Taylor, D.G., Bushell, M.C.: The spatial mapping of translational diffusion coefficients by the NMR imaging technique. Phys. Med. Biol. **30**(4), 345 (1985)
65. Tournier, J.-D., Calamante, F., Connelly, A.: Robust determination of the fibre orientation distribution in diffusion MRI: non-negativity constrained super-resolved spherical deconvolution. NeuroImage **35**(4), 1459–1472 (2007)
66. Tristán-Vega, A., Westin, C.-F., Aja-Fernández, S.: Estimation of fiber orientation probability density functions in high angular resolution diffusion imaging. NeuroImage **47**(2), 638–650 (2009)
67. Tristan-Vega, A., Westin, C.-F., Aja-Fernandez, S.: A new methodology for the estimation of fiber populations in the white matter of the brain with the funk–radon transform. NeuroImage **49**(2), 1301–1315 (2010)
68. Tuch, D.S.: Q-ball imaging. Magn. Reson. Med. **52**(6), 1358–1372 (2004)
69. Tuch, D.S., Reese, T.G., Wiegell, M.R., Makris, N., Belliveau, J.W., Wedeen, V.J.: High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. Magn. Reson. Med. **48**(4), 577–582 (2002)
70. Tuch, D.S., Reese, T.G., Wiegell, M.R., Wedeen, V.J.: Diffusion MRI of complex neural architecture. Neuron **40**(5), 885–895 (2003)
71. Waxman, S.G.: Determinants of conduction velocity in myelinated nerve fibers. Muscle Nerve **3**(2), 141–150 (1980)
72. Wedeen, V.J., Hagmann, P., Tseng, W.-Y.I., Reese, T.G., Weisskoff, R.M.: Mapping complex tissue architecture with diffusion spectrum magnetic resonance imaging. Magn. Reson. Med. **54**(6), 1377–1386 (2005)
73. Zhang, H., Schneider, T., Wheeler-Kingshott, C.A., Alexander, D.C.: NODDI: practical in vivo neurite orientation dispersion and density imaging of the human brain. NeuroImage **61**(4), 1000–1016 (2012)

Measuring Microscopic Anisotropy with Diffusion Magnetic Resonance: From Material Science to Biomedical Imaging

Andrada Ianuş, Noam Shemesh, Daniel C. Alexander, and Ivana Drobnjak

Abstract Diffusion magnetic resonance provides a non-invasive probe of material structure at the micro-scale in porous media including emulsions, rocks, catalysts and biological tissue. The quantification of microscopic anisotropy aims to reflect the size and shape of individual pores, separating the effect of their orientation distribution in the imaging voxel, which is of great importance in many applications.

The single diffusion encoding (SDE) sequence, which consists of a pair of diffusion gradients applied before and after the refocusing pulse in a spin-echo preparation, is the standard pulse sequence for acquiring diffusion MRI data. SDE sequences, which have one gradient orientation per measurement, have been used in various studies to estimate microscopic anisotropy, mainly assuming that the underlying substrate consists of identical pores. In order to discriminate between more complex systems, which may include pores of various sizes and shapes, more sophisticated techniques which use diffusion gradients with varying orientation within one measurement, such as double diffusion encoding, isotropic encoding or q-space trajectory imaging, have been proposed in the literature. In addition to the these techniques which aim to estimate microscopic anisotropy, a different approach to characterize pore shape directly is to take the inverse Fourier transform of the reciprocal pore shape function which can be measured with diffusion gradients that are highly asymmetric.

This work provides a review of various diffusion magnetic resonance techniques which have been proposed in the literature to measure the microscopic shape of pores, both in material science as well as in biomedical imaging.

A. Ianuş (✉)
Department of Computer Science, University College London, Gower Street, WC1E 6BT London, UK

Champalimaud Neuroscience Programme, Champalimaud Centre for the Unknown, Lisbon, Portugal
e-mail: a.ianus@ucl.ac.uk

N. Shemesh
Champalimaud Neuroscience Programme, Champalimaud Centre for the Unknown, Lisbon, Portugal

D.C. Alexander • I. Drobnjak
Department of Computer Science, University College London, London, UK

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_10

1 Introduction

A non-invasive measurement of pore size and shape is of great interest both in material science, as well as in biomedical applications. Microscopic anisotropy (μA) aims to characterize the elongation of individual pores, independently of their organization at the macroscopic level, and its quantification has many different applications. For example, in material science, μA can be used to distinguish between different phases of liquid crystals [4, 8] or to investigate the structure of biological porous media such as starch, cellulose, yeast cells, etc. [9, 11, 60]. More recent developments in biomedical imaging have highlighted the importance of μA as a potential biomarker for white matter tracts integrity [36, 38] as well as for mapping differences in tumour microstructure [58, 59].

Diffusion magnetic resonance (dMR) is a modality of choice for assessing pore morphology at the microscopic scale, non-invasively. The diffusion weighted signal is sensitive to the displacement of probe molecules within the substrate, and can be used to extract information regarding the porous structure, such as μA.

The standard pulse sequences for acquiring dMR data is the single diffusion encoding (SDE) sequence [57], which applies a pair of magnetic field gradients to sensitise the signal to the molecular displacement in the direction of the gradient. When the pores are coherently aligned, the diffusion measurements at macroscopic scale reflect the microscopic properties of individual pores. However, this is not the case in other situations when the presence of different pore orientations has a significant effect on the measured signal. One way to eliminate the influence of orientation distribution is to calculate the powder average of the signal, i.e. the average over measurements with different gradient directions uniformly distributed on a sphere. Assuming the material consists of identical pores which differ only in orientation, microscopic anisotropy can be estimated from powder averaged signal acquired at several diffusion weightings [9, 23, 60]. However, when this assumption is not valid, SDE measurements fail to distinguish substrates featuring microscopic anisotropy and/or a distribution of pore sizes [21, 60]. To overcome this limitation, different techniques which use sequences with varying gradient orientation within one measurement have been proposed in the literature.

One approach is to use double diffusion encoding (DDE) sequences [11, 39], which concatenate two independent pairs of diffusion gradients, separated by a mixing time. Thus measurements which vary the angle between the two gradients probe the correlation of molecular displacements in different directions and provide sensitivity to microscopic anisotropy even in heterogeneous substrates with a distribution of pore sizes [48]. Various approaches have been proposed to estimate μA in macroscopically isotropic substrates [6, 11, 48], however if the substrates are not macroscopically isotropic, the values depend on the particular choice of gradient directions. To overcome this issue, recent studies have introduced rotationally invariant acquisitions which allow for a consistent estimation of μA in macroscopically anisotropic materials [22, 37].

Another approach that has been recently proposed for quantifying μA is to combine measurements which provide isotropic and linear encoding at several diffusion weightings [32]. These concepts have been generalized in the q-space trajectory imaging approach [61, 62] in order to disentangle variations in size from variations in shape and orientation of the underlying structures.

This work aims to discuss the mathematical concepts behind the various approaches for estimating microscopic anisotropy, the advantages and limitations of the different techniques, as well as possible directions for future research. The chapter is organized in two parts: in the first part we discuss approaches which assume that diffusion in the microscopic pores can be approximated by a diffusion tensor, while in the second part we discuss the techniques which explicitly assume that the pores exhibit restricted diffusion and model spin displacements instead.

2 Microscopic Anisotropy in the Gaussian Regime

The first part presents various techniques that have been proposed in the diffusion MRI literature to estimate microscopic anisotropy, assuming that diffusion in individual pores can be described by a diffusion tensor. In this case, the pore size and shape is reflected by the diffusivity values, i.e. the eigenvalues of the diffusion tensor, and the effect of the boundaries is not explicitly modelled. We present different approaches, with their benefits and limitations, in the order of their acquisition complexity, from methods which use single diffusion encoding, to double diffusion encoding and generalized gradient waveforms.

2.1 Single Diffusion Encoding

This section presents various approaches which aim to estimate microscopic anisotropy from diffusion measurements which were acquired using standard SDE sequences. One common assumption of these techniques is that the underlying substrate consists of identical pores which vary only in orientation.

The SDE sequence, illustrated in Fig. 1a, consists of a pair of magnetic field gradients which sensitise the signal to the displacement of the probe molecules in one dimension defined by the direction of the applied gradient. When molecules diffuse in an unbounded, isotropic environment, i.e. free diffusion, their displacements have a Gaussian distribution with zero mean and a time dependent variance $2Dt$ where D is the diffusion coefficient and t is the time. In this case the signal attenuation from a dMR measurement is given by:

$$E(b) = \exp(-bD), \tag{1}$$

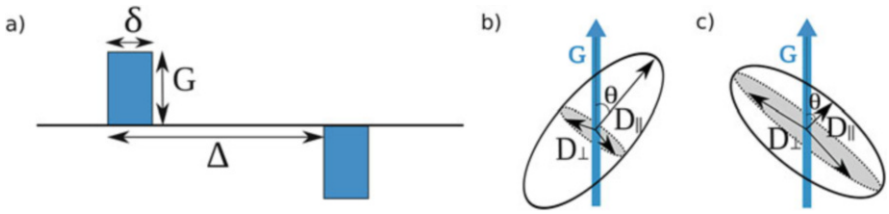


Fig. 1 Schematic representation of (**a**) SDE diffusion sequence, (**b**) microdomain with $D_{\parallel} > D_{\perp}$ (prolate tensor) and (**c**) microdomain with $D_{\parallel} < D_{\perp}$ (oblate tensor)

where b is the diffusion weighting factor, which for an SDE sequence has the following expression [57]:

$$b = (\gamma G\delta)^2(\Delta - \delta/3), \tag{2}$$

where γ is the gyromagnetic ratio of the nuclear spin and G, δ and Δ are the gradient strength, duration and separation, respectively.

When diffusion is no longer isotropic and has a preferential direction, the measured signal depends on the direction of the measurement. A common way of modelling anisotropic microdomains is to assume they are cylindrically symmetric and exhibit Gaussian diffusion with diffusivity $D_{\parallel}$ in the direction parallel to the cylinder axis and $D_{\perp}$ in the transverse plane, as illustrated in Fig. 1b, c for the cases $D_{\parallel} > D_{\perp}$ and $D_{\parallel} < D_{\perp}$, respectively. In this case the apparent diffusion coefficient along the direction of the diffusion gradient $\hat{\mathbf{g}}$ depends on the angle θ between this direction and the main axis of the microdomain [9]:

$$D_{\theta} = D_{\parallel}\cos^2\theta + D_{\perp}\sin^2\theta. \tag{3}$$

Thus, the signal attenuation is:

$$\begin{aligned} E_{\hat{\mathbf{g}}}(b) &= \exp(-bD_{\theta}) = \exp\left(-b(D_{\parallel}\cos^2\theta + D_{\perp}\sin^2\theta)\right) \\ &= \exp(-bD_{\perp})\exp\left(-b\cos^2\theta(D_{\parallel} - D_{\perp})\right). \end{aligned} \tag{4}$$

The signal measured along the gradient direction $\hat{\mathbf{g}}$ for a macroscopic sample with many microdomains which vary only in orientation can be obtained by integrating Eq. (4) over the orientation distribution [60]:

$$\bar{E}_{\hat{\mathbf{g}}}(b) = \exp(-bD_{\perp})\int_0^{\pi} P_{\hat{\mathbf{g}}}(\theta)\exp\left(-b\cos^2\theta(D_{\parallel} - D_{\perp})\right)d\theta, \tag{5}$$

where $P_{\hat{\mathbf{g}}}(\theta)$ is the probability distribution of θ.

For a powder average, i.e. when the microdomains are isotropically orientated, the probability distribution $P_{\hat{\mathbf{g}}}(\theta) = 1/2 \sin\theta$ and no longer depends on the gradient direction.

$$\bar{E}(b) = \exp(-bD_\perp) \int_0^\pi \exp\left(-b\cos^2\theta(D_\parallel - D_\perp)\right) \frac{1}{2} \sin\theta d\theta. \tag{6}$$

This approach has been used to study the structure of different materials. For instance, water diffusion in each microdomain of a matrix of long polymer chains can be approximated as one dimensional ($D_\parallel > 0, D_\perp = 0$), while the diffusion within lamellar structures can be approximated as two dimensional ($D_\parallel = 0, D_\perp > 0$). Based on the different signal attenuation in the two cases, [9] has shown that the endosperm tissue from wheat grains can be described as an ensemble of randomly oriented thin capillaries, while diffusion in the Lamellar Phase of Aerosol OT/Water system is two dimensional [8].

Although quantifying the shape of microdomains using the powder averaged signal has been studied for decades in material science, only recently, a similar idea has been used for biomedical imaging, where the tissue structure in many cases is not macroscopically isotropic. A recent study has shown that measuring the SDE signal along many gradient directions which are isotropically oriented on a sphere and taking their average (i.e. mean spherical signal) is equivalent to the powder average [23]. The mean spherical signal has been derived in [2] and reads as:

$$\bar{E}(b) = \exp(-bD_\perp) \frac{\sqrt{\pi}\,\mathrm{erf}\left(\sqrt{b(D_\parallel - D_\perp)}\right)}{2\sqrt{b(D_\parallel - D_\perp)}}, \tag{7}$$

where erf is the error function.

Equation (7) has only two unknowns, which can be estimated when the mean spherical signal is measured with at least two different b values. Furthermore, constraining $D_\parallel > D_\perp$, which is more biologically plausible, improves the stability of the fitting. After estimating $D_\parallel$ and $D_\perp$, the microscopic fractional anisotropy (μFA), which is a normalized quantity with respect to the size of the diffusion tensor of individual microdomains, is defined as [3, 52]:

$$\mu FA = \sqrt{\frac{3}{2} \frac{(D_\parallel - \bar{D})^2 + 2(D_\perp - \bar{D})^2}{D_\parallel^2 + 2D_\perp^2}}, \tag{8}$$

where $\bar{D} = (D_\parallel + 2D_\perp)/3$ is the mean diffusivity of the microdomain. This approach is known as the spherical mean technique (SMT) and aims to estimate the per-axon diffusivity in the brain using SDE measurements [23]. The main advantage of this approach is that it can be applied retrospectively to already acquired data sets which has at least two different diffusion weighting shells.

Probability Distribution of Diffusion Coefficients

The main limitation of these approaches is the assumption that all microdomains are identical. The effect of this assumption can be better understood in an alternative approach of computing the signal attenuation. The average signal computed from a probability distribution of orientations $P_{\hat{\mathbf{g}}}(\theta)$ in (5) can be reinterpreted as the signal given by a probability distribution of diffusion coefficients $P_{\hat{\mathbf{g}}}(D)$:

$$P_{\hat{\mathbf{g}}}(D) = \int_0^{\pi} P_{\hat{\mathbf{g}}}(\theta)\delta(D - D_\theta)d\theta, \tag{9}$$

where $\delta(D - D_\theta)$ is the Dirac delta function. Then, the signal attenuation can be written as the Laplace transform of $P_{\hat{\mathbf{g}}}(D)$

$$\bar{E}_{\hat{\mathbf{g}}}(b) = \int_0^{\infty} P_{\hat{\mathbf{g}}}(D)\exp(-bD)dD. \tag{10}$$

Although in a macroscopically anisotropic system $P_{\hat{\mathbf{g}}}(D)$ depends on the gradient direction, when the signal is averaged over isotropically oriented gradient directions (powder average), $P(D)$ is independent on orientation. Given the measured data, the probability distribution $P(D)$ can be calculated using an inverse Laplace transform, however, it is commonly recognized that such an inversion of experimental data is very sensitive to noise and depends on the choice of algorithm [60].

Analysing this approach from the perspective of $P(D)$ highlights the main issue of estimating microscopic anisotropy based on SDE measurements, namely that the same probability distribution can be obtained from an infinite number of microdomain ensambles. For instance, this situation is presented in Fig. 2, which

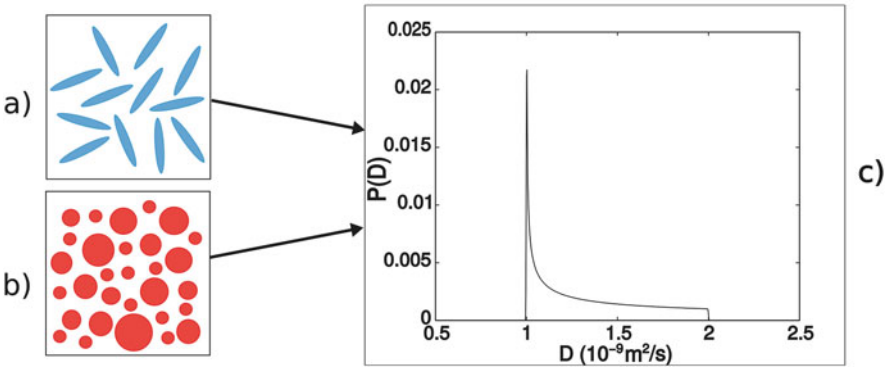


Fig. 2 Two very different substrates with either (**a**) randomly oriented identical microdomains ($D_\parallel = 2 \cdot 10^{-9}\,\mathrm{m}^2/2$, $D_\perp = 10^{-9}\mathrm{m}^2/2$) or (**b**) with a distribution of diffusion coefficients (e.g. produced by a distribution of pore sizes in the Gaussian regime or a mixture of molecular weights such mono-, oligo- or polymers), can yield the same $P(D)$ depicted in (**c**)

illustrates the probability distribution obtained from either an ensemble of randomly oriented anisotropic microdomains or an ensemble of microdomains with a range of diffusion coefficients, which can be the result of a distribution of pore sizes in the Gaussian regime or a mixture of molecular weights. Moreover, various distributions $P(D)$ can yield identical signal attenuation curves [60].

Nevertheless, if there is prior information regarding which situation is more plausible, SDE measurements could be used to quantify microscopic anisotropy, which is also the case when explicitly accounting for restricted diffusion inside the pores [21]. If there is no prior information regarding the microstructure of the material, one possible solution is to combine different modalities which provide complementary information. For instance, to differentiate the lamellar structures of lyotropic liquid crystals, [4] have combined SDE diffusion data with MR measurements of ^2H spectra. Another possibility is to go beyond single diffusion encoding and use diffusion sequences which vary the gradient orientation within one measurement, which will be discussed in the following sections.

2.2 *Double Diffusion Encoding*

One way to disentangle microscopic anisotropy from a distribution of diffusivities is to sensitize the signal to diffusion in different directions in one measurement. This can be achieved using double diffusion encoding (DDE) [12], and in general multiple diffusion encoding (MDE) [39], by concatenating gradient pulses which have different orientations. The DDE sequence is schematically illustrated in Fig. 3a.

Aiming to estimate the parallel and perpendicular diffusivities $D_\parallel$ and $D_\perp$ in the Gaussian regime for a substrate with randomly oriented microdomains, the work of Callaghan [7] shows that DDE sequences with parallel and orthogonal gradients yield different attenuation curves when the microdomains are anisotropic ($D_\parallel \neq D_\perp$). Example signal attenuation curves for microdomains with $D_\parallel > D_\perp$ and vice versa are illustrated in Fig. 3c.

Following the derivations in Eqs. (4)–(6), the average signal for a DDE sequence with parallel gradients along the (arbitrarily chosen) z direction is:

$$\bar{E}_{zz}(b_1, b_2) = \int_0^\pi \exp\left(-(b_1 + b_2)(D_\parallel \cos^2\theta + D_\perp \sin^2\theta)\right) \frac{1}{2} \sin\theta d\theta, \qquad (11)$$

where b_1 and b_2 are the diffusion weighting produced by the first and second gradient pair, respectively. When the gradients have the same orientation, only the polar angle θ appears in the exponential of Eq. (11). When the gradients have different orientations, the signal also depends on the azimuth angle ϕ. Thus, for a choice of coordinates as illustrated in Fig. 3b, the diffusion signal from a DDE

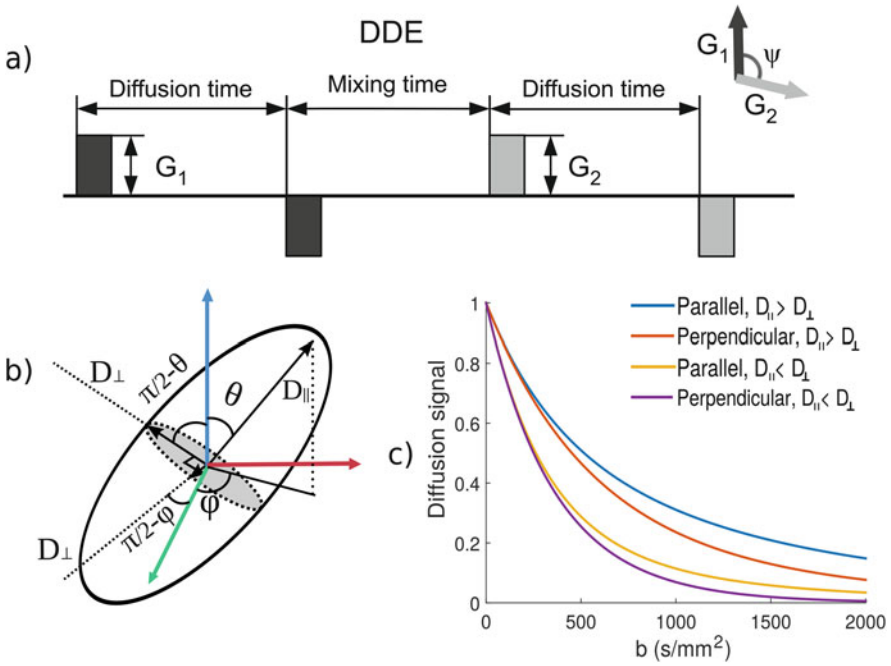


Fig. 3 Schematic representation of (**a**) double diffusion encoding (DDE) sequence and (**b**) anisotropic microdomain featuring Gaussian diffusion with DDE encoding along z and x direction. (**c**) Example attenuation curves for DDE sequences with parallel and perpendicular gradients when $D_{\parallel} > D_{\perp}$ or $D_{\parallel} < D_{\perp}$. The larger diffusion coefficient has a value of $2{\cdot}10^{-9}$ m$^2/s$ and the smaller one $2 \cdot 10^{-10}$ m^2/s

sequence with orthogonal gradients along the z and x axis is given by:

$$\bar{E}_{zx}(b_1, b_2) = \frac{1}{2\pi} \int_0^{2\pi} \int_0^{\pi} \exp[-b_1(D_{\parallel} \cos^2\theta + D_{\perp} \sin^2\theta) - b_2(D_{\parallel} \sin^2\theta \cos^2\phi + D_{\perp} \sin^2\phi + D_{\perp} \cos^2\theta \cos^2\phi)] \frac{1}{2} \sin\theta d\theta d\phi. \tag{12}$$

The difference in attenuation curves for parallel and perpendicular gradients has been experimentally measured in a polydomain lyotropic liquid crystal system (the lamellar phase of Aerosol OT-water) [7], microcapillary phantoms [27] as well as biological systems, such as fixed monkey gray matter [27] and pig spinal cord [28].

A generalization of this approach is the diffusion-diffusion correlation spectroscopy (DDCOSY) experiment [6], which consist of DDE sequences with parallel and perpendicular gradients that vary the gradient strengths G_1 and G_2 independently. The aim of this experiment is to estimate the anisotropy of the system from

a 2D Laplace transform:

$$E(b_1, b_2) = \exp(-b_1 D_1 - b_2 D_2). \tag{13}$$

The expressions of D_1 and D_2 for substrates consisting of randomly oriented microdomains are provided in Eqs. (11) and (12) for sequences with parallel and perpendicular gradients, respectively. For measurements with parallel gradients, the 2D diffusion spectrum computed from the inverse Laplace transform has only diagonal elements and provides similar information as a 1D experiment. When the gradients are orthogonal, the off diagonal peaks reflect the microscopic anisotropy of the system. Although many studies focused on DDE sequences, and their contrast is quite well understood by now, they are not the only measurements which can be used to estimate μA.

2.3 Isotropic Diffusion Encoding and q-Space Trajectories

Aiming to disentangle between the effects of orientation dispersion and a distribution of diffusivities, Lasič et al. have proposed to combine measurements with isotropic and directional diffusion encoding [32].

Isotropic Encoding and the Trace of Diffusion Tensor

The idea of isotropic diffusion encoding is not new and has been introduced earlier by Mori et al. to image the trace of the diffusion tensor in one acquisition [40]. For a substrate characterized by a diffusion tensor $\mathbf{D}$, the signal measured with a generalized gradient which may vary in amplitude and orientation, can be written as:

$$E = \exp\left(-\mathbf{b} : \mathbf{D}\right), \tag{14}$$

where $\mathbf{b} : \mathbf{D}$ denotes a generalized tensor product defined as:

$$\mathbf{b} : \mathbf{D} = \sum_i \sum_j b_{ij} D_{ij}, \text{ with } i, j \in \{x, y, z\}. \tag{15}$$

In the above equation $\mathbf{D}$ is the diffusion tensor and $\mathbf{b}$ is the diffusion encoding tensor

$$\mathbf{b} = \int_0^{TE} \mathbf{q}(t)\mathbf{q}^T(t)dt, \tag{16}$$

where $\mathbf{q}(t) = \int_0^t \mathbf{G}(t')dt'$ is the time integral of the effective gradient waveform. The standard b-value is the trace of $\mathbf{b}$.

Isotropic diffusion encoding is achieved when the off-diagonal elements of **b** are zero (i.e. $\int_0^{TE} q_i(t)q_j(t)dt = 0$ for $i \neq j$) and the diagonal elements are equal (i.e. $\int_0^{TE} q_i^2(t)dt = \int_0^{TE} q_j^2(t)dt > 0$ for any i, j). For example, one simple sequence which satisfies this condition consists of three consecutive equal gradient pairs along orthogonal axis. Many different gradient waveforms satisfy this condition which can be solved in Cartesian coordinates, e.g. [40], as well as in spherical coordinates [16]. Isotropic encoding has been initially proposed for measuring the trace of the diffusion tensor ($\mathrm{Tr}(\mathbf{D}) = \sum D_{ii} = 3\bar{D}$, where $\bar{D}$ is the mean diffusivity) in one single scan [40]. More recent work has shown the benefits of such encoding for estimating microscopic anisotropy [32].

A Probability Distribution Perspective

As presented in Sect. 2.1, assuming Gaussian diffusion in each microdomain, the diffusion signal can be written in terms of the probability distribution $P(D)$ as

$$\bar{E} = \int_0^{\infty} P(D) \exp(-bD) dD, \tag{17}$$

and can be expanded in terms of its cumulants:

$$\ln(\bar{E}) = -b\mu_1 + \frac{b^2}{2}\mu_2 + \ldots \tag{18}$$

The first cumulant μ_1 is given by the mean diffusivity $\mu_1 = \bar{D} = \int_0^{\infty} DP(D)dD$ and the second cumulant $\mu_2 = \int_0^{\infty}(D - \bar{D})^2 P(D)dD$ corresponds to the variance of $P(D)$. As illustrated in Fig. 2, in an ensemble of pores, the probability $P(D)$ describing the powder average signal, depends both on the distribution of mean diffusivities within the ensemble as well as on the anisotropy of individual microdomains. Thus, the second cumulant μ_2^{pa} of the distribution measured from a powder averaged signal reflects both these properties. By using a gradient which provides isotropic encoding, the signal depends only on the distribution of mean diffusivities in the ensemble which is reflected by the second cumulant μ_2^{iso}. Thus, by comparing the variances μ_2^{pa} and μ_2^{iso} measured with directional and isotropic encoding [32, 55], the effect of size distribution and microscopic anisotropy can be separated. In analogy to the fractional anisotropy, Lasič et al. define the microscopic fractional anisotropy as:

$$\mu FA = \sqrt{\frac{3}{2}}\left(1 + \frac{2}{5}\frac{1}{\Delta\tilde{\mu}_2}\right), \tag{19}$$

where

$$\Delta\tilde{\mu}_2 = \frac{\mu_2^{pa} - \mu_2^{iso}}{\bar{D}^2} \tag{20}$$

is the scaled difference in variance between powder averaged and isotropically encoded measurements. This approach has been used to show differences in microscopic anisotropy in various samples such as liquid crystals, yeast cells, or pureed asparagus [32], as well as in biomedical applications to investigate microscopic anisotropy in the brain and brain tumours [58]. As shown in [58] μFA provides valuable information to distinguish between different tumour types, e.g. glioblastoma and meningioma, which have very different microstructures, yet on a standard *FA* map look very similar.

Although μFA measured with directional and isotropic diffusion encoding disentangles the effects of pore elongation and size distribution, it is not sensitive enough to distinguish between pores with prolate or oblate shapes. To differentiate the two cases, Eriksson et al. [17] have proposed to extend the measurement space and vary both the trace of the diffusion encoding tensor (b-value), as well as its anisotropy, covering the parameter space between directional and isotropic encoding. This approach has been applied to distinguish between various phases (lamellar, cubic and reverse hexagonal) of the AOT/water liquid crystals which have different microstructural morphologies. Almeida Martins and Topgaard have used a similar set of measurements to recover the microstructural characteristics in a multi-compartment liquid crystal / yeast cell phantom [13]. The two dimensional data acquired by varying the linear and the isotropic components of the diffusion encoding tensor can be analysed using numerical techniques similar to a 2D Inverse Laplace Transform as well as by fitting a multi-compartment model. The peaks in the resulting 2D spectra represent the different compartments, with diagonal peaks reflecting isotropic diffusion with various diffusivities and off-diagonal peaks representing anisotropic components. In phantom and ex-vivo studies the acquired data set can span a wide parameter range, as time-constraints are not as crucial. However, for biomedical imaging, translating such approaches to in-vivo and/or clinical application is very challenging due to hardware and time limitations.

Building on the previous concepts, Westin et al. have proposed the q-space trajectory imaging (QTI) technique to estimate microscopic anisotropy which is feasible for clinical applications. QTI is a mathematical framework which combines an acquisition consisting of diffusion encoding tensors with various shapes and a data model consisting of a distribution of diffusion tensors to represent the substrate [62]. The acquisition consists of measurements with different b-values and different shapes of the **b** tensors yielding linear, prolate, spherical and planar encoding. Furthermore, several orientations for each encoding tensor have been used in order to obtain a rotationally invariant acquisition, which is important for brain imaging applications. From the perspective of the data model, using a distribution of tensors allows to separate the isotropic variance arising due to a distribution of sizes and the

anisotropic variance caused by microscopic anisotropy. A recent study [58] showed a very good correlation between diffusion-derived metrics and histology in two types of brain tumours, namely meningioma and glioblastoma. Thus, microscopic anisotropy correlated very well with cell eccentricity and the isotropic heterogeneity with cell density. Moreover, the two metrics show significant differences between the two tumour types and help understand the differences observed in the total variance.

3 Restricted Diffusion

The techniques presented above assume that diffusion within each microdomain is Gaussian, however in many systems diffusion is restricted within the pore boundaries. Thus, the measured signal is influenced by the size, shape and orientation of the pores, as well as by the timing parameters of the diffusion acquisition, not only the b-value (or the generalized **b** tensor). This section starts by describing the theoretical aspects of restricted diffusion for SDE and DDE sequences, then it presents various applications of DDE techniques as well as model-based approaches for estimating μA. We further compare the estimates of μA from two different methods, one based on DDE sequences and one on isotropic sequences, and we test the effect of assuming Gaussian diffusion when the underlying substrate is restricted. In the last part of this section, we discuss the recently proposed techniques to recover pore shape from diffusion MRI measurements acquired with various sequences.

3.1 Restricted Signal Model for SDE Sequences

For a particle which moves from point $\mathbf{r}_0$ to $\mathbf{r}_1$ during the diffusion time Δ, an ideal SDE sequence with short gradient duration induces a net phase shift $\Delta\phi(\mathbf{r}_1 - \mathbf{r}_0) = \gamma\delta\mathbf{G} \cdot (\mathbf{r}_1 - \mathbf{r}_0)$ and contributes with $\exp(i\Delta\phi)$ to the measured signal. Thus, the diffusion attenuation is computed as the ensemble average of signal contributions:

$$\langle \exp(i\Delta\phi) \rangle = \langle \exp(i\gamma\delta\mathbf{G} \cdot (\mathbf{r}_1 - \mathbf{r}_0)) \rangle \,. \tag{21}$$

Its value can be computed according to [56]:

$$E(\mathbf{G}, \Delta) = \iint \rho(\mathbf{r}_0) P(\mathbf{r}_0|\mathbf{r}_1, \Delta) \exp(i\gamma\delta\mathbf{G} \cdot (\mathbf{r}_1 - \mathbf{r}_0)) d\mathbf{r}_0 d\mathbf{r}_1, \tag{22}$$

where $\rho(\mathbf{r}_0)$ is the initial distribution of particles and $P(\mathbf{r}_0|\mathbf{r}_1, \Delta)$ is the diffusion propagator which represents the probability that a particle moves from position $\mathbf{r}_0$

to $\mathbf{r}_1$ in the time interval Δ. Many times Eq. (22) is written in terms of the wave vector $\mathbf{q} = \gamma\delta\mathbf{G}$, which yields a Fourier relationship between the diffusion signal and the average propagator.

3.2 DDE Sequences

Signal Model

Equation (22) can be easily extended for multiple gradient pulses [39, 43]. For DDE sequences with short gradient pulses and the same gradient separation Δ, the diffusion signal written in terms of $\mathbf{q}$ has the following form:

$$E(\mathbf{q}_1, \mathbf{q}_2, \Delta, \tau_m) = \int\int\int\int d\mathbf{r}_0 \rho(\mathbf{r}_0) e^{-i\mathbf{q}_1\cdot\mathbf{r}_0} \times d\mathbf{r}_1 P(\mathbf{r}_0|\mathbf{r}_1, \Delta) e^{i\mathbf{q}_1\cdot\mathbf{r}_1} \times d\mathbf{r}_2 P(\mathbf{r}_1|\mathbf{r}_2, \tau_m) e^{i\mathbf{q}_2\cdot\mathbf{r}_2} \times d\mathbf{r}_3 P(\mathbf{r}_2|\mathbf{r}_3, \Delta) e^{-i\mathbf{q}_2\cdot\mathbf{r}_3} \tag{23}$$

where $\mathbf{r}_i$ indicates the position during the application of the gradient pulses and $P(\mathbf{r}|\mathbf{r}', T)$ is the diffusion propagator which indicates the probability of a particle to move from position $\mathbf{r}$ to $\mathbf{r}'$ during time interval T.

A well known experiment for measuring diffusion correlation in different directions is angular DDE [11, 25, 39, 42, 46] which varies just one parameter, namely the angle between the two gradient orientations. In this case, the gradient amplitude, duration and diffusion time are the same for the two pulses. At long τ_m the difference between parallel and perpendicular gradient orientation increases with pore eccentricity, and thus is sensitive to microscopic anisotropy. To better understand this fact we can further analyse Eq. (23) for long diffusion and mixing times.

In the long time limit $P(\mathbf{r}|\mathbf{r}', \infty) = \rho(\mathbf{r}')$, and defining the reciprocal pore space function as $\tilde{\rho}(\mathbf{q}) = \int d\mathbf{r}\rho(\mathbf{r}) \exp(-i\mathbf{q}\cdot\mathbf{r})$, Eq. (23) becomes:

$$E(\mathbf{q}_1, \mathbf{q}_2) = |\tilde{\rho}(\mathbf{q}_1)|^2 |\tilde{\rho}(\mathbf{q}_2)|^2 \tag{24}$$

which gives the DDE signal for one pore. When we consider an ensemble of pores, the total signal is given by summing the individual contributions:

$$E_{tot}(\mathbf{q}_1, \mathbf{q}_2) = \sum_n |\tilde{\rho}_n(\mathbf{q}_1)|^2 |\tilde{\rho}_n(\mathbf{q}_2)|^2 \tag{25}$$

In the case of randomly oriented pores, the signal does not depend on the absolute orientation of $\mathbf{q_i}$, but may vary with the angle between $\mathbf{q}_1$ and $\mathbf{q}_2$ [39]. For spherical pores $\tilde{\rho}_n(\mathbf{q})$ depends only on the magnitude of $\mathbf{q}$, thus $E_{tot}(\mathbf{q}_1, \mathbf{q}_2)$ does not depend on the angle between the gradients. In contrast, for ellipsoids, $\tilde{\rho}_n(\mathbf{q})$ depends on the relative orientation of $\mathbf{q}$ to the main axis. After averaging over all possible pore

orientation, the signal $E_{tot}(\mathbf{q}_1, \mathbf{q}_2)$ still depends on the relative angle between $\mathbf{q}_1$ and $\mathbf{q}_2$, as the product $|\tilde{\rho}_n(\mathbf{q}_1)|^2|\tilde{\rho}_n(\mathbf{q}_1)|^2$ is taken before the summation. The effect appears in the fourth order expansion of the signal in $\mathbf{q}$ and is described in detail for various geometries in [42, 44].

Mitra showed theoretically that DDE sequences with varying angle between the two gradients provide sensitivity to pore elongation at long mixing times [39], a fact that was first experimentally verified in different types of yeast cells [11]. Özarslan presents a comprehensive analysis which uses a Taylor expansion of the DDE signal in idealized experimental conditions (narrow gradient pulses, long diffusion time and long or vanishing mixing times) in closed pores [42] for simple geometries of spheres, ellipsoids and capped cylinders. For arbitrary sequence parameters, the DDE signal can be calculated using other approaches to compute the restricted diffusion signal, such as Gaussian Phase Distribution [21, 41], Multiple Correlation Function (MCF) formalism [18, 42, 45], a semi-analytical Matrix Method approach [5, 15, 20] or numerical Monte Carlo simulations. To better understand the contrast in DDE sequences, numerical simulations have been used to analyse the dependence of DDE signal on sequence parameters and substrate properties such as pore size and eccentricity [26]. For instance, Fig. 4 plots the angular DDE signal for substrates consisting of randomly oriented finite cylinders of different size and eccentricities when the DDE sequences have short mixing time (a) or long mixing time (b). In the short mixing time regime the angular modulation of the signal exhibits a bell shaped curve and strongly depends on the size of the compartment, while in the long mixing time there is an angular signal modulation only for substrates which consist of anisotropic pores.

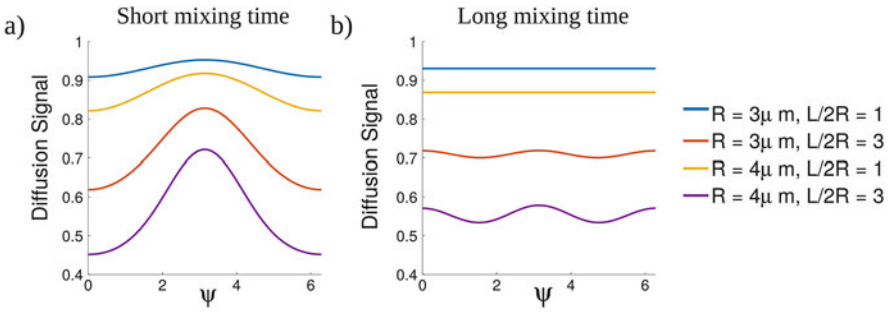


Fig. 4 Angular dependence of the DDE signal on the angle ψ between the gradient directions for substrates consisting of randomly oriented finite cylinders with different radii (R) and eccentricities ($L/2R$, where L is the length). In (**a**) the mixing time of the DDE sequence is short ($\tau_m = \delta =$ 1 ms), while in (**b**) the mixing time is long ($\tau_m = \Delta =$ 100 ms). The signal was generated using the matrix formalism described in [15], and the rest of the sequence parameters are $G =$ 500 mT/m, $\delta =$ 1 ms, $\Delta =$ 100 ms, and the gradient orientations are varied in the x-y plane

Estimating μA with Angular DDE

All these theoretical developments lead to a number of studies which aim to recover microscopic anisotropy using angular DDE experiments in various applications. Shemesh et al. investigated experimentally the angular dependence of the DDE signal in various phantoms consisting of isotropic pores (yeast cells) as well as randomly oriented anisotropic pores (e.g. water-filled microcapillaries) [46, 48, 49] and estimated the pore diameter and length by fitting the theoretical expressions of the DDE signal derived in [42] to the measured data. Information regarding the restriction length scale can also be obtained from the diffusion-diffraction patterns which are preserved in DDE measurements even in the presence of a size distribution [10, 47]. In [46], the authors perform DDE experiments to investigate the presence of microscopic anisotropy in grey and white matter of the pig spinal cord, ex-vivo. The data, which reflects the tissue structure in the transverse plan perpendicular to the main fibre orientation, shows that grey matter also exhibits microscopic anisotropy, while in this particular plane the white matter structure is less anisotropic. The angular DDE technique has been applied for imaging the rat brain ex-vivo and in-vivo in [51]. Even though the DDE signal measured in the x-y plane depends on the macroscopic fibre orientation, the study presents a data analysis scheme which maps the effect of macroscopic anisotropy as a residual phase in the angular dependence of the signal and uses this to estimate a corrected apparent eccentricity.

Rotationally Invariant Metrics of μA

Although some effects of macroscopic anisotropy can be corrected in angular DDE experiments [51], acquiring measurements as described above provides information regarding pore size and eccentricity only in the plane spanned by the gradient vectors. Thus, this is not directly a suitable technique to capture the full structure of macroscopically anisotropic substrates. To address this issue, Lawrenz et al. [37] use a fourth order Taylor expansion of the signal in $\mathbf{q}$ to provide a tensor form of the DDE signal in the presence of macroscopic anisotropy. Furthermore, they use the tensor elements to derive a rotationally invariant metric of microscopic anisotropy (I_{MA}) and its normalized counterpart (MA) based on the difference between DDE measurements with parallel and perpendicular gradients, which require a minimum of 15 combinations of the two gradient directions. In recent studies, Lawrenz et al. have used DDE sequences on a clinical scanner to investigate microscopic anisotropy in the human brain, in-vivo [35, 36]. In [35] they analyse both angular DDE measurements acquired in the three orthogonal planes as well as a rotationally invariant DDE acquisition. A later study [36] compares the effect of using different number of gradient directions in the rotationally invariant acquisition, showing that a protocol with an increased number of direction yields a smaller variance of the estimated MA. Both studies show that microscopic anisotropy values are more uniform across white matter compared to the standard

fractional anisotropy *FA* which is influenced by the orientation distribution of the fibres.

Jespersen et al. [22] further extends the DDE acquisition to a rotationally invariant 5-design scheme that can be applied to the cumulant expansion of the signal, which is more accurate than the Taylor expansion assumed in the previous work. The acquisition protocol consists of 12 measurements with parallel gradients pointing towards the vertices of a dodecahedron and 60 measurements with perpendicular gradients, 5 for each orientation. Using the difference between DDE measurements, they derive a rotationally invariant eccentricity metric (ϵ) and its normalized counterpart, fractional eccentricity (*FE*), which is the same as the macroscopic fractional anisotropy *FA* in the case of coherently oriented pores:

$$\epsilon = \frac{1}{q^4}\left[\log\left(\frac{\sum S_{\parallel}}{12}\right) - \log\left(\frac{\sum S_{\perp}}{60}\right)\right], \tag{26}$$

where $S_{\parallel}$ is the signal for the DDE sequence with parallel gradients, $S_{\perp}$ is the signal for the DDE sequence with perpendicular gradients, $q = \gamma G\delta$ is the wavenumber, γ the gyromagnetic ratio, G the gradient strength, δ the pulse duration and ϵ depends on pore size and eccentricity. For spherical pores $\epsilon = 0$. Fractional eccentricity normalizes ϵ with respect to size:

$$FE = \sqrt{\frac{\epsilon}{\epsilon + 3\Delta^2 ADC^2/5}} \tag{27}$$

which varies between 0 (spherical pores) and 1 (elongated pores), where Δ is the diffusion time and *ADC* is the apparent diffusion coefficient which reflects the length scale of the substrate. The eccentricity metric ϵ derived in [22] and the I_{MA} index from [37] are related by $\epsilon = \frac{2}{15} I_{MA}$.

The QTI framework discussed in Sect. 2.3 estimates as well a number of rotational invariant features (e.g bulk and sheer) from the covariance of the tensor distribution, under the assumption of Gaussian diffusion in each microdomain. Such invariants are then used to derive more familiar metrics, like μFA.

3.3 Double Oscillating Diffusion Encoding

The DDE theory presented in Sect. 3.1 provides the diffusion signal for ideal DDE sequences, in the limit of short pulses and long diffusion and mixing times. However, such sequences can rarely be achieved in practice, and do not necessarily provide the highest sensitivity to microscopic anisotropy [19]. Ianus et al. has shown that double oscillating diffusion encoding (DODE) sequences, which replace each pair of gradients in a DDE sequence with oscillating gradient waveforms, can improve the sensitivity of the measurements to pore diameter.

Thus when estimating pore size and eccentricity, a protocol which has both DODE and DDE sequences provides the optimal measurements. Moreover, the time dependence of μFA can be probed by varying the frequency of DODE measurements, in order to analyse the microstructural properties at different lengths scales.

3.4 Comparison of Two μA Metrics

In order to compare two μA metrics, for instance FE and μFA, we require diffusion sequences that are as similar as possible in terms of gradient waveform. This can be achieved by using DODE sequences and adapting the DODE protocol for each approach ensuring that the sequences have the same gradient waveform and vary only in direction.

To estimate FE we use the 5-design protocol presented in Sect. 3.2 and we adapt the definition of FE for DODE sequences:

$$FE = \sqrt{\frac{\epsilon}{\epsilon + \frac{3}{5}\left(\frac{\delta}{3N}\right)^2 \left(\frac{Tr(\mathbf{D})}{3}\right)^2}}, \tag{28}$$

where ϵ is defined in Eq. (26), $\mathbf{D}$ is the effective diffusion tensor computed from the parallel measurements and $\delta/3N$ is the effective diffusion time, with δ the gradient duration and N the number of periods.

To compute μFA, we use the expression in Eq. (19). Thus, We fit Eq. (18) to the isotropically encoded measurements and to the powder averaged data with two different assumptions:

- We enforce the same mean diffusivity when fitting the isotropic encoded and the powder averaged data, as assumed in [32].
- We fit the cumulant expansion separately to the two data sets and we obtain two different values of mean diffusivity and variance.

Simulation parameters: the DODE sequences have three periods $N = 3$, a gradient duration $\delta_{DODE} = 60$ ms, mixing time $\tau_m = 20$ ms and gradient strength G_{DODE} that we modify depending on the substrates we investigate. For estimating FE we use the 5-design scheme explained in Sect. 3.2. For estimating μFA we use 16 different gradient strengths between 0 and G_{DODE}, and for each gradient strength we have 15 isotropically oriented directions to compute the directional average and 15 repeats for the isotropic encoding, to match the acquisition described in [32]. For the sequences providing isotropic encoding, the gradient duration in each direction is $\delta_{DODE}/2N = 10$ ms. The two acquisition protocols have the same gradient waveform and maximum b-value and are illustrated in Fig. 5a.

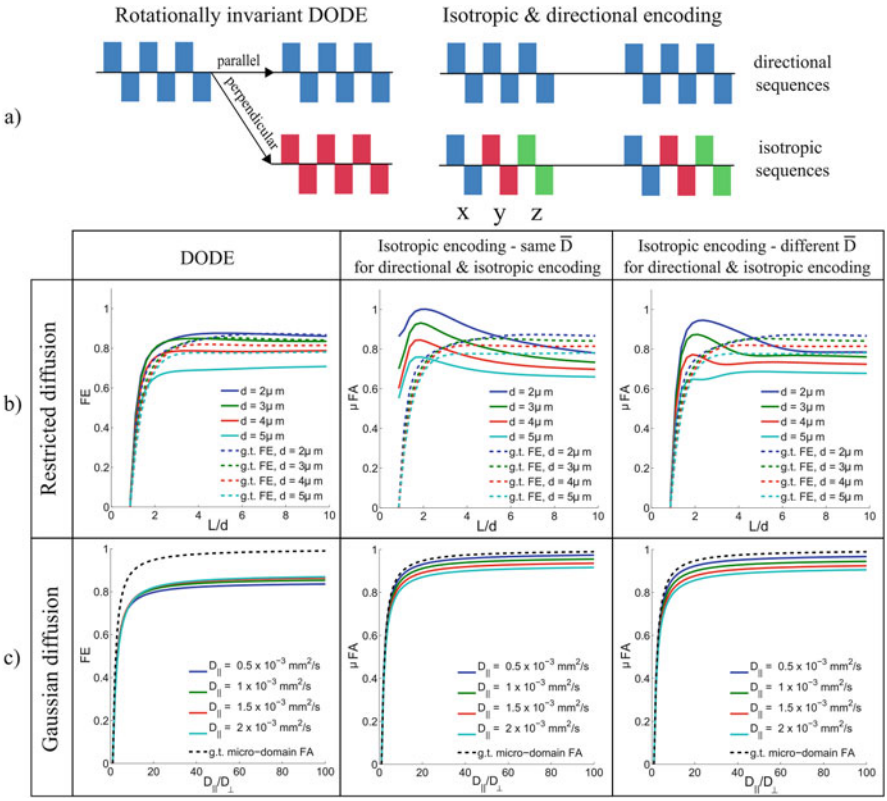


Fig. 5 (**a**) Schematic representation of diffusion sequences used to compute fractional eccentricity (*left*) and microscopic fractional anisotropy (*right*); (**b**) *Restricted diffusion:* dependence of *FE* and μFA on pore elongation for substrates which consist of randomly oriented pores. μFA is calculated in two different ways: data from directional and isotropic encoding is fitted using the same mean diffusivity and different variances, as in [32] (*middle*) and the data sets from directional and isotropic encoding are fitted separately (*right*). Gold standard *FE* values computed from an ideal DDE protocol are shown as well (*dashed line*). The maximum gradient strength is 300 mT/m corresponding to $b = 25{,}780$ s/mm^2 (**c**) *Gaussian diffusion:* dependence of *FE* and μFA on the ratio between parallel and perpendicular diffusivities for substrates which consist of randomly oriented domains. μFA is calculate in two different ways as explained above. The ground truth micro-domain *FA* is plotted as well (*dashed line*). In this case, the maximum gradient strength is 100 mT/m, corresponding to $b = 2865$ s/mm^2

We investigate two different types of diffusion substrates and adapt the maximum gradient strength accordingly:

- Randomly oriented anisotropic pores, which exhibit restricted diffusion. The maximum gradient strength for this simulation is $G_{DODE} = 300$ mT/m, which yields a b-value of 25,780 s/mm^2.
- Randomly oriented anisotropic domains which exhibit Gaussian diffusion with different parallel and perpendicular diffusivities. The maximum gradient strength

is $G_{DODE} = 100\,\text{mT/m}$, which yields a b-value of $2865\,\text{s/mm}^2$. This is similar to the simulations in [32, 58].

Figure 5b illustrates the dependence of FE and μFA on pore elongation for a substrate of anisotropic pores exhibiting restricted diffusion. In this case, the FE and μFA value computed assuming the same mean diffusivity for isotropic encoding and the directional average overestimates the microscopic anisotropy of pores with low eccentricity. Relaxing this assumption improves the values of μFA, nevertheless, it is not monotonically increasing with pore eccentricity. In case of restricted diffusion, the fractional eccentricity computed from DODE measurements is closer to the gold standard values. Figure 5c illustrates the dependence of FE and μFA on the ratio between parallel and perpendicular diffusivities in randomly oriented domains which exhibit Gaussian diffusion. In this case μFA is slightly closer to the ground truth FA values of the microdomains compared to FE. Moreover, the assumption that the mean diffusivity is the same for the directional average and isotropic encoding holds. This situation is representative of the simulations from [32, 58]. The extended models described in Sect. 2.3 which have a distribution of tensors, might be able to better represent restricted diffusion, by mapping it as an isotropic variance of the tensors, nevertheless, this needs to be investigated.

3.5 Model-Based Estimation of Pore Size and Shape

A different approach to estimate pore size and eccentricity is to develop a mathematical model which relates the substrate parameters to the acquired signal and to fit the data in order to estimate the parameters of interest. A recent simulation study showed that estimates of pore size and eccentricity can be simultaneously recovered from diffusion measurements using a model-based approach [21]. The technique uses a geometric model of finite cylinders with either one radius or gamma distributed radii to represent pores of various sizes and elongations. The two different models are fitted to synthetic SDE and DDE data generated using Monte Carlo simulations with geometric meshes consisting of cuboids with various size distributions and eccentricities, as illustrated in Fig. 6a, b. When the substrates consist of identical pores which differ only in orientation, then both SDE and DDE measurements provide similarly accurate estimates of pore size and eccentricity, and the simple model with one radius can be used to accurately recover microstructure parameters. However, when the substrates consist of elongated pores with a distribution of sizes, then DDE measurements yield more accurate values, and the size distribution needs to be explicitly accounted for in the tissue model in order to obtain accurate parameter estimates, as illustrated in Fig. 6c, d. A similar model-based approach can also be extended to account for orientation distribution in macroscopically anisotropic substrates [21]. These results are consistent with the previous studies discussed in Sect. 2.1 which assumed an ensemble of identical

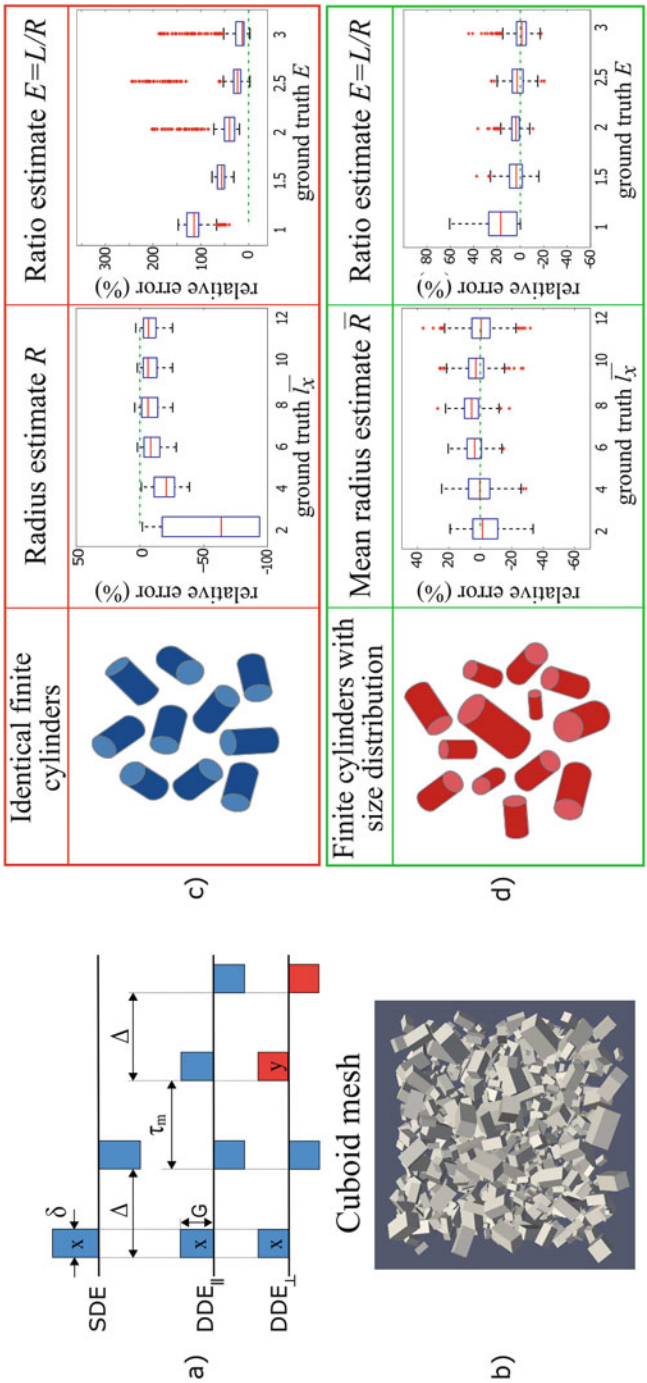


Fig. 6 Schematic representation of (**a**) diffusion sequences and (**b**) substrates used in the simulations. Median values of parameter estimates (radius and eccentricity) from a model with (**c**) one radius or (**d**) a gamma distribution of radii for various diffusion substrates and DDE sequences with parallel and perpendicular gradients. An in-depth analysis is presented in [21]

microdomains with Gaussian diffusion to estimate μFA from the parallel and perpendicular diffusivity values.

3.6 Direct Estimation of the Pore Shape Function

A very different approach to measure pore shape from diffusion data is to recover the pore shape function discussed in the beginning of this section. According to Eq. (22), the SDE signal in the long diffusion time limit is:

$$E = \int \rho(\mathbf{r}_0) e^{i\mathbf{q}\cdot\mathbf{r}_0} d\mathbf{r}_0 \int \rho(\mathbf{r}_1) e^{-i\mathbf{q}\cdot\mathbf{r}_1} d\mathbf{r}_1 = |\tilde{\rho}(\mathbf{q})|^2, \tag{29}$$

where the reciprocal pore space function $\tilde{\rho}(\mathbf{q}) = \int d\mathbf{r}\rho(\mathbf{r}) \exp(-i\mathbf{q}\cdot\mathbf{r})$ is the Fourier transform of the spin density in the pore space $\rho(\mathbf{r})$. For SDE measurements, the signal depends only on the magnitude $|\tilde{\rho}(\mathbf{q})|$, therefore the pore space function cannot be recovered from an inverse Fourier transform since the phase information is lost [34]. In SDE measurements this is a direct consequence of the anti-symmetry of the diffusion gradient (i.e. $\mathbf{G}_1 = -\mathbf{G}_2$).

To be able to recover the pore space function $\rho(\mathbf{r})$, Laun et al. proposed a different diffusion acquisition which consists of a weak and long gradient pulse followed by a strong and narrow one, which satisfy the echo condition $\mathbf{G}_1 \cdot \delta_1 = \mathbf{G}_2 \cdot \delta_2 = \mathbf{q}$ [34]. In this case the phase accumulated during the first gradient depends on the pore centre of mass $\phi_1 = \int \mathbf{G}_1(t) \cdot \mathbf{r}(t) dt = \mathbf{G}_1 \cdot \mathbf{r}_{\mathbf{cm}}$, and the signal becomes:

$$E = e^{i\mathbf{q}\cdot\mathbf{r}_{cm}} \int d\mathbf{r}_1 \rho(\mathbf{r}_1) e^{-i\mathbf{q}\cdot\mathbf{r}_1} = e^{i\mathbf{q}\cdot\mathbf{r}_{cm}} \tilde{\rho}(\mathbf{q}). \tag{30}$$

Thus, the phase information of $\tilde{\rho}(\mathbf{q})$ is preserved and the pore space function can be directly evaluated by inverse Fourier transform. Long-narrow gradient combinations are not the only ones which can be used for this purpose. Shemesh et al. showed that DDE sequences with equal gradient strengths $\mathbf{q_1} = \mathbf{q_2}$, zero mixing time $\tau_m = 0$ and long diffusion time, can also be used to measure the pore space function [50]. In this case the measured signal is:

$$\begin{aligned} E^{DDE} &= \int \rho(\mathbf{r}_0) e^{-i\mathbf{q}\cdot\mathbf{r}_0} d\mathbf{r}_0 \int \rho(\mathbf{r}_1) e^{2i\mathbf{q}\cdot\mathbf{r}_1} d\mathbf{r}_1 \int \rho(\mathbf{r}_2) e^{-i\mathbf{q}\cdot\mathbf{r}_2} d\mathbf{r}_2 \\ &= \tilde{\rho}(\mathbf{q})^2 \tilde{\rho}^*(2\mathbf{q}), \end{aligned} \tag{31}$$

and it has a different dependence on the reciprocal pore space function compared to the SDE measurements which depend only on the magnitude. This also implies that the signal can cross the zero axis and take negative values, which has been shown in [43, 47, 50]. By dividing the signal measured from a DDE sequence and the signal

from an SDE sequence with the same $\mathbf{q}$, one obtains:

$$\frac{E^{DDE}}{E^{SDE}} = \frac{\tilde{\rho}(\mathbf{q})^2 \tilde{\rho}^*(2\mathbf{q})}{|\tilde{\rho}(\mathbf{q})|^2}. \tag{32}$$

For point symmetric pores, e.g. ellipsoids, $\tilde{\rho}(\mathbf{q})$ is real, and the pore function can be estimated by taking the inverse Fourier transform of the quotient in Eq. (32), which has been experimentally used for estimating the pore space function in microcapillaries [50]. In the general case of non-symmetric pores, the phase of $\tilde{\rho}(\mathbf{q})$ can be iteratively reconstructed as described by Kuder et al. [29] and the performance of various gradient waveforms to estimate the pore function has been analysed in [33]. The reconstruction of the pore space function for identical, triangular pores has been experimentally validated in a polarized Helium phantom [31], while the effect of a distribution of shapes is considered in [30].

4 Summary

This chapter presents the mathematical background and possible applications of various techniques used in diffusion NMR and MRI to estimate the microscopic anisotropy of pores in the underlying substrates. The different approaches can be classified in terms of the diffusion sequences used for acquisition as well as in terms of the signal model which describes the diffusion process in the material.

In Sect. 2 we present techniques which assume that diffusion in each pore of an ensemble is in the Gaussian regime and the signal model is accurately described by a diffusion tensor. Thus, the ensemble signal is given by a distribution of diffusion tensors, which can vary in orientation and/or size depending on the assumption of each method. In Sect. 3, we consider the diffusion to be restricted within the pore boundaries and thus microscopic anisotropy is directly influenced by geometrical properties of the pore such as diameter and length. Thus, the diffusion signal is written in terms of the spin displacements in the substrate, instead of a distribution of diffusivities. In this case, as the spin displacements are time dependent, the measured signal also depends on the timings of the diffusion sequence and not only the b-value (b tensor). Besides the choice of signal model to represent the diffusion process in the underlying substrate, the choice of acquisition sequences has a great impact on the accuracy and applicability of various techniques.

If there is prior knowledge that the substrates consist of identical microdomains, then SDE measurements with at least two different b-values can be used to estimate microscopic anisotropy. The main advantage of such an approach, especially for clinical imaging applications, is the fact that it does not require custom acquisition protocols and can be used retrospectively to analyse various datasets [23]. The

drawback of this assumption is that the presence of a size distribution and/or multiple compartments with various diffusivities will get interpreted as increased microscopic anisotropy. To overcome this issue when mapping neurite density in the brain, Kaden et al. proposed to extend the original spherical mean technique to a two-compartment tissue model, with the intracellular compartment described by a "stick" model (diffusion tensor with $D_\perp = 0$) and hindered diffusion in the extracellular space [24], instead of identical microdomains as in [23]. Although, this assumptions is appropriate for brain imaging in the range of diffusion times and b-values used in the clinic, such as model might not applicable in case of pathology, such as tumours.

Using diffusion encoding sequences which vary the gradient direction within one measurement can disentangle the effects of a size distribution from microscopic anisotropy. The DDE sequence proposed by Mitra [39] can probe the correlation of diffusion in different directions, when the orientations of the two gradient pairs are non-collinear. Moreover, in the long diffusion and mixing time limit, the amplitude modulation of the DDE signal as the angle between the two gradients is varied reflects the presence of microscopic anisotropy. Such angular DDE experiments have been performed in phantoms [11, 46, 48], ex-vivo and in-vivo biological tissue [51] as well as in humans in-vivo [35] to characterize the microscopic anisotropy. Rotationally invariant extensions for the DDE acquisition have also been proposed for accurate estimation of μFA in macroscopically anisotropic substrates [22, 37]. In these techniques μFA is calculated based on the signal difference between measurements with parallel and perpendicular gradients. As taking the difference of two noisy measurements has a negative impact on the signal-to-noise ratio (SNR), the estimated μFA can be a very noisy measure depending on the quality of the original data. Alternatively, a signal model which accounts for anisotropic compartments can be fitted to the measured data in order to estimate μFA without computing a signal difference [21]. Moreover, the acquisition protocol can be significantly shortened by optimising the gradient waveform of the diffusion sequence. Such an approach has been beneficial for estimating pore diameter in cylindrical restriction [1, 14, 53, 54] and can be simply adapted to models which include microscopic anisotropy.

A different technique for estimating microscopic anisotropy proposed by Lasic et al. [32] combines measurements with isotropic and directional encoding acquired at several b-values. The variance of the probability distribution of diffusivities calculated from the powder average of the directional measurements reflects both the size distribution and the microscopic anisotropy of the microdomains, while the variance calculated from isotropic measurements reflects only the size distribution in the substrate. Thus, the difference in variance obtained from the two sets of measurements reflects the microscopic anisotropy. This approach disentangles the effects of μFA and size distribution, however, it cannot differentiate between an ensemble of oblate or prolate microdomains. Thus, more recent studies [13, 17] have extended the acquisition to a range of $\mathbf{b}$ tensors which span the intermediary values between linear and isotropic encoding. Westin et al. have adapted a similar acquisition for clinical practice and proposed a mathematical framework to

estimate microscopic anisotropy and other invariant metrics from a signal model which consists of a distribution of tensors [62]. All these techniques assume that diffusion in each microdomain is Gaussian and the estimated parameters do not depend on the timings of the sequence. Thus, if the data is acquired with different q-space trajectories, the estimated parameters might be different even if the **b** tensors are the same. This effect should be considered when comparing the results from different acquisitions. Moreover, the simulations in Sect. 3.4 show that μA might be overestimated if diffusion is highly restricted in isotropic pores. One drawback of these techniques is the fact that they require custom-made diffusion sequences, which can be very difficult to implement in a standard clinical set-up.

A distinct approach to estimate pore shape from diffusion MR data, is to use an acquisition with gradients that are not anti-symmetric and to compute the inverse Fourier transform of the reciprocal pore space function. Some measurements like long-narrow pulses yield directly the reciprocal pore space function, while for a variety of other gradient waveforms, including DDE sequences, an iterative estimation of the phase is necessary, as explained in Sect. 3.6.

Estimating microscopic anisotropy, and in general pore shape, from diffusion MRI data is an active research field with a plethora of applications. There have been many recent developments both from the data acquisition perspective as well as from the modelling side in order to obtain more robust and meaningful estimates of microstructural features.

Acknowledgements This study was supported by EPSRC grants M507970, G007748, H046410, K020439, and M020533 and the Leverhulme trust. Funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 657366 supports NS's work on this topic.

References

1. Alexander, D.C.: A general framework for experiment design in diffusion MRI and its application in measuring direct tissue-microstructure features. Magn. Reson. Med. **60**, 439–448 (2008)
2. Anderson, A.W.: Measurement of fiber orientation distributions using high angular resolution diffusion imaging. Magn. Reson. Med. **54**, 1194–1206 (2005)
3. Basser, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative diffusion tensor MRI. J. Magn. Reson. B **111**, 209–19 (1996)
4. Bernin, D., Koch, V., Nydén, M., Topgaard, D.: Multi-scale characterization of lyotropic liquid crystals using ^{2}H and diffusion MRI with spatial resolution in three dimensions. PLoS One **9**, e98752 (2014)
5. Callaghan, P.T.: Pulsed-gradient spin-echo NMR for planar, cylindrical and spherical pores under conditions of wall relaxation. J. Magn. Reson. **113**, 53–59 (1995)
6. Callaghan, P.T., Furo, I.: Diffusion-diffusion correlation and exchange as a signature for local order and dynamics. J. Chem. Phys. **120**, 4032–4037 (2004)
7. Callaghan, P.T., Komlosh, M.E.: Locally anisotropic motion in a macroscopically isotropic system: displacement correlations measured using double pulsed gradient spin-echo NMR. Magn. Reson. Chem. **40**, 15–19 (2002)

8. Callaghan, P.T., Söderman, O.: Examination of the lamellar phase of aerosol OT/water using pulsed field gradient nuclear magnetic resonance. J. Phys. Chem. **87**, 7737–1744 (1983)
9. Callaghan, P.T., Jolley, K.W., Lelievre, J.: Diffusion of water in the endosperm tissue of wheat grains as studied by pulsed field gradient nuclear magnetic resonance. Biophys. J. **28**, 133–141 (1979)
10. Callaghan, P.T., Coy, A., MacGowan, D., Packer, K.J., Zelaya, F.O.: Diffraction-like effects in NMR diffusion studies of fluids in porous solids. Nature **351**, 467–469 (1991)
11. Cheng, Y., Cory, D.: Multiple scattering by NMR. J. Am. Chem. Soc. **121**, 7935–7396 (1999)
12. Cory, D.G., Garroway, A.N., Miller, J.B.: Applications of spin transport as a probe of local geometry. Polym. Prepr. **31**, 149–150 (1990)
13. de Almeida Martins, J.P., Topgaard, D.: Two-dimensional correlation of isotropic and directional diffusion using NMR. Phys. Rev. Lett. **116**, 087601 (2016)
14. Drobnjak, I., Alexander, D.C.: Optimising time-varying gradient orientation for microstructure sensitivity in diffusion-weighted MR. J. Magn. Reson. **212**, 344–354 (2011)
15. Drobnjak, I., Zhang, H., Hall, M.G., Alexander, D.C.: The matrix formalism for generalised gradients with time-varying orientation in diffusion NMR. J. Magn. Reson. **210**, 151–157 (2011)
16. Eriksson, S., Lasič, S., Topgaard, D.: Isotropic diffusion weighting in PGSE NMr by magic-angle spinning of the q-vector. J. Magn. Reson. **226**, 13–8 (2013)
17. Eriksson, S., Lasič, S., Nilsson, M., Westin, C.F., Topgaard, D.: NMR diffusion-encoding with axial symmetry and variable anisotropy: distinguishing between prolate and oblate microscopic diffusion tensors with unknown orientation distribution. J. Chem. Phys. **142**, 104201 (2015)
18. Grebenkov, D.: NMR survey of reflected Brownian motion. Rev. Mod. Phys. **79**, 1077–1137 (2007)
19. Ianuş, A., Shemesh, N., Alexander, D.C., Drobnjak, I.: Double oscillating diffusion encoding and sensitivity to microscopic anisotropy. Magn. Reson. Med. (2016). Early view
20. Ianuş, A., Alexander, D.C., Drobnjak, I.: Microstructure imaging sequence simulation toolbox. In: Proceedings of SASHIMI workshop of MICCAI, Athens (2016)
21. Ianuş, A., Drobnjak, I., Alexander, D.C.: Model-based estimation of microscopic anisotropy using diffusion MRI: a simulation study. NMR Biomed. **29**, 627–685 (2016)
22. Jespersen, S.N., Lundell, H., Sonderby, C.K., Dyrby, T.B.: Orientationally invariant metrics of apparent compartment eccentricity from double pulsed field gradient diffusion experiments. NMR Biomed. **26**, 1647–1662 (2013)
23. Kaden, E., Kruggel, F., Alexander, D.C.: Quantitative mapping of the per-axon diffusion coefficients in brain white matter. Magn. Reson. Med. **75**, 1752–63 (2015)
24. Kaden, E., Kelm, N.D., Carson, R.P., Does, M.D., Alexander, D.C.: Multi-compartment microscopic diffusion imaging. NeuroImage **139**, 346–359 (2016)
25. Koch, M.A., Finsterbusch, J.: Compartment size estimation with double wave vector diffusion-weighted imaging. Magn. Reson. Med. **60**, 90–101 (2008)
26. Koch, M.A., Finsterbusch, J.: Numerical simulation of double-wave vector experiments investigating diffusion in randomly oriented ellipsoidal pores. Magn. Reson. Med. **62**, 247–254 (2009)
27. Komlosh, M.E., Horkay, F., Freidlin, R.Z., Nevo, U., Assaf, Y., Basser, P.J.: Detection of microscopic anisotropy in gray matter and in a novel tissue phantom using double pulsed gradient spin echo MR. J. Magn. Reson. **189**, 38–45 (2007)
28. Komlosh, M.E., Lizak, M.J., Horkay, F., Freidlin, R.Z., Basser, P.J.: Observation of microscopic diffusion anisotropy in the spinal cord using double-pulsed gradient spin echo MRI. Magn. Reson. Med. **59**, 803–809 (2008)
29. Kuder, T.A., Laun, F.B.: NMR-based diffusion pore imaging by double wave vector measurements. Magn. Reson. Med. **70**, 836–841 (2013)
30. Kuder, T.A., Laun, F.B.: Effects of pore-size and shape distributions on diffusion pore imaging by nuclear magnetic resonance. Phys. Rev. Lett. **92**, 022706 (2015)
31. Kuder, T.A., Bachert, P., Windschuh, J., Laun, F.B.: Diffusion pore imaging by hyperpolarized xenon-129 nuclear magnetic resonance. Phys. Rev. Lett. **111**, 028101 (2013)

32. Lasič, S., Szczepankiewicz, F., Eriksson, S., Nilsson, M., Topgaard, D.: Microanisotropy imaging: quantification of microscopic diffusion anisotropy and orientational order parameter by diffusion MRI with magic-angle spinning of the q-vector. Front. Phys. **2**, 11 (2014)
33. Laun, F.B., Kuder, T.A.: Diffusion pore imaging with generalized temporal gradient profiles. Magn. Reson. Imaging **31**, 1236–44 (2013)
34. Laun, F.B., Kuder, T.A., Semmler, W., Stieltjes, B.: Determination of the defining boundary in nuclear magnetic resonance diffusion experiments. Phys. Rev. Lett. **107**, 048102 (2011)
35. Lawrenz, M., Finsterbusch, J.: Double-wave-vector diffusion-weighted imaging reveals microscopic diffusion anisotropy in the living human brain. Magn. Reson. Med. **69**, 1072–1082 (2013)
36. Lawrenz, M., Finsterbusch, J.: Mapping measures of microscopic diffusion anisotropy in human brain white matter in vivo with double-wave-vector diffusion-weighted imaging. Magn. Reson. Med. (2014). Early view
37. Lawrenz, M., Koch, M.A., Finsterbusch, J.: A tensor model and measures of microscopic anisotropy for double-wave-vector diffusion-weighting experiments with long mixing times. J. Magn. Reson. **202**, 43–56 (2010)
38. Lawrenz, M., Brassen, S., Finsterbusch, J.: Microscopic diffusion anisotropy in the human brain: Age-related changes. NeuroImage **141**, 313–325 (2016)
39. Mitra, P.P.: Multiple wave-vector extensions of the NMR pulsed-field-gradient spin-echo diffusion measurement. Phys. Rev. B **51**(21), 15074–15078 (1995)
40. Mori, S., van Zijl, P.C.M.: Diffusion weighting by the trace of the diffusion tensor within a single scan. Magn. Reson. Med. **33**, 41–52 (1995)
41. Neuman, C.H.: Spin echo of spins diffusing in a bounded medium. J. Chem. Phys. **60**, 4508–4511 (1974)
42. Özarslan, E.: Compartment shape anisotropy (CSA) revealed by double pulsed field gradient MR. J. Magn. Reson. **199**, 56–67 (2009)
43. Özarslan, E., Basser, P.J.: MR diffusion/diffraction phenomenon in multi-pulse-field-gradient experiments. J. Magn. Reson. **188**, 285–294 (2007)
44. Özarslan, E., Besser, P.J.: Microscopic anisotropy revealed by NMR double pulsed field gradient experiments with arbitrary timing parameters. J. Chem. Phys. **128**, 154511 (2009)
45. Özarslan, E., Shemesh, N., Basser, P.J.: A general framework to quantify the effect of restricted diffusion on the NMR signal with applications to double pulsed field gradient NMR experiments. J. Chem. Phys. **130**, 104702 (2009)
46. Shemesh, N., Cohen, Y.: Microscopic and compartment shape anisotropies in grey and white matter revealed by angular bipolar double-PFG. Magn. Reson. Med. **65**, 1216–1227 (2011)
47. Shemesh, N., Özarslan, E., Basser, P.J., Cohen, Y.: Detecting diffusion-diffraction patterns in size distribution phantoms using double-pulsed field gradient NMR: theory and experiments. J. Chem. Phys. **132**, 034703 (2010)
48. Shemesh, N., Özarslan, E., Adiri, T., Basser, P.J., Cohen, Y.: Noninvasive bipolar double-pulsed-field-gradient NMR reveals signatures for pore size and shape in polydisperse, randomly oriented, inhomogeneous porous media. J. Chem. Phys. **133**, 044705 (2010)
49. Shemesh, N., Özarslan, E., Basser, P.J., Cohen, Y.: Accurate noninvasive measurement of cell size and compartment shape anisotropy in yeast cells using double-pulsed field gradient MR. NMR Biomed. **25**, 236–246 (2011)
50. Shemesh, N., Westin, C.-F., Cohen, Y.: Magnetic resonance imaging by synergistic diffusion-diffraction patterns. Phys. Rev. Lett. **108**, 058103 (2012)
51. Shemesh, N., Barazany, D., Sadan, O., Bar, L., Zur, Y., Barhum, Y., Sochen, N., Offen, D., Assaf, Y., Cohen, Y.: Mapping apparent eccentricity and residual ensemble anisotropy in the gray matter using angular double-pulsed-field-gradient MRI. Magn. Reson. Med. **68**, 795–806 (2012)
52. Shemesh, N., Jespersen, S.N., Alexander, D.C., Cohen, Y., Drobnjak, I., Dyrby, T.B., Finsterbusch, J., Koch, M.A., Kuder, T., Laun, F., Lawrenz, M., Lundell, H., Mitra, P.P., Nilsson, M., Özarslan, E., Topgaard, D., Westin, C.F.: Conventions and nomenclature for double diffusion encoding (DDE) NMR and MRI. Magn. Reson. Med. **75**, 82–87 (2016)

53. Siow, B., Drobnjak, I., Chatterjee, A., Lythgoe, M.F., Alexander, D.C.: Estimation of pore size in a microstructure phantom using the optimised gradient waveform diffusion weighted NMR sequence. J. Magn. Reson. **214**, 51–60 (2012)
54. Siow, B., Drobnjak, I., Ianuş, A., Christie, I.N., Lythgoe, M.F., Alexander, D.C.: Axon radius estimation with oscillating gradient spin echo (OGSE) diffusion MRI. Diffus. Fundam. **18**, 1–6 (2013)
55. Sjölund, J., Szczepankiewicz, F., Nilsson, M., Topgaard, D., Westin, C.F., Knutsson, H.: Constrained optimization of gradient waveforms for generalized diffusion encoding. J. Magn. Reson. **261**, 157–68 (2015)
56. Stejskal, E.O.: Use of spin echoes in a pulsed magnetic-field gradient to study anisotropic, restricted diffusion and flow. J. Chem. Phys. **43**, 3597–3603 (1965)
57. Stejskal, E.O., Tanner, T.E.: Spin diffusion measurements: spin echoes in the presence of a time-dependent field gradient. J. Chem. Phys. **42**, 288–292 (1965)
58. Szczepankiewicz, F., Lasič, S., van Westen, D., Sundgren, P.C., Englund, E., Westin, C.F., Ståhlberg, F., Lätt, J., Topgaard, D., Nilsson, M.: Quantification of microscopic diffusion anisotropy disentangles effects of orientation dispersion from microstructure: applications in healthy volunteers and in brain tumors. NeuroImage **104**, 241–52 (2015)
59. Szczepankiewicz, F., van Westen, D., Englund, E., Westin, C.F., Ståhlberg, F.: Lätt, J., Sundgren, P.C., Nilsson, M.: The link between diffusion MRI and tumor heterogeneity: mapping cell eccentricity and density by diffusional variance decomposition (DIVIDE). NeuroImage (2016). Early view
60. Topgaard, D., Söderman, O.: Self-diffusion in two- and three-dimensional powders of anisotropic domains: an NMR study of the diffusion of water in cellulose and starch. J. Phys. Chem. **106**, 11887–11892 (2002)
61. Westin, C.F., Szczepankiewicz, F., Pasternak, O., Özarslan, E., Topgaard, D., Knutsson, H., Nilsson, M.: Measurement tensors in diffusion MRI: generalizing the concept of diffusion encoding. In: Proceedings of MICCAI, Boston, pp. 209–216. Springer, Berlin (2014)
62. Westin, C.F., Knutsson, H., Pasternak, O., Szczepankiewicz, F., Özarslan, E., van Westen, D., Mattisson, C., Bogren, M., O'Donnell, L.J., Kubicki, M., Topgaard, D., Nilsson, M.: Q-space trajectory imaging for multidimensional diffusion MRI of the human brain. NeuroImage **135**, 345–362 (2016)

Bayesian Heteroscedastic Regression for Diffusion Tensor Imaging

Bertil Wegmann, Anders Eklund, and Mattias Villani

Abstract We propose a single-diffusion tensor model with heteroscedastic noise and a Bayesian approach via a highly efficient Markov Chain Monte Carlo (MCMC) algorithm for inference. The model is very flexible since both the noise-free signal and the noise variance are functions of diffusion covariates, and the relevant covariates in the noise are automatically selected by Bayesian variable selection. We compare the estimated diffusion tensors from our model to a homoscedastic counterpart with no covariates in the noise, and to commonly used linear and nonlinear least squares methods. The estimated single-diffusion tensors within each voxel are compared with respect to fractional anisotropy (FA) and mean diffusivity (MD). Using data from the Human Connectome Project, our results show that the noise is clearly heteroscedastic, especially the posterior variance for MD is substantially underestimated by the homoscedastic model, and inferences from the homoscedastic model are on average spuriously precise. Inferences from commonly used ordinary and weighted least squares methods (OLS and WLS) show that it is not adequate to estimate the single-diffusion tensor from logarithmic measurements.

B. Wegmann (✉) • M. Villani
Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, SE-581 83 Linkoping, Sweden
e-mail: bertil.wegmann@liu.se; mattias.villani@liu.se

A. Eklund
Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, SE-581 83 Linkoping, Sweden

Division of Medical Informatics, Department of Biomedical Engineering, Linköping University, SE-581 85 Linkoping, Sweden
e-mail: anders.eklund@liu.se

Center for Medical Image Science and Visualization (CMIV), Linköping University, Linkoping, Sweden

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*, Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_11

1 Introduction

Diffusion weighted imaging is a popular modality for studying structural brain connectivity, as it can non-invasively measure how easily water can diffuse along different directions; see [17] for a collection of very accessible papers in diffusion MRI. While recent work (e.g. [32]) is focused on improving and extending the rather simple diffusion tensor model, diffusion tensor imaging (DTI) is still the most common choice for studies looking at differences between healthy controls and subjects with some disease [9, 21, 24]; the TBSS approach (tract-based spatial statistics) [26] for voxel-wise inference of fractional anisotropy (FA) has since 2006 received more than 2800 citations, with about 500 citations in 2015. A drawback of the TBSS approach is that it ignores the uncertainty of the FA for each subject, which is in contrast to functional magnetic resonance imaging (fMRI) where heteroscedastic group analyses are common [5, 8]. For both fMRI and DTI, the assumption of homoscedasticity (constant variance over samples) at the single subject level has been overlooked, although weighted least squares is rather common for DTI [4].

There are several reasons why diffusion data can be heteroscedastic. One of the most obvious reasons is that the signal to noise ratio (SNR) is in general lower with a higher b-value, and multi-shell diffusion data are therefore heteroscedastic. For single-shell diffusion data, heteroscedasticity can be introduced by taking the logarithm of the measurements before fitting a tensor [4]. Head motion is another potential source of heteroscedasticity in diffusion data. The analysis in [11] shows that motion spikes can give rise to heteroscedasticity in fMRI data, even after correcting for head motion and using motion covariates (and their temporal derivative) to capture residual motion effects. Another source of unequal noise variances is model misspecification, e.g. deviations from mono-exponential decay in the MR signal. The commonly used Gaussian approximation to the underlying Rician distributed signal in diffusion MR may also give rise to heteroscedastic data. This explanation is supported by the comparison between a Gaussian and Rician DTI model in [31], where covariates in the variance are much more frequent in the Gaussian model than in the Rician model.

We therefore propose a Bayesian heteroscedastic regression approach, and a highly efficient Markov Chain Monte Carlo (MCMC) algorithm, to estimate a single-diffusion tensor model. Our model is very flexible as it can include any covariate to model the noise variance, and it automatically selects the relevant covariates by Bayesian variable selection. While the common weighted least squares approach uses the signal intensity as weights, our algorithm instead estimates optimal weights from the data. The Bayesian approach also has the obvious advantage of capturing the full uncertainty of the parameters through the posterior distribution. The subject and voxel specific uncertainty of the FA can thereby easily be propagated to the group analysis.

Compared to the early work by Behrens et al. [6], our regression approach is more general, as it can include covariates for both the mean and the variance, and

uses a logarithmic link function instead of taking the logarithm of the measurements. Instead of using a maximum a posteriori approach [1], which only provides a point estimate of the posterior, our MCMC approach gives the full posterior distribution of the diffusion tensor. We use the single tensor model in this chapter, but our methodology applies to any diffusion model that can be expressed as regression model with a link function. One such model is the regression model for q-space trajectory imaging used in [32] to estimate a diffusion tensor and a fourth order covariance matrix in every voxel.

It is crucial to have a very efficient MCMC algorithm for diffusion tensor imaging since the algorithm needs to be iterated some 1000–10,000 times for each of several thousand voxels. Our proposed MCMC algorithm generates an effective number of posterior draws per minute which often is tens or hundreds times larger than for commonly used algorithms in DTI, see [31] for a comparison, with the additional benefit of performing variable selection in the same run.

Using data from the Human Connectome Project [13], our results show that the commonly used ordinary and weighted least squares methods greatly underestimate FA and MD, and are therefore not adequate for estimating the single-diffusion tensor. The main reason for this is that these methods take the logarithm of the measurements, instead of using the logarithmic link function dictated by the theoretical single-diffusion tensor model. Our results also clearly show that DTI noise is heteroscedastic, and inferences under the homoscedastic special case of our heteroscedastic DTI model are on average similar, although the variance for especially MD is substantially underestimated by the homoscedastic DTI model. This reflects well-known results that homoscedastic models give unbiased estimates, but underestimate standard errors when the data are heteroscedastic. The variability in the posterior distributions of FA and MD is in our case small, which is partly due to our use of DWI data with many gradient directions for the different b-values.

2 The Heteroscedastic Diffusion Tensor Model

We first present the homoscedastic DTI model for the single-diffusion tensor in Sect. 2.1, extend this model to our heteroscedastic DTI model for the noise variance in Sect. 2.2 and then present our prior distributions in Sect. 2.3.

2.1 The Homoscedastic DTI Model

The DTI model assumes that the noise-free signal μ_i for measurement i is of the form

$$\mu_i = \mu_0 \exp\left(-b_i \mathbf{g}_i^T \mathbf{D} \mathbf{g}_i\right), \tag{1}$$

where μ_0 is the signal in absence of any diffusion, b_i is the b-value and $\mathbf{g}_i = (g_{ix}, g_{iy}, g_{iz})^T$ is the gradient vector for the ith measurement, and

$$\mathbf{D} = \begin{pmatrix} d_{xx} & d_{xy} & d_{xz} \\ d_{xy} & d_{yy} & d_{yz} \\ d_{xz} & d_{yz} & d_{zz} \end{pmatrix}$$

is the diffusion tensor. The signal is theoretically observed in Rician noise [15], but is approximated by a Gaussian distribution in most applications. The models proposed here can be estimated with Rician noise (see [31]), but we will use Gaussian noise here for two main reasons. First, the analysis of heteroscedasticity in DTI noise is more transparent when the Gaussian distribution is used since it is explicitly parametrized by a mean and variance. Second, we can directly compare to the estimation methods traditionally used for the DTI model (see Sect. 3.1), which assume Gaussian noise, or are known to be optimal under such an assumption.

The diffusion tensor model in Eq. (1) with homoscedastic Gaussian noise can be written as a generalized linear regression model with a logarithmic link function, according to

$$\begin{aligned} y_i|\mathbf{x}_i &\sim N(\mu_i, \sigma^2) \\ \ln \mu_i &= \beta_0 + \mathbf{x}_i^T \beta, \end{aligned} \tag{2}$$

for $i = 1, \ldots, n$, where $\beta_0 = \ln \mu_0$,

$$\beta = \left(d_{xx}, d_{yy}, d_{zz}, d_{xy}, d_{yz}, d_{xz}\right),^T$$

and

$$\mathbf{x}_i^T = -\left(b_i g_{ix}^2, b g_{iy}^2, b_i g_{iz}^2, 2b_i g_{ix} g_{iy}, 2b_i g_{iy} g_{iz}, 2b_i g_{ix} g_{iz}\right). \tag{3}$$

However, since the tensor $\mathbf{D}$ is required to be positive definite, the parameter space of β in Eq. (2) is restricted. One can impose the positive definiteness restriction explicitly by a reparametrization of the model parameters. We will here use the log-Cholesky representation of the diffusion tensor [18]

$$\mathbf{D}(\omega) = \Omega^T \Omega,$$

where

$$\Omega = \begin{pmatrix} e^{\omega_1} & \omega_4 & \omega_6 \\ 0 & e^{\omega_2} & \omega_5 \\ 0 & 0 & e^{\omega_3} \end{pmatrix},$$

and $\omega = (\omega_1, \ldots, \omega_6) \in \mathbb{R}^6$ are the unrestricted parameters. In this parametrization

$$\mathbf{D}(\omega) = \begin{pmatrix} e^{2\omega_1} & \omega_4 e^{\omega_1} & \omega_6 e^{\omega_1} \\ \omega_4 e^{\omega_1} & \omega_4^2 + e^{2\omega_2} & \omega_4\omega_6 + \omega_5 e^{\omega_2} \\ \omega_6 e^{\omega_1} & \omega_4\omega_6 + \omega_5 e^{\omega_2} & \omega_6^2 + \omega_5^2 + e^{2\omega_3} \end{pmatrix},$$

so that the vector of regression coefficients $\beta(\omega)$ in Eq. (2) is given by

$$\beta(\omega) = (e^{2\omega_1}, \omega_4^2 + e^{2\omega_2}, \omega_6^2 + \omega_5^2 + e^{2\omega_3}, \omega_4 e^{\omega_1}, \omega_4\omega_6 + \omega_5 e^{\omega_2}, \omega_6 e^{\omega_1}). \tag{4}$$

We demonstrate empirically in Sect. 5.2 that the log-Cholesky parametrization has the additional advantage of making the likelihood function more regular and Gaussian compared to the unrestricted parametrization.

2.2 The Heteroscedastic DTI Model

The model in Eq. (2) implicitly assumes that the noise variance σ^2 is constant over all observations. The following heteroscedastic DTI model drops this assumption, and lets the noise variance be driven by a set of explanatory variables, according to

$$\begin{aligned} y_i | x_i, z_i &\sim N\left(\mu_i, \sigma_i^2\right) \\ \ln \mu_i &= \beta_0 + \mathbf{x}_i^T \beta, \\ \ln \sigma_i^2 &= \gamma_0 + \mathbf{z}_i^T \gamma, \end{aligned} \tag{5}$$

for $i = 1, \ldots, n$, where β_0, β and $\mathbf{x}$ are defined as in the homoscedastic model in Eq. (2) and $\mathbf{z}$ may include all of the covariates in $\mathbf{x}$ and other covariates which may be useful for modeling the heteroscedasticity. The variable selection presented in Sect. 2.3 will automatically select the relevant covariates in $\mathbf{z}$.

2.3 Prior Distribution and Variable Selection

2.3.1 Priors on the Regression Coefficients

We analyze both the homoscedastic and heteroscedastic DTI models using a Bayesian approach, which formulates a prior distribution for all model parameters and then updates this prior distribution with observed data. We choose non-informative priors for both models, such that the amount of prior information does not affect comparison between the models. We assume the priors for the intercepts $\beta_0 \sim N(m_\beta, c)$ and $\gamma_0 \sim N(m_\gamma, c)$, independently of the priors for the regression coefficients $\omega \sim N(0, cI)$ and $\gamma \sim N(0, cI)$, where $c = 100$ to induce non-

informative priors and I is the identity matrix. Note that the prior expected value of 0 for γ implies that the heteroscedastic DTI model is centered on the homoscedastic case a priori. The models for μ and σ^2 in Eqs. (2) and (5) become $\beta_0 = \ln \mu_i$ and $\gamma_0 = \ln \sigma_i^2$ when $b = 0$. Therefore, we set the prior expected values, m_β and m_γ, for β_0 and γ_0 in each voxel i by taking the logarithm of the mean and variance of y_i when $b = 0$, respectively.

2.3.2 Variable Selection Priors

Our MCMC algorithm can perform Bayesian variable selection among both sets of covariates (i.e. $\mathbf{x}$ and $\mathbf{z}$). In the empirical results, however, we only apply variable selection to $\mathbf{z}$ for two reasons. First, in our example, we only include the tensor covariates (Eq. (3)) in $\mathbf{x}$, and excluding covariates with variable selection does not make much sense there. Second, in the log-Cholesky parametrization, variable selection amounts to zero-restrictions on the ω, which have no clear interpretation. We nevertheless present our algorithm for the case where we allow for variable selection in both $\mathbf{x}$ and $\mathbf{z}$, since other applications of our model may involve additional covariates in $\mathbf{x}$ (e.g. head motion covariates) where variable selection can be very useful.

The intercepts in $\ln \mu$ and $\ln \sigma^2$ are always assumed to be included in the model. We focus here on an unrestricted β in the equation for μ (exactly the same general algorithm is used for variable selection in the noise variance σ^2). Define the vector with binary indicators $\mathcal{I} = \{I_1, \ldots I_p\}$ such that $I_j = 0$ means that the jth element in β is zero and that the corresponding covariate drops out of the model. Let $\mathcal{I}^c$ denote the complement of $\mathcal{I}$. Let $\beta_\mathcal{I}$ denote the subset of regression coefficients selected by $\mathcal{I}$. To allow for variable selection we take the prior $\beta \sim N(0, c\Sigma)$ on the model with all covariates, and condition on the zeros dictated by $\mathcal{I}$:

$$\beta_\mathcal{I} | \mathcal{I} \sim N\left[0, c(\Sigma_{\mathcal{I},\mathcal{I}} - \Sigma_{\mathcal{I},\mathcal{I}^c} \Sigma_{\mathcal{I}^c,\mathcal{I}^c}^{-1} \Sigma_{\mathcal{I}^c,\mathcal{I}}^T)\right],$$

and $\beta_{\mathcal{I}^c} | \mathcal{I}$ is identically zero. In our special case here with $\Sigma = I$, this conditional prior simplifies to $\beta_\mathcal{I} | \mathcal{I} \sim N\left(0, cI_{|\mathcal{I}|}\right)$. To complete the variable selection prior we let the elements of $\mathcal{I}$ to be a priori independent and Bernoulli distributed, i.e $\Pr(I_i = 1) = \pi$, and π is allowed to be different for the covariates in μ and σ^2. Other priors on $\mathcal{I}$ are just as easily handled, and one can easily also use a prior on π and estimate it from the data.

3 Inference Methods

Common least squares estimation methods in DTI are presented in Sect. 3.1 and our Bayesian approach with a highly efficient Markov Chain Monte Carlo (MCMC) algorithm for the heteroscedastic DTI model in Eq. (5) is outlined in Sect. 3.2.

3.1 Traditional Estimation Methods

One of the most common estimation method for the homoscedastic DTI model is to use *Ordinary Least Squares* (*OLS*) fitting on the log responses [3], $\ln y_i$,

$$\hat{\beta}_{OLS} = \left(\mathbf{X}^T\mathbf{X}\right)^{-1}\mathbf{X}^T \ln \mathbf{y},$$

where $\mathbf{y} = (y_i)_{i=1}^n$, $\mathbf{X} = (\mathbf{x}_i^T)_{i=1}^n$. However, since OLS is only technically correct under the assumption of log normally distributed homoscedastic multiplicative errors, the use of OLS for DTI is likely to be suboptimal. It is possible to improve on OLS by using *Weighted Least Squares* (*WLS*), according to [4]

$$\hat{\beta}_{WLS} = \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{W} \ln \mathbf{y},$$

where $\mathbf{W} = Diag\left(y_1^2, \ldots, y_n^2\right)$ is the weight matrix; see [18] for a motivation for using the squared signals y_i^2 as weights. While OLS and WLS are convenient closed form estimators with great practical appeal, it is actually straight-forward to correctly estimate the homoscedastic DTI model by using the iteratively reweighted least squares (IRLS) algorithm (see for example [27]) typically used to estimate generalized linear models.

We have so far assumed that the tensor parameters are unrestricted, in which case the implied estimated tensor from OLS, WLS and IRLS are not guaranteed to be positive definite. Imposing the positive definiteness restriction by parametrizing β using the log-Cholesky representation makes the OLS, WLS and IRLS optimization problem for the unrestricted parameters ω non-linear, and the solutions are no longer available in closed form. Using Nonlinear Least Squares (NLS), Koay [18] develops more general methods that also provide the uncertainty of any derived quantity [19, 20]. It is, however, not clear how to extend these least squares methods to the heteroscedastic case. We therefore turn to a Bayesian approach via MCMC.

3.2 Bayesian Inference Using MCMC Sampling with Variable Selection

The aim of a Bayesian analysis is the joint posterior distribution of all model parameters

$$p(\beta, \gamma|\mathbf{y}, \mathbf{X}, \mathbf{Z}) \propto p(\mathbf{y}|\beta, \gamma, \mathbf{X}, \mathbf{Z})p(\beta, \gamma),$$

where $\mathbf{y} = (y_i)_{i=1}^n$, $\mathbf{X} = (\mathbf{x}_i)_{i=1}^n$, $\mathbf{Z} = (\mathbf{z}_i)_{i=1}^n$, $p(\mathbf{y}|\beta, \gamma, \mathbf{X}, \mathbf{Z})$ is the likelihood function for the DWI measurements in the n-dimensional vector $\mathbf{y}$, and $p(\beta, \gamma)$ is the prior. If the log-Cholesky parametrization is used, β is replaced by ω. The joint posterior $p(\beta, \gamma|\mathbf{y}, \mathbf{X}, \mathbf{Z})$ is intractable and we instead simulate from the joint

posterior using an efficient MCMC algorithm given below. The simulated draws from the posterior $p(\beta, \gamma|\mathbf{y}, \mathbf{X}, \mathbf{Z})$ can be used to easily compute the posterior distribution of the tensor $\mathbf{D}$ and any derived measure such as fractional anisotropy (FA), or even the joint posterior distribution of any number of derived measures. The posterior distribution for $\mathbf{D}$ provides a very natural way to quantify the uncertainty, and draws from it can be used as input to tractography algorithms [7].

We use a Metropolis-within-Gibbs algorithm to simulate from the joint posterior distribution. Our posterior sampling algorithm is based on a general framework presented in [28] and [29], which we here extend to the log-Cholesky parametrization. We will outline the major steps in the algorithm here, and refer the reader to a more detailed description in [29] and in [31], where we apply the same framework to the Rician model for fMRI and DTI. Our proposed Metropolis-within-Gibbs algorithm samples iteratively from the following pair of full conditional posteriors:

1. $p\left(\beta, \mathcal{I}_\beta|\gamma, \mathcal{I}_\gamma, \mathbf{y}, \mathbf{X}, \mathbf{Z}\right)$,
2. $p\left(\gamma, \mathcal{I}_\gamma|\beta, \mathcal{I}_\beta, \mathbf{y}, \mathbf{X}, \mathbf{Z}\right)$.

Note that we sample β and $\mathcal{I}_\beta$ jointly given the other parameters. The full conditional posteriors $p(\beta, \mathcal{I}_\beta|\cdot)$ and $p(\gamma, \mathcal{I}_\gamma|\cdot)$ are highly non-standard distributions, but can be efficiently sampled using Metropolis-Hastings (MH) updates. The sampling of the pair $(\gamma, \mathcal{I}_\gamma)$ is analogous to the sampling of $(\beta, \mathcal{I}_\beta)$, so we will only describe the update of $(\beta, \mathcal{I}_\beta)$. The conditional posterior of β is simply denoted by $p(\beta|\mathcal{I}, \mathbf{y}, \mathbf{X})$ in the remainder of this section. The MH proposal distribution is of the form

$$J(\beta_p, \mathcal{I}_p|\beta_c, \mathcal{I}_c) = J_1(\beta_p|\mathcal{I}_p, \beta_c)J_2(\mathcal{I}_p|\beta_c, \mathcal{I}_c), \tag{6}$$

where $(\beta_c, \mathcal{I}_c)$ denotes the current and $(\beta_p, \mathcal{I}_p)$ the proposed posterior draw. Following [28] and [29], we choose J_2 to be a simple proposal of $\mathcal{I}$ where a subset of the indicators is randomly selected and a change of the selected indicators is proposed, one variable at a time. The proposal of β, the J_1 distribution, is a tailored multivariate-t distribution with ν degrees of freedom:

$$\beta_p|\mathcal{I}_p, \beta_c \sim t_\nu \left[\hat{\beta}, -\left(\frac{\partial^2 \log p(\beta|\mathcal{I}, \mathbf{y}, \mathbf{X})}{\partial\beta\partial\beta^T}\right)^{-1}\Bigg|_{\beta=\hat{\beta}}\right],$$

where $\log p(\beta|\mathcal{I}, \mathbf{y}, \mathbf{X})$ denotes the (conditional) posterior density, $\hat{\beta}$ is the terminal point of a small number of Newton iterations to climb towards the mode of the full conditional $p(\beta_p|\mathcal{I}_p, \cdot)$, and $-\left(\frac{\partial^2 \log p(\beta|\mathcal{I}, \mathbf{y}, \mathbf{X})}{\partial\beta\partial\beta^T}\right)^{-1}\Big|_{\beta=\hat{\beta}}$ is the negative inverse Hessian of the full conditional posterior evaluated at $\beta = \hat{\beta}$. There are a number of different aspects of these Newton-based proposals. First, the number of Newton iterations can be kept very small (one or two steps is often enough), since each iteration always starts at β_c, which is typically not far from the mode. Second,

$\hat{\beta}$ is often not exactly the mode, but the posterior draws from the algorithm will nevertheless converge to the target posterior. Third, the update $(\beta_c, \mathcal{I}_c) \rightarrow (\beta_p, \mathcal{I}_p)$ is accepted with probability

$$\min\left(1, \frac{p(\mathbf{y}|\beta_p, \mathcal{I}_p, \mathbf{X})p(\beta_p|\mathcal{I}_p)p(\mathcal{I}_p)/J_1(\beta_p|\mathcal{I}_p, \beta_c)J_2(\mathcal{I}_p|\beta_c, \mathcal{I}_c)}{p(\mathbf{y}|\beta_c, \mathcal{I}_c, \mathbf{X})p(\beta_c|\mathcal{I}_c)p(\mathcal{I}_c)/J_1(\beta_c|\mathcal{I}_c, \beta_p)J_2(\mathcal{I}_c|\beta_p, \mathcal{I}_p)}\right),$$

where the factor $J_1(\beta_c|\mathcal{I}_c, \beta_p)$ is again computed by Newton iterations, this time starting from the proposed point β_p. Fourth, to implement the Newton iterations we need to be able to compute the gradient $\frac{\partial \log p(\mathbf{y}|\mathcal{I},\beta,\mathbf{X})}{\partial\beta}$ and the Hessian $\frac{\partial^2 \log p(\beta|\mathcal{I},\mathbf{y},\mathbf{X})}{\partial\beta\partial\beta^T}$ efficiently. Villani et al. [29] derive the gradient and the Hessian for a general posterior of the form

$$p(\beta|\mathcal{I}, \mathbf{y}, \mathbf{X}) \propto \prod_{i=1}^{n} p(y_i|\phi_i, \mathbf{x}_i)p(\beta), \tag{7}$$

where $k(\phi_i) = \mathbf{x}_i'\beta$ is a smooth link function, and $\mathbf{x}_i$ is a covariate vector for the ith observation; the full conditional posteriors for β and γ in the heteroscedastic DTI model with log links are clearly of this form. The gradient of the likelihood in Eq. (7) can be expressed as

$$\frac{\partial \ln p(\mathbf{y}|\mathcal{I}, \beta, \mathbf{X})}{\partial\beta} = \mathbf{X}^T\mathbf{g}, \tag{8}$$

where $\mathbf{g} = (g_1, \ldots, g_n)^T$, and

$$g_i = \frac{\partial \ln p(y_i|\phi_i, \mathbf{x}_i)}{\partial\phi_i}\left[k'(\phi_i)\right]^{-1}.$$

The Hessian of the likelihood is

$$\frac{\partial^2 \ln p(\mathbf{y}|\mathcal{I}, \beta, \mathbf{X})}{\partial\beta\partial\beta'} = X^T\left(A_1 + A_2\right)X, \tag{9}$$

where $A_1 = \mathrm{Diag}(a_{1i})$, $A_2 = \mathrm{Diag}(a_{2i})$,

$$a_{1i} = \frac{\partial^2 \ln p(y_i|\phi_i, \mathbf{x}_i)}{\partial\phi_i^2}\left[k'(\phi_i)\right]^{-2},$$

and

$$a_{2i} = -\frac{\partial \ln p(y_i|\phi_i, \mathbf{x}_i)}{\partial\phi_i}k''[k'(\phi_i)^{-1}]k'(\phi_i)^{-2}.$$

It is very important to note that the posterior gradient and the Hessian only require derivatives for the scalar parameters of the log-likelihood, i.e. $\partial \ln p(y_i|\phi_i, \mathbf{x}_i)/\partial\phi_i$ and $\partial^2 \ln p(y_i|\phi_i, \mathbf{x}_i)/\partial\phi_i^2$, which are straightforward for the heteroscedastic DTI model, see [29]. When the log-Cholesky parametrization is used, the mapping from ω to β is nonlinear, and the gradient in Eq. (8) is modified to

$$\frac{\partial \ln p(\mathbf{y}|\mathcal{I}, \omega, \mathbf{X})}{\partial\omega} = \left(\mathbf{X}\frac{\partial\beta(\omega)}{\partial\omega}\right)^T \mathbf{g},$$

where

$$\frac{\partial\beta(\omega)}{\partial\omega} = \begin{pmatrix} 2e^{2\omega_1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 2e^{2\omega_2} & 0 & 2\omega_4 & 0 & 0 \\ 0 & 0 & 2e^{2\omega_3} & 0 & 2\omega_5 & 2\omega_6 \\ \omega_4 e^{\omega_1} & 0 & 0 & e^{\omega_1} & 0 & 0 \\ 0 & \omega_5 e^{\omega_2} & 0 & \omega_6 & e^{\omega_2} & \omega_4 \\ \omega_6 e^{\omega_1} & 0 & 0 & 0 & 0 & e^{\omega_1} \end{pmatrix}.$$

The Hessian in Eq. (9) is modified in the same fashion.

In summary, our proposed MCMC algorithm consists of a two-block Metropolis-Hastings within Gibbs sampler, where each step updates a set of regression coefficients simultaneously with their binary variable selection indicators. The multivariate student-t proposal is tailored to the full conditional posterior at each step, using a generalized Newton method to approximate the conditional posterior mode and curvature (Hessian). The gradient and the Hessian for the Newton steps can be computed very efficiently in compact matrix form. Only a very small number of Newton steps is needed, since each iteration starts at the previously accepted parameter draw which is typically an excellent starting value.

4 Data

We use the freely available MGH adult diffusion dataset from the Human Connectome Project (HCP) [13, 23][1] to test our algorithms. The dataset comprises DWI data collected with several different b-values, and the downloaded data have already been corrected for gradient nonlinearities, subject motion and eddy currents [2, 14]. The DWI data were collected using a spin-echo EPI sequence and a 64-channel array coil, yielding volumes of $140 \times 140 \times 96$ voxels with an isotropic voxel size of 1.5 mm. The data collection was divided into 5 runs, giving data with four different b-values: 1000, 3000, 5000, and 10,000 s/mm^2. However, we do not use the data collected with a b-value of 10,000 s/mm^2 in any analysis, because it is unusual in

[1] http://www.humanconnectome.org/documentation/MGH-diffusion/.

DTI analysis. The number of gradient directions was 64 for $b = 1000\,\text{s/mm}^2$ and $b = 3000\,\text{s/mm}^2$, and 128 for $b = 5000\,\text{s/mm}^2$.

Prior to any statistical analysis, the function FAST in FSL was used to generate a mask of white brain matter, gray brain matter and cerebrospinal fluid (CSF). In general, voxels in CSF do not have a mono-exponential signal decay, which is assumed by the DTI models with a log-link function. Therefore, we find it often very problematic to estimate the parameters in CSF voxels for the log-link models NLS and the homoscedastic and heteroscedastic DTI models. In addition, it is not of interest to estimate the diffusion parameters in CSF. Thus, we avoid running the analysis on voxels in CSF.

Data used in the preparation of this work were obtained from the Human Connectome Project (HCP) database (https://ida.loni.usc.edu/login.jsp). The HCP project (Principal Investigators: Bruce Rosen, M.D., Ph.D., Martinos Center at Massachusetts General Hospital; Arthur W. Toga, Ph.D., University of Southern California, Van J. Weeden, MD, Martinos Center at Massachusetts General Hospital) is supported by the National Institute of Dental and Craniofacial Research (NIDCR), the National Institute of Mental Health (NIMH) and the National Institute of Neurological Disorders and Stroke (NINDS). HCP is the result of efforts of co-investigators from the University of Southern California, Martinos Center for Biomedical Imaging at Massachusetts General Hospital (MGH), Washington University, and the University of Minnesota.

5 Results

The differences in estimation results between our heteroscedastic DTI model and the other models are discussed in Sect. 5.1 and the more Gaussian posterior from the log-Cholesky parametrization is illustrated in Sect. 5.2.

5.1 The Effect of Allowing for Heteroscedasticity and Comparison to Traditional Estimation Methods

We first compare our heteroscedastic DTI model in Eq. (5) to the homoscedastic counterpart in Eq. (2), using the whole dataset with all b-values up to $b = 5000\,\text{s/mm}^2$. Figure 1 shows that our MCMC algorithm exhibits excellent convergence with high acceptance probabilities for μ and σ^2. Figure 2 shows posterior inclusion probabilities for the explanatory variables $\mathbf{z}$ in the noise variance in Eq. (5). In many voxels the inclusion probabilities for the covariates corresponding to (d_{xx}, d_{yy}, d_{zz}) are close or equal to 1, while the inclusion probabilities for the remaining covariates in $\mathbf{z}$ are mostly very close to zero. This clearly shows that

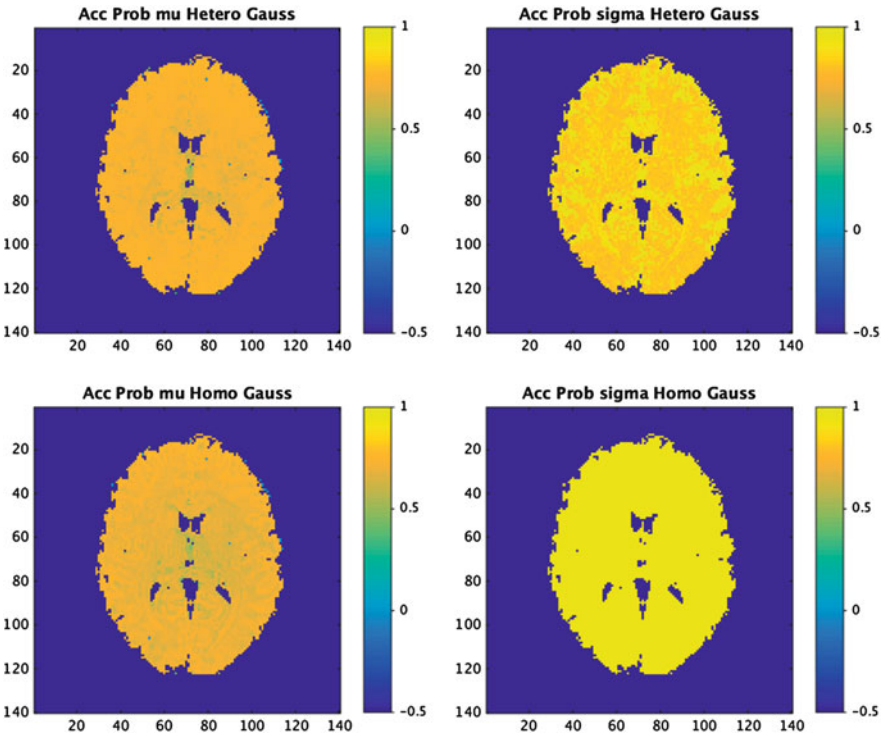


Fig. 1 Acceptance probabilities for μ and σ^2 in our MCMC of the heteroscedastic and homoscedastic DTI models

diffusion covariates affect the noise variance, and results can be distorted by not allowing for a heteroscedastic noise variance.

We compare the estimated single-diffusion tensors across voxels for the heteroscedastic DTI model in Eq. (5) to the homoscedastic counterpart in Eq. (2), and the traditional estimation methods OLS, WLS and NLS in Sect. 3.1 with respect to the DTI scalar measures mean diffusivity (MD) and fractional anisotropy (FA). FA and MD are both functions of the eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3$ of the single-diffusion tensor:

$$MD = \frac{\lambda_1 + \lambda_2 + \lambda_3}{3}, \; FA = \sqrt{\frac{3}{2}} \frac{\sqrt{\sum_{i=1}^{3} (\lambda_i - MD)^2}}{\sqrt{\sum_{i=1}^{3} \lambda_i^2}}.$$

Using the whole dataset with all b-values up to 5000 s/mm^2, Fig. 3 shows the estimated values of FA, for OLS, WLS and NLS and the posterior mean from the heteroscedastic and homoscedastic DTI models. The differences in FA between the heteroscedastic and homoscedastic DTI models and NLS are small, but compared to

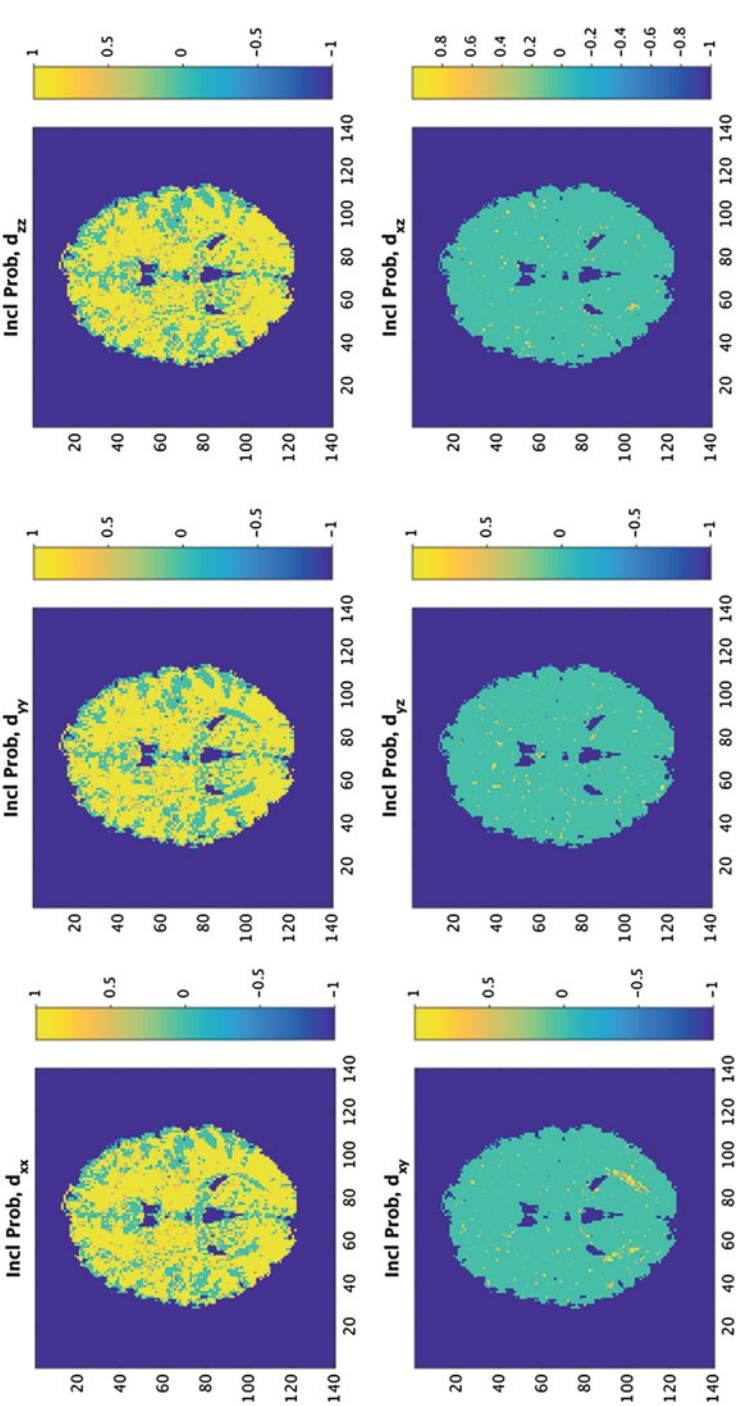


Fig. 2 Posterior inclusion probabilities for the explanatory variables z corresponding to the diffusion directions $\left(d_{xx}, d_{yy}, d_{zz}, d_{xy}, d_{yz}, d_{xz}\right)$ in the noise variance in Eq. (5). Note that the inclusion probability is close to 1 for a large number of voxels for the diagonal elements $\left(d_{xx}, d_{yy}, d_{zz}\right)$

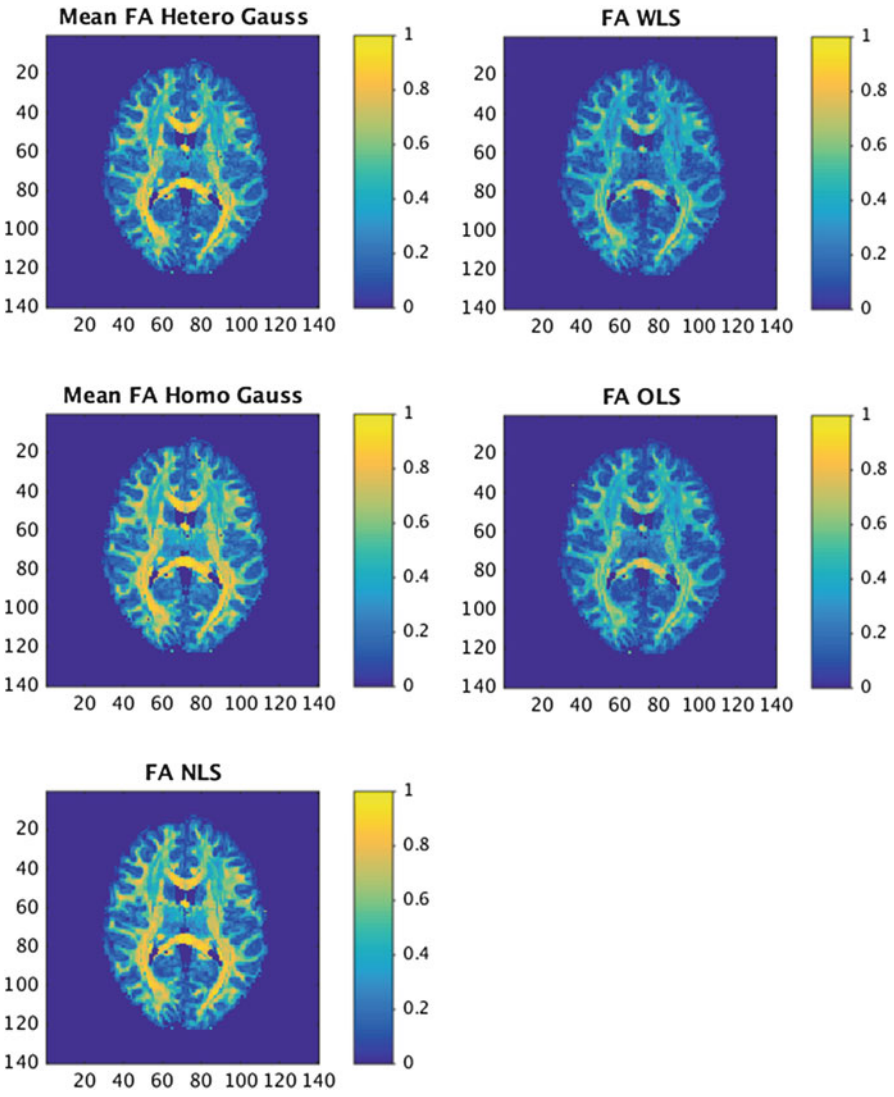


Fig. 3 Posterior mean estimates of FA for the heteroscedastic and homoscedastic DTI models and point estimates for OLS, WLS and NLS, using the whole dataset with all b-values up to $b = 5000\,\text{s/mm}^2$

WLS and OLS the estimated values differ considerably, especially in white matter regions with high FA values. This suggests that it is not adequate to use simple OLS and WLS methods on the logarithm of the signal. Figure 4 shows the ratios of the estimated FA between the heteroscedastic DTI model and the other models.

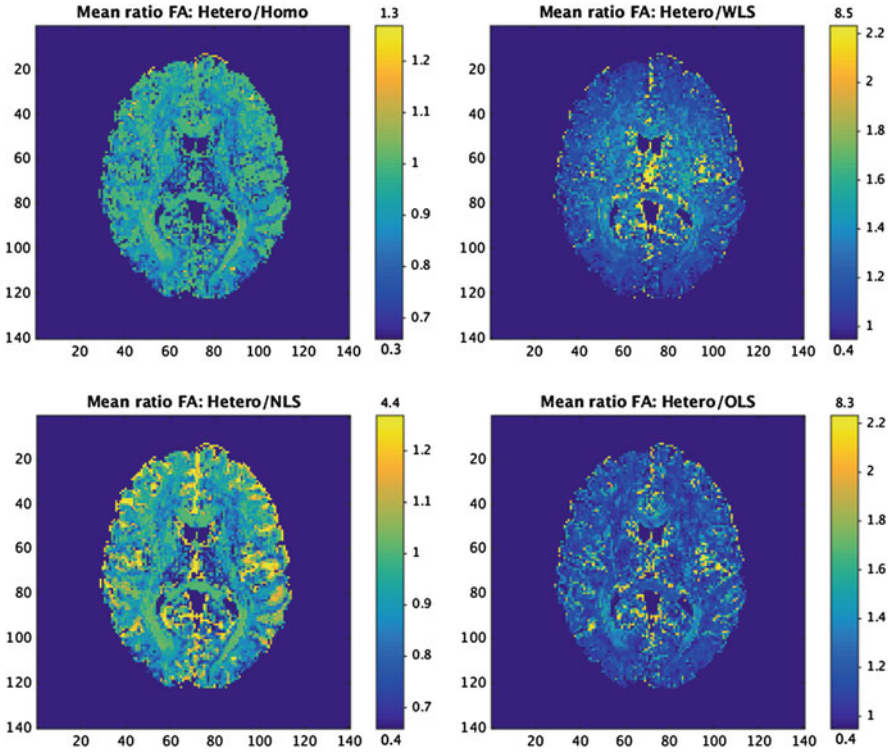


Fig. 4 The ratio between the posterior means of FA for the heteroscedastic DTI model to the homoscedastic counterpart and to the point estimates for OLS, WLS and NLS, using the whole dataset with all b-values up to $b = 5000\,\text{s/mm}^2$. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

Figure 5 shows that there are some small differences in MD among the three models with log link, but even larger differences between the models with and without log link. Figure 6 shows the ratios of the estimated MD between the heteroscedastic DTI model and the other models.

The small differences in FA and MD between our heteroscedastic DTI model to NLS are mainly due to our more general approach where the relevant explanatory variables in the noise variance are automatically selected by Bayesian variable selection. However, our Bayesian approach is also able to quantify the full uncertainty in FA and MD from the estimated posterior distribution compared to NLS, which further explains the differences between the models.

Figures 7 and 8 also show ratios of estimated values of FA and MD as in Figs. 4 and 6, but only for the subset of the dataset with b-values up to $b = 3000\,\text{s/mm}^2$. The differences between the models are smaller for this subset of the data, and when

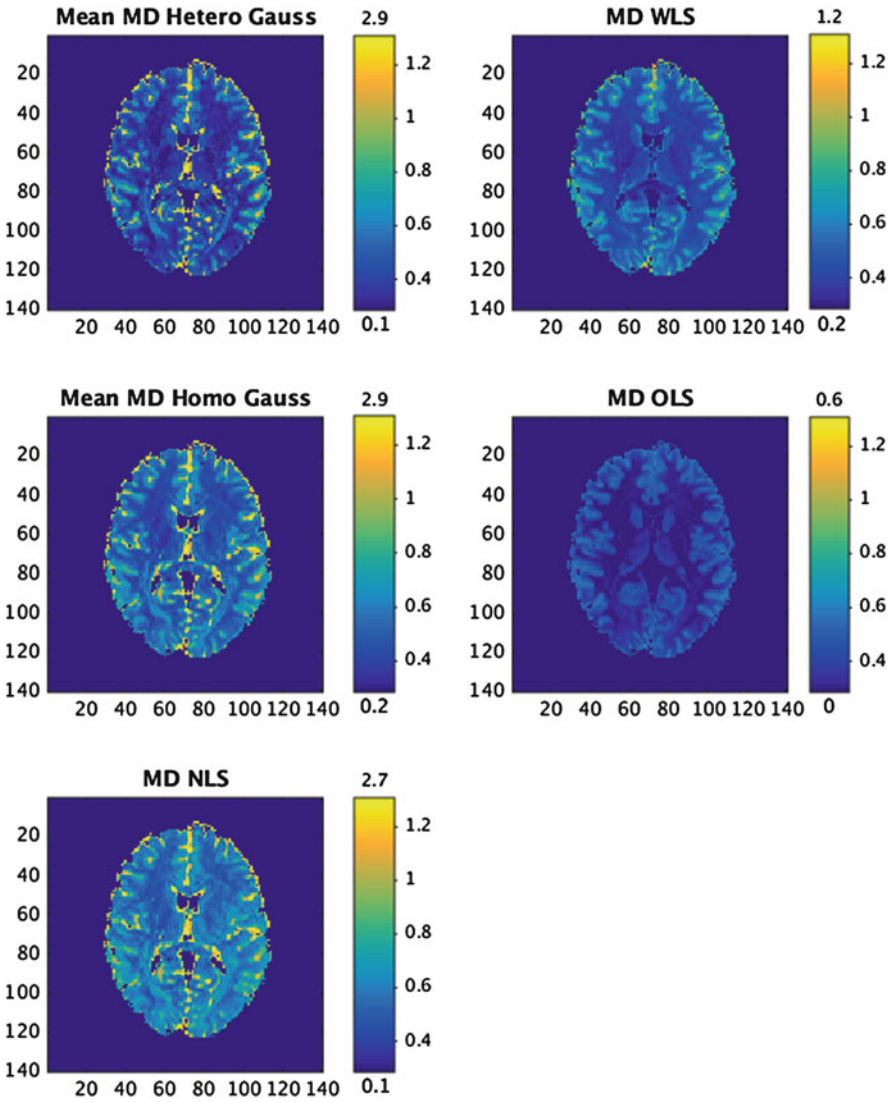


Fig. 5 Posterior mean estimates of MD for the heteroscedastic and homoscedastic DTI models and point estimates for OLS, WLS and NLS, using the whole dataset with all b-values up to $b = 5000\,\mathrm{s/mm^2}$. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

comparing the estimates for an even smaller subset with b-value $b = 1000\,\mathrm{s/mm^2}$, the differences between the models are negligible (not shown here).

Figure 9 shows the posterior standard deviations of FA and MD for the heteroscedastic and homoscedastic DTI models, and the ratio of the standard deviations between the models. The standard deviations are small for both models. However,

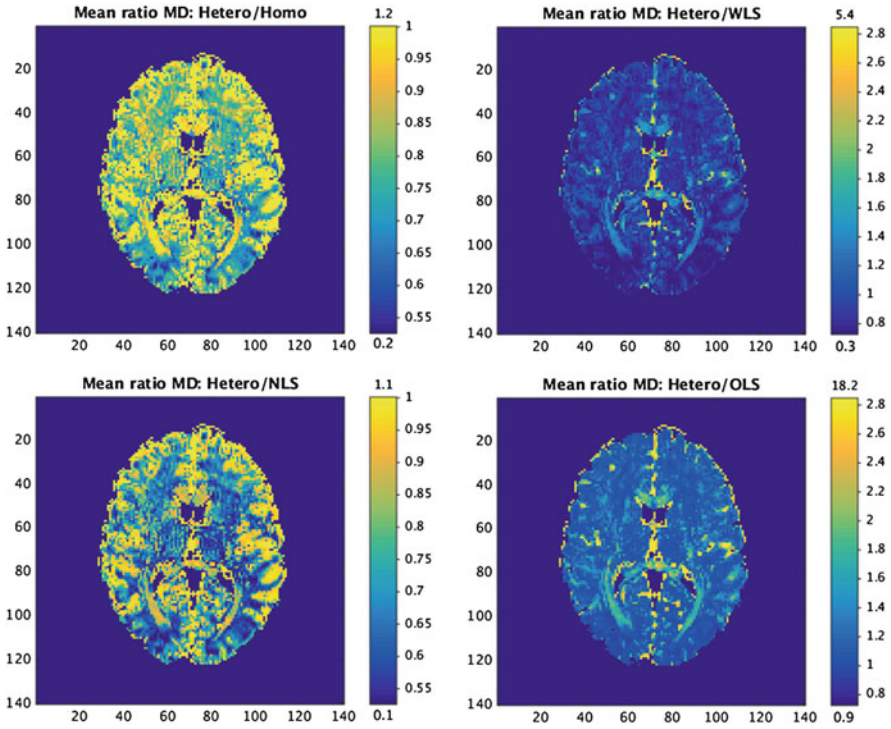


Fig. 6 The ratio between the posterior means of MD for the heteroscedastic DTI model to the homoscedastic counterpart and to the point estimates for OLS, WLS and NLS, using the whole dataset with all b-values up to $b = 5000\,\text{s/mm}^2$. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

the standard deviations for the heteroscedastic DTI model vary more across voxels than for the homoscedastic counterpart, especially for MD. In addition, there is a substantial amount of voxels where the standard deviations of MD are much higher for the heteroscedastic DTI model compared to the homoscedastic DTI model. This reflects well-known results that standard errors for homoscedastic models are known to be too small for data with heteroscedastic noise variance. The standard deviations for the part of the dataset with b-values up to $b = 3000\,\text{s/mm}^2$ are similar to the ones in Fig. 9 for the whole dataset, but the standard deviations for the even smaller subset of data with b-value $b = 1000\,\text{s/mm}^2$ are substantially larger, and the differences in standard deviations between the models are substantially lower (not shown here).

The small standard deviations in Fig. 9 are partly due to using a large amount of DWI data, with a large number of gradient directions for the different b-values. We examine further how the standard deviations increase for a subset of the dataset with b-values up to $b = 3000\,\text{s/mm}^2$. The subset consists of the first 32 gradient directions

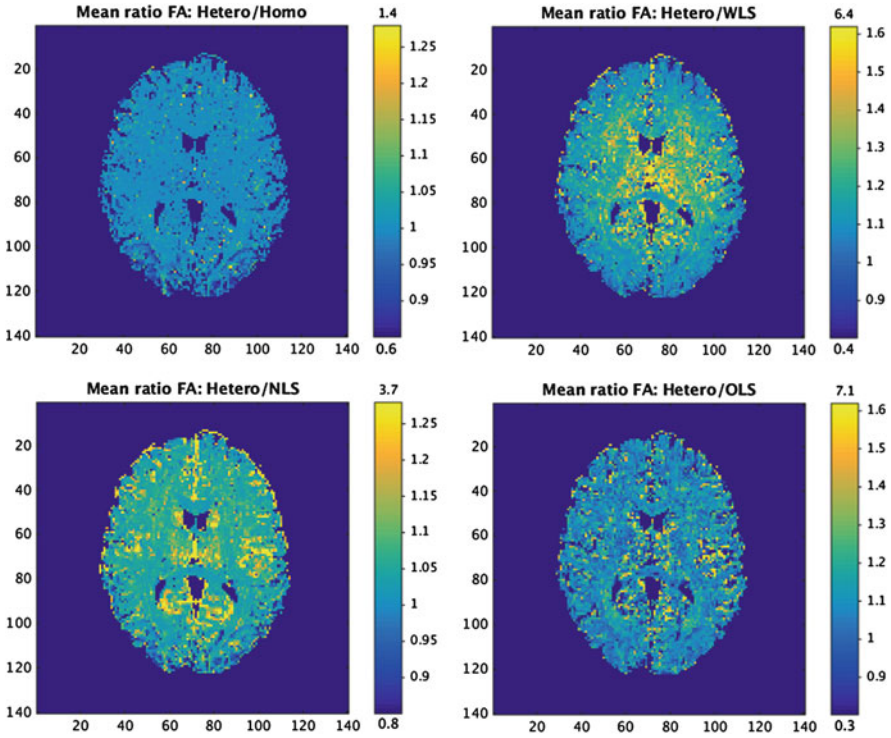


Fig. 7 The ratio between the posterior means of FA for the heteroscedastic DTI model to the homoscedastic counterpart and to the point estimates for OLS, WLS and NLS, using the part of the dataset with all b-values up to $b = 3000\,\text{s/mm}^2$. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

for each of the datasets with b-value $b = 1000\,\text{s/mm}^2$ and $b = 3000\,\text{s/mm}^2$. We find our selection of data subset acceptable as a representative 50% part of the data, as the selected gradient directions for each subset are rather uniformly distributed across the unit sphere, see Fig. 10 for an illustration. In Fig. 11 the posterior standard deviations for the heteroscedastic DTI model are shown for the data subset and the whole dataset with b-values up to $b = 3000\,\text{s/mm}^2$. In general, the standard deviations for the data subset are substantially larger than the ones for the whole dataset, especially for FA in regions close to CSF. The corresponding standard deviations for an even smaller subset with the first 16 gradient directions are in general even higher (not shown here), but in some voxels the estimation of the single-diffusion tensor become unstable due to the low number of measurements. Hence, with less DTI data the DTI scalar measures become less precise for the heteroscedastic DTI model.

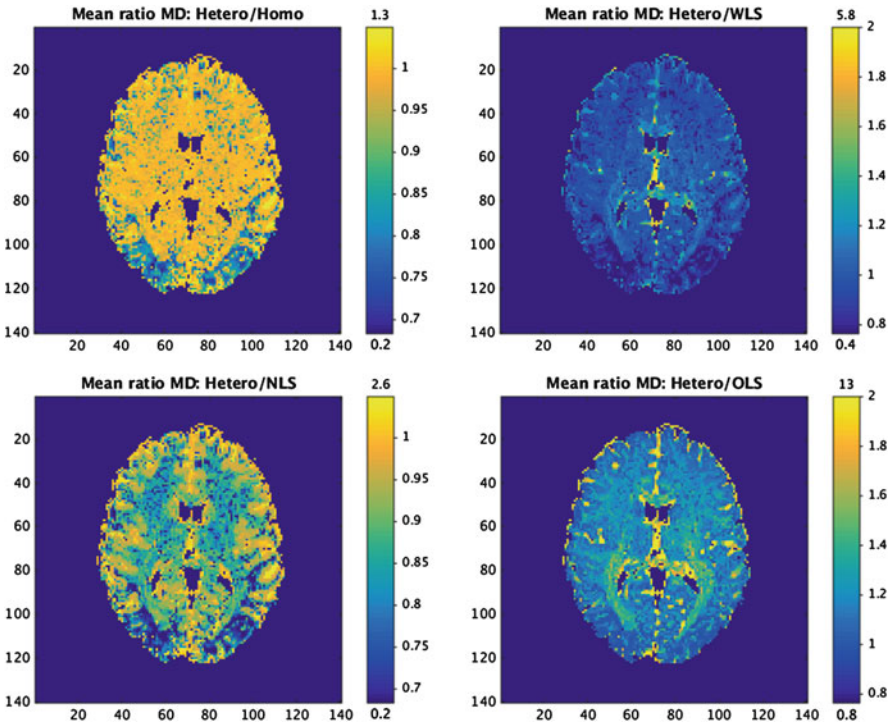


Fig. 8 The ratio between the posterior means of MD for the heteroscedastic DTI model to the homoscedastic counterpart and to the point estimates for OLS, WLS and NLS, using the part of the dataset with all b-values up to $b = 3000$ s/mm^2. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

5.2 Properties of the Log-Cholesky Parametrization

The log-Cholesky parametrization has the clear advantage of guaranteeing a positive definite tensor. Nevertheless, using least squares methods OLS, WLS or NLS to estimate the tensor, there are at most 22 out of 6751 voxels in the shown slice that do not meet the criteria of a positive definite tensor. The new parameters, ω, from using the log-Cholesky parametrization appear very non-linearly in the DTI regression, however, and one may suspect that this will cause the posterior density to be more non-Gaussian and problematic for optimization and sampling. We have found that the opposite is true; the posterior distribution is closer to Gaussian in ω-space than it is in β-space in the original parametrization. To illustrate this we have selected a voxel where the original parametrization is particularly far from Gaussian, and the log posterior distribution is plotted both in β-space (Fig. 12) and ω-space (Fig. 13). It is clear that the log-Cholesky parametrization gives a more Gaussian posterior. Figure 14 shows that this result holds for a large majority of the voxels.

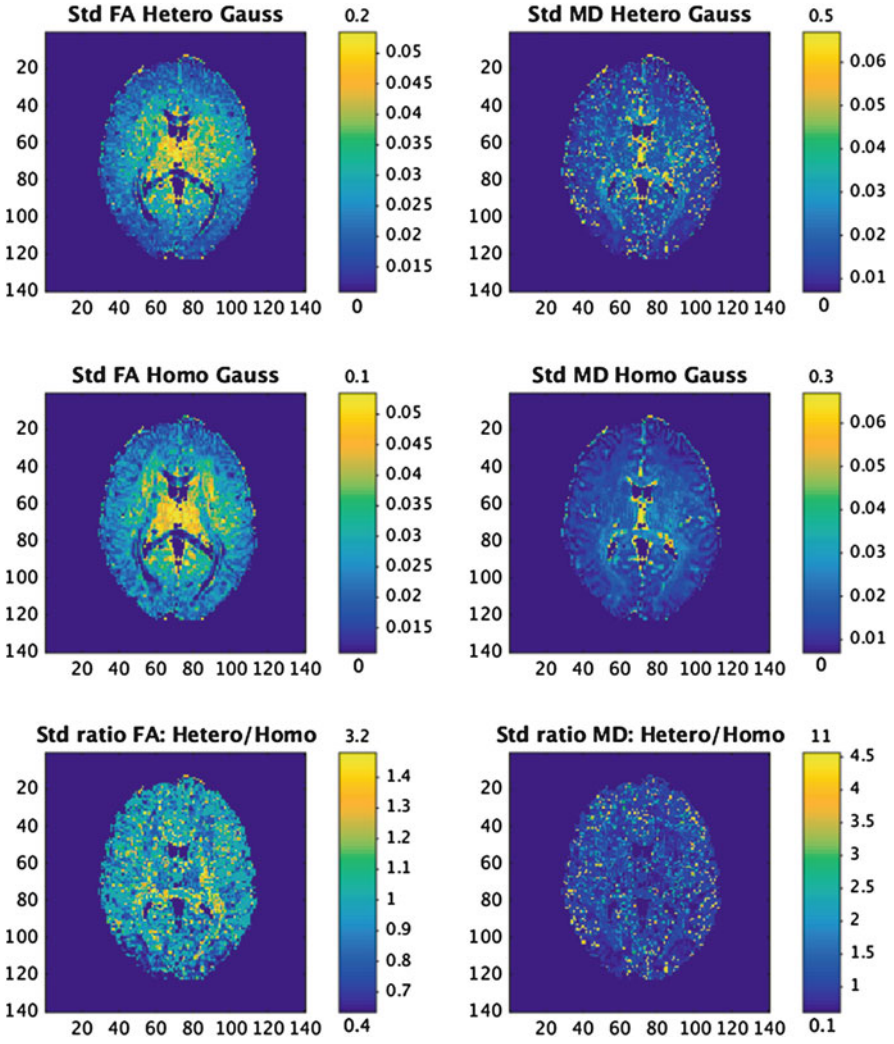


Fig. 9 Posterior standard deviations for the heteroscedastic and homoscedastic DTI models of the DTI scalar measures FA and MD, using the whole dataset with all b-values up to $b = 5000$ s/mm^2. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

6 Discussion

We propose a Bayesian heteroscedastic regression approach for estimating single-diffusion tensors, and apply it to data from the Human Connectome Project. Compared to traditional homoscedastic DTI models such as OLS, WLS and NLS, our approach is more general as it makes it possible to include arbitrary

Fig. 10 Diffusion directions for a 50% part (*left*) and for all diffusion directions (*right*) of the dataset with all b-values up to $b = 3000\,\mathrm{s/mm}^2$

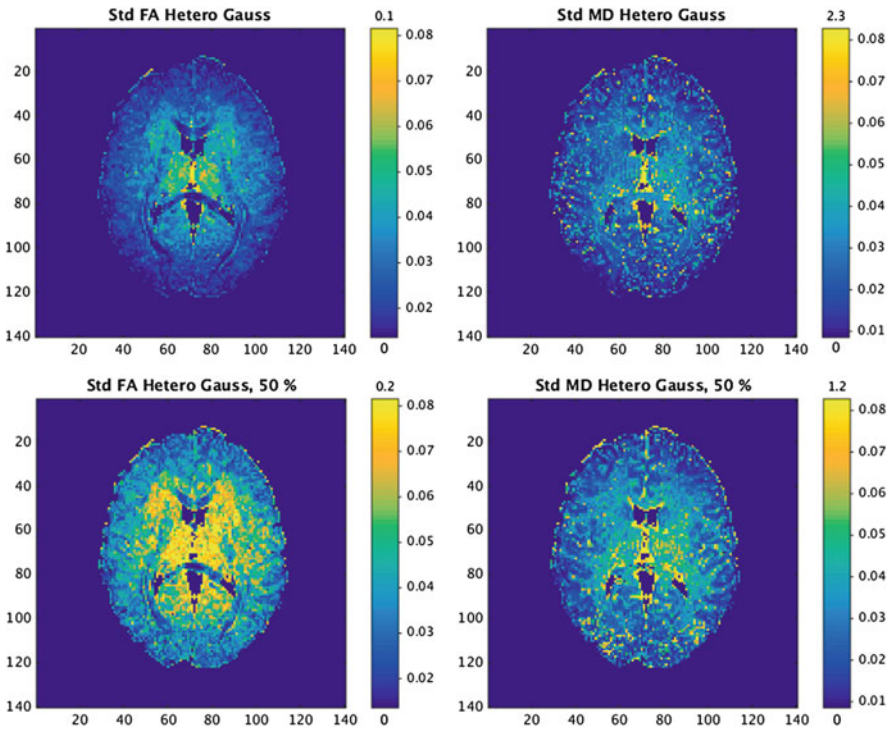


Fig. 11 Posterior standard deviations for the heteroscedastic DTI model of FA and MD, using the data subset and the whole dataset with b-values up to $b = 3000\,\mathrm{s/mm}^2$. The *colorbars* are shown for the mid 95% values and the minimum and maximum values are marked out at the *bottom* and *top of the colorbars*, respectively

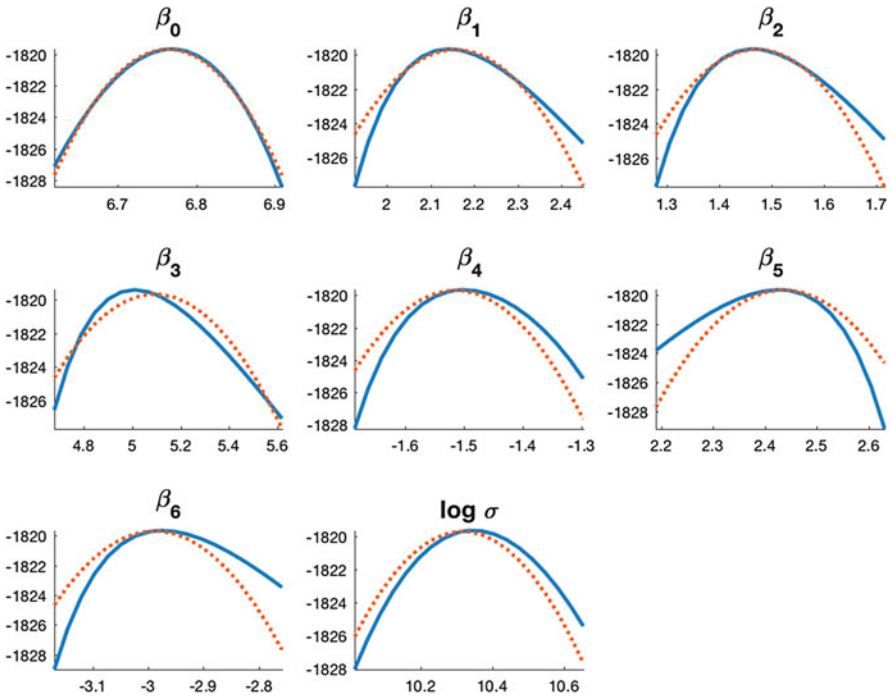


Fig. 12 Investigating the Gaussianity of the posterior distribution in the homoscedastic DTI model in the original parametrization. The *solid blue line* is a slice of the log posterior for a given parameter with the other parameters kept fixed at the posterior mode. The *dotted red line* is the log density of the normal approximation, conditional on the other parameters being at the mode

covariates for the mean as well as for the variance. We are therefore able to model a heteroscedastic noise variance, and show that our approach replicates well-known results that homoscedastic models underestimate standard errors but are still unbiased when data are heteroscedastic. Another difference is that we use a logarithmic link function, instead of first calculating the logarithm of the signal intensity. We show that using the logarithm of the signal and estimating the single-diffusion tensor by ordinary or weighted least squares greatly underestimates the DTI scalar measures FA and MD. This raises some concerns regarding the use of OLS or WLS for estimating the diffusion tensor.

Similar to the work in [19, 20], our algorithm can provide the uncertainty of any derived quantity, such as FA and MD. We obtain the full posterior distribution of FA and MD, as well as the joint distribution of FA and MD or any other DTI measure. As expected, our results show that using a subset of the diffusion gradients leads to a higher variance of FA. By saving the full posterior of the FA in each voxel, a Bayesian group analysis can automatically down-weight subjects with a higher uncertainty.

A drawback of our approach is the computational complexity. In a representative voxel, it took 3.4 s to run 1000 MCMC iterations for the homoscedastic model, and

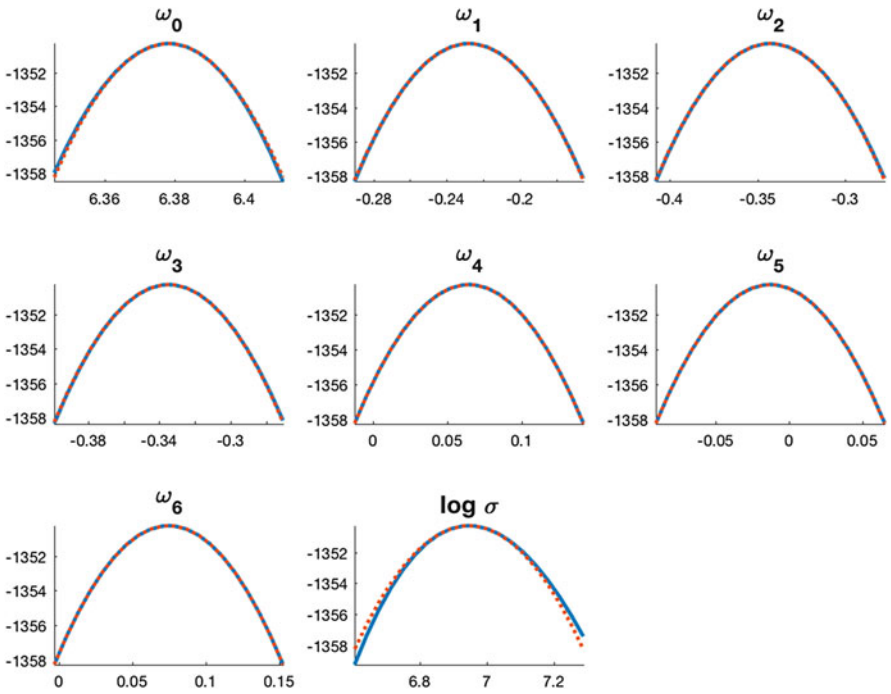


Fig. 13 Investigating the Gaussianity of the posterior distribution in the homoscedastic DTI model in the log-Cholesky parametrization. The *solid blue line* is a slice of the log posterior for a given parameter with the other parameters kept fixed at the posterior mode. The *dotted red line* is the log density of the normal approximation, conditional on the other parameters being at the mode

5.6 s for the heteroscedastic model. For a typical DTI dataset with 20,000 brain voxels, this gives a total processing time of 18.9 h for the homoscedastic model, and 31.1 h for the heteroscedastic model. To run OLS, WLS and NLS for 20,000 brain voxels takes 6, 72, and 112 s, respectively. For this reason, we have only analyzed a single subject, as a group analysis with 20 subjects would be rather time consuming. Fast variational inference can in principle be used instead of MCMC, but variable selection is then no longer straightforward. As each voxel is analyzed independently, it is in theory straightforward to run MCMC on the voxels in parallel, using a CPU or a GPU [10, 16].

To only perform motion correction of diffusion weighted volumes is not sufficient, as the interpolation operation will mix voxels with low and high signal intensity (especially at the border between white matter and gray matter), and a motion correlated signal will therefore still be present. Although not investigated in this work, estimated head motion parameters, or their temporal derivative, can also be included as covariates for the variance, to for example automatically down-weight volumes corresponding to motion spikes [11] (any volume with a high variance will automatically be down-weighted in our framework). Such an approach

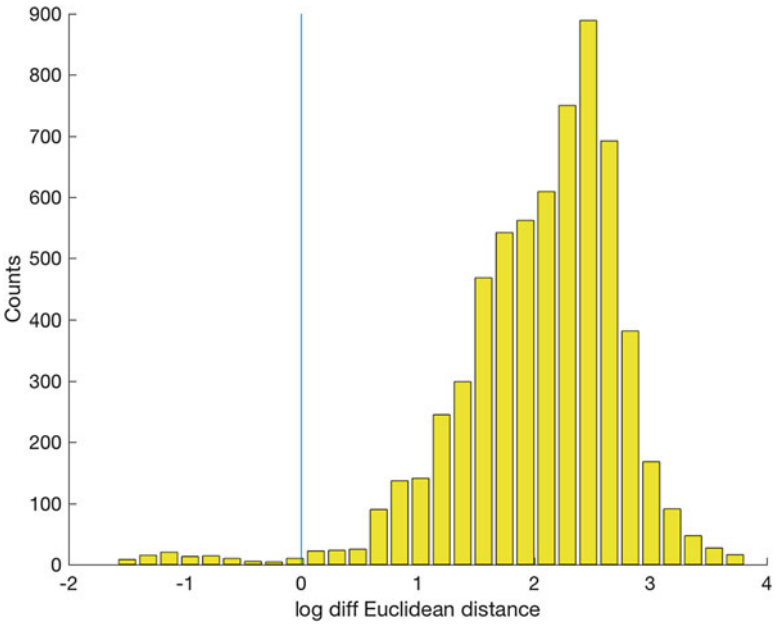


Fig. 14 Comparing the Gaussianity of the posterior in the original and log-Cholesky parametrizations for all voxels. The figure plots the log ratio of the Euclidean distance between the log posterior and its normal approximation in Figs. 12 and 13 over all parameters. A value larger than zero means that the log-Cholesky parametrization produces a more normal posterior than the original parametrization in that particular voxel

would be rather different compared to the common scrubbing approach used for both fMRI and DTI [12, 22, 25], where an arbitrary motion threshold is used to remove or censor volumes with too much head motion. It is also possible to include other covariates that may be important for the variance, e.g. recordings of breathing and pulse [30].

Through the use of variable selection, a large number of covariates can be used for the mean as well as the variance, even if the number of diffusion volumes is rather low. This becomes more important for higher order diffusion models, where a large number of parameters need to be estimated in every voxel. For example, a fourth order covariance matrix in each voxel [32] requires estimation of 21 additional parameters. Our algorithm can therefore also be applied to higher order diffusion models.

Acknowledgements The authors would like to thank professor Hans Knutsson for his code for visualizing diffusion directions.

Anders Eklund was supported by the Information Technology for European Advancement (ITEA) 3 Project BENEFIT (better effectiveness and efficiency by measuring and modelling of interventional therapy) and by the Swedish research council (grant 2015-05356, “Learning of sets of diffusion MRI sequences for optimal imaging of micro structures“). Anders Eklund and Bertil

Wegmann were supported by the Swedish research council (grant 2013-5229, "Statistical analysis of fMRI data").

Data collection and sharing for this project was provided by the Human Connectome Project (HCP; Principal Investigators: Bruce Rosen, M.D., Ph.D., Arthur W. Toga, Ph.D., Van J. Weeden, MD). HCP funding was provided by the National Institute of Dental and Craniofacial Research (NIDCR), the National Institute of Mental Health (NIMH), and the National Institute of Neurological Disorders and Stroke (NINDS). HCP data are disseminated by the Laboratory of Neuro Imaging at the University of Southern California.

References

1. Andersson, J.: Maximum a posteriori estimation of diffusion tensor parameters using a Rician noise model: Why, how and but. NeuroImage **42**, 1340–1356 (2008)
2. Andersson, J.L., Sotiropoulos, S.N.: An integrated approach to correction for off-resonance effects and subject movement in diffusion MR imaging. NeuroImage **125**, 1063–1078 (2016)
3. Basser, P., Pierpaoli, C.: A simplified method to measure the diffusion tensor from seven MR images. Magn. Reson. Med. **39**, 928–934 (1998)
4. Basser, P., LeBihan, D., Mattiello, J.: Estimation of the effective self-diffusion tensor from the NMR spin echo. J. Magn. Reson. **103**, 247–254 (1994)
5. Beckmann, C.F., Jenkinson, M., Smith, S.M.: General multilevel linear modeling for group analysis in FMRI. NeuroImage **20**(2), 1052–1063 (2003)
6. Behrens, T., Woolrich, M., Jenkinson, M., Johansen-Berg, H., Nunes, R., Clare, S., Matthews, P., Brady, J., Smith, S.: Characterization and propagation of uncertainty in diffusion-weighted MR imaging. Magn. Reson. Med. **50**, 1077–1088 (2003)
7. Behrens, T., Berg, H.J., Jbabdi, S., Rushworth, M., Woolrich, M.: Probabilistic diffusion tractography with multiple fibre orientations: what can we gain? NeuroImage **34**, 144–155 (2007)
8. Chen, G., Saad, Z.S., Nath, A.R., Beauchamp, M.S., Cox, R.W.: FMRI group analysis combining effect estimates and their variances. NeuroImage **60**(1), 747–765 (2012)
9. Eierud, C., Craddock, R.C., Fletcher, S., Aulakh, M., King-Casas, B., Kuehl, D., LaConte, S.M.: Neuroimaging after mild traumatic brain injury: review and meta-analysis. NeuroImage: Clin. **4**, 283–294 (2014)
10. Eklund, A., Dufort, P., Forsberg, D., LaConte, S.M.: Medical image processing on the GPU - past, present and future. Med. Image Anal. **17**(8), 1073–1094 (2013)
11. Eklund, A., Lindquist, M., Villani, M.: A Bayesian Heteroscedastic GLM with Application to fMRI Data with Motion Spikes. arXiv:1612.00690 (2016)
12. Elhabian, S., Gur, Y., Vachet, C., Piven, J., Styner, M., Leppert, I.R., Pike, G.B., Gerig, G.: Subject-motion correction in HARDI acquisitions: choices and consequences. Front. Neurol. **5**, 240 (2014)
13. Essen, D.C.V., Smith, S.M., Barch, D.M., Behrens, T.E., Yacoub, E., Ugurbil, K.: The WU-Minn human connectome project: an overview. NeuroImage **80**, 62–79 (2013)
14. Glasser, M.F., Sotiropoulos, S.N., Wilson, J.A., Coalson, T.S., Fischl, B., Andersson, J.L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J.R., Essen, D.C.V., Jenkinson, M.: The minimal preprocessing pipelines for the human connectome project. NeuroImage **80**, 105–124 (2013)
15. Gudbjartsson, H., Patz, S.: The Rician distribution of noisy MRI data. Magn. Reson. Med. **34**(6), 910–914 (1995)
16. Guo, G.: Parallel statistical computing for statistical inference. J. Stat. Theory Pract. **6**(3), 536–565 (2012)
17. Jones, D.: Diffusion MRI: Theory, Methods and Applications. Oxford University Press, Oxford (2011)

18. Koay, C.G.: Least squares approaches to diffusion tensor estimation. In: Jones, D. (ed.) Diffusion MRI: Theory, Methods and Applications, pp. 272–284. Oxford University Press, Oxford (2011)
19. Koay, C., Chang, L., Pierpaoli, C., Basser, P.: Error propagation framework for diffusion tensor imaging via diffusion tensor representations. IEEE Trans. Med. Imaging **26**, 1017–1034 (2007)
20. Koay, C., Nevo, U., Chang, L., Pierpaoli, C., Basser, P.: The elliptical cone of uncertainty and its normalized measures in diffusion tensor imaging. IEEE Trans. Med. Imaging **27**, 834–846 (2008)
21. Kubicki, M., Westin, C.F., Maier, S.E., Mamata, H., Frumin, M., Ersner-Hershfield, H., Kikinis, R., Jolesz, F.A., McCarley, R., Shenton, M.E.: Diffusion tensor imaging and its application to neuropsychiatric disorders. Harv. Rev. Psychiatry **10**(6), 324–336 (2002)
22. Power, J.D., Mitra, A., Laumann, T.O., Snyder, A.Z., Schlaggar, B.L., Petersen, S.E.: Methods to detect, characterize, and remove motion artifact in resting state fMRI. NeuroImage **84**, 320–341 (2014)
23. Setsompop, K., Kimmlingen, R., Eberlein, E., Witzel, T., Cohen-Adad, J., McNab, J., Keil, B., Tisdall, M., Hoecht, P., Dietz, P., Cauley, S., Tountcheva, V., Matschl, V., Lenz, V., Heberlein, K., Potthast, A., Thein, H., Horn, J.V., Toga, A., Schmitt, F., Lehne, D., Rosen, B., Wedeen, V., Wald, L.: Pushing the limits of in vivo diffusion MRI for the human connectome project. NeuroImage **80**, 220–233 (2013)
24. Shenton, M.E., Hamoda, H.M., Schneiderman, J.S., Bouix, S., Pasternak, O., Rathi, Y., Vu, M.A., Purohit, M.P., Helmer, K., Koerte, I., Lin, A.P., Westin, C.F., Kikinis, R., Kubicki, M., Stern, R.A., Zafonte, R.: A review of magnetic resonance imaging and diffusion tensor imaging findings in mild traumatic brain injury. Brain Imaging Behav. **6**(2), 137–192 (2012)
25. Siegel, J.S., Power, J.D., Dubis, J.W., Vogel, A.C., Church, J.A., Schlaggar, B.L., Petersen, S.E.: Statistical improvements in functional magnetic resonance imaging analyses produced by censoring high-motion data points. Hum. Brain Mapp. **35**(5), 1981–1996 (2014)
26. Smith, S.M., Jenkinson, M., Johansen-Berg, H., Rueckert, D., Nichols, T.E., Mackay, C.E., Watkins, K.E., Ciccarelli, O., Cader, M.Z., Matthews, P.M., Behrens, T.E.: Tract-based spatial statistics: Voxelwise analysis of multi-subject diffusion data. NeuroImage **31**(4), 1487–1505 (2006)
27. Veraart, J., Sijbers, J., Sunaert, S., Leemans, A., Jeurissen, B.: Weighted linear least squares estimation of diffusion MRI parameters: strengths, limitations, and pitfalls. NeuroImage **81**, 335–346 (2013)
28. Villani, M., Kohn, R., Giordani, P.: Regression density estimation using smooth adaptive gaussian mixtures. J. Econom. **153**(2), 155–173 (2009)
29. Villani, M., Kohn, R., Nott, D.: Generalized smooth finite mixtures. J. Econom. **171**(2), 121–133 (2012)
30. Walker, L., Chang, L.C., Koay, C.G., Sharma, N., Cohen, L., Verma, R., Pierpaoli, C.: Effects of physiological noise in population analysis of diffusion tensor MRI data. NeuroImage **54**(2), 1168–1177 (2011)
31. Wegmann, B., Eklund, A., Villani, M.: Non-central chi regression for neuroimaging. arXiv:1612.07034 (2016)
32. Westin, C.F., Knutsson, H., Pasternak, O., Szczepankiewicz, F., Ozarslan, E., van Westen, D., Mattisson, C., Bogren, M., O'Donnell, L.J., Kubicki, M., Topgaard, D., Nilsson, M.: Q-space trajectory imaging for multidimensional diffusion MRI of the human brain. NeuroImage **135**, 345–362 (2016)

Multi-Fiber Reconstruction Using Probabilistic Mixture Models for Diffusion MRI Examinations of the Brain

Snehlata Shakya, Nazre Batool, Evren Özarslan, and Hans Knutsson

Abstract In the field of MRI brain image analysis, Diffusion tensor imaging (DTI) provides a description of the diffusion of water through tissue and makes it possible to trace fiber connectivity in the brain, yielding a map of how the brain is wired. DTI employs a second order diffusion tensor model based on the assumption of Gaussian diffusion. The Gaussian assumption, however, limits the use ofDTI in solving intra-voxel fiber heterogeneity as the diffusion can be non-Gaussian in several biological tissues including human brain. Several approaches to modeling the non-Gaussian diffusion and intra-voxel fiber heterogeneity reconstruction have been proposed in the last decades. Among such approaches are the multi-compartmental probabilistic mixture models. These models include the discrete or continuous mixtures of probability distributions such as Gaussian, Wishart or von Mises-Fisher distributions. Given the diffusion weighted MRI data, the problem of resolving multiple fibers within a single voxel boils down to estimating the parameters of such models.

In this chapter, we focus on such multi-compartmental probabilistic mixture models. First we present a review including mathematical formulations of the most commonly applied mixture models. Then, we present a novel method based on the mixture of non-central Wishart distributions. A mixture model of central Wishart distributions has already been proposed earlier to resolve intra-voxel heterogeneity. However, we show with detailed experiments that our proposed model outperforms the previously proposed probabilistic models specifically for the challenging scenario when the separation angles between crossing fibers (two or three) are small. We compare our results with the recently proposed probabilistic models of mixture of central Wishart distributions and mixture of hyper-spherical von Mises-Fisher distributions. We validate our approach with several simulations including fiber orientations in two and three directions and with real data. Resistivity to noise is also demonstrated by increasing levels of Rician noise in simulated data.

S. Shakya (✉) • E. Özarslan • H. Knutsson
Department of Biomedical Engineering, Linköping University, SE-581 85 Linköping, Sweden
e-mail: sneh022@gmail.com; evren.ozarslan@liu.se; hans.knutsson@liu.se

N. Batool
School of Technology and Health, KTH Royal Institute of Technology, Stockholm, Sweden
e-mail: naz.bat@kth.se

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_12

The experiments demonstrate the superior performance of our proposed model over the prior probabilistic mixture models.

1 Introduction

Magnetic resonance imaging (MRI) is a non-invasive technique to evaluate the anatomical structures of various organs. There is a loss in MR signal due to thermal random motion of water molecules in the presence of magnetic field gradients. Diffusion weighted magnetic resonance imaging (DW-MRI) is based on measuring the random Brownian motion of water molecules within a voxel of tissue. By producing quantitative data of water molecule motion that naturally occurs in brain tissues as part of the physical diffusion process, DW-MRI has also been used to map the fiber orientation in the brain's white matter tracts.

Diffusion tensor imaging (DTI) was introduced by Basser et al. [3] in 1994; basic principles and applications of DTI have been reported earlier [16, 26, 29, 30]. Regarding brain anatomy, diffusion appears mostly isotropic in gray matter and cerebrospinal fluid where signal is independent of the direction in which the gradients are applied. It becomes anisotropic in white matter where brain water diffuses freely along axonal fiber direction. A second order diffusion tensor is used in DTI to represent anisotropic diffusion that is based on the assumption of Gaussian-distributed diffusion. Displacement probability characterized by an oriented Gaussian probability distribution function, provides a relatively simple way of quantifying diffusional anisotropy as well as predicting the local fiber direction within the tissue. However, the major drawback of DTI is that it can reveal a single fiber orientation in each voxel and fails in voxels with orientation heterogeneity. Several image acquisition strategies and reconstruction methods have been developed and modified to overcome the limitation of DTI [43]. The q-space imaging (QSI), also referred to as diffusion spectrum imaging (DSI), utilizes the Fourier relation between the diffusion signal and the average particle displacement probability density function [42]. High angular resolution diffusion imaging (HARDI) is an approach that utilizes measurements along many directions [38]. HARDI allows the computation of water molecule displacement probabilities over the sphere thus incorporating the diffusion in several directions. The probability distribution given as spherical function is often referred as the orientation distribution function (ODF) and several probabilistic models have been proposed to compute ODF's. Tuch et al. [40] proposed q-ball imaging method that attempts to transform limited number of multidirectional signals into an ODF. Recently Scherrer et al. [35] have presented a diffusion compartmental model, Distribution of Anisotropic Microstructural Environments in Diffusion-Compartment Imaging (DIAMOND) to characterize the brain tissues. In Sect. 2.2, we will review how the mixtures of von Mises-Fisher distributions are used to construct ODF for the purpose of estimating intra-voxel multi-fiber structure.

Probabilistic Multi-Compartmental Models Multi-compartment models have been proposed as alternative diffusion models for the intrinsic voxel-wise diffusion heterogeneity due to fibers oriented in several directions or crossing fibers. In a multi-compartmental model, diffusion in a number of pre-specified compartments is explicitly modeled as a mixture (discrete/continuous) of basis functions to approximate the MR signal attenuation where the compartments represent groups of cells/fibers inducing similar diffusion properties. Specifically, in a probabilistic multi-compartmental model, the underlying basis functions consist of modular probability distributions. Each voxel is associated with underlying probability distribution(s) defined on the space of diffusion tensors (the manifold of 3×3 positive-definite matrices). Under this probabilistic multi-compartmental model, the problem of extracting the intra-voxel multi-fiber structure then boils down to estimating the mixture of probabilistic distributions for tensors that best explains the observed diffusion-weighted images.

In this chapter, we review mixture models proposed to date for multi-fiber reconstruction such as mixture of Gaussian distributions [39] (Sect. 2.1), mixture of central Wishart distributions (MoCW) by Jian et al. [13, 14] (Sect. 2.3) and mixture of von Mises-Fisher distributions by Kumar et al. [4, 18, 19, 28] (Sect. 2.2). Overall, these methods work well with large separation angles between two or three crossing fibers but have limitations (i.e. the angular error in reconstruction is significantly large) when the separation angles are small. We address this specific problem and present our work on the mixture of *non*-central Wishart distribution (MoNCW) which is shown to perform better in case of smaller separation angles. The motivation to use MoNCW is the increased flexibility in terms of mathematical structure imposed on the underlying probability distribution. Our detailed experiments and analysis presented in Sect. 3 show that despite the simple change in the centrality assumption of Wishart distributions, the proposed method performs much better as compared to the previous methods [13, 14, 18, 19] when separation angles between fibers are small for many simulations.

2 Theory

Let G and $\mathbf{g}$ be the magnitude and direction of the diffusion sensitizing gradient $\mathbf{G}$, and b be the 'b-value' or diffusion weighting given as $b = (\gamma\delta\mathbf{G})^2 t$, where γ is gyromagnetic ratio, δ is the diffusion gradient duration and t is the effective diffusion time. Traditionally, DTI assumes a displacement probability characterized by an oriented Gaussian probability distribution function which yields to the following well-known Stejskal-Tanner signal decay formula (assuming $\delta \ll t$):

$$S(\mathbf{q}) = S_0 \exp(-b\mathbf{g}^T\mathbf{D}\mathbf{g}), \tag{1}$$

where S_0 is the signal in the absence of any diffusion weighting gradient, $\mathbf{D}$ is the apparent diffusion tensor and $\mathbf{q} = \frac{1}{2\pi}\gamma\delta\mathbf{G}$ is the coordinate vector in q-space which

describes the direction and strength of diffusion gradient. Traditionally, diffusional anisotropy in brain white matter is characterized by estimating the diffusion tensor **D** by fitting the tensor to the diffusion data via Eq. (1) using linear regression. However, this model can detect a single fiber orientation from the apparent **D** only. In probabilistic framework, an underlying density, $f(\mathbf{D})$, is defined on the space of diffusion tensors for each voxel and the indirectly observed **D** via DTI measurements is considered to be an instance of the underlying $f(\mathbf{D})$. In this probabilistic framework, Eq. (1) can be rewritten as:

$$S(\mathbf{q}) = S_0 \int_{Sym_+^d} f(\mathbf{D}) \exp(-b\mathbf{g}^T\mathbf{D}\mathbf{g}) d\mathbf{D}, \tag{2}$$

where Sym_+^d is the manifold of symmetric positive definite matrices.

2.1 Mixture of Gaussian Distributions

A multiple compartment model has been proposed earlier based on a *discrete* mixture of Gaussian distributions where diffusion within each compartment/region of a voxel is assumed to be Gaussian [10, 39]. Then the diffusion signal from multiple diffusion compartments is given as a finite mixture of Gaussian distributions as follows:

$$S(\mathbf{q})/S_0 = \sum_j w_j \exp(-b\mathbf{g}^T\mathbf{D}_j\mathbf{g}). \tag{3}$$

Here w_j's are the apparent volume fractions of the voxel with diffusion tensors $\mathbf{D}_j$'s. Eigenvalues of the individual tensors $\mathbf{D}_j$ were specified *a priori* according to the physiological constraints of diffusion in fibers. An optimization scheme based on gradient descent was then used to solve for the eigenvectors and volume fractions w_j that gave the lowest error between the predicted and observed diffusion signals. This discrete mixture model, however, had the model selection limitation where the number of mixture components had to be set arbitrarily. To mitigate that, Leow et al. [22] employed the tensor distribution model introduced by Jian et al. [14, 15] to estimate the tensor distribution function (TDF) that best explained the observed diffusion-weighted data. Given multiple observations of **q**, the TDF was calculated by solving the following equation for the optimal $P^*(\mathbf{D})$:

$$P^*(\mathbf{D}) = \arg\min_P \sum_j \left[S_{obs}(\mathbf{q_j}) - S_{cal}(\mathbf{q_j})\right]^2,$$

$$\text{where } \int P(\mathbf{D}) d\mathbf{D} = 1 \text{ and } P(\mathbf{D}) \geq 0. \tag{4}$$

$$S_{cal}(\mathbf{q}) = \int_{Sym_+^d} P(\mathbf{D}) \exp(-b\mathbf{g}^T\mathbf{D}\mathbf{g}) d\mathbf{D} \tag{5}$$

The constraints on the TDF were imposed using exponential mapping and the TDF was estimated using the methods of Lagrange multipliers and gradient descent.

2.2 Mixture of Von Mises-Fisher Distributions

As mentioned earlier, multi-direction diffusion is represented as a multivariate spherical function ODF. One approach to specifying ODF has been the mixture models using von Mises-Fisher distributions. These models are claimed to have several advantages over other spherical distributions [28]. The von Mises-Fisher (vMF) distribution is the analog of the Gaussian distribution on a sphere and is parameterized by a principal (mean) direction $\boldsymbol{\mu}$ and a concentration parameter κ. The concentration parameter κ quantifies how tightly the function is distributed around the mean direction $\boldsymbol{\mu}$. For example, a set of vectors which are more or less in the same direction would have a vMF distribution which would be unimodal and symmetric around that mean direction. For $\kappa = 0$ the distribution is uniform over the sphere. The vMF distribution over a hypersphere in space S_{p-1} in the random variable $\mathbf{x}$ is given as:

$$M_p(\mathbf{x}|\boldsymbol{\mu}, \kappa) = \left(\frac{\kappa}{2}\right)^{p/2-1} \frac{1}{2\pi\Gamma(p/2)I_{p/2-1}(\kappa)} \exp(\kappa \boldsymbol{\mu}^T \mathbf{x}), \tag{6}$$

where $|\mathbf{x}| = 1$, $|\boldsymbol{\mu}| = 1$ and I_k denotes the modified Bessel function of the first kind and order k. For $p = 3$ the distribution can be written as follows:

$$M_3(\mathbf{x}|\boldsymbol{\mu}, \kappa) = \frac{\kappa}{4\pi \sinh(\kappa)} \exp(\kappa \boldsymbol{\mu}^T \mathbf{x}). \tag{7}$$

The vMF distribution is inherently unimodal. To allow multiple peaks over sphere depicting diffusion in multiple directions, McGraw et al. [28] proposed a discrete mixture of vMF distributions to specify the ODF as follows:

$$P(\mathbf{x}|\boldsymbol{\mu}, \kappa) = \sum_j w_j M_3(\mathbf{x}|\boldsymbol{\mu}_j, \kappa_j), \quad \text{with } \sum_j w_j = 1 \text{ and } w_j \geq 0, \tag{8}$$

where $(w_j, \boldsymbol{\mu}_j, \kappa_j)$ constituted the model parameters and were estimated using the Expectation-Maximization algorithm. One detail regarding the vMF distributions is that the Gaussian models do not extend to the sphere in a straightforward way because of the problem of 'warping' of 3D angles modulo π and 2π whereas the ODF is antipodally symmetric. Hence, they proposed to use the antipodal pairs of

$\boldsymbol{\mu}_1 = -\boldsymbol{\mu}_2$ and assumed the total number of mixture vMF distributions to be four a priori.

To alleviate the problem of fixing the number of components a priori in the discrete mixture models, Kumar et al. [18, 19] proposed to use a continuous mixture of vMF distribution functions as follows:

$$f(\mathbf{x}) = \int_{S_{p-1}} D(\boldsymbol{\mu}) K(\mathbf{x}, \boldsymbol{\mu}) d\boldsymbol{\mu}. \tag{9}$$

where $D(\boldsymbol{\mu})$ denotes the continuous mixing density parametrized by the angle $\boldsymbol{\mu}$ in the spherical domain S_{p-1} and $K(\mathbf{x}, \boldsymbol{\mu})$ are the mixture components, vMF distributions in this case. Following McGraw et al. [28], they proposed to achieve antipodal symmetry by pairing each vMF distribution with principal direction $\boldsymbol{\mu}$ with its antipodal counterpart $-\boldsymbol{\mu}$. Thus each direction $\boldsymbol{\mu}$ was assigned a modified vMF distribution, $\tilde{M}_3(\mathbf{x}|\boldsymbol{\mu}, \kappa)$ which was an average of the two vMF distributions with principal directions $\boldsymbol{\mu}$ and $-\boldsymbol{\mu}$. $\tilde{M}_3$ has the following expression:

$$\tilde{M}_3(\pm\mathbf{x}|\boldsymbol{\mu}, \kappa) = \frac{\kappa}{4\pi \sinh(\kappa)} \cosh(\kappa \boldsymbol{\mu}^T \mathbf{x}). \tag{10}$$

Interestingly, they proposed to use vMF as the mixing density for the vMF mixing distributions as well; doing so results in a closed form expression for the diffusion weighted MR signal. However, a unimodal single vMF as mixing density would yield a single mode as the convolution of two unimodal functions $(D(\cdot), K(\cdot))$ yields a unimodal function. Hence, to allow for multi-modal (corresponding to multi-fibers) spherical distribution function, a *discrete mixture* of vMF densities was used as the mixing density.

$$D(\boldsymbol{\mu}) = \sum_{i=1}^{M} w_i \tilde{M}_3(\pm\boldsymbol{\mu}|\boldsymbol{\mu}'_i, \kappa'_i), \tag{11}$$

where $(\boldsymbol{\mu}'_i, \kappa'_i)$ are the parameters for the vMF component distributions over the space of $\boldsymbol{\mu}$. For simplicity, κ'_i's were all set to be equal to 1. By substituting the values of $D(\boldsymbol{\mu})$ and $\tilde{M}_3(\pm\mathbf{x}|\boldsymbol{\mu}, \kappa)$ in Eq. (9), their model for the MR signal is given as:

$$\frac{S(\mathbf{q})}{S_0} = \sum_{i=1}^{M} w_i \frac{\kappa}{4\pi \sinh(1) \sinh(\kappa)} \left[\frac{\sinh(\|\kappa \boldsymbol{\mu}_i - \mathbf{q}\|)}{\|\kappa \boldsymbol{\mu}_i - \mathbf{q}\|} + \frac{\sinh(\|\kappa \boldsymbol{\mu}_i + \mathbf{q}\|)}{\|\kappa \boldsymbol{\mu}_i + \mathbf{q}\|} \right], \tag{12}$$

where M is the number of directions which determines the resolution of spherical discretization and $\boldsymbol{\mu}_i$'s are the mean directions selected for M directions. The $\boldsymbol{\mu}_i$'s are pre-defined using some sampling scheme of the unit sphere, κ is the constant concentration and w_i's are the unknown weights to be estimated. For

sparse and positive weights, the iterative algorithm of Non-Negative Least Squares (NNLS) [21] was used for estimation.

From von Mises-Fisher to *Hyperspherical* von Mises-Fisher Models One approach to enforce antipodal symmetry in mixture models has been the use of *Knutsson mapping* [4, 18]. Knutsson [17] proposed a mapping which leads to direction independent magnitude in the higher-dimensional mapped space. Thus, it is useful for handling data which depend only on axes and not directions. In the context of DTI continuous mixture models, this mapping alleviated the need to pair antipodal bases into one but at the expense of increased complexity of problem definition in a higher dimensional space. Given a vector lying in S_2 with 3D spherical parameters (r, θ, ϕ), the Knutsson mapping $\nu(\cdot)$ maps the vector to S_4 by the following 5D parametrization:

$$\begin{aligned}\nu : (r,\theta,\phi) \rightarrow (\mathbf{p_1},\mathbf{p_2},\mathbf{p_3},\mathbf{p_4},\mathbf{p_5}) \; ; \; \mathbf{p_1} = \sin^2\theta\cos 2\phi, \; \mathbf{p_2} = \sin^2\theta\sin 2\phi, \\ \mathbf{p_3} = \sin^2\theta\sin\phi, \; \mathbf{p_4} = \sin^2\theta\cos\phi, \; \mathbf{p_5} = \sqrt{3}(\cos^2\theta - \frac{1}{3}).\end{aligned} \tag{13}$$

Bhalerao and Westin [4] proposed a discrete mixture of Hyper-spherical vMF distributions based on Knutsson mapping. Their M-modal mixture model is given as:

$$\begin{aligned}P(\mathbf{x}|w,\Theta) &= \sum_j w_j M_5(\mathbf{x}|\boldsymbol{\mu}_j,\kappa_j), \\ &\text{with } \sum_j w_j = 1 \text{ and } \Theta = (\boldsymbol{\mu}_j,\kappa_j),\end{aligned} \tag{14}$$

where the Hyper-spherical vMF distributions, $M_5(\mathbf{x}|\boldsymbol{\mu}_j,\kappa_j)$, are as defined in Eq. (6). The Expectation Maximization algorithm was used for Maximum Likelihood (ML) parameter estimation. However, their model required a pre-determined value of number of mixtures, M. Given an ML fit to the samples, the model selection criterion, Akaike information criterion (AIC), was used to select M. Furthermore, the values AIC(1) and AIC(2) were compared to select between single or multi-fiber regions within the voxel.

Based on their earlier work [19] on the continuous mixture of vMF distributions Kumar et al. extended their work [18] by replacing continuous mixture of vMF by continuous mixture of Hyper-spherical vMF (MoHvMF) based on Knutsson's higher dimensional mapping described above. Thus, by using new parametrization in Eq. (13) and by replacing $\tilde{M}_3(\pm\mathbf{x}|\boldsymbol{\mu},\kappa)$ with $\tilde{M}_5(\pm\mathbf{x}|\boldsymbol{\mu},\kappa)$ in Eqs. (9), (11), they

arrived at the following closed-form MRI signal model:

$$\frac{S(\mathbf{q})}{S_0} = \sum_{i=1}^{N} w_i \frac{\sqrt{2}\kappa^{\frac{3}{2}}/(3\sqrt{\pi} I_{\frac{3}{2}}(1) I_{\frac{3}{2}}(\kappa))}{(\kappa \cosh(\kappa) - \sinh(\kappa))} \left[\frac{\cosh(\|\kappa \boldsymbol{\mu}_i + \mathbf{q}\|)}{\|\kappa \boldsymbol{\mu}_i + \mathbf{q}\|^2} - \frac{\sinh(\|\kappa \boldsymbol{\mu}_i + \mathbf{q}\|)}{\|\kappa \boldsymbol{\mu}_i + \mathbf{q}\|^3} \right]. \tag{15}$$

Kumar et al. [19] compared their results with those of Jian et al. [13, 14] (described in next Section) and showed superior performance in terms of fiber separation in the presence of noise. In Sect. 3 we will show that our model is able to produce better results than the MoHvMF model.

2.3 Mixture of Central Wishart Distributions (MoCW)

The family of Wishart probability distributions are defined over symmetric, non-negative definite matrix-valued random variables (or random matrices). A random matrix with central Wishart distribution having parameters p and Σ can be seen as a sum of outer products of p independent multivariate normal random vectors having zero mean ('centrality') and covariance matrix Σ.

Definition 1 Let $\mathbf{X}_{p \times n} = [X_1, X_2, \cdots, X_p]^T$ be a $p \times n$ matrix where each row is independently drawn from an n-variate Gaussian distribution, $X_i \sim N(\mathbf{0}, \Sigma)$. Then $\mathbf{Z} = \mathbf{X}^T\mathbf{X}$ is a random $n \times n$ symmetric matrix having central Wishart distribution, $W_n(p, \Sigma)$, with probability density defined as:

$$f(\mathbf{Z}) = \frac{\left|\mathbf{Z}\right|^{\frac{p-n-1}{2}} \exp\left[-\frac{1}{2}\text{trace}(\Sigma^{-1}\mathbf{Z})\right]}{\left|\Sigma\right|^{\frac{p}{2}} 2^{\frac{np}{2}} \Gamma_n(\frac{p}{2})}, \tag{16}$$

where Γ_n is the multivariate gamma function given as:

$$\Gamma_n(p) = \int_{Sym_+^d} \exp\left(-\text{trace}(\mathbf{Z})\right) \left|\mathbf{Z}\right|^{p-(n+1)/2} d\mathbf{Z}.$$

The parameter $\Sigma \in Sym_+^d$ is known as scale matrix, n corresponds to the space Sym_+^d of diffusion tensors and p is the degree of freedom and has to be $p \geq n$. If this condition does not hold, the resulting Wishart distribution is called a singular Wishart distribution due to Σ being a singular matrix. And if $n = 1$, the one dimensional Wishart distribution is a chi-square distribution with p degrees of freedom and the scale parameter $\Sigma \geq 0$. □

Jian et al. [13, 14] proposed to model the underlying distribution of diffusion tensors through the parametric probability family of 'central' Wishart distributions, $W_n(p, \Sigma)$, for a single fiber case and then a mixture of central Wishart distributions for compartmental or multi-fiber case. The Laplace transform of central Wishart distribution is [23]:

$$\int \exp\left(-\text{trace}(\Theta u)\right)\gamma_{p,\mathbf{\Sigma}}\, du = \left|I_n + \Theta\Sigma\right|^{-p} \tag{17}$$

where $(\Theta + \Sigma^{-1}) \in Sym_+^d$, $\gamma_{p,\mathbf{\Sigma}}$ is the generalized gamma distribution and $\gamma_{p/2,2\Sigma} = W_n(p, \Sigma)$. By utilizing this definition of Laplace transform of Wishart distribution and noting $\mathbf{B} = b\mathbf{g}\mathbf{g}^T$, the Eq. (2) can be written as:

$$S(\mathbf{q})/S_0 = \left|I_n + \mathbf{B}\Sigma\right|^{-p} = (1 + \text{trace}(\mathbf{B}\Sigma))^{-p} = (1 + b\mathbf{g}^T\Sigma\mathbf{g})^{-p}. \tag{18}$$

By using the expected value of Wishart distribution ($\hat{\mathbf{D}} = p\Sigma$) in the above expression, we reach at the following expression:

$$S(\mathbf{q})/S_0 = (1 + b\mathbf{g}^T\hat{\mathbf{D}}\mathbf{g}/p)^{-p}. \tag{19}$$

This is referred to as the Rigaut-type asymptotic fractal expression [34]. The value of p depends on the dimension of space and the signal should have the asymptotic behavior in three-dimensional space as $S(q) \sim q^{-4}$ [36]. Therefore, the value of p is chosen to be 2 as $b \propto q^2$. This model can be used to estimate the single fiber orientation in a voxel but cannot resolve intra-voxel heterogeneity based on the existence of several fibers. Hence, Jian et al. [14] proposed a mixture of central Wishart distributions (MoCW) as a probabilistic multi-compartmental model:

$$S(\mathbf{q})/S_0 = \sum_{i=1}^{N} w_i(1 + \text{trace}(\mathbf{B}\Sigma_i))^{-p_i}, \tag{20}$$

where N is the number of mixture components, w_i are the mixture weights and (p_i, Σ_i) are assumed to be the parameters of Wishart basis densities in the mixture. If p_i's and Σ_i's are fixed, the above system of linear equations can be solved to determine w_i. The assumption of fixed (p_i, Σ_i), in addition to the parametric Wishart probability models, adds further constraints to the ill-posed problem of the estimation of underlying probability distribution. By applying these constraints, the problem reduces to the estimation of w_i's only.

2.4 Proposed Model of Mixture of Non-central Wishart Distributions (MoNCW)

The two mixture models based on Gaussian and central Wishart distribution have the similarity of fixing eigenvalues for the diffusion tensor. But the mixture model based on Wishart distribution performs better as diffusion is not constrained to be Gaussian. In this work, we propose to use a mixture of 'non'-central Wishart distribution as the parametric underlying distribution for diffusion tensors $\mathbf{D} \sim W_n(p, \Sigma, \Omega)$ where an additional non-centrality parameter, $\Omega \in Sym_+^d$ is also introduced. The approach is similar to that used in central Wishart distribution model but the non-centrality makes the current model more general.

Definition 2 Let $\mathbf{X}_{p\times n} = [X_1, X_2, \cdots, X_p]^T$ be a $p \times n$ matrix where each row is independently drawn from an n-variate Gaussian distribution, $X_i \sim N(\mu_i, \Sigma)$. Define a matrix $\mathbf{M}_{p\times n}$ which is derived from the population means μ_i as, $\mathbf{M}_{p\times n} = [\mu_1, \mu_2, \cdots, \mu_p]^T$. Then the random $n \times n$ symmetric matrix defined by $\mathbf{Z} = \mathbf{X}^T\mathbf{X}$ has the non-central Wishart distribution, $W_n(p, \Sigma, \Omega)$ with the expected value $\mathrm{E}(\mathbf{Z}) = p\Sigma + \Omega$. □

The interested reader is referred to [11, 24, 25, 33] for detailed description of non-central Wishart distribution including density function. When $\Omega = \mathbf{0}$, the above reduces to the central Wishart distribution $W_n(p, \Sigma)$. Laplace transform of non-central Wishart distribution is given as follows [27]:

$$\int \exp\left(-\mathrm{trace}(\Theta u)\right)\gamma_{p,\mathbf{\Sigma},\Omega}\,du = \left|I_n + \Theta\Sigma\right|^{-p} \exp\left[-\mathrm{trace}\left(\Theta(I_n + \Theta\Sigma)^{-1}\Omega\right)\right], \quad (21)$$

where $p \geq 0$ is the shape parameter and $\Omega \in Sym_+^d$ is the non-centrality parameter. This expression has an additional factor, $\exp\left[-\mathrm{trace}(\Theta(I_n + \Theta\Sigma)^{-1}\Omega)\right]$, due to the non-centrality parameter in comparison to central Wishart distribution. Hence our assumption of non-centrality changes the signal equation for multi-compartmental model in Eq. (20) to:

$$\frac{S(\mathbf{q})}{S_0} = \sum_{i=1}^{N} w_i \left[1 + \mathrm{trace}(\mathbf{B}\Sigma_i)\right]^{-p} \exp\left[-\mathrm{trace}\left(\mathbf{B}\left\{I_n + \mathbf{B}\Sigma_i\right\}^{-1}\Omega_i\right)\right]. \quad (22)$$

The next section details the estimation of the model parameters.

2.4.1 Estimation of Model Parameters

It has been shown earlier [9, 32, 37] that p should belong to Gindikin ensemble $\Lambda = \{\frac{j}{2}, j = 1, 2, \cdots, (n-2)\} \cup [\frac{n-1}{2}, \infty)$. Following Jian et al., we fix all $p_i = 2$ and the eigen values of $\mathbf{D}_i = p\Sigma_i + \Omega_i$ to $(\lambda_1, \lambda_2, \lambda_3) = (1.5, 0.4, 0.4)\,\mu\mathrm{m}^2/\mathrm{ms}$ which are commonly observed in the white matter [39]. This follows from the

assumption that fiber tracts are cylindrical, thus enforcing two eigenvalues (out of three) to be equal for each individual tensor. Furthermore, each tensor is associated by one (principal) unit direction (defined on the unit sphere), which corresponds to the third eigenvalue. Principal directions of Σ are chosen to be N unit vectors evenly distributed on unit sphere and the values for the weight factors, w_i, are estimated for N directions.

In case of MoNCW, we have one additional unknown parameter namely, Ω, to be estimated. In a more rigorous estimation approach (e.g. see Sections 9.3 and 9.4 of [24]) both non-centrality and scale parameters would be estimated from the observed samples, $\mathbf{Z}_i \sim W_n(p, \Sigma, \Omega)$, where $\mathbf{Z} = \mathbf{D}$ in our case. However, based on the signal decay model in Eq. (1), we are able to observe $\mathbf{D}$ indirectly only through signal measurements $S(\mathbf{q})$. Therefore, in the absence of direct observations of the random matrix $\mathbf{D}$, we resort to selecting the non-centrality parameters to be factors of $\mathbf{D}_i$, i.e. $\Omega_i = \alpha\mathbf{D}_i$ with $0 \leq \alpha < 1$. If $\alpha = 0$, the model turns into central Wishart distribution and if $\alpha = 1$, the scale matrix Σ vanishes. This selection ensures that, once the matrices $\mathbf{D}_i$ are assumed (as mentioned earlier), the principal directions of Σ_i are the same as those of $\mathbf{D}_i$ via the relation $\Sigma_i = (\mathbf{D}_i - \Omega_i)/p$.

To select an appropriate value of α, an ad-hoc approach was adopted where the value of α was varied from 0 to 0.99 for a variety of fiber orientations and separation angles. However, for the sake of clarity, we omit the display of results for all fiber orientations and include results for two crossing fibers with separation angles of 60° and 50° only in Fig. 1. As can be observed in Fig. 1, the error in the estimated angle decreases when $\alpha > 0.5$ and is minimum when α is close to 1. Hence, α was selected to be 0.99 for this study. It should be noted that using $\alpha = 0.99$ yields $\Sigma_i = 0.01\mathbf{D}_i/p = 0.005\mathbf{D}_i$ for $p_i = 2$. This new estimate of Σ has much smaller eigenvalues, on a factor of $1/100$, as compared to MoCW model and yields much less angular errors as compared to MoCW and MoHvMF models.

Finally, Eq. (22) leads to a linear system of equations $A\mathbf{w} = \mathbf{s}$, where $\mathbf{s} = S(\mathbf{q})/S_0$ is the normalized signal vector. A is the matrix with elements A_{ji}

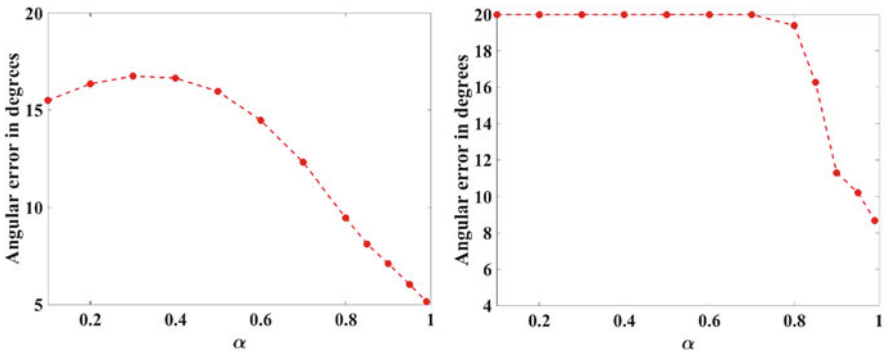


Fig. 1 Angular error in degrees with changing the value of constant α when separation angle between two crossing fibers is (*left*) 60° and (*right*) 50°

given as,

$$A_{ji} = \left[1 + \text{trace}(\mathbf{B}_j \Sigma_i)\right]^{-p} \exp\left[-\text{trace}\left\{\mathbf{B}_j \left(I_n + \mathbf{B}_j \Sigma_i\right)^{-1} \Omega_i\right\}\right], \tag{23}$$

Here $j = 1, 2, \ldots, K$ are the DTI measurements and the vector $\mathbf{w} = \{(w_i); i = 1, 2, \ldots, N\}$ is the vector of unknown parameters to be estimated and $N \geq K$ to avoid the under-determined system. The matrix A needs to be calculated once as it depends on the sampling scheme only. We employ the non-negative least square method (NNLS) [21] to estimate $\mathbf{w}$.

2.5 *Visualization of Estimation Results with Displacement Probabilities*

Fiber orientations in voxels are usually visualized by the displacement probabilities calculated in the direction of the displacement vector $\mathbf{r}$ as follows:

$$P(\mathbf{r}) = \int_{R^3} \int_{Sym^d_+} f(\mathbf{D}) \exp(-\mathbf{q}^T \mathbf{D} \mathbf{q} t) d\mathbf{D} \exp(-i\mathbf{q} \cdot \mathbf{r}) d\mathbf{q} \tag{24}$$

$$\approx \sum_{i=1}^{N} \frac{w_i}{\sqrt{(4\pi t)^3 \left|\hat{\mathbf{D}}_\mathbf{i}\right|}} \exp(-\mathbf{r}^T \hat{\mathbf{D}}_i^{-1} \mathbf{r} / 4t), \tag{25}$$

where $\hat{\mathbf{D}}_i = p\Sigma_i + \Omega_i$ are the expected values of non-central Wishart distribution. The displacement probability profiles are computed as real valued function on the sphere and represented in terms of spherical harmonics series where the spherical harmonics coefficients are used for visualization. We use 8^{th} order spherical harmonics expansion for visualization. The finite order of spherical harmonics and noise lead to several local maxima which are usually removed by applying some technique. However, we decided not to remove these local maxima which appear as small spikes at the center of displayed fibers in our results.

3 Results

In this section, we present results of our experiments on the reconstruction of intra-voxel crossing fibers with and without noise. We present results from simulated and real data and compare our results with both the MoCW and MoHvMF mixture models. The MATLAB$^{\text{TM}}$ open library [1, 2] was used to simulate data for crossing fibers where adaptive kernels were used to approximate the accurate continuous DW-MRI signal. The total number of measurements (K) and mixture components

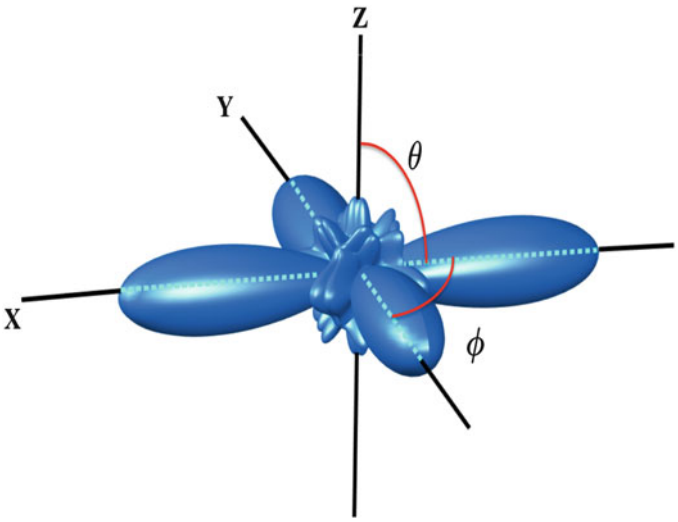


Fig. 2 Display of azimuthal angles ϕ and polar angles θ for two crossing fibers

(N) were set to be 81 and 321 respectively for all simulations. The simulated DTI data were computed for the b-value of 1500 s/mm^2. A built-in MATLAB^{TM} function *lsqnonneg* was used to compute mixture weights in Eq. (22). We used the results of MoHvMF model from the work reported in [19] for comparison and produced results for MoCW and MoNCW models using MATLAB^{TM} software.

Experiments on Fiber Separation First, we present results on the estimation and visualization of displacement probability distributions using Eq. (25) for two or three crossing fibers without noise. The estimation accuracy is calculated in terms of the errors in the polar angle, θ, and the azimuthal angle, ϕ, of the crossing fibers. For clarity, Fig. 2 includes a three dimensional display of two crossing fibers showing θ and ϕ. One fiber is oriented at the azimuthal angle $\phi_1 = 0°$ and the other at $\phi_2 = 90°$. The polar angle θ is 90° for both fibers. The smaller spikes near the origin represent truncation error in the spherical harmonics reconstruction we mentioned before. The angular error is calculated as follows:

$$Error = \frac{\sum_{i=1}^{L} \left| \phi_i^{Original} - \phi_i^{Estimated} \right|}{L},$$

where L is the number of fibers in a voxel.

Figure 3 includes plots of mean angular errors in azimuthal angle ϕ with increasing separation angles for two and three crossing fibers. We also calculated angular errors for the same experimental setups using MoCW modeling. It can be observed in the plots that the proposed MoNCW model offers significant reduction in angular error (except for 30° and 40°) especially at small separation angles. For

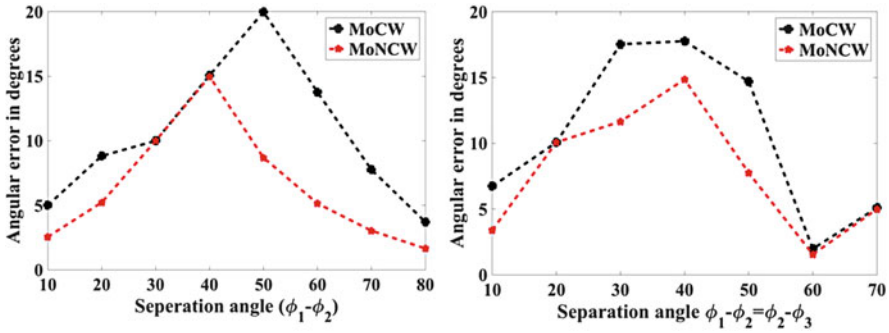


Fig. 3 Mean of angular errors in degrees with increasing separation angles between (*left*) two crossing fibers and (*right*) three crossing fibers

Table 1 Comparison of angular errors in (ϕ, θ) for the two mixture models, MoCW and MoNCW

$\lvert\phi_1 - \phi_2\rvert$	MoCW	MoNCW
10°	(5.00°, 10.0°)	(5.00°, 10.0°)
20°	(6.00°, 7.50°)	(5.70°, 11.7°)
30°	(13.0°, 6.00°)	(13.0°, 6.00°)
40°	(21.0°, 7.00°)	(17.0°, 7.00°)
60°	(28.0°, 6.00°)	(13.0°, 3.00°)
70°	(23.0°, 7.00°)	(9.00°, 2.00°)
80°	(15.0°, 4.00°)	(7.00°, 2.00°)

these cases it becomes more challenging to separate two crossing fibers. For a more difficult setup, we also conducted simulation for the two crossing fibers residing not in the $xy-$ plane i.e. having different polar angles $(\theta_1, \theta_2) = (40°, 60°)$. Table 1 details the angular errors in (ϕ, θ) in this case with increasing separation angle $(\phi_1 - \phi_2)$. The error is significantly low with MoNCW model for some instances.

Experiments with Noise Next, we conducted experiments in the presence of noise. We imposed different levels of Rician-distributed noise on the synthetic MR signal data for two and three crossing fibers with different separation angles. Rician-distributed noise is added to signal as follows:

$$S = \sqrt{(S + \sigma * randn(1))^2 + (\sigma * randn(1))^2},$$

where *randn* is a built-in MATLAB^{TM} function which generates normally distributed random numbers. The experiments on noise sensitivity analysis were set up with increasing levels of Rician-distributed noise having standard deviations in the range $(\sigma = 0.01 - 0.09)$.

First, we generated 100 simulation samples with the noise of $\sigma = 0.08$ with fibers orientation as $(\phi_1, \phi_2) = (10°, 90°)$. Polar angle was set to be 90° for all simulations. Figure 4 shows estimated probability displacement profiles for the 100 simulations with the two models of MoCW and MoNCW. It can be observed that

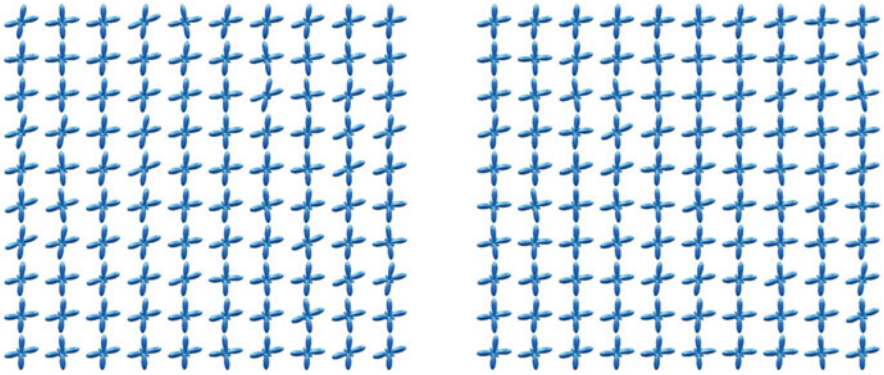


Fig. 4 Resistance to noise with large separation angle (80°) between two crossing fibers, $\sigma = 0.08$ (*left*) MoCW model and (*right*) MoNCW model

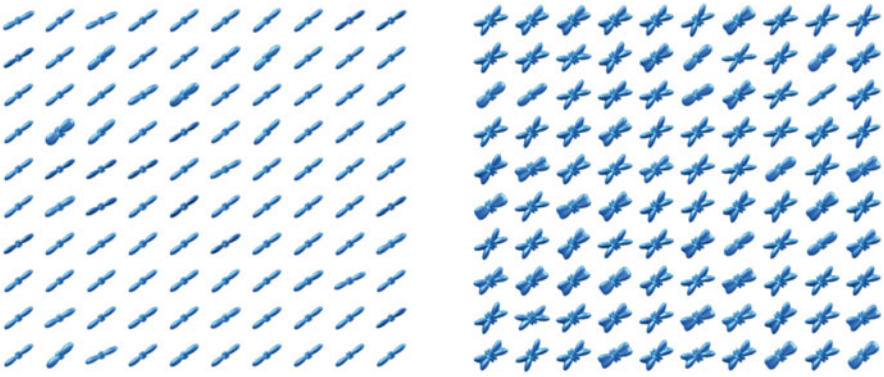


Fig. 5 Resistance to noise with small separation angle (50°) between two crossing fibers, $\sigma = 0.06$ (*left*) MoCW model and (*right*) MoNCW model

both models are stable in the presence of noise when the separation angle between the two fibers is large $|\phi_1 - \phi_2| = 80°$. Next, we performed similar simulations with a smaller separation angle of $|\phi_1 - \phi_2| = 50°$ where fibers are oriented at $(\phi_1, \phi_2) = (10°, 60°)$ and the noise level is set to be $\sigma = 0.06$. From the results shown in Fig. 5, it can be observed that the MoCW model is not able to distinguish the two crossing fibers, even in the presence of small amount of noise in the data. On the other hand, the proposed model is well able to separate the two fibers. We also performed simulations for three crossing fibers. The orientation of fibers was set to be $(\phi_1, \phi_2, \phi_3) = (10°, 60°, 120°)$ and the noise level was $\sigma = 0.04$. The 100 samples reconstructed with MoCW and MoNCW are shown in Fig. 6. It can be observed from the figure that the MoNCW model performs much better than the MoCW model in many instances in separating the three crossing fibers as well.

Figure 7 includes plots of mean and standard deviation of the azimuthal angular errors with small and large separation angles in case of two crossing fibers. The

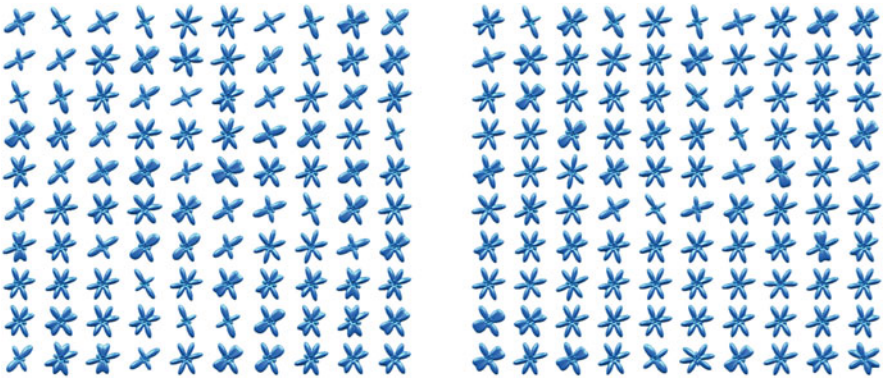


Fig. 6 Resistance to noise in case of three crossing fibers ($(\phi_1, \phi_2, \phi_3) = (10°, 60°, 120°)$), $\sigma = 0.04$ (*left*) MoCW model and (*right*) MoNCW model

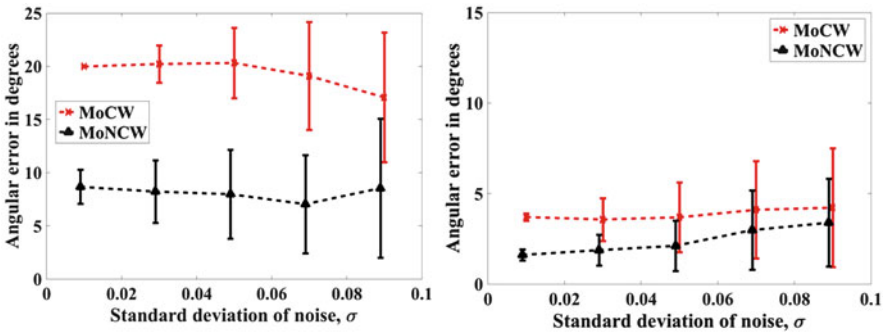


Fig. 7 Mean and standard deviation of angular error in azimuthal angles ϕ when separation angle is (*left*) 50° and (*right*) 80°

improvement in the estimation of azimuthal angle ϕ using MoNCW model is significant. We also plotted the mean and standard deviation for azimuthal angles for three crossing fibers. These are displayed in Fig. 8 with orientations $(\phi_1, \phi_2, \phi_3) = (10°, 30°, 60°)$ and $(\phi_1, \phi_2, \phi_3) = (10°, 60°, 120°)$. The reduction in error is large at some instances with the proposed model when fibers are closely oriented. This reduction, however, is not significant when fibers are sufficiently apart. Table 2 shows the mean and standard deviation values of angular errors from original fiber orientation and computed fiber orientation with increasing Rician-distributed noise. The errors are presented for 1, 2 and 3-fibers in a voxel using two models, MoCW and MoNCW. It is observed that the error in case of 1 fiber (orientation is at 0°) is slightly higher than that for MoCW model but the overall error is not significant ($<2°$) with MoNCW. In case of two fibers with separation angle 70°, the mean and standard deviation is always $<5°$ with MoNCW model, while the mean with MoCW model is always $>6°$. It is also observed that for three fibers (oriented as $(\phi_1, \phi_2, \phi_3) = (10°, 40°, 80°)$), the error is lower with MoNCW model.

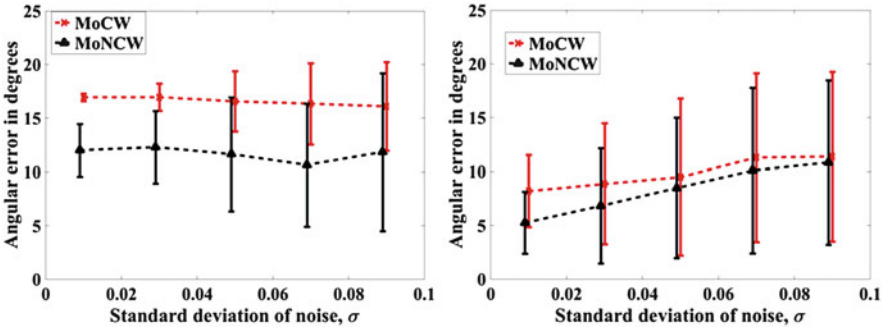


Fig. 8 Mean and standard deviation of angular error in azimuthal angles ϕ when fibers are orientated at (*left*) $(\phi_1, \phi_2, \phi_3) = (10°, 30°, 60°)$ and (*right*) $(\phi_1, \phi_2, \phi_3) = (10°, 60°, 120°)$

Table 2 Mean and standard deviation values of angular errors in degrees with increasing noise levels, σ

MoCW						
σ	0	0.01	0.03	0.05	0.07	0.09
SNR	*inf*	19.1	14.3	12.1	10.6	9.52
1 fiber	0.1	0.2 ± 0.1	0.3 ± 0.2	0.4 ± 0.3	0.6 ± 0.5	0.8 ± 0.9
2 fibers	8.7	9.1 ± 1.3	8.2 ± 2.7	7.8 ± 3.5	6.7 ± 4.0	6.4 ± 4.7
	6.8	6.1 ± 1.4	6.5 ± 2.5	6.8 ± 2.8	6.7 ± 3.2	6.6 ± 4.2
3 fibers	22.6	22.6 ± 0.0	22.4 ± 1.4	21.7 ± 3.2	19.7 ± 5.6	18.2 ± 7.3
	2.60	2.60 ± 0.0	4.2 ± 4.1	5.9 ± 5.5	7.1 ± 6.4	10.5 ± 8.4
	25.0	25.0 ± 0.2	26.0 ± 3.6	25.3 ± 6.4	23.1 ± 8.7	19.6 ± 9.9
MoNCW						
σ	0	0.01	0.03	0.05	0.07	0.09
SNR	*inf*	19.1	14.3	12.1	10.6	9.52
1 fiber	0.5	0.6 ± 0.1	0.6 ± 0.2	0.7 ± 0.6	1.0 ± 0.8	1.4 ± 1.0
2 fibers	2.5	2.5 ± 0.0	2.5 ± 0.9	3.2 ± 2.3	3.0 ± 2.8	3.6 ± 2.9
	3.5	3.7 ± 0.7	3.6 ± 1.5	3.5 ± 1.7	4.0 ± 2.9	4.4 ± 3.3
3 fibers	12.6	13.4 ± 1.6	13.0 ± 2.3	12.0 ± 4.8	12.2 ± 5.9	11.4 ± 7.2
	7.7	14.3 ± 8.6	15.1 ± 8.5	15.1 ± 8.9	15.3 ± 9.0	19.3 ± 13.3
	13.7	11.9 ± 4.1	9.1 ± 4.5	9.2 ± 5.2	10.1 ± 10.2	14.2 ± 17.3
MoHvMF						
SNR	*inf*	10.1	7.4	6.5	4.8	3.6
1 fiber	0.43	4.15	4.96	7.63	9.94	13.58
2 fibers	6.06	7.56	8.63	10.21	12.87	13.75
3 fibers	14.81	15.49	15.47	17.25	17.85	18.28

Finally, we compared our results with the MoHvMF model [19] as well and the comparison is shown in the lower rows of Table 2. This comparison could not be very detailed as we did not have the mean and standard deviations of errors for the MoHvMF model. But it was possible to compare results in the absence of noise

and also for the single fiber case. The error is comparatively lower with MoNCW model when no noise is added to data for single fiber case. In the presence of noise, the error is never higher than 2° for MoNCW while it is significantly high with MoHvMF model. The results of the two models, MoNCW and MoHvMF, could be compared for two crossing fibers with the separation angle of 70° as well due to the availability of MoHvMF results. The results are more stable with MoNCW model in the presence of noise.

Experiments on Simulated Tensor Field Next we applied the MoNCW model to a tensor field simulation with two crossing fiber bundles. The original tensor field and the probability surfaces from MoCW and MoNCW are shown in Figs. 9 and 10 respectively. It can be noticed from the original tensor field that the crossing fibers have large separation angles. As described in earlier sections, the two models do not

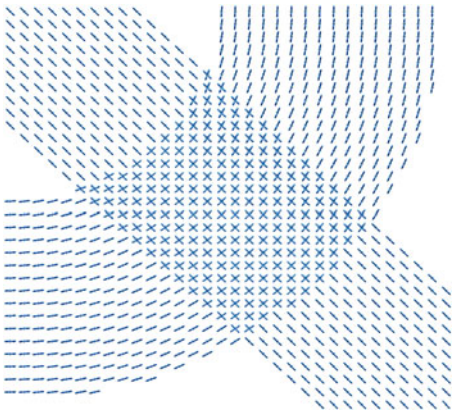


Fig. 9 Original tensor filed of two crossing fiber bundles

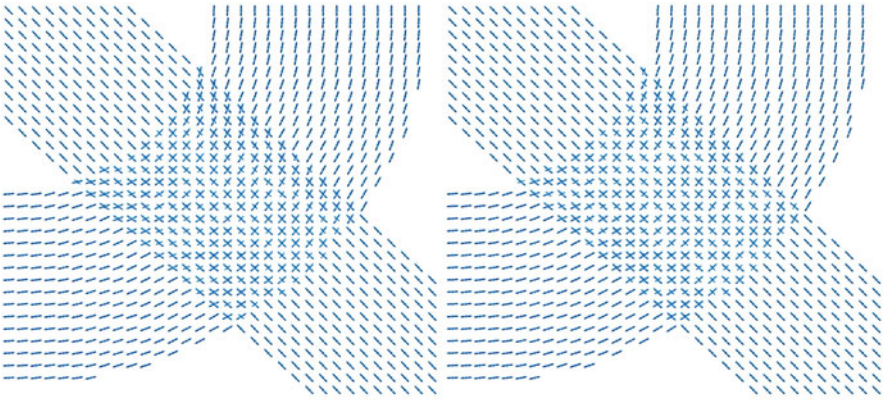


Fig. 10 Probability maps of tensor field of two crossing fiber bundles with MoCW model (*left*) and MoNCW model (*right*)

show much difference in angular errors when separation angle is large. Therefore, we do not notice significant difference in the visualizations of the outcomes of the two models.

Comparison with Other Methods Recently Daducci et al. [5] published a quantitative comparison of 20 reconstruction algorithms for intra-voxel fiber recovery. They proposed to assess the accuracy of fiber reconstruction quantitatively by introducing two measures: angular accuracy and correct assessment of number of fibers within a voxel. The angular accuracy is defined as follows [5]:

$$\bar{\theta} = \frac{180}{\pi} \arccos(|\mathbf{d}_{true} \cdot \mathbf{d}_{estimated}|)$$

where the vectors $\mathbf{d}_{true}$ and $\mathbf{d}_{estimated}$ are true and estimated fiber orientations, respectively, and $(\cdot)$ represents the dot product. We compared the performance of our proposed model with the 20 algorithms reported [5]. We generated a dataset consisting of 910 voxels with two fibers in each voxel crossing at separation angles in the range $(0° - 90°)$. The SNR was set to be 30dB and 10 repetitions were conducted for each crossing fiber angle. We used two different methods namely Gaussian mixture model [5] and adaptive kernels [2] to simulate the dataset. The polar angles for two fibers were kept fixed at 90°. The number of measurements in the data, (K), and the number of mixture components in MoNCW, (N), were set to be 60 and 300, respectively. The dataset was computed for the b-value of 3000 s/mm^2. Figure 11 (top left) includes the angular accuracy plots for 20 algorithms (color coding shown at right) in terms of $\bar{\theta}$ with SNR $=$ 30 and is reproduced from [5, 6]. The angular accuracy for the proposed MoNCW model is given on the bottom left of Fig. 11. It can be observed from the plot that our model performs equivalent to the state-of-the-art for the dataset generated using the adaptive kernels [2]. However, when it comes to the dataset generated using the Gaussian mixture model and which was also used to evaluate the 20 algorithms [5, 6], the MoNCW model shows mixed performance when compared to other models. It performs better than the reported DTI and CSD algorithms for the complete range of separation angles, while in the separation angle ranges of $(0° - 25°)$ and $(45° - 90°)$, it performs better than several algorithms. However, in the separation angle range of $(25° - 45°)$, DSI-based algorithms exhibit better performance than our algorithm. Overall, the angular error for the MoNCW model is quite less, $(5° - 7°)$, when using adaptive kernels to simulate the dataset and is in the mid range $(20° - 50°)$ when using Gaussian mixture model to simulate the dataset. It should be noted that the data acquisition parameters (b-value and number of measurements) are quite different for all 20 algorithms. The error behaviour might be different if these parameters are changed. Also if the adaptive kernels have been used to simulate the signal, one might expect a different error behaviour for 20 algorithms.

Experiments on Real Data Next, we conducted experiments on four different sets of real data. First, we used the rat optic chiasm DW-MRI data set to validate our model. This experiment was performed to test the performance of diffusion

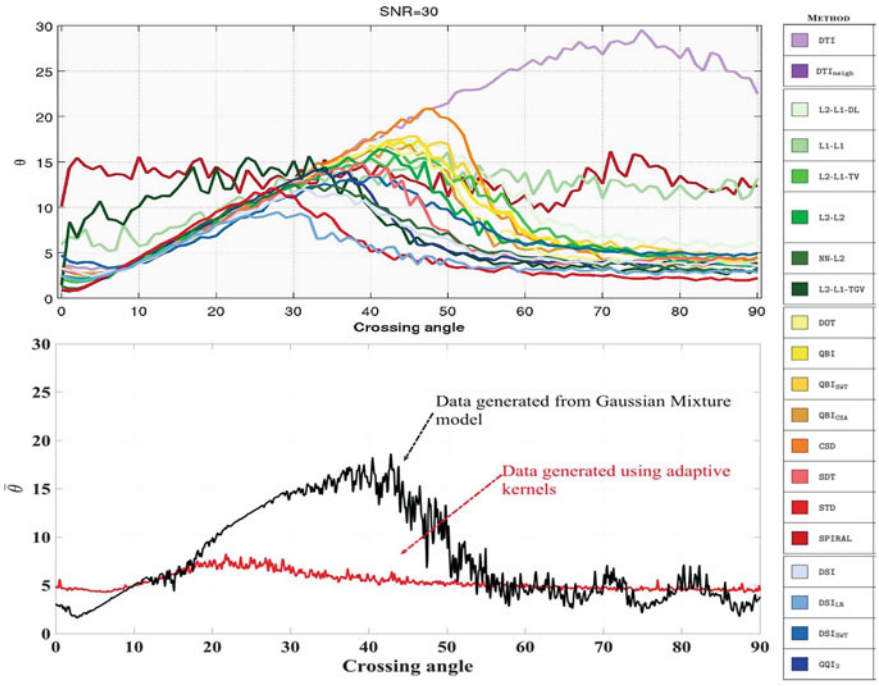


Fig. 11 Angular accuracy in the fiber orientation: comparison of 20 algorithms [5] (*top left*), MoNCW model (*bottom left*) and color coding for various algorithms (*right*)

orientation transform (DOT) [31]. The dataset was downloaded from [2]. The images were acquired for 46 directions with a b-value of 1250 s/mm^2 and one single image was acquired at $b \approx 0$ s/mm^2. The other experimental parameters, TE, TR, Δ, δ and bandwidth were set as 25 ms, 1.17 s, 17.5 ms, 1.5 ms. The reference image and the displacement probabilities are shown in Fig. 12. Our model is able to detect the orientations of optic nerves and their crossings at the center region although the directions of crossing fibers are not always coherent.

Second, we used a two-shell healthy human brain DWI-MRI dataset from the MyConnectome project [20]. The dataset was obtained using a multiband EPI sequence on a Siemens Skyra 3T scanner. Diffusion weighted images were acquired with two b-values of 1000 and 2000 s/mm^2 along 64 gradient directions with 4 volumes without diffusion weighting. The matrix size was 128×128 with 78 slices and the resolution was $1.74 \times 1.74 \times 1.7$ mm^3 and the values of TR, TE and multiband factor were equal to 5000 ms, 108 ms and 3 respectively. We chose the 42nd slice for analyzing brain's white matter tract and used FSL software [12] to calculate the fractional anisotropy images. Figure 13 shows the displacement probabilities computed for corpus callosum splenium (left) and corpus callosum genu (right) of human brain. The calculated probability maps using the proposed MoNCW

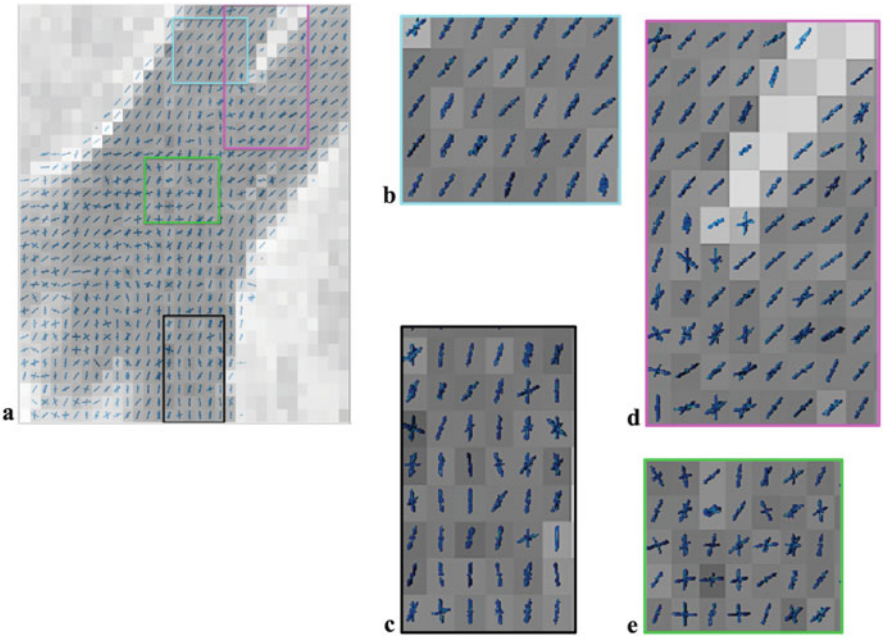


Fig. 12 (**a**) Probability maps for rat optic chiasm overlaid on the reference image, (**b**)–(**e**) the zoomed in versions of regions of interest in (**a**) with the same boundary colors

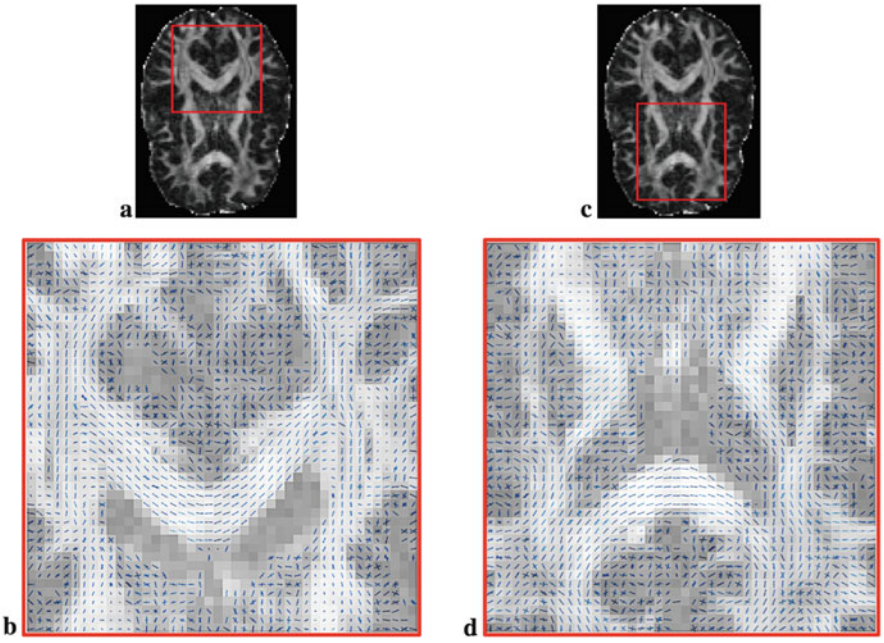


Fig. 13 (**a**) FA map of a human brain MRI slice with corpus callosum splenium enclosed in a box, (**c**) with genu enclosed in a box, (**b**) and (**d**) probability maps for the enclosed regions in (**a**) and (**c**) overlaid on FA maps respectively

model are overlaid on the fractional anisotropy images. These images show that the proposed model is able to extract the orientations of the anisotropic fibers in real data as well. We can also observe that the reconstruction results are more coherent for human brain images as compared to the rat chiasm image given in Fig. 12.

Next, we conducted experiments on three-shell real human brain data. The dataset was acquired from a healthy subject on a customized Siemens 3T 'Connectome Skyra' MR scanner. The interested reader is referred to [41] for details of the dataset such as b-values and scanning parameters. The voxel size was 1.25 mm and the matrix size was 145×174 with 145 slices. We chose the 60th slice for our experiments. Figure 14 includes reconstructed fiber orientations for this particular slice. Figure 14a shows the fractional anisotropy (FA) map of the whole slice where the region of interest enclosed in red box contains the corpus callosum splenium. Figure 14b shows the reconstructed fiber orientations corresponding to the region where major white matter tracts oriented in different directions can be observed. For better visualizion, the FA map of a smaller region of 'kissing fibers' is shown in Fig. 14c and the corresponding reconstructions with MoNCW and MoCW models are shown in Fig. 14d and e, respectively. The highlighted regions show crossing

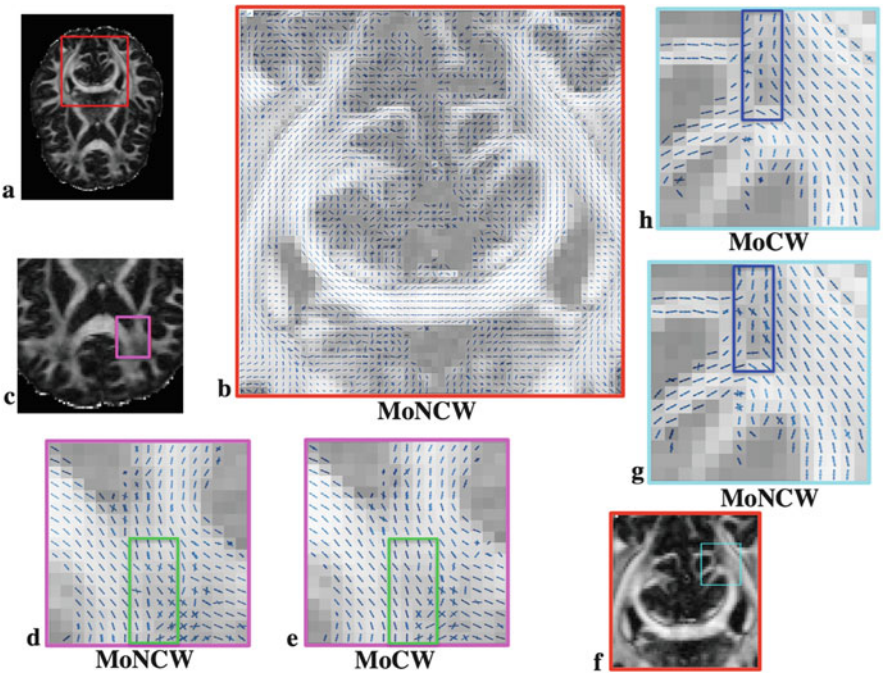


Fig. 14 (**a**) FA map of a human brain MRI slice with corpus callosum splenium enclosed in a box, (**b**) probability maps overlaid on FA maps for the region of interest in (**a**), (**c**) FA map for a smaller region in the corpus callosum, (**d**)–(**e**) probability maps for the region of interest in (**c**) reconstructed with MoNCW and MoCW respectively, (**f**) FA map for another smaller region, (**g**)–(**h**) probability maps for the region of interest in (**f**) reconstructed with MoNCW and MoCW respectively

fibers and it can be observed that the MoNCW model is able to capture the presence of two fibers at severals voxels with accurate fiber dispersions in those voxels. Another smaller region with crossing fibers is highlighted enclosed in a blue box in Fig. 14f. The corresponding probability maps for the selected region reconstructed with MoNCW and MoCW models are shown in Fig. 14g and h, respectively. It can be observed that the proposed model is better able to identify the presence and orientation of crossing fibers.

Finally, we conducted experiments on four-shell dataset [7, 8]. The dataset was collected via the customized Siemens 3T Connectom scanner (MAGNETOM Skyra Siemens Healthcare) from healthy adults. Diffusion scans were obtained at four different b-values of 1000, 3000, 5000 and 10,000 s/mm^2 and the number of diffusion directions were 64, 64, 128 and two sets of 128 for the four aforementioned b-values, respectively. The interested reader is referred to [7, 8] for details of the dataset. We chose the 50th slice for our experiments. Since the value of K was $= 552$ for the dataset, we chose the value of N to be 569 to avoid under determined system. Figure 15a and e are the FA maps of the whole slice with regions of corpus callosum genu and splenium enclosed in red and magenta boxes. The reconstructed

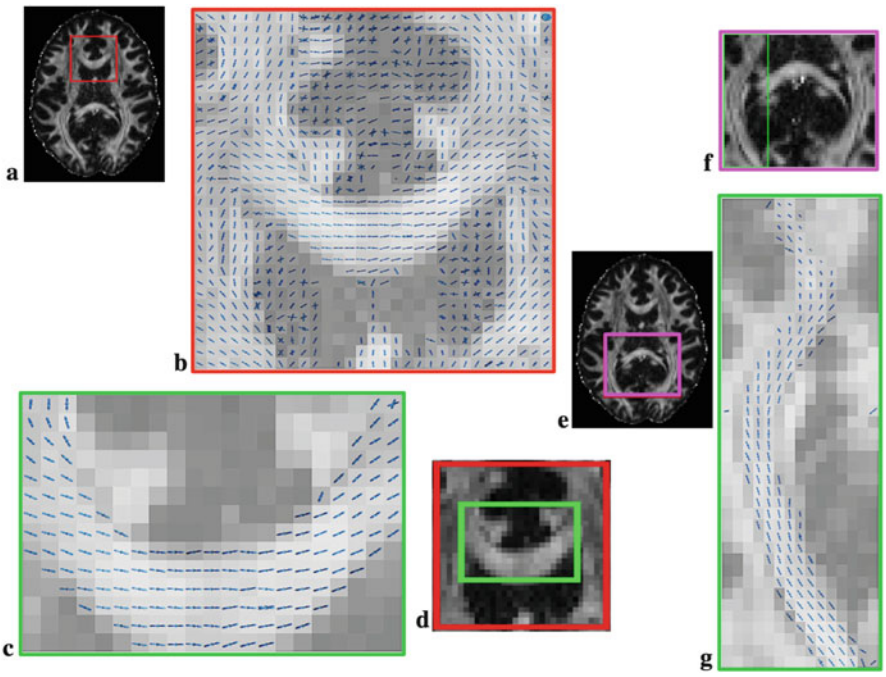


Fig. 15 (**a**) FA map of a human brain MRI slice with corpus callosum genu enclosed in a box, (**b**) probability map for the region of interest in (**a**) overlaid on the FA maps, (**c**) probability maps overlaid on FA map for the region of interest in (**d**), (**d**) the FA map for a smaller region of interest, (**e**) FA map of a human brain MRI slice with corpus callosum splenium enclosed in a box, (**f**) the FA map for a smaller region of interest and (**g**) the reconstructed probability maps for the regions of interest shown in (**f**)

probability map corresponding to the region of corpus callosum genu is shown in Fig. 15b. The bundles of straight, bending and crossing fibers are visible. Figure 15d and f include smaller sections of the slice with the corresponding reconstructed fiber tracts in Fig. 15c and g. The changing orientation and intersections of the fiber tracts is clearly visible.

4 Conclusion

In this work, we presented a new probabilistic model, namely, the mixture of non-central Wishart distributions to model intra-voxel fiber heterogeneity of white matter of brain. The proposed model is general in the sense that the center of the distribution need not to be at zero. We proposed a way to estimate the newly introduced non-centrality model parameters Ω. Our experiments on the simulated and real data including a rat optic chiasm and healthy human brain show that the proposed model is promising specially in distinguishing crossing fibers with smaller separation angles and in the presence of noise when compared to the previous models of mixture of central Wishart distribution and hyper-spherical von Mises-Fisher distributions.

Acknowledgements This research is a part of the project "Seeing Organ Function" funded by "The Knut and Alice Wallenberg Foundation". EÖ acknowledges support by Linköping University Center for Industrial Information Technology (CENIIT). We would also like to thank Russell Poldrack and his colleagues for sharing diffusion MRI data from the project MyConnectome. The HCP Data were provided (in part) by the Human Connectome Project, WU-Minn Consortium (Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54MH091657) funded by the 16 NIH institutes and centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University. The MGH data were provided (in part) by the Human Connectome Project, MGH-USC Consortium (Principal Investigators: Bruce R. Rosen, Arthur W. Toga and Van Wedeen; U01MH093765), funded by the NIH Blueprint Initiative for Neuroscience Research grant; the National Institutes of Health grant P41EB015896; and the Instrumentation Grants S10RR023043, 1S10RR023401, 1S10RR019307.

References

1. Barmpoutis, A.: Adaptive Kernels for multi-fiber reconstruction. Inf. Proces. Med. Imaging **21**, 338–349 (2009)
2. Barmpoutis, A.: Tutorial on Diffusion Tensor MRI using Matlab. Electronic edn. University of Florida, Gainesville (2010)
3. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophys. J. **66**, 259–267 (1994)
4. Bhalerao, A., Westin, C.F.: Hyperspherical von Mises-Fisher mixture (HvMF) modelling of high angular resolution diffusion MRI. Med. Image Comput. Comput. Assist. Interv. **10**(Pt 1), 236–243 (2007)

5. Daducci, A., Canales-Rodriguez, E.J., Descoteaux, M., Garyfallidis, E., Gur, Y., Lin, Y.-C., Mani, M., Merlet, S., Paquette, M., Ramirez-Manzanares, A., Reisert, M., Rodrigues, P., Sepehrband, F., Caruyer, E., Choupan, J., Deriche, R., Jacob, M., Menegaz, G., Prckovska, V., Rivera, M., Wiaux, Y., Thiran, J.-P.: Quantitative comparison of reconstruction methods for Intra-Voxel fiber recovery from diffusion MRI. IEEE Trans. Med. Imaging **33**(2), 384–399 (2014)
6. EPFL, Switzerland: ISBI 2012 – HARDI reconstruction workshop [Online] Available: http://hardi.epfl.ch/static/events/2012_ISBI/download.html (2012)
7. Fan, Q., Nummenmaa, A., Witzel, T., Zanzonico, R., Keil, B., Cauley, S., Polimeni, J.R., Tisdall, D., Van Dijk, K.R., Buckner, R.L., Wedeen, V.J., Rosen, B.R., Wald, L.L.: Investigating the capability to resolve complex white matter structures with high b-value diffusion magnetic resonance imaging on the MGH-USC Connectom scanner. Brain Connect. **4**(9), 718–726 (2014)
8. Fan, Q., Witzel, T., Nummenmaa, A., Van Dijk, K.R., Van Horn, J.D., Drews, M.K., Somerville, L.H., Sheridan, M.A., Santillana, R.M., Snyder, J., Hedden, T., Shaw, E.E., Hollinshead, M.O., Renvall, V., Zanzonico, R., Keil, B., Cauley, S., Polimeni, J.R., Tisdall, D., Buckner, R.L., Wedeen, V.J., Wald, L.L., Toga, A.W., Rosen, B.R.: MGH-USC Human Connectome Project datasets with ultra-high b-value diffusion MRI. Neuroimage **124**(Pt B), 1108–1114 (2016)
9. Gindikin, S.G.: Invariant generalized functions in homogeneous domains. Funct. Anal. Appl. **9**, 50–52 (1975)
10. Inglis, B.A., Bossart, E.L., Buckley, D.L., Wirth, E.D., Mareci, T.H.: Visualization of neural tissue water compartments using biexponential diffusion tensor MRI. Magn. Reson. Med. **45**, 580 (2001)
11. James, A.T.: The non-central Wishart distribution. R. Soc. Lond. Ser. A Math. and Phys. Sci. **229**(1178), 364–366 (1955)
12. Jenkinson, M., Beckmann, C.F., Behrens, T.E.J., Woolrich, M.W., Smith, S.M.: FSL. Neuroimage **62**(2), 782–790 (2011)
13. Jian, B., Vemuri, B.C.: Multi-fiber reconstruction from diffusion MRI using mixture of Wisharts and sparse deconvolution. Int. Conf. Inf. Process. Med. Imaging **20**, 384–395 (2007)
14. Jian, B., Vemuri, B.C., Özarslan, E., Carney, P.R., Mareci, T.H.: A novel tensor distribution model for the diffusion-weighted MR signal. NeuroImage **37**, 164–176 (2007)
15. Jian, B., Vemuri, B.C., Özarslan, E.: A mixture of Wisharts (MOW) model for multi-fiber reconstruction. In: Visualization and Processing of Tensor Fields, vol. 9, pp. 39–55. Springer, Berlin (2012)
16. Jones, D.K., Leemans, A.: Diffusion tensor imaging. Methods Mol. Biol. **11**, 127–144 (2011)
17. Knutsson, H.: Producing a continuous and distance preserving 5-D vector representation of 3-D orientation. In: Proceedings of IEEE Computer Society Workshop on Computer Architecture for Pattern Analysis and Image Database Management, vol. 175, p. 182 (1985)
18. Kumar, R., Barmpoutis, A., Vemuri, B.C., Carney, P.R., Mareci, T.H.: Multi-fiber reconstruction from DW-MRI using a continuous mixture of von Mises-Fisher distributions. In: Mathematical Methods in Biomedical Image Analysis (MMBIA), pp. 1–8 (2008)
19. Kumar, R., Vemuri, B.C., Wang, F., Syeda-Mahmood, T., Carney, P.R., Mareci, T.H.: Multi-fiber reconstruction from DW- MRI using a continuous mixture of hyperspherical von Mises-Fisher distributions. Inf. Proces. Med. Imaging **5636**, 139–150 (2009)
20. Laumann, T.O., Gordon, E.M., Adeyemo, B., Snyder, A.Z., Joo, S.J., Chen, M.Y., Gilmore, A.W., McDermott, K.B., Nelson, S.M., Dosenbach, N.U., Schlaggar, B.L., Mumford, J.A., Poldrack, R.A., Petersen, S.E.: Functional system and areal organization of a highly sampled individual human brain. Neuron **87**, 657–670 (2015)
21. Lawson, C.L., Hanson, R.J.: Solving Least-Squares Problems, Chapter 23, p. 161. Upper Saddle River, NJ, Prentice Hall (1974)
22. Leow, A.D., Zhu, S., Zhan, L., McMahon, K., de Zubicaray, G.I., Meredith, M., Thompson, P.M.: The tensor distribution function. Magn. Reson. Med. **61**(1), 205–214 (2009)

23. Letac, G., Massam, H.: Quadratic and inverse regressions for Wishart distributions. Ann. Stat. **26**(2), 573–595 (1998)
24. Letac, G., Massam, H.: A tutorial on noncentral Wishart distributions. Laboratory of Statistics and Probability, Toulouse, France (2004)
25. Li, K., Zhi, G.: The noncentral Wishart distribution and related distributions. Commun. Stat. Theory Methods **32**(1), 33–45 (2003)
26. Luypaert, R., Boujraf, S., Sourbron, S., Osteaux, M.: Diffusion and perfusion MRI: basic physics. Eur. J. Radiol. **38**, 19–27 (2001)
27. Mayerhofer, E.: On the existence of non-central Wishart distributions. J. Multivar. Anal. **114**, 448–456 (2013)
28. McGraw, T., Vemuri, B.C., Yezierski, B., Mareci, T.: Von Mises-Fisher mixture model of the diffusion ODF. In: Proceedings of IEEE ISBI, pp. 65–68 (2006)
29. Mori, S., Barker, P.B.: Diffusion magnetic resonance imaging: its principle and applications. Anat. Rec. **257**, 102–109 (1999)
30. Mori, S., Zhang, J.: Principles of diffusion tensor imaging and its applications to basic neuroscience research. Neuron **51**, 527–539 (2006)
31. Özarslan, E., Shepherd, T.M., Vemuri, B.C., Blackband, S.J., Mareci, T.H.: Resolution of complex tissue microarchitecture using the diffusion orientation transform (DOT). NeuroImage **31**(3), 1086–1103 (2006)
32. Peddada, S.D., Richards, D.S.: Proof of a conjecture of M. L. Eaton on the characteristic function of the Wishart distribution. Ann. Probab. **19**, 868–874 (1991)
33. Pham-Gia, T., Thanh, D., Phong, D.: Trace of the Wishart matrix and applications. Open J. Stat. **5**, 173–190 (2015)
34. Rigaut, J.P.: An empirical formulation relating boundary lengths to resolution in specimens showing "non-ideally fractal" dimensions. J. Microsc. **133**, 41–54 (1984)
35. Scherrer, B., Schwartzman, A., Taquet, M., Sahin, M., Prabhu, S.P., Warfield, S.K.: Characterizing brain tissue by assessment of the distribution of anisotropic microstructural environments in diffusion-compartment imaging (DIAMOND). Magn. Reson. Med. **76**, 963–977 (2016)
36. Sen, P.N., Hürlimann, M.D., de Sweit, T.M.: Debye-Porod law of diffraction for diffusion in porous media. Phys. Rev. B **51**(1), 601–604 (1995)
37. Shanbhag, D.: The Davidson-Kendall Problem and related results on the structure of the Wishart distribution. Aust. J. Stat. **30**(A), 272–280 (1988)
38. Tuch, D.S., Weisskoff, R.M., Belliveau, J.W., Wedeen, V.J.: High angular resolution diffusion imaging of the human brain. In: 7th ISMRM, p. 321 (1999)
39. Tuch, D.S., Reese, T.G., Wiegell, M.R., Makris, N., Belliveau, J.W., Wedeen, V.J.: High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. Magn. Reson. Med. **48**, 577–582 (2002)
40. Tuch, D.S., Reese, T.G., Wiegell, M.R., Wedeen, V.J.: Diffusion MRI of complex neural architecture. Neuron **40**, 885–895 (2003)
41. Van Essen, D.C., Ugurbil, K., Auerbach, E., Barch, D., Behrens, T.E.J., Bucholz, R., Chang, A., Chen, L., Corbetta, M., Curtiss, S.W., Penna, S.D., Feinberg, D., Glasser, M.F., Harel, N., Heath, A.C., Larson-Prior, L., Marcus, D., Michalareas, G., Moeller, S., Oostenveld, R., Petersen, S.E., Prior, F., Schlaggar, B.L., Smith, S.M., Snyder, A.Z., Xu, J., Yacoub, E., WU-Minn, H.C.P., Consortium: The human connectome project: a data acquisition perspective. NeuroImage **62**, 2222–2231 (2012)
42. Wedeen, V.J., Reese, T.G., Tuch, D.S., Weigell, M.R., Dou, J.G., Weiskoff, R.M., Chessler, D.: Mapping fiber orientation spectra in cerebral white matter with Fourier-transform diffusion MRI. In: 8th ISMRM, p. 82 (2000)
43. Westin, C.F., Szczepankiewicz, F., Pasternak, O., Özarslan, E., Topgaard, D., Knutsson, H., Nilsson, M.: Measurement tensors in diffusion MRI: generalizing the concept of diffusion encoding. Med. Image Comput. Assist. Interv. **17**(03), 209–216 (2014)

Part IV
Tractography

Edge Detection in Diffusion Weighted MRI Using a Tangent Curve Similarity Metric

Zi'Ang Ding, Xavier Tricoche, and Yaniv Gur

Abstract We present a technique to automatically characterize the geometry of important anatomical structures in diffusion weighted MRI (DWI) data. Our approach is based on the interpretation of diffusion data as a superimposition of multiple line fields that each form a continuum of space filling curves. Using a dense tractography computation, our method quantifies the spatial variations of the geometry of these curves and use the resulting measure to characterize salient structures as edges. Anatomically, these structures have a boundary-like nature and yield a clear picture of major fiber bundles. In particular, the application of our algorithm to high angular resolution imaging (HARDI) data yields a precise geometric description of subtle anatomical configurations associated with the local presence of multiple fiber orientations. We evaluate our technique and study its robustness to noise in the context of a phantom dataset and present results obtained with two diffusion weighted brain images.

1 Introduction

Diffusion weighted imaging (DWI) is a medical imaging technique that measures the anisotropic Brownian motion of water molecules in fibrous tissues and enables their in-vivo investigation. The modeling of the measured multidirectional diffusion information through a second-order tensor, known as diffusion tensor MRI (or DTI), is an important tool for the analysis of the brain's white matter structure [2, 3, 22] and the heart's myocardium [18, 19, 36, 41]. Yet, the gaussian diffusion model used in DTI is unable to adequately model complex diffusion patterns that are common in the white matter such as crossing, fanning, or bent fibers. In such cases, an alternative imaging modality known as high angular resolution diffusion imaging

Z. Ding • X. Tricoche (✉)
Purdue University, West Lafayette, IN, USA
e-mail: ding29@purdue.edu; xmt@purdue.edu

Y. Gur
IBM Research, San Jose, CA, USA
e-mail: guryaniv@us.ibm.com

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_13

(HARDI) proves superior. In HARDI, the measured information is amenable to an orientation distribution function (ODF) that may be described by a higher-order tensor [16, 20, 33] or a band-limited expansion of spherical harmonics [14, 37, 38], which allows one to identify several co-existing significant diffusion directions within a voxel.

Irrespective of the considered diffusion model, the interpretation of the resulting images requires the challenging analysis of a high-dimensional data space. The two main approaches used to facilitate this task are tractography and scalar measures. In tractography one integrates along the dominant diffusion direction(s) to approximate fiber tracts and derive a white matter connectivity map. Both fiber tracts geometry and corresponding connectivity map, in turn, can be used to identify major fiber bundles, which has various clinical applications [39]. The second approach exploits scalar measures derived from the diffusion data, such as fractional anisotropy (FA), or Generalized FA (GFA), for segmentation and analysis [24, 25, 34].

In recent years significant advances in the structural analysis of diffusion tensor fields have been achieved through the extraction of so-called ridge and valley (jointly, *crease*) manifolds from tensor invariants [23, 25]. While crease manifolds have proven successful at characterizing major white matter structures in DTI, no similar investigation was carried out in the context of HARDI problems. Following a different approach, methods considering the end positions of fiber traces in a dense tractogram have been shown to characterize interesting anatomical structures in DTI datasets of the brain's white matter and the heart's myocardium [15, 17].

The approach presented in this paper builds upon a new model of the boundaries of anatomical structures in diffusion weighted MRI as edges of a continuous mapping between spatial locations and the geometric signature of the fiber traces that run through them. By adapting to this geometry-valued setting edge detection techniques devised for scalar images, our method is able to properly characterize subtle anatomical structures in both DTI and HARDI.

Our work advances the state of the art in three significant ways. First, our edge strength measurement is fundamentally nonlocal while prior methods that consider scalar invariants [23, 25] focus on local properties. Second, in contrast to methods that focus on the end points of fiber traces [15, 17], we do not rely on any particular model of curve separation to measure fiber distances and derive a spatial gradient. Third, unlike fiber clustering methods [5, 6, 32], we are not interested in forming bundles from a discrete set of fibers though we are able to explicitly characterize the geometric structures that form the boundaries of fiber bundles.

The main contributions of this paper are

- A novel model that defines structure boundaries as edges of a fiber-valued mapping;
- A tractography-based edge detection method that extracts structures from diffusion weighted MRI;
- A simple conceptual framework applicable both to DTI and HARDI data, that performs well in regions with challenging fiber structures.

The rest of this paper is organized as follows. Section 2 summarizes related work. The details of our edge detection method are provided in Sect. 3 while Sect. 4 presents a number of visualization approaches derived from the measured edge strength to improve the understanding of the anatomical structure information from the DWI signals; Sect. 5 documents the results of our edge detection method on several DWI datasets; and Sect. 6 presents our conclusions and discusses future work.

2 Related Work

We briefly review in the following relevant prior work in DTI and HARDI visualization and analysis.

2.1 Tractography in DTI and HARDI

Tractography is a technique that estimates the trajectories of neural tracts from diffusion weighted MRI data. It provides an effective way to model and analyze the fiber tracts in the white matter, and has further been used to study the structure and connectivity of the human brain [21, 26]. Both in the visualization and medical imaging communities, many methods have been proposed based on the streamline algorithm to perform the tractography in DTI. Assuming the major eigenvector is parallel to the local fiber orientation in each voxel, it is possible to integrate a pathway using numerical integration methods include Euler's method or Runge-Kutta method [4, 8, 31]. Later, several methods using the local diffusion tensor to deflect the incoming direction instead of the major eigenvector are introduced to solve the problem when tracking trough regions of planar anisotropy [27, 47]. Also, streamtubes and streamsurfaces were used to visualize diffusion weighted MRI data [48]. Moreover a MLS-based regularization technique was used to allow tracking to cross noisy regions and gaps [51].

Tracking fibers in higher order tensor (HOT) was first proposed Hlawitschka and Scheuermann as HOT-lines [16]. Schultz and Seidel [33] and Jiao et al. [20] later improved these kind of techniques by introducing tensor decomposition methods to find the local orientations in each step during the integration.

2.2 Diffusion Weighted MRI Analysis

In the medical imaging community, a number of clustering methods which group fiber tracts into anatomical meaningful bundles were used to analyze and investigate information from diffusion weighted MRI. O'Donnell et al. [32] presented a fiber grouping approach that delineates fiber tracts that can be further analyzed for

clinical research purposes. Brun et al. [6] proposed a fiber clustering method to create a weighted undirected graph by comparing fiber tracts pairwise, and perform segmentation in high dimensional space. Also, the same author used laplacian eigenmaps to create a mapping from DTI fiber tracts to a low dimensional Euclidean space, thereby enabling a color coding of fiber tracts that enhances the perception of fiber bundles and connectivity in the human brain [5]. Liang et al. [28] introduced a technique to group fiber tracts into bundles using Nonegative Matrix Factorization (NMF) of the frequency-tract matrix. Mai et al. [30] proposed a method to segment fiber tracts based on a shape similarity measure. To that end they introduced a new technique called Warped Longest Common Subsequence (WLCS), which was used to speed up the segmentation process. Instead of explicitly delineating anatomical structures such as boundaries between fiber bundles, all these fiber clustering methods are interested in organizing a discrete set of fibers into bundles.

In the scientific visualization community, researchers have applied ridge and edge detection methods which were originally developed by computer vision community to the analysis of diffusion weighted MRI data. Kindlmann et al. [23, 25] applied crease surfaces of FA to characterize important anatomical structures in the brain. Extension of this work to ridge lines of FA as models of core lines in fiber bundles was discussed in Tricoche et al. [40]. By defining gradients of shape invariants and rotation tangents [24], Schultz and Seidel successfully extended image processing techniques such as edge detection to diffusion tensor images [34]. However, all these methods focus on local properties and are not able to reveal structures in sub-voxel level.

Also Schultz et al. considered tensor topology [10, 49] in the context of DTI [35] but found the results to lack a clear interpretation. Instead they proposed an alternative topological definition for DTI [35].

In a recent study, a generalized framework for creating super-resolution track-weighed imaging (TWI) was introduced [7]. The intensity of an individual pixel on the resulting image could be determined by a specific property, such as the fractional anisotropy (FA), of the tensorlines which traverse this pixel or the spatial coordinates of those tensorlines.

Most germane to the ideas developed in this paper are recent works applying to tensor field and DTI visualization a technique previously used in flow visualization. Specifically, Hlawitschka et al. [17] and Hlawatsch et al. [15] proposed to use the rate of separation of neighboring fiber tracts as a measure of coherence in DTI volumes. The resulting scalar quantity was able to show certain anatomical structures in human brain and in dog heart.

3 Method

We aim to extract the boundaries of individual fiber bundles as edges of a fiber trace-valued image that we derive from the DWI dataset via dense tractography. First, we wish to motivate some of the choices made in the design of our method by briefly discussing edge detection.

3.1 Edges as Ridges

Edges are fundamental image descriptors in image processing and computer vision and many different techniques have been devised for their extraction [29]. A commonly used approach in that context characterizes edges in a 2-step process: it first computes at each pixel an edge strength measure from which the edge geometry can then be obtained as curves (or surfaces in 3D) along which that edge strength is locally largest.

While edge strength can be measured in scalar images in a variety of ways, we only consider here the simplest possible definition, namely gradient magnitude. Leaving aside for the time being the question of how to robustly compute this gradient, we can see that identifying edges in fiber-valued images necessitates a metric to measure distances, that is dissimilarities between neighboring fibers, which in turn requires the choice of a fiber encoding that lends itself to meaningful distance measures. Once a suitable edge strength has been computed across the dataset, the geometry of the edges can be extracted as ridges of the corresponding field [11].

3.2 Fiber-Valued Image Computation

To create a fiber-valued volume, we first compute a dense, full brain tractogram. The 2nd-order Runge-Kutta method [4] is used to integrate tangent curves along the major eigenvector of the diffusion tensor in DTI datasets. To increase the robustness of our integration, a moving least-squares regularization procedure first proposed by Zhukov and Barr [51, 53] is applied to the tensor field along the integration path. This procedure has the double benefit of increasing the robustness of the integration to the noise inherently present in the data and also to partially mitigate the limitations of the tensor model in regions exhibiting fiber crossing. Indeed, similar to the *tensorline* method [47], the MLS regularization effectively uses the shape of the previously computed filtered tensor value along the curve to constrain the range of directions that the next integration step may take.

Once the integration has been performed, each voxel is assigned an array of 3D positions that describe the geometry of the computed fiber. For the need of subsequent processing, however, a different fiber encoding is needed.

3.3 Feature Encoding

With about a hundred vertices per fiber trace on average, the information associated with each voxel is expressed in a fairly high-dimensional data space. In that space, the straightforward Eulerean metric is both costly to compute and ineffective as dissimilarity measure. Furthermore, the numerical criteria used to control the

progression of the integration (e.g., a lower bound on fractional anisotropy as stop criterion), produce fibers with a varying number of vertices.

To avoid these issues, we map the raw geometric information produced by the fiber tracking step to a low-dimensional representation comprised of the first and second moments of each fiber description [6]. The corresponding set of coefficients f is given in Eq. (1).

$$\mathbf{f} = (m_x, m_y, m_z, h_{xx}, h_{xy}, h_{xz}, h_{yy}, h_{yz}, h_{zz})^\intercal, \tag{1}$$

where $\mathbf{m} = (m_x, m_y, m_z)^\intercal$ is the mean vector of the vertices in 3D space, and the terms $h_{\cdot,\cdot}$ are the independent coefficients of the (symmetric) square root of their covariance matrix $\mathbf{H}$. As a result, each fiber trace is represented by only 9 coefficients that are invariant under flipped fiber orientation. In all datasets we have tested in Sect. 5, this simplification is good enough to capture the geometry of fiber traces. For applications which the first and the second order moment is not enough to distinguish the geometric difference between neighboring fiber traces, any higher order moment could be employed to provide more dimensions in the feature space.

3.4 Edge Strength in Vector-Valued Images

The previous steps of the algorithm yield a volume dataset that associates each data point with a 9D feature vector. To detect edges in this vector-valued image, we need to evaluate its gradient. Our solution consists in computing a linear least squares fit over the 26 neighbors of each voxel. Let f_p denote the feature vector associated with the voxel at position p, the desired linear fit at p is the solution of the following expression:

$$\mathbf{A}_p = \min_{\forall \mathbf{A} \in \mathbb{R}^{9\times 3}} \sum_{q \in N^1(p)} || \mathbf{A}\,(q-p) - \left(\mathbf{f}_q - \mathbf{f}_p\right) ||^2. \tag{2}$$

Here N^1 designates the 1-neighborhood of p, which is comprised of its 26 direct neighbors.

While the corresponding solution $\mathbf{A}_p$ could be used directly as approximation of the gradient $\nabla \mathbf{f}_p$, we adopt in this work a more robust approach. Following prior work on tensor-based feature detection in color images [45], we frame our gradient estimation problem as the construction of a structure tensor on a multichannel volume with 9 parameters. Here, our structure tensor $\mathbf{G}$ is given by

$$\mathbf{G} = \begin{pmatrix} \overline{\mathbf{f}_x^\intercal \mathbf{f}_x} & \overline{\mathbf{f}_x^\intercal \mathbf{f}_y} & \overline{\mathbf{f}_x^\intercal \mathbf{f}_z} \\ \overline{\mathbf{f}_y^\intercal \mathbf{f}_x} & \overline{\mathbf{f}_y^\intercal \mathbf{f}_y} & \overline{\mathbf{f}_y^\intercal \mathbf{f}_z} \\ \overline{\mathbf{f}_z^\intercal \mathbf{f}_x} & \overline{\mathbf{f}_z^\intercal \mathbf{f}_y} & \overline{\mathbf{f}_z^\intercal \mathbf{f}_z} \end{pmatrix}, \tag{3}$$

where the partial derivatives $\mathbf{f}_x$, $\mathbf{f}_y$, and $\mathbf{f}_z$ at p are obtained directly from A_p as its 1st, 2nd, and 3rd columns. The bar $\bar{\cdot}$ indicates the convolution with a Gaussian filter for increased robustness.

Finally, the edge strength s can be measured by a scalar invariant of the structure tensor. Specifically, in our approach, we measure the edge strength as the major eigenvalue of the structure tensor.

$$s = \lambda_{max}(\mathbf{G}) \tag{4}$$

3.5 Extension to HARDI

The DT-MRI model has been shown to fail in regions containing multiple distinct orientations [1, 12, 42]. The HARDI model presented by Tuch et al. [43, 44] extends the single-tensor model to multi-tensor models or ODFs which can capture multiple independent fiber orientations in each voxel, thereby allowing one to track fibers across regions of complex crossing white-matter structures.

In this work, we define the nth higher order tensor which represents the estimated fiber-ODF from DWI signals as follows.

$$\mathbf{D} = \sum_{r=1}^{R} \mathbf{v}_r^1 \otimes \mathbf{v}_r^2 \otimes \cdots \otimes \mathbf{v}_r^n, \tag{5}$$

where $\mathbf{v}$ are vectors and $R = rank(\mathbf{D})$. Further, following prior work we assume that the tensors are supersymmetric, i.e., $\mathbf{v}_r^1 = \mathbf{v}_r^2 = \cdots = \mathbf{v}_r^n$. Unlike the DTI case, the rank of the estimated higher order tensor is unknown. Assuming we are interested in a low-rank approximation of the original higher order tensor, for a given rank $k < R$, $\mathbf{D}$ can be decomposed into k rank-1 tensors.

$$\mathbf{D} \approx \sum_{r=1}^{k} w_r \left(\mathbf{v}_r^1 \otimes \mathbf{v}_r^2 \otimes \cdots \otimes \mathbf{v}_r^n\right), \tag{6}$$

where $\|\mathbf{v}_r\| = 1$, and $\mathbf{v}_r$ represent the possible fiber orientations at the location where $\mathbf{D}$ is estimated. The number k is determined through a simple heuristic method described in [33].

Here we extend our edge detection approach discussed in the DTI Sect. 3.4 to HARDI model and improve the edge detection results by accounting for the presence of multiple orientations in voxels.

For tractography we follow the deterministic higher-order tensor tracking algorithm presented by Hlawitschka and Scheuermann [16] and Schultz and Seidel [33]. More specifically, at each step of the fiber integration, the $rank - k$ decomposition approach is employed to extract the possible fiber orientations from the higher-order

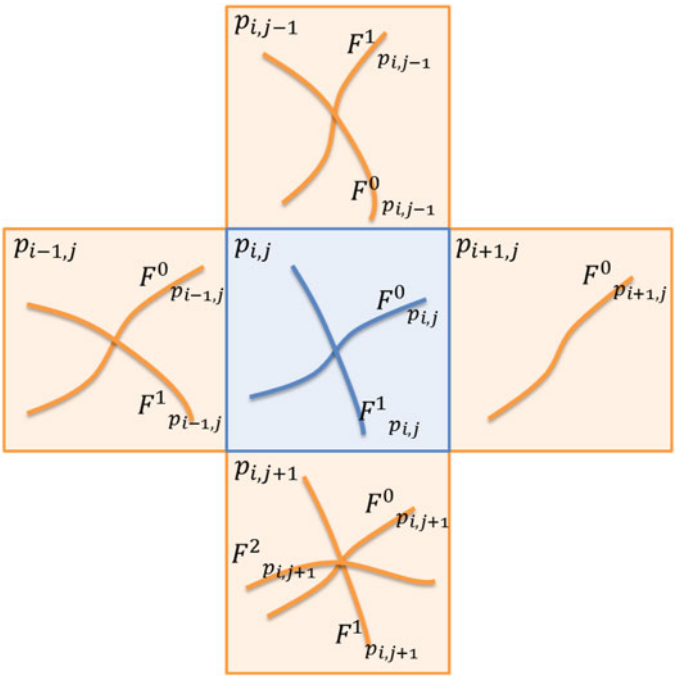


Fig. 1 Case in which neighbor voxels have different number of fibers

tensor. Then an evaluation scheme is applied to find the best orientation, defined as the one forming the smallest angle with the last integration step.

Similar to fiber tracking in DT-MRI, we perform a MLS-based regularization to filter the higher order tensor at each integration step, namely we compute the weighted average of the higher order tensor value within the filter kernel which in this case is defined by the local diffusion ODF.

Unlike DTI, the result of our fiber tracking method on HARDI, for a particular voxel p, could contain M fibers F_p^0, F_p^1, …, F_p^{M-1} corresponding to distinct local orientations $\mathbf{o}_p^0$, $\mathbf{o}_p^1$, …, $\mathbf{o}_p^{M-1}$. Figure 1 illustrates the case in which a different number of fibers is found in voxel $p_{i,j}$ and its neighborhood in 2D.

Since multiple local line fields may coexist, we measure the overall edge strength at a particular voxel p as the sum of individual edge strengths corresponding to fibers F_p^m and their respective local orientation $\mathbf{o}_p^m$. For each possible local orientation $\mathbf{o}_{p_{i,j}}^m$ at voxel $p_{i,j}$ we first construct a local linear fit of feature vectors associated with matching fiber orientations in its neighborhood. Practically, in each surrounding voxel, the fiber with closest orientation to $\mathbf{o}_{p_{i,j}}$ is included in the least squares fit.

As illustrated in Fig. 1, fiber $F_{p_{i-1,j}}^0$, $F_{p_{i+1,j}}^0$, $F_{p_{i,j-1}}^1$, and $F_{p_{i,j+1}}^0$ are selected by fiber $F_{p_{i,j}}^0$ while fiber $F_{p_{i-1,j}}^1$, $F_{p_{i+1,j}}^0$, $F_{p_{i,j-1}}^0$, and $F_{p_{i,j+1}}^1$ are selected by fiber $F_{p_{i,j}}^1$. Note that fiber $F_{p_{i-1,j}}^0$ is selected by both $F_{p_{i,j}}^0$ and $F_{p_{i,j}}^1$, but fiber $F_{p_{i,j+1}}^2$ is never

selected. Applying the above procedure, different fiber functions $\mathbf{f}_p^m$ are estimated from different local fields of space filling curves, then the general derivatives are computed as the mean of derivatives obtained from each $\mathbf{f}_p^m$. As in the DTI case, the structure tensor is employed for a robust estimate of the general derivatives.

$$\mathbf{G} = \sum_{m=0}^{M-1} \begin{pmatrix} \overline{\mathbf{f}_x^{m\top} \mathbf{f}_x^m} & \overline{\mathbf{f}_x^{m\top} \mathbf{f}_y^m} & \overline{\mathbf{f}_x^{m\top} \mathbf{f}_z^m} \\ \overline{\mathbf{f}_y^{m\top} \mathbf{f}_x^m} & \overline{\mathbf{f}_y^{m\top} \mathbf{f}_y^m} & \overline{\mathbf{f}_y^{m\top} \mathbf{f}_z^m} \\ \overline{\mathbf{f}_z^{m\top} \mathbf{f}_x^m} & \overline{\mathbf{f}_z^{m\top} \mathbf{f}_y^m} & \overline{\mathbf{f}_z^{m\top} \mathbf{f}_z^m} \end{pmatrix} \tag{7}$$

Similar to edge detection in DT-MRI, the edge strength is measured by the major eigenvalue of the structure tensor $\mathbf{G}$ in Eq. (4).

4 Visualization

The per voxel edge strength measured by our approach yields a scalar field by applying the measurement to the entire domain of a dataset. Classical scalar field visualization methods, such as isosurfaces and volume rendering are applicable on the edge strength field. However the visualization technique introduced by Hlawitschka et al. [17] which augments the edge strength by overlaying an anisotropy-scaled RGB color map provides the information to identify typical anatomical structures in DWI datasets, the high density of edges detected by our approach could potentially cause visual clutter, especially in the brain dataset. Therefore, in this paper, we split the result visualization into two steps:

1. Characterization of edges' geometry by performing ridge extraction on a user defined sweeping plane or region of interest. The extracted ridge lines/ridge surfaces can further be filtered by the edge strength and ridge strength.
2. Visualizing the ridge lines/ridge surfaces by superimposing an anisotropy-scaled RGB color map which shows the local orientation and fractional anisotropy (FA) to provide the context information of the sweeping plane or the region of interest.

Furthermore, we propose to enhance the visualization result by rendering fiber trajectories with ridge lines/ridge surfaces characterized from our edge strength result (cf. Fig. 8). Hence, the ridge lines/ridge surfaces emphasize the boundary of different anatomical structures while fiber trajectories convey the shape and connectivity of neural tracts.

5 Results and Discussions

Both synthetic and real data were considered to test the presented edge detection approach. First, we focused on a publicly available phantom data to validate our method against a known ground truth and investigate its robustness to various noise levels (Sect. 5.1). Then we studied two human brain datasets (Sects. 5.2 and 5.3).

5.1 Phantom Data

We tested our approach on the phantom data used in the HARDI reconstruction challenge 2013 [9]. The dataset with a $b-value$ of 1200 s/mm^2 and 32 directions was used to test our method on DTI, and the dataset with a $b-value$ of 3000 s/mm^2 and 64 directions was studied to test our method on HARDI. For both DTI and HARDI, two different signal-to-noise-ratio (SNR), 10 and 30, were tested to validate our approach against noise. The original spatial resolution of the phantom is $50 \times 50 \times 50$ with isotropic voxel size as 1.0 mm and we measured the edge strength on a discrete domain with spatial resolution $400 \times 400 \times 400$ in both studies on DTI and HARDI.

Figure 2 shows the edge strength computed from our method both on DTI and HARDI as well as the ground truth. In single orientation regions, both DTI and HARDI can reconstruct the correct orientation, and our approach can successfully detect edges in those regions. In regions which contain crossing fiber tracts from different fiber bundles, DTI leads to incorrect fiber traces. In contrast, the tensor

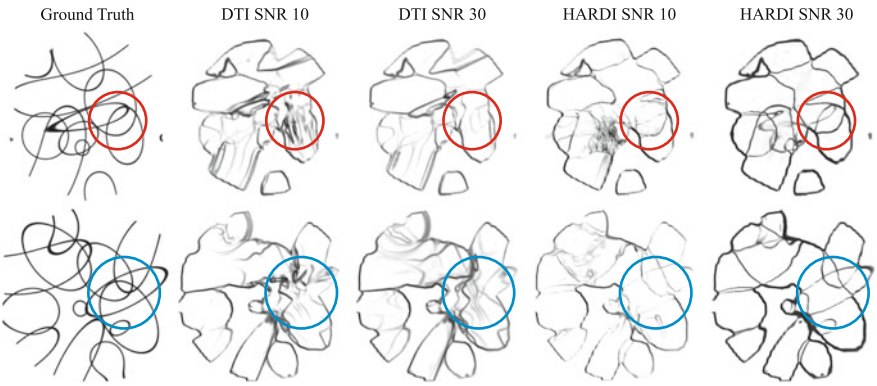


Fig. 2 A comparison of the ground truth and the results obtained by applying our edge detection approach on the phantom dataset with different SNRs. Both *red and blue cycles* highlight the regions where different fiber bundles intersect with each other. It is clear that the edge detection results using DTI fail in those regions. On the other hand, the edge detection results using HARDI delineate meaningful boundaries of fiber bundles similar to the ground truth

decomposition method used in HARDI properly reconstructs the multi-orientation in those regions. Our results in crossing fiber regions shown in Fig. 2 confirm that the DTI results are not close to the ground truth while the HARDI results are comparable to it.

A comparison of our method to the end-position tractography method [15] on DTI is proposed in Fig. 3. Although both methods were implemented with the MLS-based fiber tracking technique [52], the inconsistency of local orientations introduced by DTI model in crossing regions causes incorrect end positions of individual fiber traces. Therefore, using these end positions results a discontinuous

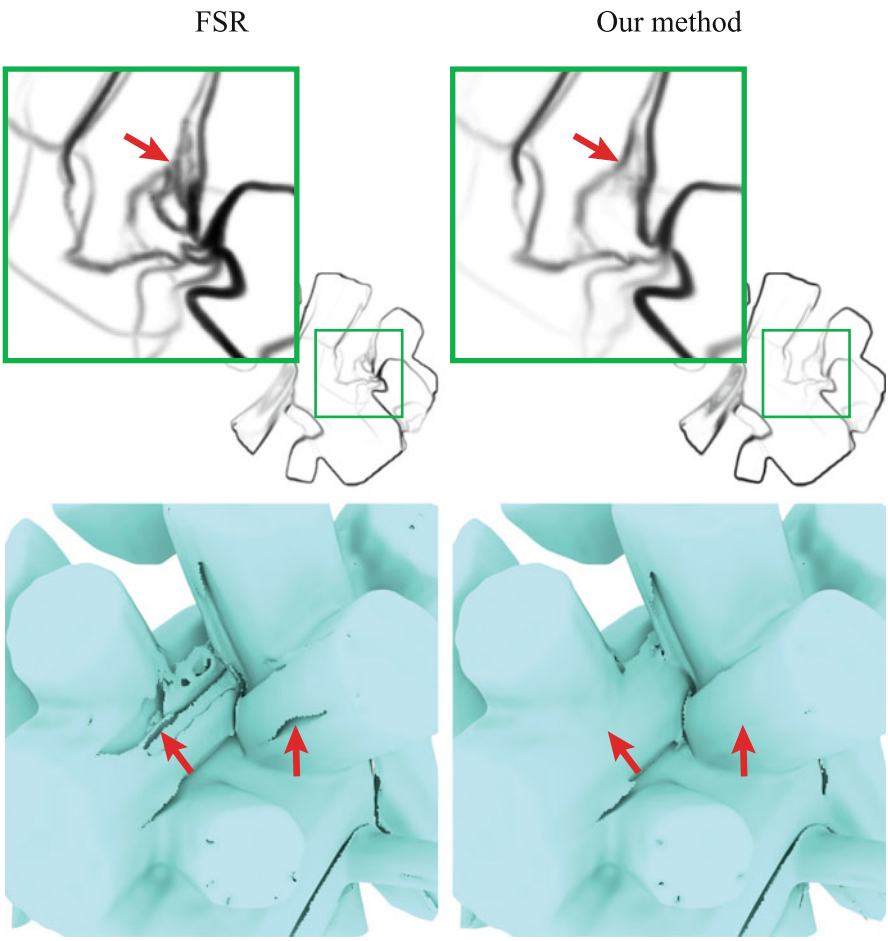


Fig. 3 A detailed comparison of the end-position method and the presented method on phantom dataset. First row visualizes the side-by-side comparison of the edge strength computed from these two methods. Second row shows the ridge surfaces characterized from the edge strength fields. Differences between these two methods are highlighted by *red arrows*

edge strength measurement in FSR. On the other hand, the fiber function estimated by entire fiber traces generates a smooth and consistent result in our approach. The first column shows the edge strength measured by end-position method and our approach on one slice of the YZ plane in the phantom dataset. In single orientation regions, the edge strength measured by our approach is smoother and less influenced by the inconsistency of local orientations caused by noise than the end-position method. The second column shows the characterized ridge surfaces using the same ridge extraction method [13] on edge strength measurement fields generated by end-position method and our approach. Comparing these two ridge surfaces, a significant number of disconnected components and cracks exist in the result of the end-position method.

Finally, the renderings in Fig. 4 show the extracted ridge surfaces from the edge strength on HARDI with SNR = 30 and the boundaries from the ground truth. A unique color was assigned to each fiber bundle, therefore, the correspondence between the characterized ridge surfaces from our edge strength measurement and the actual boundaries of different fiber bundles is clearly shown in these renderings.

5.2 IIT2 Human Brain DTI Template

The first in vivo human brain used to test our approach is the public IIT2 human brain DTI template [50]. The original spatial resolution is $181 \times 217 \times 181$ with isotropic voxel size as 1 mm and the edge strength was measured on a discrete domain with spatial resolution $724 \times 868 \times 724$ which is 4 times larger in each dimension. The relevant fiber tracking parameters are: step-size = 0.1 mm and maximum fiber length = 30.0 mm. The stopping criteria in fiber tracking are set as the maximum angle between steps = 45° and the FA value threshold for white matter = 0.15.

Ridge surfaces rendered as Fig. 5a and b are extracted from the edge strength measured by our edge detection approach in the brainstem (the posterior part of the brain). Complex fiber traces with distinct directions pass through this region. Similar to (f) which is taken from Fig. 5 in Kindlmann et al. [25], boundaries of different fiber traces including the *middle cerebellar peduncle* (mcp), *corticospinal tract*(cst), *transverse pontine fibers* (tpf), *medial lemniscus* (ml), *superior cerebellar peduncle* (scp), and *inferior cerebellar peduncle* (icp) are identified and visualized with the difference that our results exhibit more comprehensive and clear boundaries. In addition, two images (d) and (e) visualize the characterized ridge lines corresponding to the cutting plane *i* and *ii* in (c).

The yellow cycle in Fig. 6a indicates another interesting region where the white matter lateral to the posterior horn of the lateral ventricle consists of three layers of tracts: the most lateral layer is the *superior longitudinal fasciulus* (slf) with a superior-inferior orientation; the most medial layer is the callosal projection to the *temporal lobe* (tapetum); and the *posterior region of the corona radiata* (pcr) can be found between them with an anterior-posterior orientation [46]. Boundaries between

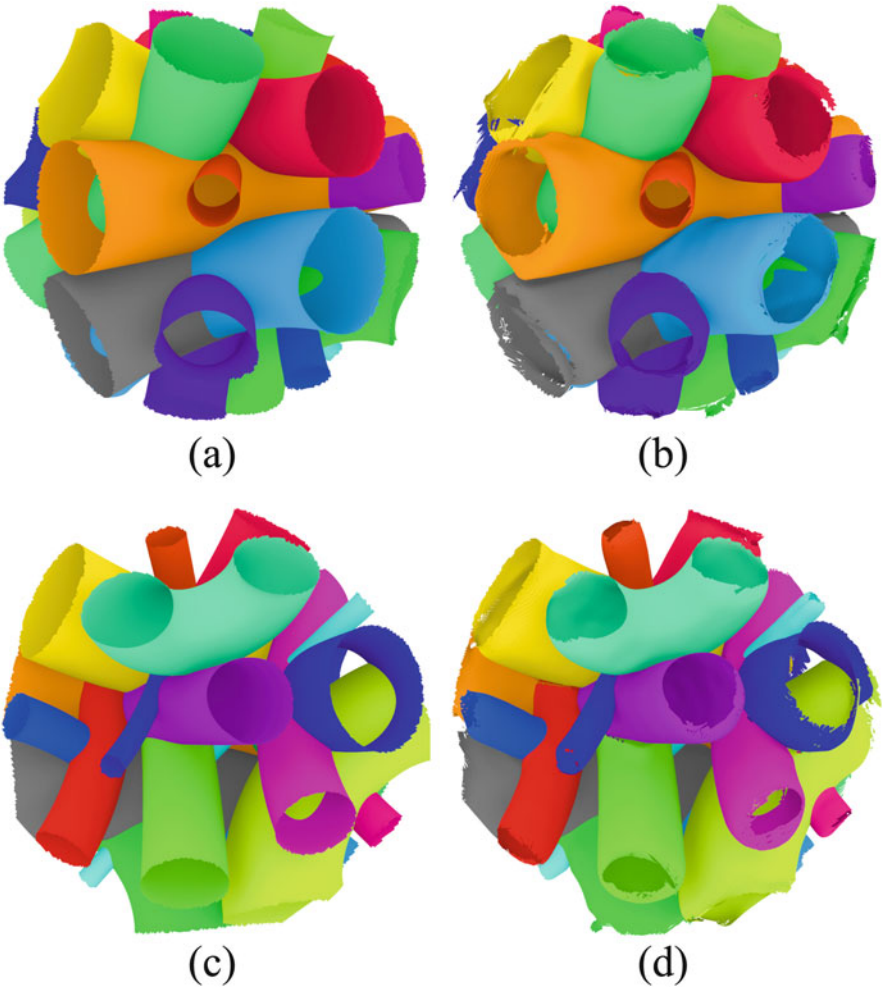


Fig. 4 Ridge surfaces characterized from the edge strength measured on HARDI with SNR=30. (**a**) and (**c**) boundaries of different fiber bundles from the ground truth. (**b**) and (**d**) extracted ridge surfaces by our approach

these three layers are successfully detected by our approach as shown in Fig. 6b. Figure 6c shows the ridge surfaces extracted from our edge strength measurement. A visualization which overlays the anisotropy-based RGB color map with ridge surfaces is shown in Fig. 6d. This result confirms that the detected edges do represent the actual boundaries between different white matter layers.

Figure 7a highlightes 9 anatomical structures on a anisotropy-based RGB color map of a coronal plane. Figure 7b and c visualize edge strength measured with 2 times and 4 times as large as the original spatial resolution in each dimension respectively. It takes about 5 s to measure the edge strength of Fig. 7b and 13 s of

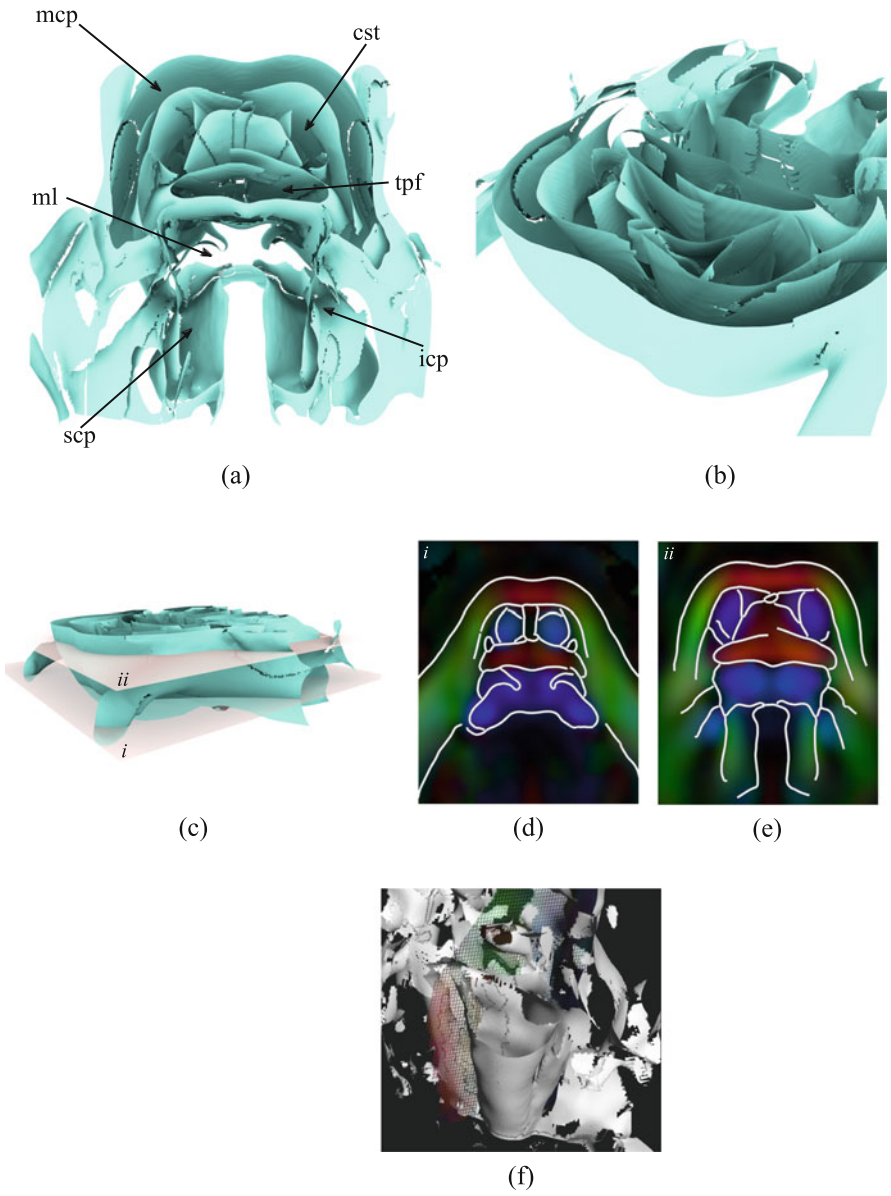


Fig. 5 Edge detection result in the brainstem by our approach. mcp: middle cerebellar peduncle; cst: corticospinal tract; tpf: transverse pontine fibers; ml: medial lemniscus; scp: superior cerebellar peduncle; icp: inferior cerebellar peduncle.

Fig. 7c on a machine with an Intel i7 quad-core CPU and a Nvidia Quadro M3000M graphics card. In general, the edge strength measured with a higher resolution

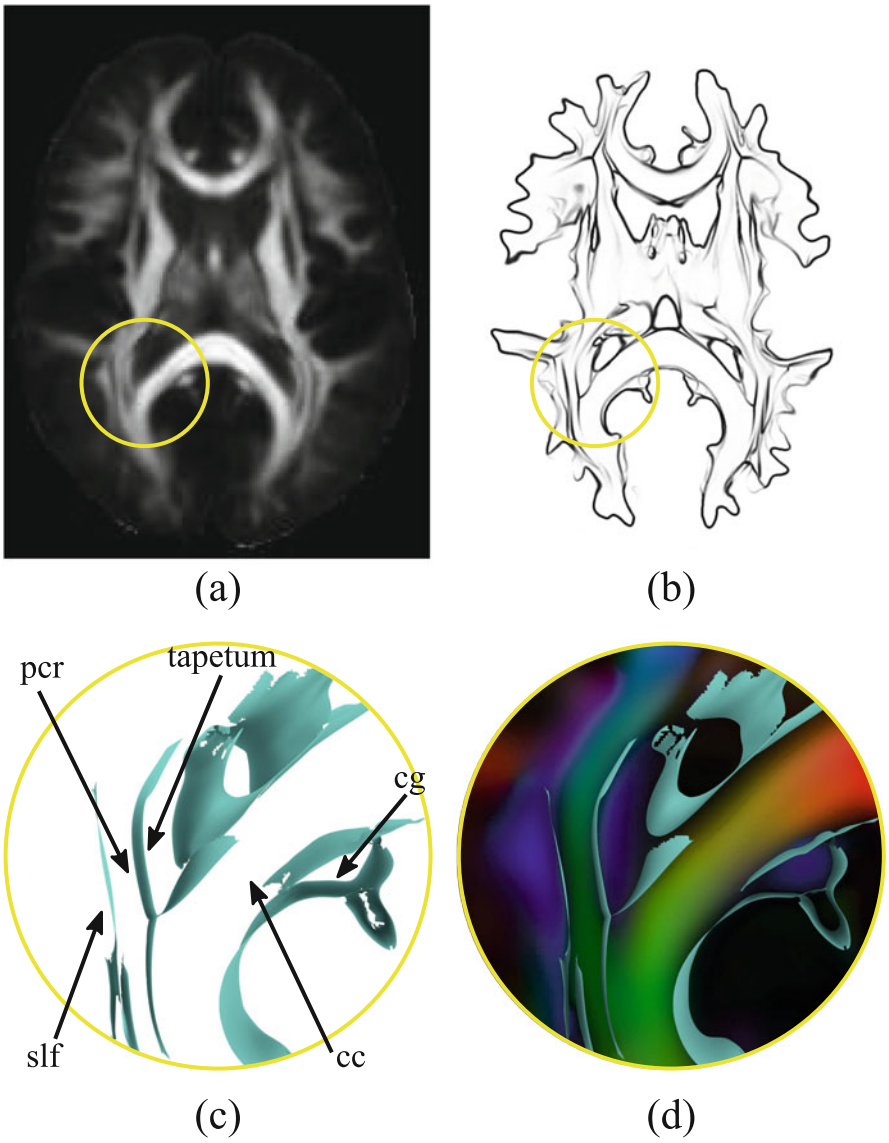
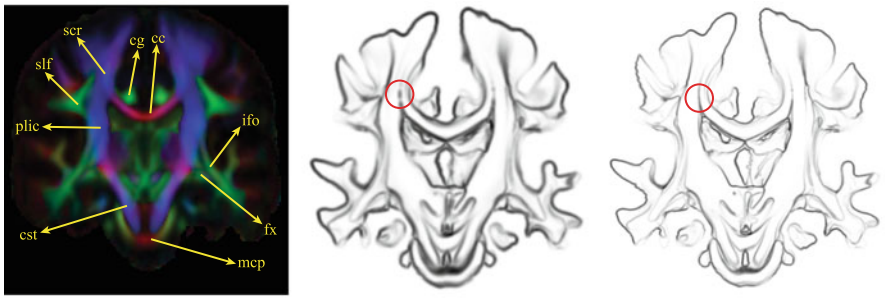


Fig. 6 Edge detection results in an interesting region where the white matter lateral to the posterior horn of the lateral ventricle consists of three layers of tracts. cc: corpus callosum; cg: cingulum; pcr: posterior region of corona radiata; slf: superior longitudinal fasciculus. (**a**) Fractional anisotropy (FA) of the brain white matter visualized on a transverse section. The *circle* indicates the considered anatomical region. (**b**) Edge strength measured by our method on the same section. (**c**) Ridge surfaces of the edge strength characterize the complex geometry of crossing fibers in the area of interest. (**d**) Same as (**c**) with FA overlaid and color-coded by the orientation of the main diffusion direction



(a) (b) (c)

Fig. 7 (**a**) Coronal anisotropy-based RGB color map; (**b**) edge strength measured with 2 times as large as the original spatial resolution in each dimension; (**c**) edge strength measured with 4 times as large as the original spatial resolution in each dimension. cc: *corpus callosum*; cg: *cingulum*; st: *corticospinal tract*; mcp: *middle cerebellar peduncle*; fx: *fornix*; ifo: *inferior fronto-occipital fasciculus*; plic: *posterior limb of internal capsule*; scr: *superior region of internal capsule*; slf: *superior longitudinal fasciculus*

exhibits sharper edges and contains less discontinuity than the one measured with a lower resolution. However, both results are able to capture the boundaries of anatomical structures highlighted in Fig. 7a.

5.3 Human Brain with DTI and HARDI

The second in vivo human brain dataset used to test our approach consists of 270 diffusion weighted images with three different $b-values$, 1000 s/mm^2, 2000 s/mm^2, and 3000 s/mm^2, as well as 18 baseline scans with $b-value = 0$. Multiple $b-value$ allows us to test our method on DTI and HARDI separately and compare the results. We use the diffusion weighted images with $b - value = 1000$ s/mm^2 to estimate a DTI dataset, and use the ones with $b - value = 3000$ s/mm^2 to estimate a HARDI dataset.

In this experiment, a small region of interest was selected where the lateral *transcallosal fibers* (tf) runs through the *corpus callosum* (cc) and intersects with the *internal capsule* (ic). The DTI model yields invalid orientation information in the fiber crossing region while HARDI successfully reconstructs the *transcallosal fibers*. Our edge detection approach on HARDI could extract edges that correctly represent the anatomical structures in this region. A side-by-side comparison of our edge detection results on DTI and HARDI is shown in Fig. 8. Figure 8 also highlights the anatomical structures that could be found by our approach both on DTI and HARDI with corresponding fiber traces. As can be seen, both DTI and HARDI allowed our method capture boundaries of *corpus callosum* (red) and *cingulum* (green), while it was only in the HARDI case that our method succeeded in reconstructing the boundaries of *transcallosal fibers* (purple). Similarly, the crossing

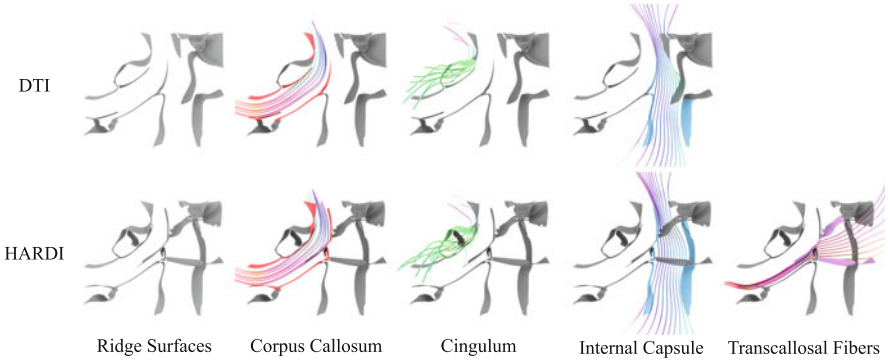


Fig. 8 A comparison of the ridge surfaces extracted from the edge detection results by our approach on DTI and HARDI

of *internal capsule* and *superior longitudinal fasciculus* (blue) was not properly characterized in DTI data, which led to the partial extraction of the boundaries of the *internal capsule* (blue) in this case. In contrast, edge extraction in the HARDI case produces correct boundaries between *internal capsule* and *superior longitudinal fasciculus*.

6 Conclusion and Future Work

In many applications of diffusion weighted imaging (DWI) analysis, extracting the boundaries of anatomical structures from the scanned DWI signals is a crucial step. In this paper, we have presented an tractography-based edge detection technique for DWI that takes the entire geometry of fiber traces into consideration to identify the contour of distinct fiber populations. An evaluation against ground truth in a phantom dataset has proved that the edges characterized by our method coincide with the boundaries of individual fiber bundles and thereby reveal major anatomical structures. In addition, experiments performed on real data have shown that subtle anatomical structures, in particular those associated with fiber crossings, can be identified by our method in noisy datasets.

Limitations of our approach and open questions remain as avenues for future work. First, the super-sampling of the original dataset combined with an on-the-fly regularization procedure at each step of fiber integration make our technique computationally expensive. Although a GPU implementation was used to accelerate the necessary computations, a strategy consisting in reusing fiber traces among neighboring voxels could dramatically reduce the computational time by exploiting redundancy. Second, standard tensor invariant information like fractional anisotropy (FA) aggregated along individual fiber traces could also be used to measure edge strength in the context of Diffusion Tensor Imaging. Finally, the basic approach

presented in this paper is not limited to the visualization of DWI data, and we would like to apply it to tensor field visualization problems in other application domains.

Acknowledgements This work was made possible in part by a NSF CAREER Program Award No. 1150000: Efficient Structural Analysis of Multivariate Fields for Scalable Visualizations. This support is here gratefully acknowledged.

References

1. Alexander, D.C., Barker, G.J., Arridge, S.R.: Detection and modeling of non-Gaussian apparent diffusion coefficient profiles in human brain data. Magn. Reson. Med. **48**(2), 331–340 (2002)
2. Basser, P.J., Pierpaoli, C.: Microstructural and physiological features of tissues elucidated by quantitative-diffusion-tensor MRI. J. Mag. Reson. Ser. B **111**(3), 209–219 (1996)
3. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophys. J. **66**(1), 259–267 (1994)
4. Basser, P.J., Pajevic, S., Pierpaoli, C., Duda, J., Aldroubi, A.: In vivo fiber tractography using DT-MRI data. Magn. Reson. Med. **44**(4), 625–32 (2000)
5. Brun, A., Park, H.-J., Knutsson, H., Westin, C.-F.: Coloring of DT-MRI fiber traces using laplacian eigenmaps. In: Diaz, R.M., Arencibia, A.Q. (eds.) Computer Aided Systems Theory (EUROCAST'03). Lecture Notes in Computer Science 2809, pp. 564–572, Las Palmas de Gran Canaria, Spain, 24–28 Feb 2003. Springer, Berlin (2003)
6. Brun, A., Knutsson, H., Park, H.J., Shenton, M.E., Westin, C.-F.: Clustering fiber tracts using normalized cuts. In: *Seventh International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'04)*. Lecture Notes in Computer Science, pp. 368–375, Rennes – Saint Malo, France, Sept 2004
7. Calamante, F., Tournier, J.-D., Smith, R.E., Connelly, A.: A generalised framework for super-resolution track-weighted imaging. NeuroImage **59**(3), 2494–2503 (2012)
8. Conturo, T.E., Lori, N.F., Cull, T.S., Akbudak, E., Snyder, A.Z., Shimony, J.S., McKinstry, R.C., Burton, H., Raichle, M.E.: Tracking neuronal fiber pathways in the living human brain. Proc. Natl. Acad. Sci. **96**(18), 10422–10427 (1999)
9. Daducci, A., Caruyer, E., Descoteaux, M., Houde, J.-C., Thiran, J.-P.: Hardi reconstruction challenge. In: IEEE ISBI (2013)
10. Delmarcelle, T., Hesselink, L.: The topology of symmetric, second-order tensor fields. In: Proceedings of IEEE Visualization 1994, pp. 140–147. IEEE Computer Society Press, Los Alamitos, CA (1994)
11. Eberly, D.: Ridges in Image and Data Analysis. Kluwer Academic Publishers, Boston (1996)
12. Frank, L.R.: Characterization of anisotropy in high angular resolution diffusion-weighted MRI. Magn. Reson. Med **47**, 1083–1099 (2002)
13. Furst, J.D., Pizer, S.M.: Marching ridges. In: Signal and Image Processing, pp. 22–26 (2001)
14. Hess, C.P., Mukherjee, P., Han, E.T., Xu, D., Vigneron, D.B.: Q-ball reconstruction of multimodal fiber orientations using the spherical harmonic basis. Magn. Reson. Med. **56**(1), 104–117 (2006)
15. Hlawatsch, M., Vollrath, J.E., Sadlo, F., Weiskopf, D.: Coherent structures of characteristic curves in symmetric second order tensor fields. IEEE Trans. Vis. Comput. Graph. **17**(6), 781–794 (2011)
16. Hlawitschka, M., Scheuermann, G.: Hot-lines: tracking lines in higher order tensor fields. In: IEEE Visualization, 2005. VIS 05, pp. 27–34, Oct. 2005
17. Hlawitschka, M., Garth, C., Tricoche, X., Kindlmann, G., Scheuermann, G., Joy, K.I., Hamann, B.: Direct visualization of fiber information by coherence. Int. J. Comput. Assist. Radiol. Surg. **5**(2), 125–131 (2010)

18. Holmes, A.A., Scollan, D.F., Winslow, R.L.: Direct histological validation of diffusion tensor MRI in formaldehyde-fixed myocardium. Magn. Reson. Med. **44**(1), 157–61 (2000)
19. Hsu, E.W., Muzikant, A.L., Matulevicius, S.A., Penland, R.C., Henriquez, C.S.: Magnetic resonance myocardial fiber-orientation mapping with direct histological correlation. Am. J. Physiol. **274**(5 Pt 2), H1627–1634 (1998). 0002-9513 (Print) Journal Article Research Support, U.S. Gov't, Non-P.H.S. Research Support, U.S. Gov't, P.H.S.
20. Jiao, F., Gur, Y., Johnson, C.R., Joshi, S.: Detection of crossing white matter fibers with high-order tensors and rank-k decompositions. In: Information Processing in Medical Imaging, pp. 538–549. Springer, London (2011)
21. Jones, D.K.: Studying connections in the living human brain with diffusion MRI. Cortex **44**(8), 936–952 (2008)
22. Jones, D.K., Simmons, A., Williams, S.C.R., Horsfield, M.A.: Non-invasive assessment of axonal fiber connectivity in the human brain via diffusion tensor MRI. Magn. Reson. Med. **42**(1), 37–41 (1999)
23. Kindlmann, G., Tricoche, X., Westin, C.-F.: Anisotropy creases delineate white matter structure in diffusion tensor MRI. In: Proceedings of Medical Imaging Computing and Computer-Assisted Intervention, MICCAI '06 (2006)
24. Kindlmann, G., Ennis, D.B., Whitaker, R.T., Westin, C.-F.: Diffusion tensor analysis with invariant gradients and rotation tangents. IEEE Trans. Med. Imaging **26**(11), 1483–1499 (2007)
25. Kindlmann, G., Tricoche, X., Westin, C.-F.: Delineating white matter structure in diffusion tensor MRI with anisotropy creases. Med. Image Anal. **11**(5), 492–502 (2007)
26. Lazar, M.: Mapping brain anatomical connectivity using white matter tractography. NMR in Biomed. **23**(7), 821–835 (2010)
27. Lazar, M., Weinstein, D.M., Tsuruda, J.S., Hasan, K.M., Arfanakis, K., Meyerand, M.E., Badie, B., Rowley, H.A., Haughton, V., Field, A., et al.: White matter tractography using diffusion tensor deflection. Hum. Brain Mapp. **18**(4), 306–321 (2003)
28. Liang, X., Wang, J., Lin, Z., Zhang, C.: White matter fiber tract segmentation using nonnegative matrix factorization. In: 3rd International Conference on Bioinformatics and Biomedical Engineering, 2009. ICBBE 2009, pp. 1–4 (2009)
29. Lindeberg, T.: Edge detection and ridge detection with automatic scale selection. Int. J. Comput. Vis. **30**(2), 77–116 (1998)
30. Mai, S.T., Goebl, S., Plant, C.: A similarity model and segmentation algorithm for white matter fiber tracts. In: 2012 IEEE 12th International Conference on Data Mining (ICDM), pp. 1014–1019 (2012)
31. Mori, S., Crain, B.J., Chacko, V.P., Van Zijl, P.: Three-dimensional tracking of axonal projections in the brain by magnetic resonance imaging. Ann. Neurol. **45**(2), 265–269 (1999)
32. O'Donnell, L., Kubicki, M., Shenton, M.E., Dreusicke, M., Grimson, W.E.L., Westin, C.-F.: A method for clustering white matter fiber tracts. AJNR Am. J. Neuroradiol. **27**(5), 1032–1036 (2006)
33. Schultz, T., Seidel, H.-P.: Estimating crossing fibers: a tensor decomposition approach. IEEE Trans. Vis. Comput. Graph. (Proc. IEEE Vis.) **14**(6), 1635–1642 (2008)
34. Schultz, T., Seidel, H.-P.: Using eigenvalue derivatives for edge detection in DT-MRI data. In: Rigoll, G. (ed.) Pattern Recognition. Lecture Notes in Computer Science, vol. 5096, pp. 193–202. Springer, New York (2008)
35. Schultz, T., Theisel, H., Seidel, H.-P.: Topological visualization of brain diffusion MRI data. IEEE Trans. Vis. Comput. Graph. (Proc. IEEE Vis.) **13**(6), 1496–1503 (2007)
36. Scollan, D.F., Holmes, A., Winslow, R., Forder, J.: Histological validation of myocardial microstructure obtained from diffusion tensor magnetic resonance imaging. Am. J. Physiol. **275**, 2308–2318 (1998)
37. Tournier, J.-D., Calamante, F., Gadian, D.G., Connelly, A.: Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. NeuroImage **23**, 1176–1185 (2004)
38. Tournier, J.-D., Calamante, F., Connelly, A.: Robust determination of the fibre orientation distribution in diffusion MRI: non-negativity constrained super-resolved spherical deconvolution. NeuroImage **35**, 1459–1472 (2007)

39. Travers, B.G., Adluru, N., Ennis, C., Tromp, D.P.M., Destiche, D., Doran, S., Bigler, E.D., Lange, N., Lainhart, J.E., Alexander, A.L.: Diffusion tensor imaging in autism spectrum disorder: a review. Autism Res. **5**(5), 289–313 (2012)
40. Tricoche, X., Kindlmann, G., Westin, C.-F.: Invariant crease lines for topological and structural analysis of tensor fields. IEEE Trans. Vis. Comput. Graph. **14**(6), 1627–1634 (2008)
41. Tseng, W.-Y.I., Wedeen, V.J., Reese, T.G., Smith, R.N., Halpern, E.F.: Diffusion tensor MRI of myocardial fibers and sheets: correspondence with visible cut-face texture. J. Mag. Reson. Imaging **17**(1), 31–42 (2003)
42. Tuch, D.S.: Diffusion MRI of complex tissue structure. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, Jan 2002
43. Tuch, D.S., Weisskoff, R.M., Belliveau, J.W., Wedeen, V.J.: Clustering fiber tracts using normalized cuts. In: Proceeding of the 7th Anmual Meeting of ISMRM, p. 321, Philadephia, PA, 1999
44. Tuch, D.S., Reese, T.G., Wiegell, M.R., Wedeen, V.J.: Diffusion MRI of complex neural architecture. Neuron **40**, 885–895 (2003)
45. van de Weijer, J., Gevers, Th.: Tensor based feature detection for color images. In: Color and Imaging Conference, vol. 2004, no. 1, pp. 100–105. Society for Imaging Science and Technology (2004)
46. Wakana, S., Jiang, H., Nagae-Poetscher, L.M., Van Zijl, P.C.M., Mori, S.: Fiber tract–based atlas of human white matter anatomy 1. Radiology **230**(1), 77–87 (2004)
47. Weinstein, D., Kindlmann, G., Lundberg, E.: Tensorlines: advection-diffusion based propagation through diffusion tensor fields. In: Proceedings of the Conference on Visualization'99: Celebrating Ten Years, pp. 249–253. IEEE Computer Society Press, Los Alamitos, CA (1999)
48. Zhang, S., Demiralp, C., Laidlaw, D.H.: Visualizing diffusion tensor MR images using streamtubes and streamsurfaces. IEEE Trans. Vis. Comput. Graph. **9**(4), 454–462 (2003)
49. Zheng, X., Parlett, B.N., Pang, A.: Topological lines in 3D tensor fields and discriminant hessian factorization. IEEE Trans. Vis. Comput. Graph. **11**(4), 395–407 (2005)
50. Zhang, S., Peng, H., Dawe, R.J., Arfanakis, K.: Enhanced icbm diffusion tensor template of the human brain. Neuroimage **54**(2), 974–984 (2011)
51. Zhukov, L., Barr, A.: Oriented tensor reconstruction: tracing neural pathways from diffusion tensor MRI. In: Proceedings of IEEE Visualization 2002, pp. 387–394 (2002)
52. Zhukov, L., Barr, A.H.: Oriented tensor reconstruction: tracing neural pathways from diffusion tensor MRI. In: IEEE Visualization, 2002. VIS 2002, pp. 387–394. IEEE, Piscataway, NJ (2002)
53. Zhukov, L., Barr, A.: Heart -muscle fiber reconstruction from diffusion tensor MRI. In: Proceedings of IEEE Visualization 2003, pp. 597–602 (2003)

Repeated Tractography of a Single Subject: How High Is the Variance?

Xuan Gu, Anders Eklund, and Hans Knutsson

Abstract We have investigated the test-retest reliability of diffusion tractography, using 32 diffusion datasets from a single healthy subject. Preprocessing was carried out using functions in FSL (FMRIB Software Library), and tractography was carried out using FSL and Dipy. The tractography was performed in diffusion space, using two seed masks (corticospinal and cingulum gyrus tracts) created from the JHU White-Matter Tractography atlas. The tractography results were then warped into MNI standard space by a linear transformation. The reproducibility of tract metrics was examined using the standard deviation, the coefficient of variation (CV) and the Dice similarity coefficient (DSC), which all indicated a high reproducibility. Our results show that the multi-fiber model in FSL is able to reveal more connections between brain areas, compared to the single fiber model, and that distortion correction increases the reproducibility.

1 Introduction

In the past few years, a number of algorithms for reconstruction of fiber tracts from diffusion-weighted images have been proposed, collectively known as tractography [10]. Tractography is an important neuroimaging tool which can be used for studying brain connectivity and aiding brain surgery [8, 11, 22, 27]. The fundamental

X. Gu (✉) • H. Knutsson
Division of Medical Informatics, Department of Biomedical Engineering, Linköping University, Linköping, Sweden

Center for Medical Image Science and Visualization, Linköping University, Linköping, Sweden
e-mail: xuan.gu@liu.se; hans.knutsson@liu.se

A. Eklund
Division of Medical Informatics, Department of Biomedical Engineering, Linköping University, Linköping, Sweden

Center for Medical Image Science and Visualization, Linköping University, Linköping, Sweden

Division of Statistics and Machine Learning, Department of Computer and Information Science, Linköping University, Linköping, Sweden
e-mail: anders.eklund@liu.se

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_14

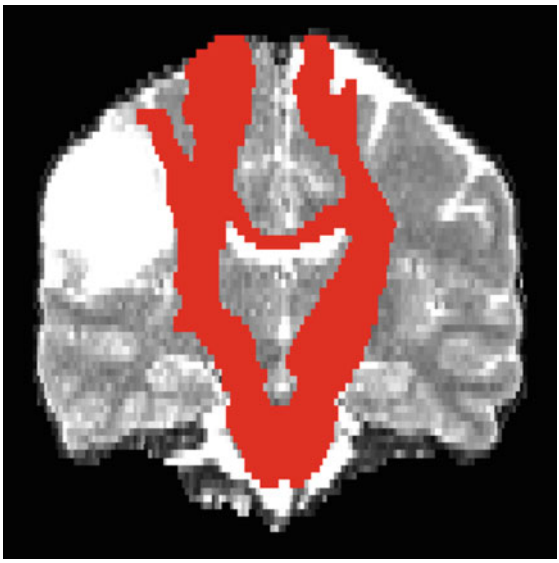


Fig. 1 A mid-axial slice of the corticospinal tract, for a subject with an Astrocytoma tumor in the right primary somatosensory area. A threshold of 0.2% was used to remove less likely tracts. While tractography can aid planning of tumor surgery, it is important to know the uncertainty of the nice images

goal of brain tumor surgery is to resect the maximum amount of tumoral tissue, while removing as little healthy tissue as possible. Figure 1 shows an example where the estimated corticospinal tracts for a patient were very close to an Astrocytoma tumor in the right primary somatosensory area [31]. The corticospinal tract is one of the major nerve fiber tracts, and should be preserved as much as possible during brain surgery. Therefore, it is important to fully evaluate the reproducibility of diffusion tractography results, so that the surgical approaches can be designed to avoid damaging important tracts. Although tractography provides very nice images [4], there has been some concerns regarding the reproducibility of the results, and these concerns are especially important for clinical applications. Reconstruction of fiber tracts can, for example, differ depending on the software package being used, the specific algorithm, and different parameter settings.

However, only a few attempts have been made so far to quantitatively investigate the reproducibility of different methods and softwares for tractography. Heiervang et al. [19] conducted studies to characterize the reproducibility and variability of diffusion tractography using datasets from eight subjects scanned three times. The mean tract fractional anisotropy (FA) and the mean diffusivity (MD) along the tracts showed a very low coefficient of variation, both below 2% for inter-session and 3–5% for inter-subject. They also found that the number of diffusion directions (60 and 12) has a limited effect on the inter-session coefficient of variation (CV). Datasets with more diffusion directions, however, produced greater tract volume. Tensaouti et al. [35] reported a maximum value of 56% of tract volume agreement for diffusion data collected in 32 directions. They drew a similar conclusion as [19] that more directions allows to detect more tracts until a certain level between 15–32 directions. Vollmar et al. [39] investigated the intra-site and inter-site reproducibility and

reported a very low (1.6%) CV for the mean tract FA within the tracts, and 6.2–8.4% CV of the tract volume. It is also claimed that nonlinear registration between scans can be used to eliminate different distortions and improve reproducibility. Vaessen et al. [38] evaluated the reproducibility of brain network connectivity by diffusion tractography, and reported low values (3.8%) for the CV of the network measures. Danielian et al. [14] assessed intra-human-rater and inter-human-rater reproducibility by measurements of tract FA, MD, axial diffusivity, transverse diffusivity and tract volume using intraclass correlation coefficient (ICC), CV and kappa (κ) statistic. They reported an ICC greater than 0.77 and a κ greater than 0.76 for inter-human-rater and an ICC greater than 0.92 and a κ greater than 0.9 for intra-human-rater. Tensaouti et al. [36] evaluated the reproducibility of tractography in terms of data acquisitions and tractography algorithms. Similarly to [19, 35], Tensaouti et al. reported an increase in reproducibility in tractography according to the number of directions used during the scans.

For reproducibility of tractography in brachial plexus and kidney, evaluations were carried out by Tagliafico et al. [34], Cutajar et al. [13]. Apart from studies that assess the longitudinal reproducibility of fiber tractography, results regarding the accuracy of fiber tractography can be found in [12, 16, 18, 29, 33].

The aim of this study was therefore to analyze the reproducibility of tractography for diffusion data repeatedly collected from a single healthy subject. This makes it possible to investigate how the whole workflow (data collection, preprocessing and tractography) affects the final results, instead of only focusing on the tractography itself. We here present results for the popular diffusion tractography softwares FSL [24] and Dipy [17] which use a probabilistic and a deterministic tractography algorithm, respectively. For the quantitative analysis of the results, we focused on the CV and the Dice similarity coefficient (DSC) which indicate how high the reproducibility is. Testing other software packages is planned for future work.

The remainder of the paper is organized as follows. In Sect. 3, the evaluated tractography software packages and image process workflow is presented in detail. The results of the evaluations are shown in Sect. 4, and the results are discussed in Sect. 5.

2 Data

Diffusion datasets were acquired from the MyConnectome [26] study (myconnectome.org), where MR imaging was performed on a fixed schedule during an entire year on a single healthy individual. In this study, we chose the 16 scan sessions (out of the total 106) containing diffusion data, obtained using a multiband EPI sequence on a Siemens Skyra 3T scanner. Each session consists of two scans, giving a total of 32 diffusion datasets, with L→R phase encoding and the other with R→L phase encoding (these two scans can be combined to correct for distortions). The following scanning parameters were used: b = 1000/2000 s · mm^{-2} (30 gradient directions per b-factor and 4 volumes without diffusion weighting), 1.74 × 1.74 × 1.7 mm voxels, 72 slices, 128 × 128 matrix, TR = 5000 ms, TE = 108 ms, multiband factor 3.

3 Methods

3.1 FSL

Preprocessing and tractography was carried out using tools in FSL [24]. Susceptibility distortions were corrected for using the function *topup* [3], while the function *eddy* [2] was used to correct for head motion and eddy-current induced distortions.

The ball and stick model [6] is in the function *bedpostx* used to model the diffusion in each voxel, with one isotropic part (ball) and N non-isotropic parts (sticks), according to

$$\frac{S_i}{S_0} = (1 - \sum_{j=1}^{N} f_j)\exp(-b_i d) + \sum_{j=1}^{N} f_j \exp(-b_i d \mathbf{r}_i^T \mathbf{R}_j \mathbf{A} \mathbf{R}_j^T \mathbf{r}_i), \quad (1)$$

where S_i is the diffusion-weighted signal for measurement i, S_0 is the signal with no diffusion gradient applied, f_j is the fraction of signal contributed by diffusion along fiber direction j, b_i and $\mathbf{r}_i$ are the b-value and the gradient direction for measurement i, d is the diffusivity, $\mathbf{RAR^T}$ is the anisotropic diffusion tensor along the fiber direction (θ_j, ϕ_j) where

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Each stick is thus represented by two angles, using a spherical coordinate system. The first term represents the diffusion of free water, and the second term represents the diffusion along the different fiber orientations. The joint posterior distribution of the parameters of interest is derived using Bayesian inference. Specifically, Markov Chain Monte Carlo (MCMC) simulation is used in *bedpostx* to generate draws from the complicated posterior distribution. In the case of multiple fiber orientations, automatic relevance determination (ARD) is used to in each voxel determine the optimal number of fibers. The GPU version of *bedpostx* [20] was used in our case, to reduce the processing time from 15 h to 40 min per analysis. Each dataset was analyzed 4 times (maximum of 1, 2, 3 or 4 crossing fibers in each voxel), resulting in a total of 128 analyses for the 32 datasets.

The function *fslmaths* was used to create two seed masks in MNI space (for corticospinal and cingulum gyrus tracts), using the JHU White-Matter Tractography atlas [21]. The transformation between standard space and diffusion space was achieved in three separate steps. First, the anatomical volume was linearly registered to MNI space using the function *flirt* [23]. Second, the diffusion data was linearly registered to the anatomical volume. Third, the two transformations were combined, to transform the seed masks from MNI space to diffusion space.

The probabilistic tractography was performed using the function *probtrackx2* [6] in diffusion space. The *bedpostx* function results in draws from the posterior distribution of the ball and stick model, which are then used by *probtrackx2* to achieve probabilistic fiber tracking. A total of 5000 streamlines were initiated from each seed voxel, and constructed by randomly selecting one draw from the ball and stick model in each voxel, and following the main fiber orientation given by that draw. Each streamline was stopped after 2000 steps with a step length of 0.5 mm, or terminated if the curvature exceeded 80°. A volume containing the output connectivity distribution from the seed mask was finally computed, where each entry denotes the number of streamlines that passed through that voxel. The results of *probtrackx2* were transformed to MNI standard space by linear transformation, inverting the previously calculated linear transformation. The results were finally normalized by the total number of streamlines from the seed mask. By removing the less likely tracts, using a threshold, it becomes easier to see the most important connections from the seed mask to the rest of the brain.

3.2 Dipy

To further extend this study, we performed deterministic tractography on the same datasets using Dipy [17]. The deterministic tractography was performed on the distortion-corrected datasets from FSL. To get directions from the diffusion dataset, we fitted each voxel to a Constant Solid Angle Model [1] using the function *CsaOdfModel*. This model will represent the orientation distribution function (ODF) in each voxel. The ODF is the distribution of water diffusion as a function of direction. The peaks of an ODF can be obtained by the function *peaks_from_model*, and they can be good estimates for the orientations of the streamlines passing through voxels. The same corticospinal and cingulum gyrus tracts were used as seed masks. One seed per voxel (in the center) was used for the tractography.

4 Results

4.1 FSL

Figures 2 and 3 show one axial slice and one sagittal slice of the tractography results for the corticospinal and cingulum gyrus tracts, for all 32 datasets. The tractography results were thresholded at a connectivity value of 0.2%. The background image is the MNI template brain. The corticospinal and cingulum gyrus seed masks, as shown in Fig. 4, were created with *fslmaths*, by thresholding the JHU White-Matter Tractography atlas [28]. Due to noise and scan artifacts, none of the datasets give identical tractography results. Nevertheless, there is clearly a high degree

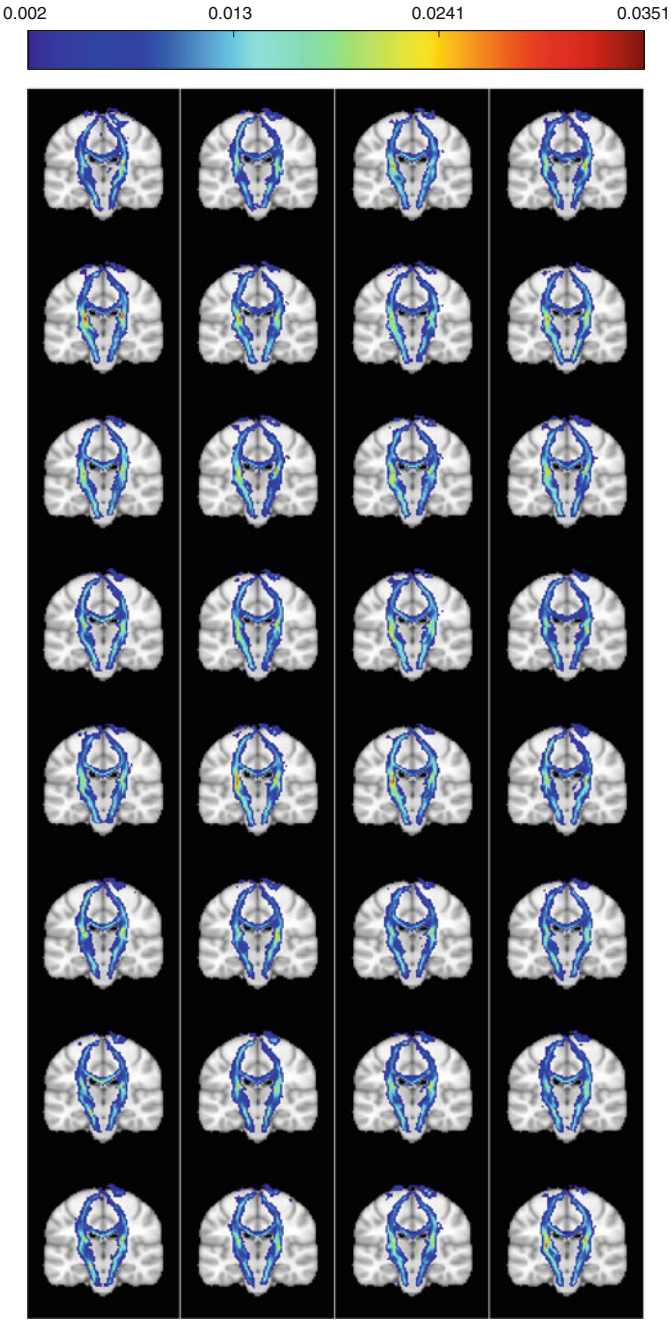


Fig. 2 Tractography results for 32 diffusion datasets (in MNI space), for a maximum of 2 fibers per voxel, when using seed masks for the corticospinal tract. A threshold of 0.2% was used to remove less likely tracts. The results represent the proportion of times a streamline passed through a voxel

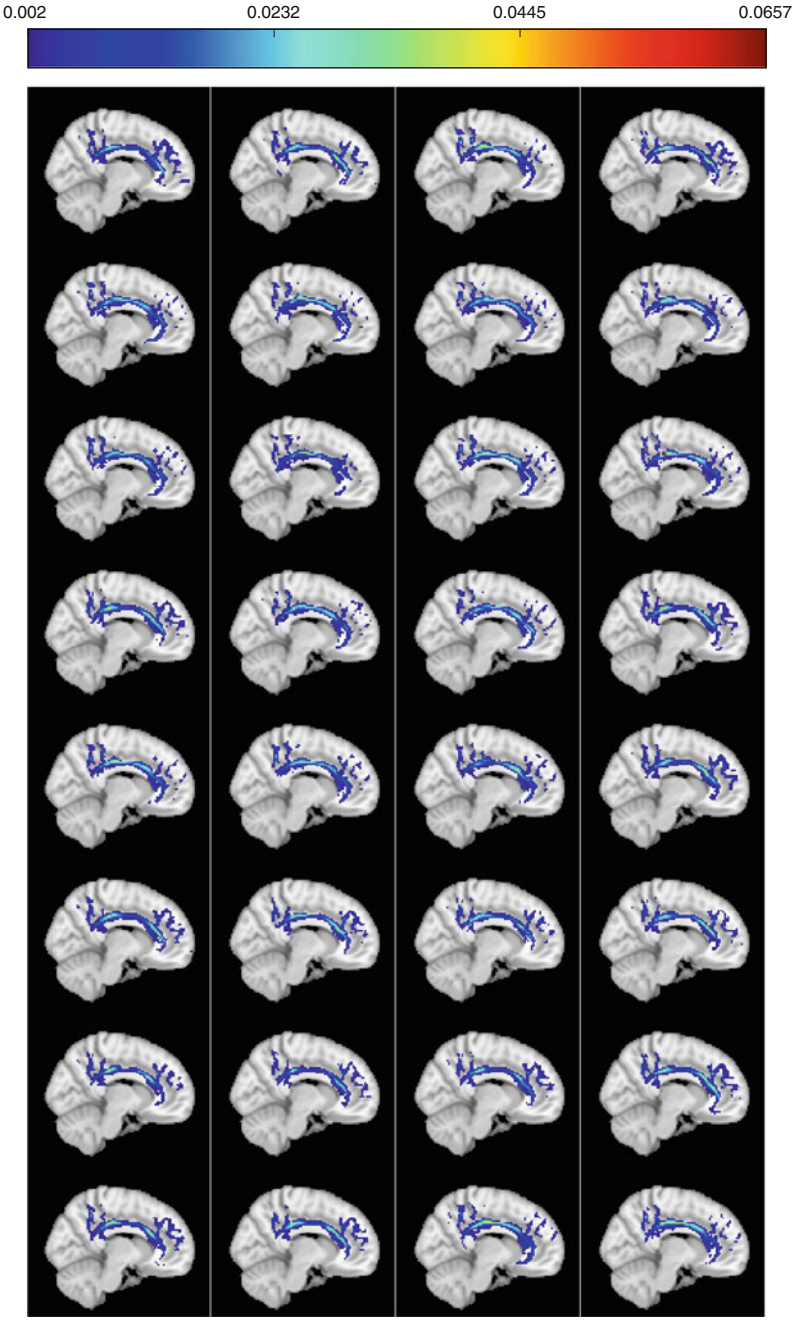


Fig. 3 Tractography results for 32 diffusion datasets (in MNI space), for a maximum of 2 fibers per voxel, when using seed masks for the cingulum gyrus tract. A threshold of 0.2% was used to remove less likely tracts. The results represent the proportion of times a streamline passed through a voxel

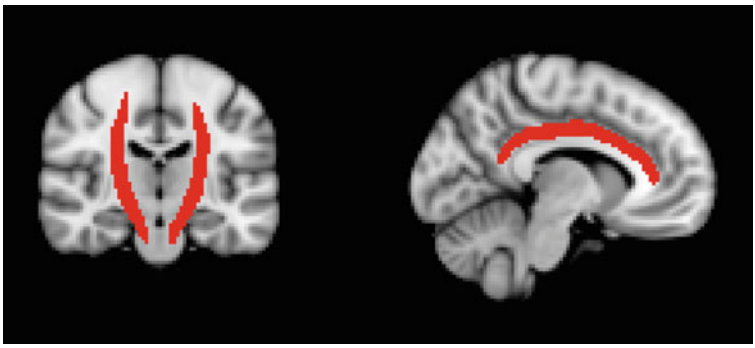


Fig. 4 The corticospinal and cingulum gyrus tracts used as seed masks. From each voxel in the seed mask, 5000 streamlines were generated, using probabilistic fiber tracking, to obtain a volume were each voxel represents the number of times a streamline passed through that voxel

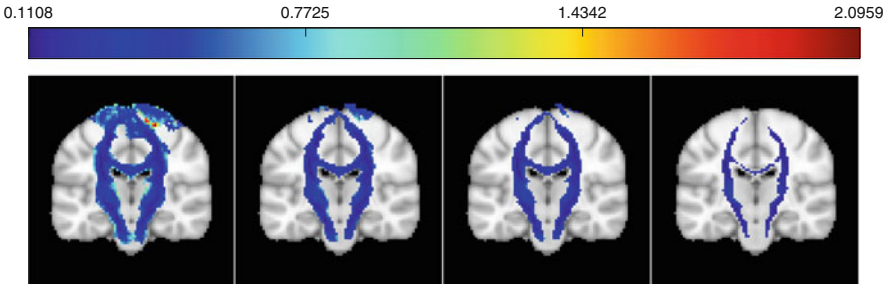


Fig. 5 CV (in MNI space) of the tractography results, for a maximum of 2 fibers per voxel, when using a seed mask for the corticospinal tract. The threshold (in %) is, *from left to right*, 0.01, 0.05, 0.1, 0.5

of similarity between the tractography maps from the different datasets. In the following results, we try to measure the degree of similarity using different metrics.

The reproducibility of the tractography was first examined using the CV, which is defined as the ratio of the standard deviation σ divided by the mean μ:

$$\mathrm{CV} = \frac{\sigma}{\mu}. \tag{2}$$

The CV is a standardized measure of dispersion of a probability distribution [9]. It gives an intuitive estimate of the measurement repeatability, expressed as a relative percentage (regardless of the absolute measurement value). The standard deviation σ and mean μ can be calculated from the 32 tractography results. Figure 5 shows the CV of the tractography results for various thresholds when using a seed mask for the corticospinal tract. By removing the voxels below the threshold, it is possible to focus on the tracts with a lower CV.

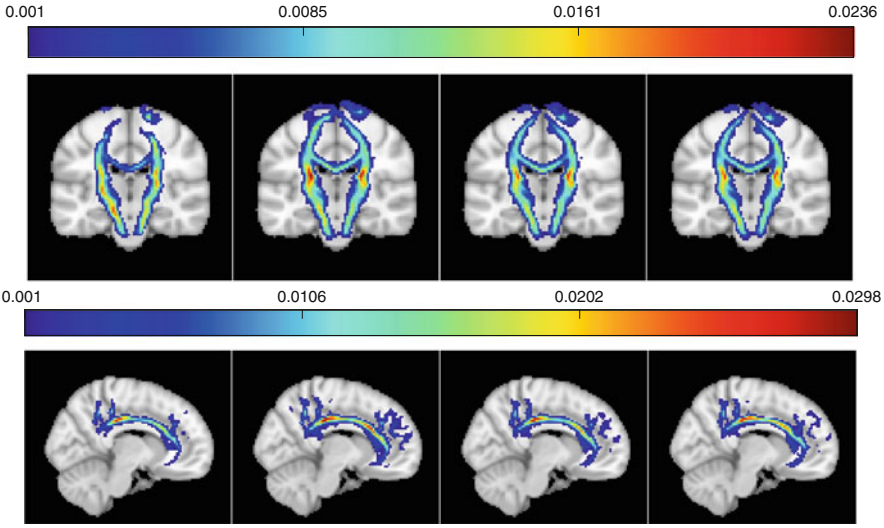


Fig. 6 Averaged tractography results over the 32 datasets (in MNI space), for a maximum of 1, 2, 3 or 4 fibers per voxel (*left to right*), when using seed masks for the corticospinal and cingulum gyrus tracts. A threshold of 0.1% was used to remove less likely tracts. Using a maximum of 2 fibers per voxel leads to including more voxels in the tracts, compared to using a maximum of 1 fiber per voxel

Figure 6 shows the average tractography results over the 32 datasets, when changing the maximum number of crossing fibers x in the *bedpostx* function. It has previously been reported that some 70% of the white matter voxels contain at least two crossing fibers [25]. Our results show that the multi-fiber model is in general able to reveal more connections between the brain areas, compared to the single fiber model, for both corticospinal and cingulum gyrus tracts. The ability to detect connections varies when different maximum number of crossing fibers is used. Nevertheless, the same cores of the tracts were found by all settings. The standard deviation, for the voxels that survived an initial threshold, was calculated over all 32 datasets and is shown in Fig. 7.

Figure 8 shows the tract volume of the tractography results, for different settings of the maximum number of crossing fibers. The tract volume was estimated as the total number of voxels that survived thresholding. The average tract volume of corticospinal tracts over the 32 datasets are 3527, 4648, 4210 and 4208 for a maximum of 1, 2, 3 and 4 crossing fibers. The average tract volume of cingulum gyrus tracts over the 32 datasets are 1892, 3153, 2548 and 2583 for a maximum of 1, 2, 3 and 4 crossing fibers.

To better understand the results in Fig. 8, we plotted the average tract volume over the 32 datasets before and after thresholding for different settings of the maximum number of crossing fibers, see Fig. 9. For both before and after thresholding, tractography results obtained using a single fiber model ($x = 1$ in the FSL function

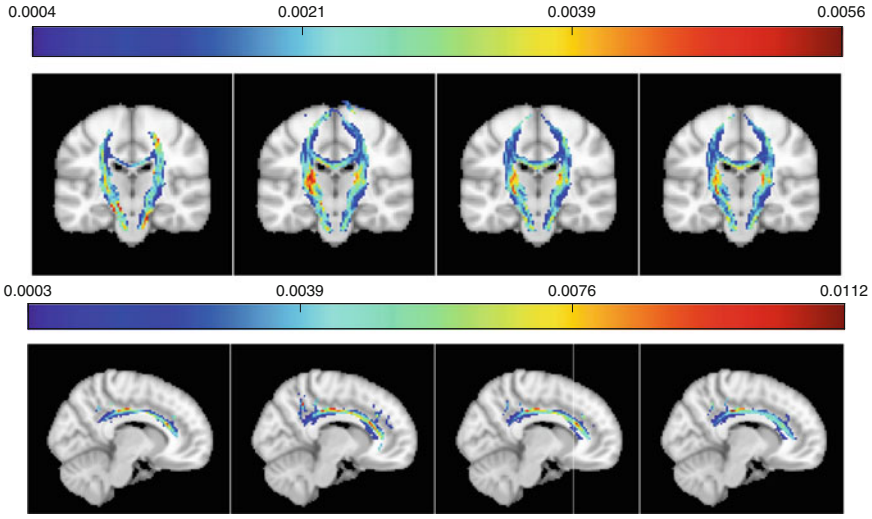


Fig. 7 Standard deviation of tractography results over the 32 datasets (in MNI space), for a maximum of 1, 2, 3 or 4 fibers per voxel (*left to right*), when using seed masks for the corticospinal and cingulum gyrus tracts. A threshold of 0.1% was used to remove less likely tracts

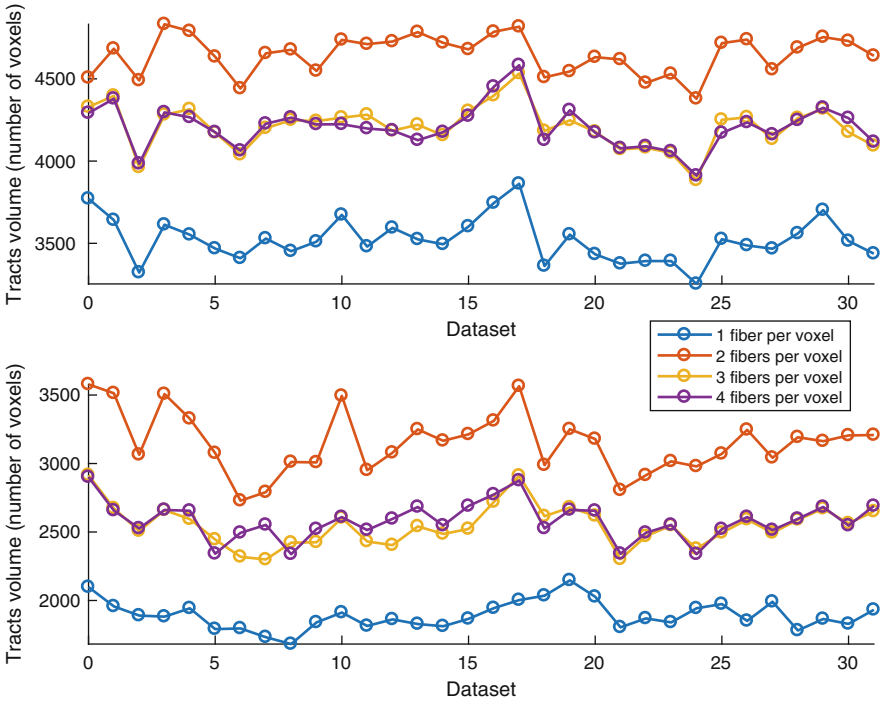


Fig. 8 Tract volume of the tractography results, for a maximum of 1, 2, 3 or 4 fibers per voxel. The *top figure* shows the results for the corticospinal tract, and the *bottom figure* shows the results for the cingulum gyrus tract. A threshold of 0.5% was used to remove less likely tracts. Using a maximum of 2 fibers per voxel reveals most brain connections, while using a maximum of 1 fiber leads to the fewest connections

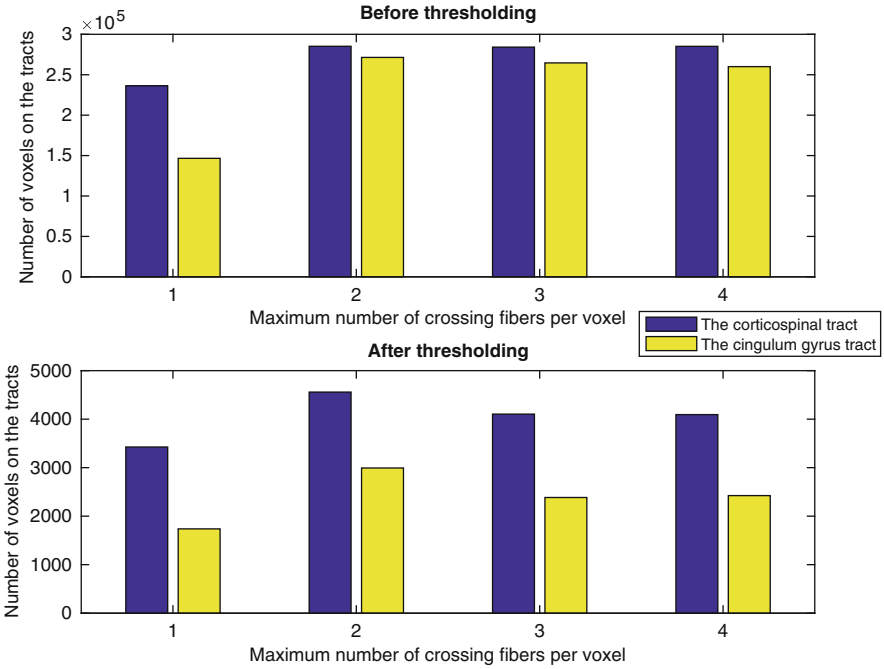


Fig. 9 Tract volume of the tractography results, for a maximum of 1, 2, 3 or 4 fibers per voxel. The *top figure* shows the results before thresholding, and the *bottom figure* shows the results after thresholding

bedpostx) are expected to represent a smaller tract, i.e. there are fewer connections. This is due to the fact that a multi-fiber model ($x = 2, 3, 4$ in the FSL function *bedpostx*) results in a greater variability of fiber orientations and tend to disperse the streamlines to more voxels. This prediction clearly holds for the results in Figs. 8 and 9, for both the corticospinal and the cingulum gyrus tracts. However, after thresholding, the tract volume for the 3 fibers per voxel case was decreased a little compared with a maximum of 2 fibers. The reason is that when a larger number of crossing fibers are fitted, the tractography results tend to be more dispersive, i.e. the connectivity value in the voxels will be smaller. Therefore, a larger amount of voxels will be removed after the thresholding, which causes a smaller tract volume. It is interesting to note that increasing the number of maximum fibers from 3 to 4, does not change the tract volume further. This is because very few voxels in the datasets support 4 crossing fibers, see Figs. 10 and 11.

Figure 10 shows an example of the multi-fiber ball and stick model fitting for one of the 32 datasets. The fraction of signal contributed by diffusion along fiber direction f_i [output of FSL function *bedpostx*, see Eq. (1)] was thresholded at 0.05 [6] and the surviving voxels were able to detect more than i crossing fibers. For the chosen dataset, 40% of voxels with FA > 0.1 were able to detect at least two

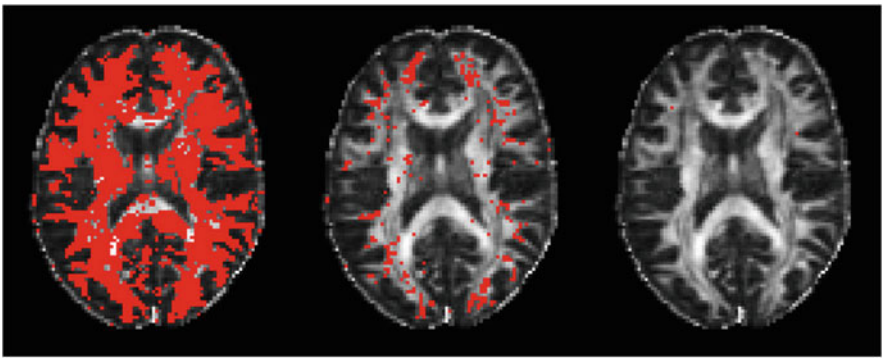


Fig. 10 Multi-fiber ball and stick model fitting. An axial slice showing where more than 1 (*left*), 2 (*middle*), and 3 (*right*) crossing fibers in each voxel were supported by the dataset. The fiber fractions f_2, f_3 and f_4 were thresholded at 0.05

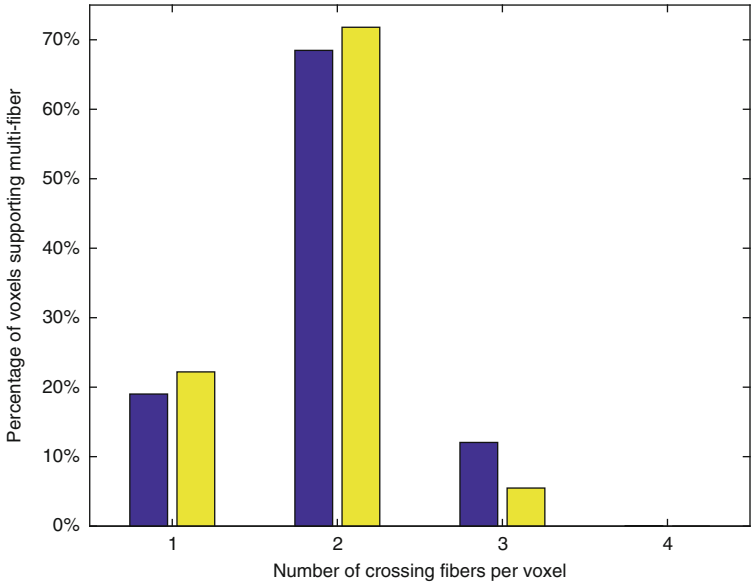


Fig. 11 Percentage of voxels supporting 1, 2, 3 and 4 crossing fibers, corticospinal tract in blue and cingulum gyrus tract in *yellow*. A threshold of 0.5% was used to remove less likely tracts

crossing fibers which is a little higher than the previous reported 33% in [6], where a different data acquisition scheme was used. For only 4.76% of voxels with FA > 0.1, more than two fibers was supported by the model, and for only 0.05% of voxels with FA > 0.1 more than three fibers was supported.

Figure 11 shows the percentage of voxels supporting 1, 2, 3 and 4 crossing fibers in corticospinal (blue) and cingulum gyrus (yellow) tracts for the same chosen dataset as in Fig. 10. For both the corticospinal and the cingulum gyrus (yellow)

tracts, around 20%, 70% and 10% of the voxels supported 1, 2 and 3 crossing fibers, respectively. The number of voxels supporting more than 3 crossing fibers is negligible. In [6] it was reported that no single voxel was able to support more than 2 crossing fibers when 60 directions was used for data acquisition, and no single voxel was able to support more than 1 crossing fibers when 12 directions was used. Tuch et al. [37] reported that it is possible to detect 3 crossing fibers in the corticospinal tract in high b-value data. In our study, the data acquisition scheme using two shells and 30 directions for each shell made it possible to detect 3 crossing fibers in 12.1 and 5.5% of the voxels in the corticospinal and cingulum gyrus tracts, respectively, as shown in Fig. 11. It is reasonable to infer that data acquisition schemes using either more shells, or more directions, or higher b-value allows more crossing fibers to be supported by the data.

Figure 12 shows the probability density of the CV when the maximum number of crossing fibers varies from 1 to 4. The CV is the precision of a measure, i.e. a smaller CV is equivalent to a higher reproducibility of the fiber tracts. The probability density was estimated using the MATLAB (Version 2016b) function *ksdensity* with 100 bins. For the corticospinal case, the total number of voxels after

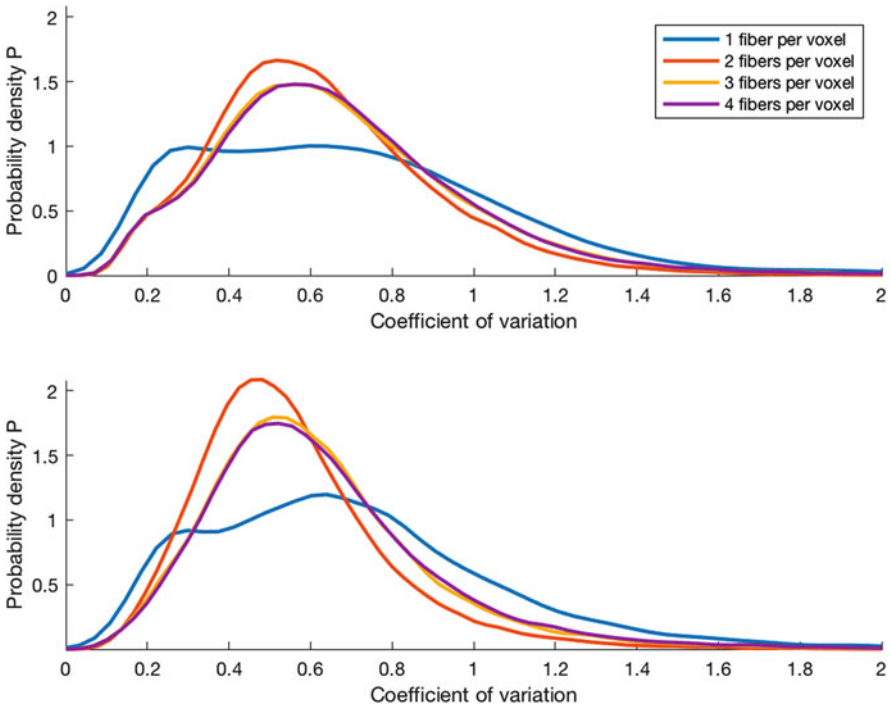


Fig. 12 Probability density of CV for the tractography results, for a maximum of 1, 2, 3 or 4 fibers per voxel. The *top figure* shows the results for the corticospinal tract, and the *bottom figure* shows the results for the cingulum gyrus tract. A threshold of 0.001% was used to remove less likely tracts

thresholding is 48010, 172799, 143935, and 146322 for a maximum of 1, 2, 3 and 4 fibers, respectively. The corresponding number of voxels for $0 < CV < 1$ is 15556, 112227, 76359, and 77328. The mean of the CV over all voxels is 1.32, 0.99, 1.15, and 1.16, for a maximum of 1–4 fibers. The choice of a maximum of 2 fibers per voxel gives the highest reproducibility, at the cost of revealing fewer connections between brain areas, compared with a maximum of 3 or 4 fibers per voxel. The single-fiber model leads to the highest variation. Figure 12 shows the probability density of the CV with and without distortion correction, when the maximum number of crossing fibers is set to 3. Correcting for susceptibility effects, eddy currents and head motion clearly increases the reproducibility (Fig. 13).

We derived mean and CV for tract FA, MD and volume (Table 1) across the 32 datasets for the corticospinal and cingulum gyrus tracts when a maximum of 1, 2, 3 and 4 fibers per voxel was applied. A threshold of 0.5% was used to remove less likely tracts. Tract FA and MD are defined as the average FA and MD values within a tract. The tract volume is the total number of voxels that survived thresholding. FA and MD of whole brain were obtained using the FSL function *dtifit* to fit a diffusion tensor model in each voxel. FA and MD results were then linearly

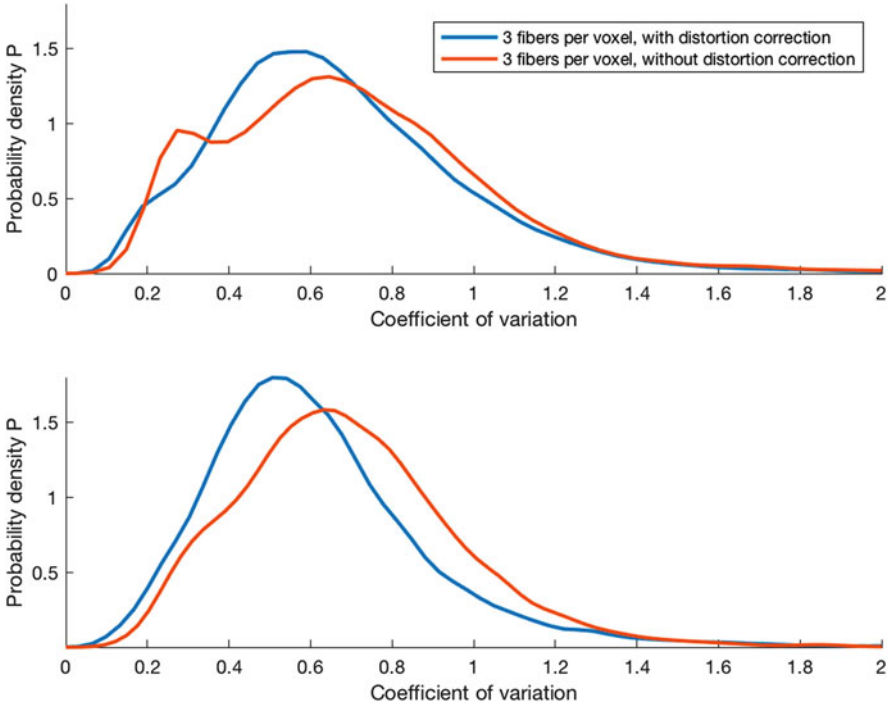


Fig. 13 Probability density of CV for the tractography results with and without distortion correction, for a maximum of 3 fibers per voxel. The *top figure* shows the results for the corticospinal tract, and the *bottom figure* shows the results for the cingulum gyrus tract. A threshold of 0.001% was used to remove less likely tracts

Table 1 Results for tracts analysis of mean tract FA, MD and tract volume

Measure	Tract	Maximum number of crossing fibers	Mean across datasets	CV across (%) datasets
FA	Corticospinal	1	0.58	1.03
		2	0.55	1.12
		3	0.57	1.04
		4	0.58	1.06
	Cingulum	1	0.50	4.29
		2	0.47	3.96
		3	0.49	3.86
		4	0.48	4.78
MD	Corticospinal	1	($\times 10^{-4}$) 5.48	1.40
		2	5.61	2.17
		3	5.52	1.78
		4	5.51	1.97
	Cingulum	1	5.86	2.01
		2	5.87	2.42
		3	5.82	1.89
		4	5.86	2.53
Tract volume	Corticospinal	1	3522	3.74
		2	4648	2.50
		3	4210	3.06
		4	4208	3.05
	Cingulum	1	1892	5.36
		2	3153	6.87
		3	2548	5.74
		4	2583	5.20

A threshold of 0.5% was used. Tract volume is defined as the number of voxels of the tract

registered to MNI space using the function *flirt* [23]. Mean tract FA across datasets ranged from 0.55 to 0.58 and 0.47 to 0.50 for the corticospinal and cingulum gyrus tracts, respectively. Measures of mean tract FA, MD and tract volume produced very low CVs, for both corticospinal (below 3.06%) and cingulum gyrus (below 6.87%) tracts. CVs for mean tract FA of cingulum gyrus across datasets (3.86–4.78%) are largely consistent with previous reported results (3.18–4.32%) [19]. Results for the corticospinal tract show a higher degree of reproducibility than the cingulum gyrus tract. It is consistent with previous research [19, 39] that larger tracts can produce a lower CV since they are less sensitive to the uncertainty of tractography, artifacts of scanning and noise.

The agreement of two tracts was quantified using the DSC [15], which quantifies the degree of overlap using a number between 0 (no overlap) to 1 (complete overlap):

$$\mathrm{DSC} = \frac{2n_{ab}}{n_a + n_b}, \tag{3}$$

where n_{ab} is the number of voxels common to both volumes, and n_a and n_b are the number of voxels in volume a and volume b. The average tracts over the 32 datasets was used as the benchmark, and the DSC was then calculated between each tractography result and the average one. The results are shown in Fig. 14. With a threshold of 0.5%, the mean of the DSC for the corticospinal tract over the 32 datasets is 0.82, 0.85, 0.84 and 0.84 for a maximum of 1, 2, 3 and 4 fibers per voxel, respectively. Bauer et al. [5] repeated deterministic tractography for the corticospinal tract and reported that a DSC above 0.8 can be achieved. This is largely consistent with our results. The mean of the DSC for the cingulum gyrus tract over the 32 datasets is 0.76, 0.78, 0.77 and 0.81 for a maximum of 1, 2, 3 and 4 fibers

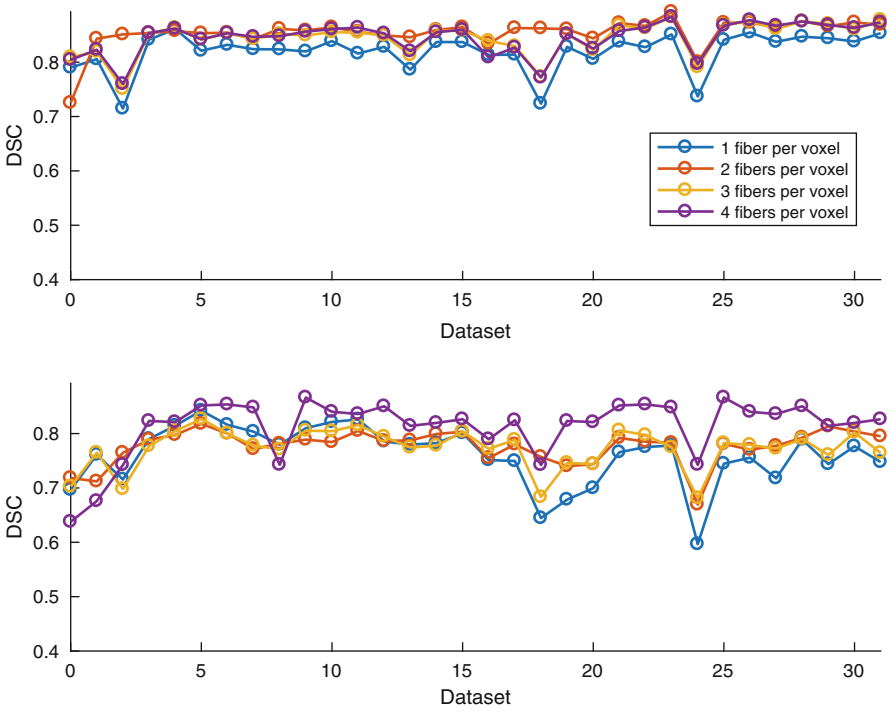


Fig. 14 DSCs for the tractography results, for a maximum of 1, 2, 3 or 4 fibers per voxel. The *top figure* shows the results for the corticospinal tract, and the *bottom figure* shows the results for the cingulum gyrus tract. A threshold of 0.5% was used to remove less likely tracts

per voxel, respectively. This is very close to the previous reported 0.8 in [7]. The DSC for the cingulum gyrus tract shows a larger variance than for the corticospinal tract. The standard deviation of the DSC for the corticospinal tract is 0.035, 0.0278, 0.03 and 0.0295 for a maximum of 1, 2, 3 and 4 fibers per voxel, respectively. The standard deviation of the DSC for the cingulum gyrus tract is 0.053, 0.031, 0.036 and 0.0532 for a maximum of 1, 2, 3 and 4 fibers per voxel, respectively. Our results thereby suggest that the reproducibility of fiber tracts depends on the seed mask used. Such an effect is natural, because different parts of the brain may suffer differently from distortions and head motion. The size of the seed mask is also a factor that has an influence on the degree of reproducibility. For the DSC, the setting of the maximum of fibers per voxel did not produce a significant difference. It it hard to tell which setting provided the highest reproducibility, but in general the multi-fiber model achieved a better performance. Figure 15 shows the DSC with and without distortion correction, for a maximum of 3 crossing fibers per voxel. Together with Fig. 13, we again draw the conclusion that distortion correction leads to a higher reproducibility.

From Figs. 6, 8 and 14 we can see that the tractography for the different datasets resulted in very similar fiber pathways, and does not reveal more information when

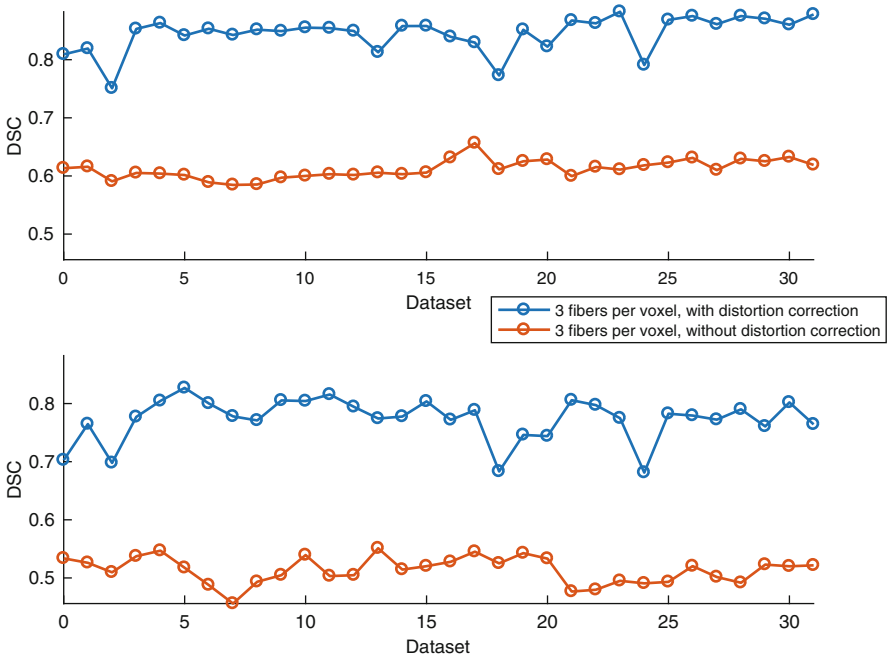


Fig. 15 DSCs for the tractography results with and without distortion correction, for a maximum of 3 fibers per voxel. The *top figure* shows the results for the corticospinal tract, and the *bottom figure* shows the results for the cingulum gyrus tract. A threshold of 0.1% was used to remove less likely tracts

the maximum number of fibers is set to 4. It can be concluded that a maximum of 3 fibers per voxel may be sufficient to reveal connections between brain areas, considering the longer computation time when a larger number of maximum fibers is used.

The Bayesian estimation of diffusion parameters implemented in *bedposts* takes approximately 15 h to complete for one diffusion dataset. Using a graphics card, the GPU version of *bedpostx* takes 10–60 min depending on the computer hardware, the chosen maximum number of fibers and the size of the datasets. The processing time for Bayesian estimation of the multi-fiber models are 10, 18, 27 and 40 min (for a maximum number of 1, 2, 3 and 4 fibers), using the graphics card NVIDIA Tesla K40c and the CPU Intel i7-5820K 3.30 GHz. The processing times for probabilistic tractography of the multi-fiber models are 68 min, 114 min, 112 min and 115 min, respectively.

4.2 Dipy

Figures 16 and 17 show one axial slice and one sagittal slice of the deterministic tractography results from Dipy, for the corticospinal and cingulum gyrus tracts for all 32 datasets. The background image is the MNI brain template. As in Figs. 2 and 3, they also show a high degree of reproducibility for the tractography results from the different datasets. In the following results, we try to measure the degree of similarity using different metrics.

Figure 18 shows the DSCs for the deterministic tractography results from Dipy. The tractography result of the first dataset was chosen as the benchmark, and the DSC was then calculated between each tractography result and the benchmark. Please note that the DSCs for the results from FSL and Dipy were calculated based on different benchmarks. Therefore the DSCs cannot be compared directly. The mean DSC for the corticospinal and cingulum gyrus tracts over the latter 31 datasets is 0.669 and 0.590, respectively. Compared with the results in Fig. 14 we can see that the deterministic tractography results from Dipy show a lower degree of reproducibility than the probabilistic tractography results from FSL. Although the deterministic tractography algorithm is very efficient, it can be sensitive to the estimated principal directions since the streamline in each voxel only follows the principal direction. The deterministic tractography algorithm fails to resolve complex fiber structure when fibers are crossing, which is one of the major limitations of the deterministic tractography algorithm. The uncertainties in the underlying fiber directions makes the tractography less reproducible than its probabilistic counterpart.

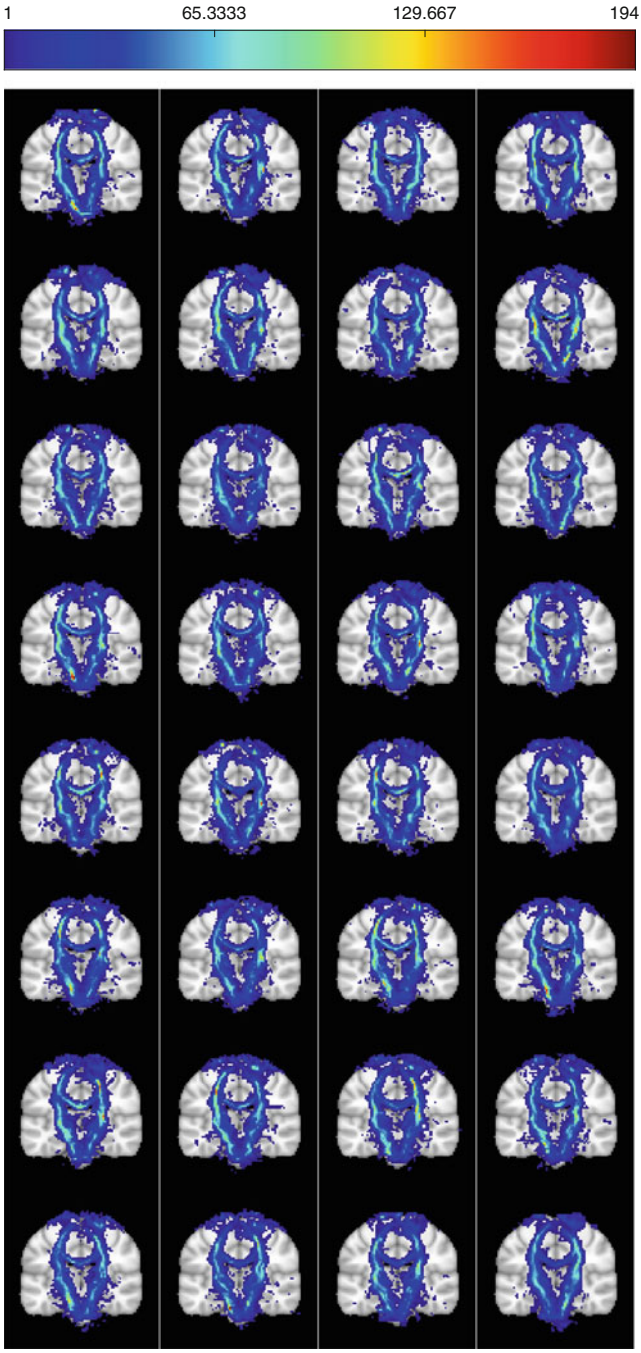


Fig. 16 Deterministic tractography results for 32 diffusion datasets (in MNI space), when using seed masks for the corticospinal tract. The results represent the number of streamlines that passed through a voxel

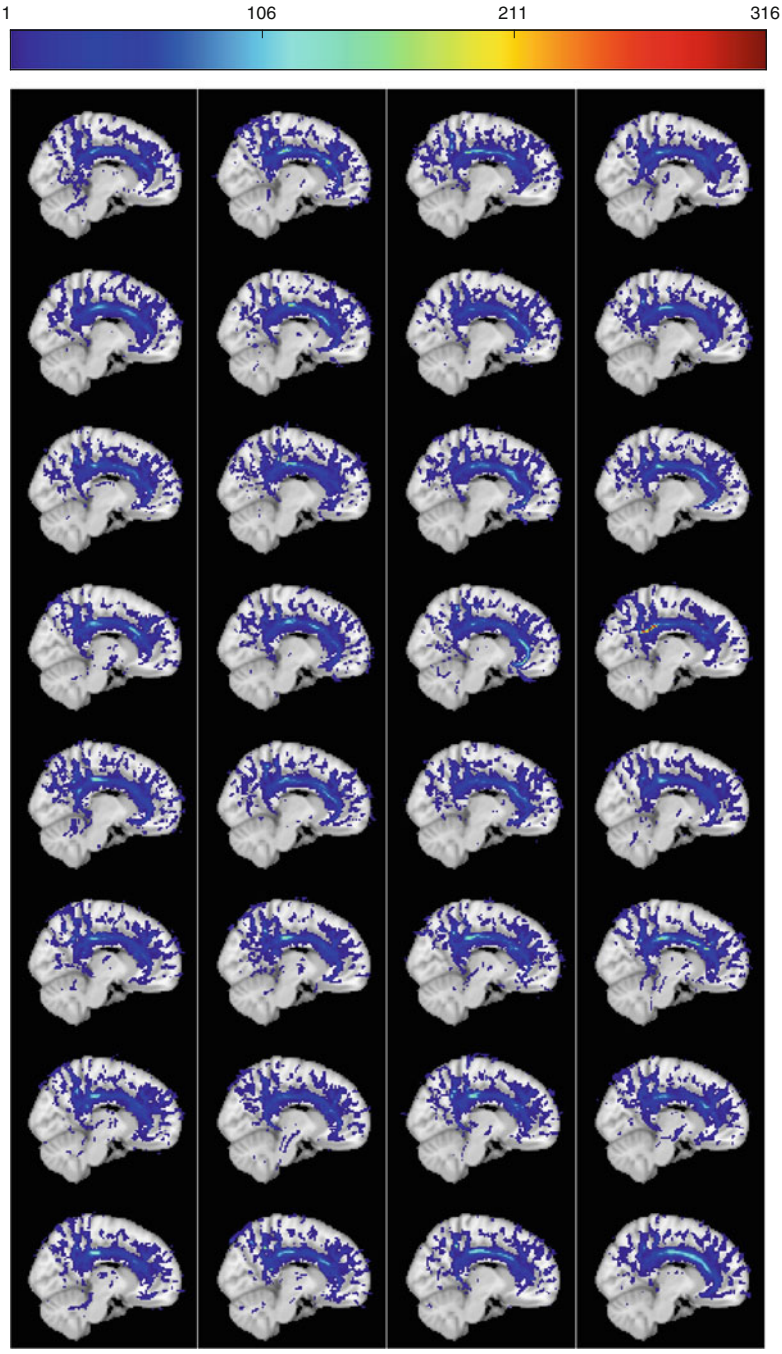


Fig. 17 Deterministic tractography results for 32 diffusion datasets (in MNI space), when using seed masks for the cingulum gyrus tract. The results represent the number of streamlines that passed through a voxel

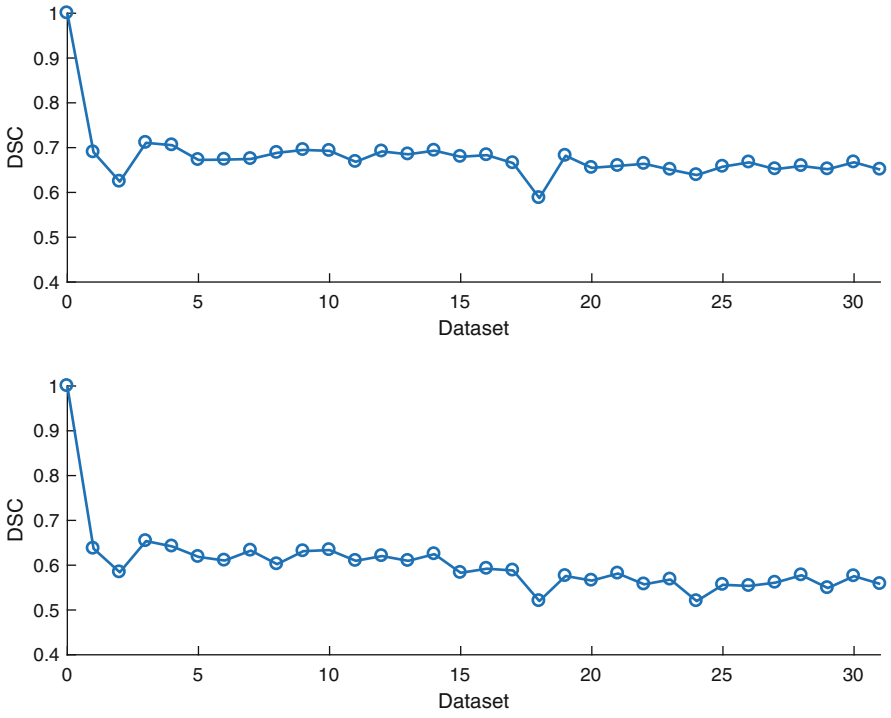


Fig. 18 DSCs for the tractography results using Dipy. The first tractography map was used as the benchmark, and the DSC was then calculated between each tractography map and the benchmark. The *top figure* shows the results for the corticospinal tract, and the *bottom figure* shows the results for the cingulum gyrus tract

5 Discussion and Conclusion

In this study, we have presumed that the brain nerve tracts do not change significantly during the scan interval (1 year). There are previous studies [30, 41] demonstrating changes of certain white matter tracts due to aging. However, it is unlikely to recognize significant age-related changes over the course of 1 year for a healthy subject.

We have investigated the test–retest reliability of diffusion tractography, using 32 diffusion datasets from a single healthy volunteer. A visual comparison of the results shows that the cores of the corticospinal and cingulum gyrus tracts are common over the 32 datasets, for both FSL and Dipy. We have reported inter-dataset overlaps of DSC = 0.6–0.9 for the probabilistic tractography results from FSL, and DSC = 0.58–0.71 for the deterministic tractography results from Dipy. The DSC values roughly fit with the range reported as 0.67–0.9 in [7]. We also observed that the DSC differs between the corticospinal and cingulum tracts. This is because the size of the tracts can be a factor influencing the degree of reproducibility. Also, different

parts of the brain may experience different distortions and head motion. The results indicate that distortions and head motion can be an important uncertainty source. It was observed that the reproducibility increases if distortion correction is used, for both the corticospinal and cingulum gyrus tracts. The results suggest that the ball and stick model representing multiple fiber orientations can reconstruct more connections, at the cost of a longer processing time. It is also demonstrated that the tractography results do not differ much when the maximum number of crossing fibers is larger than 3 in the FSL function *bedpostx*, but a higher number of crossing fibers may be optimal for DWI data collected with a higher number of gradient directions and shells.

Based on the presented results we conclude that the tractography results obtained with different software packages, and different parameter settings, show a rather high reproducibility. It is important to note that the reproducibility of tractography by no means can be interpreted as the accuracy of tractography. Nonexisting fiber pathways can, in theory, be reconstructed with a high reproducibility. Future work will be focused on evaluating the reproducibility of other tractography softwares, such as TORTOISE [32] and DSI-Studio [40].

Acknowledgements We thank Russell Poldrack and his colleagues for sharing the data from the MyConnectome project. We also thank Cyril Pernet and his colleagues for sharing neuroimaging data from brain tumor patients. The Nvidia Corporation is acknowledged for the donation of the Tesla K40 graphics card. This research was supported by the Information Technology for European Advancement (ITEA) 3 Project BENEFIT (better effectiveness and efficiency by measuring and modelling of interventional therapy) and the Swedish Research Council (grant 2015-05356, "Learning of sets of diffusion MRI sequences for optimal imaging of micro structures"). Anders Eklund was also supported by Swedish Research Council Grant 2013-5229 ("Statistical Analysis of fMRI Data").

References

1. Aganj, I., Lenglet, C., Sapiro, G., Yacoub, E., Ugurbil, K., Harel, N.: Reconstruction of the orientation distribution function in single-and multiple-shell q-ball imaging within constant solid angle. Magn. Reson. Med. **64**(2), 554–566 (2010)
2. Andersson, J.L., Sotiropoulos, S.N.: An integrated approach to correction for off-resonance effects and subject movement in diffusion MR imaging. NeuroImage **125**, 1063–1078 (2016)
3. Andersson, J.L., Skare, S., Ashburner, J.: How to correct susceptibility distortions in spin-echo echo-planar images: application to diffusion tensor imaging. Neuroimage **20**(2), 870–888 (2003)
4. Basser, P.J., Pajevic, S., Pierpaoli, C., Duda, J., Aldroubi, A.: In vivo fiber tractography using DT-MRI data. Magn. Reson. Med. **44**(4), 625–632 (2000)
5. Bauer, M.H., Kuhnt, D., Barbieri, S., Klein, J., Becker, A., Freisleben, B., Hahn, H.K., Nimsky, C.: Reconstruction of white matter tracts via repeated deterministic streamline tracking–initial experience. PloS One **8**(5), e63082 (2013)
6. Behrens, T., Berg, H.J., Jbabdi, S., Rushworth, M., Woolrich, M.: Probabilistic diffusion tractography with multiple fibre orientations: what can we gain? Neuroimage **34**(1), 144–155 (2007)

7. Besseling, R.M., Jansen, J.F., Overvliet, G.M., Vaessen, M.J., Braakman, H.M., Hofman, P.A., Aldenkamp, A.P., Backes, W.H.: Tract specific reproducibility of tractography based morphology and diffusion metrics. PloS One **7**(4), e34125 (2012)
8. Bizzi, A.: Diffusion imaging with MR tractography for brain tumor surgery. In: Clinical Functional MRI, pp. 179–228. Springer, Berlin (2015)
9. Brown, C.E.: Coefficient of variation. In: Applied Multivariate Statistics in Geohydrology and Related Sciences, pp. 155–157. Springer, Berlin (1998)
10. Caan, M.W.: DTI analysis methods: fibre tracking and connectivity. In: Diffusion Tensor Imaging, pp. 205–228. Springer, Berlin (2016)
11. Castellano, A., Bello, L., Michelozzi, C., Gallucci, M., Fava, E., Iadanza, A., Riva, M., Casaceli, G., Falini, A.: Role of diffusion tensor magnetic resonance tractography in predicting the extent of resection in glioma surgery. Neuro-Oncology **14**(2), 192–202 (2012)
12. Côté, M.-A., Girard, G., Boré, A., Garyfallidis, E., Houde, J.-C., Descoteaux, M.: Tractometer: towards validation of tractography pipelines. Med. Image Anal. **17**(7), 844–857 (2013)
13. Cutajar, M., Clayden, J.D., Clark, C.A., Gordon, I.: Test–retest reliability and repeatability of renal diffusion tensor MRI in healthy subjects. Eur. J. Radiol. **80**(3), e263–e268 (2011)
14. Danielian, L.E., Iwata, N.K., Thomasson, D.M., Floeter, M.K.: Reliability of fiber tracking measurements in diffusion tensor imaging for longitudinal study. Neuroimage **49**(2), 1572–1580 (2010)
15. Dice, L.R.: Measures of the amount of ecologic association between species. Ecology **26**(3), 297–302 (1945)
16. Fillard, P., Descoteaux, M., Goh, A., Gouttard, S., Jeurissen, B., Malcolm, J., Ramirez-Manzanares, A., Reisert, M., Sakaie, K., Tensaouti, F., et al.: Quantitative evaluation of 10 tractography algorithms on a realistic diffusion MR phantom. Neuroimage **56**(1), 220–234 (2011)
17. Garyfallidis, E., Brett, M., Amirbekian, B., Rokem, A., Van Der Walt, S., Descoteaux, M., Nimmo-Smith, I.: Dipy, a library for the analysis of diffusion MRI data. Front. Neuroinform. **8**, 8 (2014)
18. Hagmann, P., Gigandet, X., Meuli, R., Kötter, R., Sporns, O., Wedeen, V.: Quantitative validation of MR tractography using the cocomac database. In: Proceedings of 16th Annual Meeting of the ISMRM, EPFL-CONF-135048, p. 427 (2008)
19. Heiervang, E., Behrens, T., Mackay, C., Robson, M., Johansen-Berg, H.: Between session reproducibility and between subject variability of diffusion MR and tractography measures. Neuroimage **33**(3), 867–877 (2006)
20. Hernández, M., Guerrero, G.D., Cecilia, J.M., García, J.M., Inuggi, A., Jbabdi, S., Behrens, T.E., Sotiropoulos, S.N.: Accelerating fibre orientation estimation from diffusion weighted magnetic resonance imaging using GPUs. PloS One **8**(4), e61892 (2013)
21. Hua, K., Zhang, J., Wakana, S., Jiang, H., Li, X., Reich, D.S., Calabresi, P.A., Pekar, J.J., van Zijl, P.C., Mori, S.: Tract probability maps in stereotaxic spaces: analyses of white matter anatomy and tract-specific quantification. Neuroimage **39**(1), 336–347 (2008)
22. Iliescu, B., Negru, D., Poeata, I.: MR tractography for preoperative planning in patients with cerebral tumors in eloquent areas. Rom. Neurosurg. **17**(4), 413–420 (2010)
23. Jenkinson, M., Smith, S.: A global optimisation method for robust affine registration of brain images. Med. Image Anal. **5**(2), 143–156 (2001)
24. Jenkinson, M., Beckmann, C.F., Behrens, T.E., Woolrich, M.W., Smith, S.M.: FSL. Neuroimage **62**(2), 782–790 (2012)
25. Jeurissen, B., Leemans, A., Tournier, J.-D., Jones, D.K., Sijbers, J.: Investigating the prevalence of complex fiber configurations in white matter tissue with diffusion magnetic resonance imaging. Hum. Brain Mapp. **34**(11), 2747–2766 (2013)
26. Laumann, T.O., Gordon, E.M., Adeyemo, B., Snyder, A.Z., Joo, S.J., Chen, M.-Y., Gilmore, A.W., McDermott, K.B., Nelson, S.M., Dosenbach, N.U., Schlaggar, B.L., Mumford, J.A., Poldrack, R.A., Petersen, S.E.: Functional system and areal organization of a highly sampled individual human brain. Neuron **87**, 657–670 (2015)

27. Mastronardi, L., Bozzao, A., D'Andrea, G., Romano, A., Caroli, M., Cipriani, V., Ferrante, M., Ferrante, L.: Use of preoperative and intraoperative magnetic resonance tractography in intracranial tumor surgery. Clin. Neurosurg. **55**, 160–164 (2008)
28. Mori, S., Wakana, S., Van Zijl, P.C., Nagae-Poetscher, L.: MRI Atlas of Human White Matter. Elsevier, Amsterdam (2005)
29. Neher, P.F., Descoteaux, M., Houde, J.-C., Stieltjes, B., Maier-Hein, K.H.: Strengths and weaknesses of state of the art fiber tractography pipelines—a comprehensive in-vivo and phantom evaluation study using tractometer. Med. Image Anal. **26**(1), 287–305 (2015)
30. Nusbaum, A.O., Tang, C.Y., Buchsbaum, M.S., Wei, T.C., Atlas, S.W.: Regional and global changes in cerebral diffusion with normal aging. Am. J. Neuroradiol. **22**(1), 136–142 (2001)
31. Pernet, C.R., Gorgolewski, K.J., Job, D., Rodriguez, D., Whittle, I., Wardlaw, J.: A structural and functional magnetic resonance imaging dataset of brain tumour patients. Sci. Data **3** (2016). doi:10.1038/sdata.2016.3
32. Pierpaoli, C., Walker, L., Irfanoglu, M., Barnett, A., Basser, P., Chang, L., Koay, C., Pajevic, S., Rohde, G., Sarlls, J., et al.: Tortoise: an integrated software package for processing of diffusion MRI data. In: ISMRM 18th Annual Meeting, Stockholm, vol. 18, p. 1597 (2010)
33. Pujol, S., Wells, W., Pierpaoli, C., Brun, C., Gee, J., Cheng, G., Vemuri, B., Commowick, O., Prima, S., Stamm, A., et al.: The DTI challenge: toward standardized evaluation of diffusion tensor imaging tractography for neurosurgery. J. Neuroimaging **25**(6), 875–882 (2015)
34. Tagliafico, A., Calabrese, M., Puntoni, M., Pace, D., Baio, G., Neumaier, C.E., Martinoli, C.: Brachial plexus MR imaging: accuracy and reproducibility of DTI-derived measurements and fibre tractography at 3.0-t. Eur. Radiol. **21**(8), 1764–1771 (2011)
35. Tensaouti, F., Delion, M., Lotterie, J.A., Clarisse, P., Berry, I.: Reproducibility and reliability of the DTI fiber tracking algorithm integrated in the Sisyphe software. In: 2008 First Workshops on Image Processing Theory, Tools and Applications (2008)
36. Tensaouti, F., Lahlou, I., Clarisse, P., Lotterie, J.A., Berry, I.: Quantitative and reproducibility study of four tractography algorithms used in clinical routine. J. Magn. Reson. Imaging **34**(1), 165–172 (2011)
37. Tuch, D.S., Reese, T.G., Wiegell, M.R., Wedeen, V.J.: Diffusion MRI of complex neural architecture. Neuron **40**(5), 885–895 (2003)
38. Vaessen, M., Hofman, P., Tijssen, H., Aldenkamp, A., Jansen, J., Backes, W.H.: The effect and reproducibility of different clinical DTI gradient sets on small world brain connectivity measures. Neuroimage **51**(3), 1106–1116 (2010)
39. Vollmar, C., O'Muircheartaigh, J., Barker, G.J., Symms, M.R., Thompson, P., Kumari, V., Duncan, J.S., Richardson, M.P., Koepp, M.J.: Identical, but not the same: intra-site and inter-site reproducibility of fractional anisotropy measures on two 3.0 t scanners. Neuroimage **51**(4), 1384–1394 (2010)
40. Yeh, F.-C., Verstynen, T.D., Wang, Y., Fernández-Miranda, J.C., Tseng, W.-Y.I.: Deterministic diffusion fiber tracking improved by quantitative anisotropy. PloS One **8**(11), e80713 (2013)
41. Yoon, B., Shim, Y.-S., Lee, K.-S., Shon, Y.-M., Yang, D.-W.: Region-specific changes of cerebral white matter during normal aging: a diffusion-tensor analysis. Arch. Gerontol. Geriatr. **47**(1), 129–138 (2008)

Automatic Atlas-Based Segmentation of Brain White Matter in Neonates at Risk for Neurodevelopmental Disorders

L. Fonseca, C. van Pul, N. Lori, R. van den Boom, P. Andriessen, J. Buijs, and A. Vilanova

Abstract Very preterm infants, <32 weeks gestation, are at high risk for brain injury. Cognitive deficits are often diagnosed at a later stage, since there are no available predictive biomarkers in the neonatal period. The maturation of specific white matter (WM) brain structures is considered a promising early-stage biomarker. With Diffusion Tensor Imaging (DTI) tractography, an in vivo and non-invasive evaluation of these anatomical structures is possible.

We developed an automatic tractography segmentation pipeline, which allows for maturation assessment of the different segmented WM structures. Our segmentation pipeline is atlas-based, specifically designed for premature neonates at term equivalent age. In order to better make use of global information from tractography, all processing is done in the fiber domain. Segmented fiber bundles are further automatically quantified with respect to volume and anisotropy. Of the 24 automatically segmented neonatal tractographies, only three contained more than 30% mislabeled fibers. Results show no dependency to WM pathology. By automatically segmenting WM, we reduced the user-dependency and bias characteristic of manual methods. This study assesses the structure of the neonatal brain based on an automatic WM segmentation in the fiber domain method using DTI tractography data.

L. Fonseca • R. van den Boom
Medical Image Analysis, Eindhoven University of Technology, Eindhoven, The Netherlands

C. van Pul
Clinical Physics, Máxima Medical Center, Veldhoven, The Netherlands

School of Medical Physics and Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

N. Lori
LANEN, INECO, INCYT (Favaloro-CONICET), Rosario, Argentina

Centre Algoritmi, University of Minho, Braga, Portugal

P. Andriessen • J. Buijs
Neonatology Maxima Medical Center, Veldhoven, The Netherlands

A. Vilanova (✉)
Computer Graphics and Visualization, Delft University of Technology, Delft, The Netherlands

Medical Image Analysis, Eindhoven University of Technology, Eindhoven, The Netherlands
e-mail: a.vilanova@tudelft.nl

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_15

1 Introduction

Each year, over half a million babies in Europe are born prematurely. Very preterm infants, <32 weeks gestation, are at high risk for cognitive deficits without major motor deficits [33]. Therapy can be effective, especially when started at early postnatal age, which is a stage characterized by high brain plasticity. However, many patients are diagnosed at a later and more rigid neurdevelopmental stage. In order to offer patients the best possible chances of rehabilitation, early postnatal predictive biomarkers are necessary [9, 13, 16, 38].

Deviations in the development of white matter structures are considered promising early-stage biomarkers. Cognitive development is related to the maturation of the brain networks, the WM structures. These structures can be in vivo and non-invasively reconstructed by Diffusion Tensor Imaging (DTI) tractography [2, 22, 23]. Diffusion of water molecules in the brain is restricted by the underlying anatomical structure, becoming anisotropic for voxels containing WM. Even in newborns, despite their lower brain maturation, tractography can be used to reconstruct WM structures. Analysis of the WM structures can be used to show abnormalities in diffusion tensor parameters and in fibers being associated with maturational problems [3, 12, 14, 29, 30, 32].

In order to assess maturation of specific WM anatomical structures, i.e., bundles, these structures need to be first identified among the complete fiber dataset. This is made possible by a segmentation process that classifies the fibers into different bundles. Although segmentation can be done manually [14, 30, 39], it requires extensive knowledge about complex WM fiber anatomy, introduces user bias, and can become too time consuming for practical use. Furthermore, some structures have challenging shapes that are difficult to segment manually.

Automatic segmentation methods have been developed for adult tractography [6, 11, 17, 26, 27, 40], however, they need to be redesigned to account for the underdeveloped stage of the neonatal brain, since neonatal tractography results in a lower number of fibers, displays smaller and broken fibers that cannot be ignored, misses anatomical structures yet to develop, and has a higher sensitivity to partial volume effects due to their smaller brain size [24, 29, 45].

Improvement of the tractography algorithm can be used to mitigate some of the previous referred problems. Nevertheless, such improvements involves making assumptions that will generate other issues. Some DTI atlas-based methods use voxel level scalar-data [1, 8, 31], which is mainly based on local information. On the other hand, fiber-wise methods employ global information, which can allow for a better overview of the WM anatomical structures, and eventual abnormalities on these structures [6, 15, 18, 20, 25, 26, 42, 46].

In addition, with the eventual goal of modeling and analyzing WM fibers, it is advantageous to process the fibers themselves, instead of DTI scalar-data. Therefore, we developed an automatic tractography segmentation pipeline tailored for neonatal brains, which allows for subsequent maturation assessment. Segmentation is atlas-based, specifically designed for premature neonates at term equivalent age (TEA).

Our automatic tractography segmentation pipeline neonate data is based on 3D distances between fibers like in O'Donnell et al. [26], and it is specifically tailored for dealing with the neonatal tractography characteristics.

2 Material and Methods

2.1 Subjects

Máxima Medical Center (MMC) provided clinical data for this study. In total, 30 patients with an MRI-scan were included. All MRIs were conducted as part of routine clinical practice. According to Dutch Law on Medical Research with Humans (WMO) a waiver for ethical assessment was provided by the local Medical Ethical Committee of MMC, considering that anonymous data collection was performed. The preterm neonates (n = 13, of whom 11 born at a gestational age less than 32 weeks) were all scanned at term equivalent age. Full-term neonates were scanned (n-17) in the second week postnatally. For each patient the relevant clinical information was available.

For preterm infants, the MRI injury was scored according to Woodward et al. [44]. The Woodward scoring system takes into account deviations or abnormalities in WM, subarachnoidal space, basal ganglia, and determines the presence of cysts ventricular dilation, and hemorrhage as described in Kooij [14] and van Pul et al. [30]. The scores were classified in normal (Woodward score for WM of 5–6), mild (Woodward score for WM of 7–9), moderate (Woodward score for WM of 10–12), and severe (Woodward score for WM of 13–15) pathology. For full-term newborns suspected from hypoxic-ischemic encephalopathy the Shankaran method [34] was used, and pathology was defined again to the groups normal (Shankaran score normal), mild (Shankaran score 1A), moderate (Shankaran score 1B) and severe (Shankaran score 2).

2.2 Data Acquisition

The data was acquired on a Philips Achieva 3.0 T MRI-scanner. The DTI-sequence was performed with b-values 0 and 800 s/mm^2 in 32 directions using a single-shot-EPI sequence, with $TE/TR = 48/7745$ ms using a SENSE factor 2, in less than 5 min scan time. The dataset consists of 50 adjacent slices, each slice with 128×128 voxels, each voxel corresponds to a size of $1.44 \times 1.41 \times 2$ mm. Correction of movement artifacts and eddy current distortions in the DWI images was performed with the Philips software from the scanner workstation. The DTI sequence was part of the regular MRI series also including T1 and T2 series.

2.3 Tractography

2.3.1 Masking

Previous to tractography, a skull-stripping method is applied in order to reduce the number of spurious fibers. For the neonatal DTI data, best results are found for a hybrid method which combines a threshold clipping approach with morphological operators.[1] The skull-stripping mask is created by processing of the $b0$ image from DTI. An empirical investigation of the optimal parameters for neonatal datasets is conducted. Optimal parameters do not result in holes in the mask, and contain no non-brain regions, like facial and neck area and regions outside the skull.

First, a histogram of the $b0$ image is computed, for which values above a lower and upper threshold are set to 0. Best results for our dataset are found for upper and lower thresholds of 85% and 99.9% of the intensity value distribution. This mask is then further processed by application of morphological operators. First, satellites are removed. We apply a connected component analysis for 26-connectivity. Components below 100 voxels are considered satellites and deleted. Second, holes are filled by using a smart closing operator. A smart closing operator is defined similar to a normal closing operator, i.e., dilation followed with erosion, having in addition the property of conserving the initial mask outside contour. It does so by comparing the after-closing mask with a complementary image of the before-closing mask [37]. The structural element comprises a 3-D 6-connectivity.

Third, the image is smoothed by a 3-D Gaussian kernel. For the tested dataset, best results were empirically found for a standard deviation, σ, of 10 voxels, which is smaller than typically used in adults because of the smaller brain size of neonatal patients. The smoothed result is again converted into a mask by assigning the value true to voxels higher than 0.5; this can be seen as a second erosion procedure. After smoothing, the mask gets its satellites removed and holes filled once more. The mask is then applied to the multiple DW images.

2.3.2 Tractography

Tractography is performed with a deterministic streamline based full brain seeding algorithm [36], implemented in the software developed at the Biomedical Image Analysis Group from the University of Eindhoven, vIST/e.[2] The tracking parameters used are specific for the patient type of neonates at term age, as suggested by van Pul et al. [29, 30]: minimum fiber length of 20 mm, C_l [43] for minimum anisotropy index of 0.12, maximum fiber angle of 10° and minimum seed distance of 0.5 voxels. The results of the tractography are tractograms, i.e., full brain fiber tracts, which are in the order of thousands of fiber tracts.

[1] http://vistalab.stanford.edu/.

[2] http://sourceforge.net/projects/viste/.

2.4 Atlas Creation

The preterm neonatal atlas is built from the tractograms of three preterm neonates without pathology imaged with DTI at TEA from a previous study (Kooij [14], van Pul [30], permission granted). These datasets were acquired using the same type of scanner and protocol as the data of this current study. The atlas represents the common fiber patterns among healthy neonatal DTI tractography results. The tractograms from two of the patients are aligned with the third patient by affine registration of the linear anisotropy, C_l maps, with the software package SPM8.[3]

After registration, all fibers from the three patients are clustered by applying an Affinity Propagation Method [5] to facilitate the manual atlas labeling, and distance metric described in Sect. 2.5.1. The parameters of the clustering are set to maximize cluster correctness. A cluster is correct if it contains fibers that belong just to one bundle, i.e., anatomical structure, however, it does not necessarily need to contain all fibers from that bundle. A bundle can contain a combination of two or more clusters.

All clusters were visually inspected by three experts (two neonatologists and an MR physicist with >10 year experience with DTI) and labeled according to anatomical structure based on the WM atlases by Mori et al. [23] and Wakana et al. [39]. In this way all fibers were divided into the following bundle labels: CC (corpus callosum), CR (corona radiata, left and right), SS (sagittal stratum, left and right), CG (cingulum), FX (fornix), MCP (middle cerebellar peduncle), cheeks, artifacts, and other. An image of the atlas is presented in Fig. 1.

2.5 Data Processing

The processing pipeline of the DTI data to segment has the following steps: Tractography; Clustered-based sampling; Registration; Labeling; and Propagation—as schematically represented in Fig. 2. The tractography set is first sampled using a

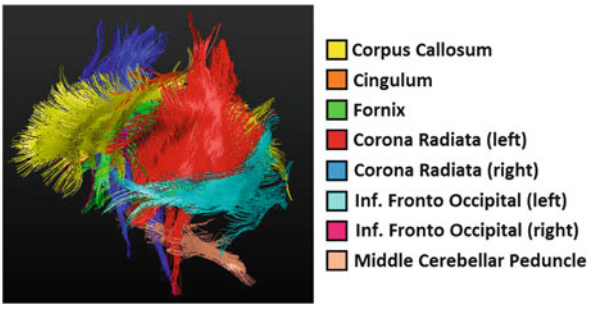


Fig. 1 Atlas of preterm neonates imaged at TEA—lateral view. Label of segmented structure is depicted by a *color-key*, at *right side* of the image

[3] http://www.fil.ion.ucl.ac.uk/spm/software/spm8/.

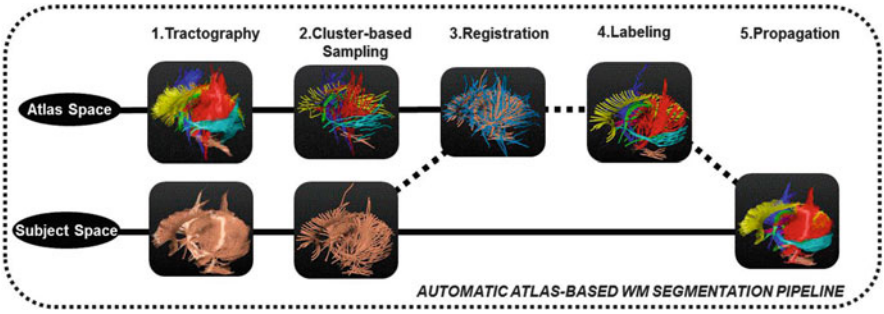


Fig. 2 Schematic representation of the processing pipeline. Columns depict the processing step and the *two lines* depict if the fibers are on subject coordinates space or atlas coordinates space

clustering algorithm, selecting the cluster fiber representatives. This first step is applied to both atlas and subject tractography sets. After, the subject sampled fiber set is registered to the atlas sampled fiber set, passing from the subject space to the atlas space. Still in the atlas space, the subject sampled fiber set is labeled. The final step of propagation is in the subject space, labeling the complete subject fiber set. Each of these steps will now be explained further. The tractography step corresponds to the method described already in Sect. 2.3.2.

2.5.1 Cluster Based Sampling

Sampling of the fibers is applied in order to add robustness and reduce computational costs in subsequent processing steps. Fibers from the same anatomical bundle can have different lengths and the amount of fibers can differ across bundles. Due to this, sampling cannot be performed randomly across structures, since we need all important bundles represented in the downsampled set. Our sampling method starts by computing similarities, i.e., within similarities, for each pair of fibers within the subject. Next, these similarities are used for clustering by Affinity Propagation [5], consequently finding the cluster fiber representatives. The sampled dataset is then downsampled to these cluster fiber representatives. A scheme of this data flow is depicted in Fig. 3.

We developed a distance measure adapted for neonatal data which is based on the Hausdorff distance. Distances in fibers are usually based on point-wise spatial distance between the reconstructed fibers points. We encountered problems with the common distance measures used for adult tractograms, e.g., average mean distance [21], due to the presence of broken reconstructed fibers in neonatal tractography. This is mainly due to the presence of low anisotropy areas. Adaptation of the tracking algorithm can mitigate the issue of broken fibers, but not without introducing assumptions, and, therefore, other issues. Ignoring short fibers, as it is commonly done for adult brains, would discard a large part of the tractogram.

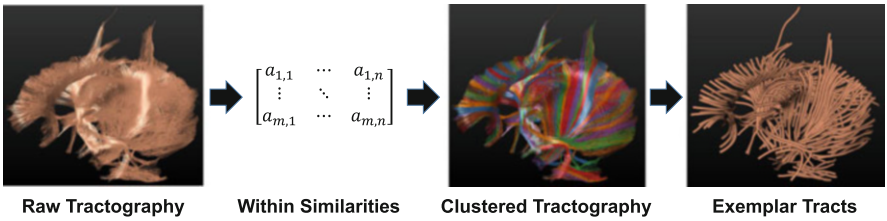


Fig. 3 Sampling data flow. Within similarities are stored in a matrix which the cluster algorithm makes use of

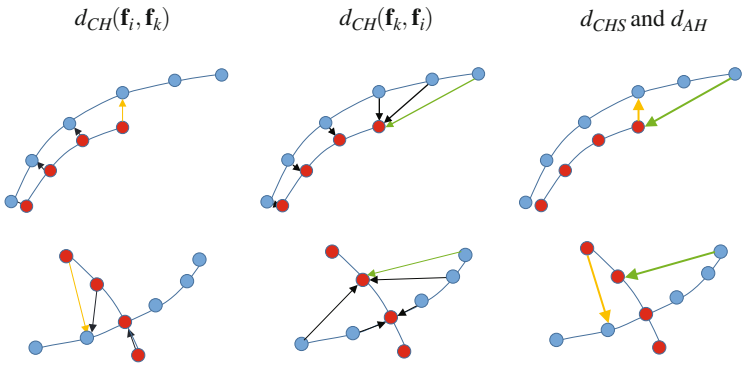


Fig. 4 Illustration of classic Hausdorff distance and our proposed adaptation. *First row* shows parallel fibers of different lengths that belong to the same bundle. *Second row* crossing fibers that belong to different bundles. First two columns shows d_{CH} from *red to blue fiber* and vice versa. Last column shows in *green* the d_{CHS} giving similar distance for parallel and crossing fiber configurations. d_{AH} is shown in *orange* with larger distance for crossing fibers than for parallel

Therefore, it is not desirable to penalize the distance between fibers due to the fiber length. On the other hand, we want to penalize crossing or diverging fibers. Most measures proposed for the adult brain [6, 15, 19, 21], e.g., average mean or Hausdorff do not fulfill our requirements. Some measures [6, 15, 19] avoid finding point to point correspondence which dominates the computational costs by mapping fiber-curve parametrizations. These measures penalize short fibers, diverging or crossing fibers equally and, therefore, these different cases cannot be discerned. Wassermann et al. [42] have a different approach where the fibers are represented implicitly. Wassermann et al. [42] distance measures looks mainly at the overlapping area, so fibers that are diverging and have a small overlap have the same distance than fibers with a full overlap but different lengths.

We developed a simple adaptation of the classic Hausdorff distance (see Fig. 4). The classic Hausdorff distance, $d_{CH}(\mathbf{f}_i, \mathbf{f}_k)$, is the maximum distance of the point-wise minimum distances between the fibers $\mathbf{f}_i$ and $\mathbf{f}_k$. In other words, it is the greatest of all the distances between each point in one fiber to its closest point in the other fiber.

This distance can be defined as:

$$d_{CH}(\mathbf{f}_i, \mathbf{f}_k) = max_{\mathbf{p}_r \in \mathbf{f}_i}(min_{\mathbf{p}_l \in \mathbf{f}_k}||\mathbf{p}_r - \mathbf{p}_l||) \tag{1}$$

where $\mathbf{p}_r$ and $\mathbf{p}_l$ stand for coordinate point belonging to the fibers $\mathbf{f}_i$ and $\mathbf{f}_k$ respectively notice that $d_{CH}(\mathbf{f}_i, \mathbf{f}_k) \neq d_{CH}(\mathbf{f}_k, \mathbf{f}_i)$. In order to make the distance symmetric the maximum is taken as shown:

$$d_{CHS}(\mathbf{f}_i, \mathbf{f}_k) = max(d_{CH}(\mathbf{f}_i, \mathbf{f}_k), d_{CH}(\mathbf{f}_k, \mathbf{f}_i)) \tag{2}$$

Our adapted distance measure simply takes the minimum instead:

$$d_{AH}(\mathbf{f}_i, \mathbf{f}_k) = min(d_{CH}(\mathbf{f}_i, \mathbf{f}_k), d_{CH}(\mathbf{f}_k, \mathbf{f}_i)) \tag{3}$$

The behaviour of d_{AH} is similar to the Hausdorff distance when fibers have similar length. On the other hand, using this adapted measure, crossing or diverging fibers of different lengths will have a relatively large distance to parallel or overlapping fibers of different lengths (see Fig. 4). This facilitates the inclusion of broken parallel fibers in the corresponding bundle, and at the same time discarding crossing or diverging fibers. Despite not being a metric and also less robust than other measures proposed in literature, we still decide to use d_{AF} since it suits our requirements.

In the next step, for the clustering we use the Affinity Propagation algorithm (AP) [5, 17]. Besides the subject fiber set, also the atlas fiber set is downsampled to improve the registration process. An image of the downsampled atlas is shown in Fig. 5 for the anatomical significant labels.

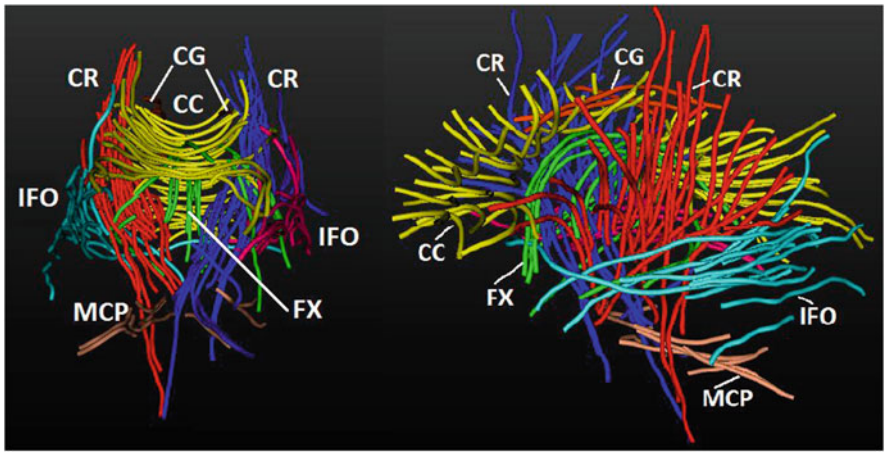


Fig. 5 Downsampled Atlas. Same *color code* as in Fig 1

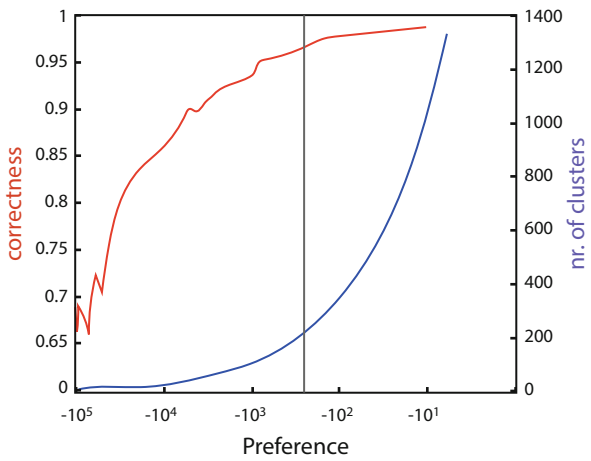


Fig. 6 Correctness study for cluster preference parameter (x-axis). Y-axis depicts correctness and number of clusters, *left* and *right* respectively

Affinity Propagation was chosen from the studied clustering methods as it allows producing clusters containing reconstructed fibers from not more than one WM anatomic structure [17]. For this clustering technique, the number of clusters is not a predefined value, but depends on previously chosen preference values. Data points with a high preference are more likely to be chosen by the clustering algorithm as a cluster center, than data points with a low preference. Because there is no a priori knowledge about which fibers are more appropriate to become cluster centers and constitute the down-sampled dataset, preference is made equal for all fibers. In order to optimize the preference value, i.e., produce cluster containing fibers from no more than one anatomical structure of interest, a correctness test is conducted (Fig. 6), for the atlas fibers, for which the ground truth of the anatomic labels is known. A balance was found between cluster correctness and the number of clusters, resulting in a choice for the preference value of -200, as it produces slightly more than 200 clusters with a correctness of 95%. After clustering, the cluster fiber representatives form the downsampled tractogram. For comparison of the subject tractogram with the atlas, also a downsampled atlas tractogram was made, shown in Fig. 5 with the anatomical labels.

2.5.2 Registration

Registration involves applying a transformation model to the subject fiber points, in order to correct for head position inside the scanner and variation of head size and shape between subjects and the atlas. Our transformation model is based on an affine transformation [20, 26, 46]. The optimal affine transformation is found in the fiber domain, i.e., fiber-wise, based on 3D fiber distances minimization.

In order to accelerate convergence, a first initialization is achieved by aligning the center of mass of the downsampled subject's tractogram and the center of mass of the atlas. After the initialization, the used registration algorithm is based on the work from O'Donnell et al. [26] with adult tractograms. O'Donnell also developed a Hausdorff inspired distance via a probability density distribution. This probability density function describes how probable it is for that subject fiber to have close-by neighbor atlas fibers. The equation is as follows:

$$\delta(\mathbf{f}_i, \mathbf{f}_k) = \frac{e^{-\left(\frac{d(\mathbf{f}_i, \mathbf{f}_k)}{\sigma}\right)^2}}{Z} \tag{4}$$

where σ is defined as a radius of interest where distances outside this radius will have similarities close to 0. Z is a normalization constant, and therefore will not influence the optimization procedure. Similarities of the subject fiber to all the atlas fibers are then combined in one similarity probability density function [26]. In order to maximize this similarity probability its entropy is minimized [35], using a cost function. Last iteration cost-function value can be further used for quantitative evaluation of segmentation results. The optimization algorithm applied is a direction set method named COBYLA, which stands for Constrained Optimization by Linear Approximation [28]. For our neonatal data, the best results are empirically found for the value of σ equal to 3 mm. This is smaller than in adults due to the smaller size of the neonatal brain.

In our application of the O'Donnell algorithm, best registration results are obtained while optimizing iteratively between four distinct deformation-types: translation, rotation, scaling and shearing. For each of these deformation types, a transformation is probed in a domain of three degrees of freedom (DoF), i.e., one for each spatial coordinate. Due to the difficulty of registering the cluster center fibers corresponding to the CG (cingulum) anatomy, translation and rotation are performed once more.

2.5.3 Labeling

After registration, computation of similarity between every registered cluster fiber representative and every cluster atlas fiber representative is calculated. The adapted Hausdorff distance presented in Sect. 2.5.1 is used again for the similarity metric. Each cluster center registered fiber receives the label of the atlas cluster center fiber towards which it has the smallest distance.

2.5.4 Label Propagation

The label of the cluster fiber representative is propagated to all the fibers that belong to the cluster it is representing. It is therefore possible to return to the

full tractography dataset, without any deformation performed to the fibers. Volume and averaged values of anisotropies can be calculated per segmented structure automatically using vIST/e.[4] Averaged anisotropy values correspond to a weighted average of the anisotropy of all voxels in that segmented structure: the anisotropy value of a voxel is included as many times as the number of fibers passing through it.

3 Evaluation

Of 30 available neonatal datasets, six contained large MRI artifacts which highly disrupted the DTI anatomical structure information and failed in the first step of the pipeline. For the remaining 24 datasets, the complete pipeline could be executed. The processed data was from 13 term borns and 11 preterms at term equivalent age, of whom eight patients were classified as having no abnormalities, seven had mild abnormalities, eight moderate abnormalities and one with severe pathology. On average, the processing took 2–8 h to produce an automatically labeled data. This time varies with the number of fibers in the subject tractograms which is in the order of thousands. The main computational costs are related to the calculation of the within similarities measurement for the cluster-based sampling which has not been optimized.

A qualitative evaluation of the segmented tractography was performed by two MRI experts. For each segmented structure, the number of incorrectly labeled fibers is evaluated, based on visual inspection. The ratio between incorrectly labeled fibers and the total number of fibers per segmented structure is further referred to as the percentage error. For minor structures, FX, CG, MCP, IFO, segmentation errors are considered less severe than segmentation errors at major structures in the neonatal brain: the CRs and CC. The segmentation performance is divided in four classes:

4 *Good*: All segmented structures have less than 10% of error;
3 *Sufficient*: Major structures until 10% of error, but minor structures until 50% of error;
2 *Moderate*: Major structures with error between 10% and 30%, minor structures until 50% of error;
1 *Bad*: All the structures, major and minor, with more than 30% of error.

Of the 24 analyzed datasets, the segmentation results were scored in these four performance classes: three were classified as *Bad* segmentation, four as *Moderate* segmentation, seven as *Sufficient* segmentation, and ten as *Good* segmentation. An example of a segmentation for each segmentation class, is shown in Fig. 7. The main structures in neonatal datasets (corpus callosum and corona radiate) are easily recognized in the patient with classification *Good*. For the *Moderate* and *Bad* examples, the automatic fiber clustering and segmentation contains visible errors.

[4] http://sourceforge.net/projects/viste/.

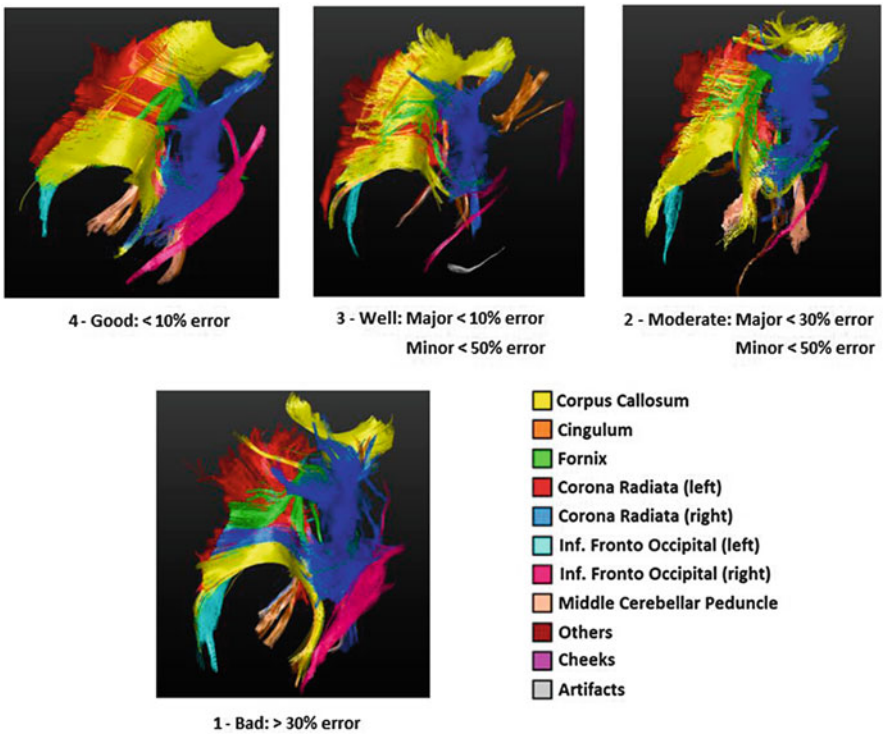


Fig. 7 Tractography segmentation examples for each quality class, posterior view. From *left to right* and *top to bottom*: *Good* segmentation performance example; *Sufficient* segmentation performance example, *Moderate* segmentation performance example; *Bad* segmentation performance example; the color of the structure and its anatomical label are displayed in the legend

Table 1 Distribution of segmentation performance classes and pathology scores

	Segmentation				
Pathology	1-Bad	2-Moderate	3-Sufficient	4-Good	
No	1	2	3	2	8
Mild	1	2	1	3	7
Moderate	1	0	3	4	8
Severe	0	0	0	1	1
Total	3	4	7	10	24

As shown in Table 1, there is no correlation between segmentation performance and presence of pathology. We can observe that patients in whom segmentation perform badly (score 1), the pathology was not severe. Patients with no abnormalities and mild abnormalities are distributed among all four classes of segmentation performance. Patients with no abnormalities are in greater number in the class *Sufficient*, and patients with mild abnormalities in the class *Good*. Patients with

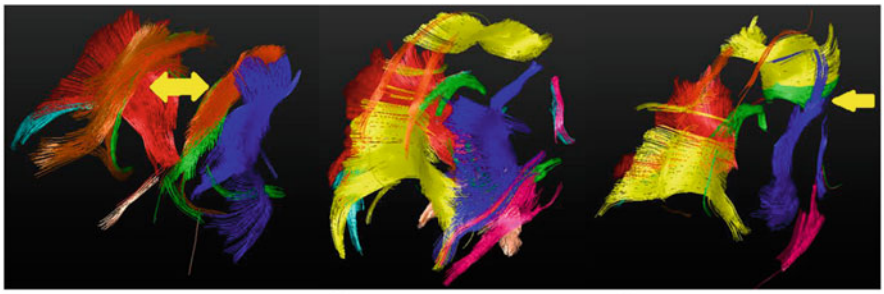


Fig. 8 Segmentation results for patients with severe WM abnormalities, posterior views. From *left to right*: (**a**) patient with CC agenesis with no further WM pathology, classified with *Good* segmentation performance; (**b**) patient with large WM abnormalities (signal intensity) and widened ventricles classified with *Good* segmentation performance; (**c**) patient with severe WM signal abnormalities classified with *Bad* segmentation performance, anterior part of the CC is segmented as FX

moderate abnormalities are also in greater number in the class *Good*. The patient with severe abnormality also had segmentation performance classified as *Good*.

Figure 8 presents the segmentation results for three patients with severe WM pathology. The first patient had no corpus callosum (CC), this is called CC-agenesis. The segmentation pipeline still processed well the tractogram and no CC was segmented for this patient but all the other structures could be observed. An arrow was inserted in the typical CC region in the image Fig. 8, for ease of interpretation. This patient had large areas of diffuse white matter signal intensity abnormalities but could still be segmented with our pipeline and was classified as having a *Good* segmentation performance. The third patient, also with white matter signal abnormalities, shows poor segmentation performance and the segmentation performance was scored as *Bad*, since the CC (a main structure) is clearly showing mislabeled fibers. For this last patient the frontal part of the CC, in yellow, is also erroneously labeled as FX, in green (also pointed out by an arrow).

Looking in more detail to the registration process, it shows that the cost-function value (CF value) obtained at the end of the registration process has a lower average value in the group with *Good* performance. A plot of the distribution of last cost-function value per segmentation performance class is presented in Fig. 9. The last cost value gives an indication of the quality of the registration, the lower the value the better the registration was matched.

Patients with a last CF-value bigger than 35 were individually assessed; this corresponded to three patients. These three patients had no severe white matter pathology: one had even no abnormalities, the other two had mild abnormalities due to residuals from a hemorrhage. Therefore, it can be observed that the deviation from a normal brain was not the reason for the lower quality results.

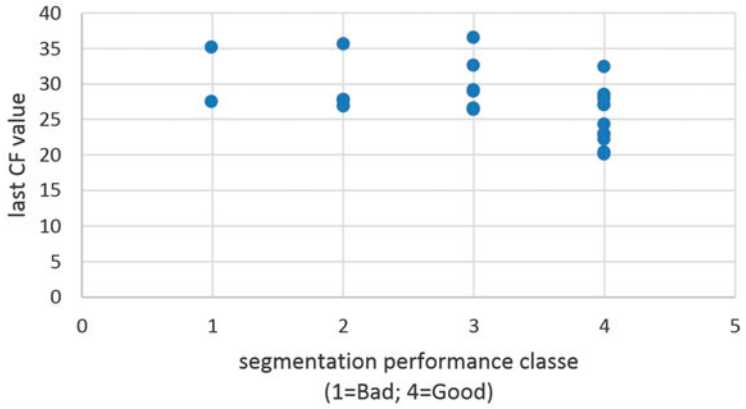


Fig. 9 Distribution of last CF value per segmentation performance class

For the complete analyzed dataset, only 30% of the patient tractography results contained spurious fibers, i.e., fibers that did not belong to any anatomical structure and only arise due to scanning or processing errors. Without skull-stripping, all tractography results contained spurious fibers. Existence of spurious fibers is almost evenly present for all the four performance classes, so segmentation performance is not dependent on this step.

4 Discussion

We have implemented an automatic pipeline that segments Diffusion Tensor Images of neonates automatically into images displaying the main WM structures in the neonatal brain. The method is automatic, without user interaction, and tailored for the evaluation of the neonatal brain. Main pipeline embedded methods and their relation to the segmentation results are here discussed: (1) the tensor fitting and tractography algorithm, (2) the tractography neonatal TEA atlas, (3) the skull-stripping method, (4) the cluster-based sampling method, and (5) the fiber-wise registration and correspondent fiber similarity metrics.

1. The tractography algorithm used is a simple FACT based method [36]. Subjects with poor segmentation performance often show partial volume effects in the clustering results (e.g., crossing over of fibers from one structure to another). Partial volume effects have not been addressed by our pipeline. Crossing over of fibers from different structures occurred usually between the CC and FX, or between CC and CG. For some patients a partial volume effect is also observed between CC and the CR. Improvement of tensor fitting accuracy is constrained by practical reasons in neonatal clinical practice. The DTI protocol on many clinical scanners is still limited to a maximum of 32 diffusion directions. If additional

packages are purchased scanning in more directions become possible, however, that requires longer scan time, making it not easy to apply in the neonatal population in a routine clinical setting. Improvement of the tractography algorithm itself can be used to mitigate but not solve some of the previous referred problems. Nevertheless, such improvements also involves making assumptions that might not suit our other constraints of our problem. An interesting direction would be to explore approaches like the one presented by Durrleman et al. [4] or Wang et al. [41] where segments of fibers are considered rather than the full length fibers. This will, however, increase the complexity of the pipeline given the considerable different nature of the proposed approaches.

2. Regarding the used atlas, it is built based on three datasets of neonates. Though, we consider these datasets as quite representative of neonatal TEA DTI data, using more data sets for the atlas construction might improve the accuracy of the results. In addition, it is important to note that although the atlas contains classes of spurious fibers, i.e., fibers that did not belong to any anatomical structure, these can only arise due to scanning or processing errors. Future atlas versions should consider removal of spurious fibers categorization and address them in the segmentation pipeline only with a good skull-stripping method or other outlier detection.
3. The applied skull-stripping reduces spurious fibers to 70%, therefore, reducing calculation of similarities for fibers without anatomical interest. However, there is empirical parameter tuning needed that should be addressed in the future.
4. The cluster-based sampling is still quite a computationally costly, as the cluster method in itself computes similarity metrics. Still, performing registration and label attribution using as input a sampled set was found to improve the accuracy of these pipeline steps, in contrast with using the complete tractogram for representing the atlas and the patient fibers. Computational costs were not the focus of this chapter, and we believe that computation costs can still be further reduced by using sparse distance matrices calculations, e.g., space subdivisions strategies like octrees.
5. For all factors probed to investigate what is affecting segmentation performance, only the registration accuracy seemed to matter. In particular a peculiar curved shape of the CC often lead to an excessive shrinking of the fibers in the registration step. Excessive shrinking might be caused due to an ineffective tuning of the registration constraint parameters, and/or due to the fact this kind of CC shape is not represented in the atlas, and/or even due to an inadequate similarity measure for the registration between atlas and subject fibers. Possible ways to tackle this can include applying a different deformation model which allows for local transformation, like elastic and fluid deformations—with the need of then defining optimal constraints for these models. Another direction would be to tune the similarity metric being used for registration specifically for dealing with neonatal data, like the one being used for the cluster-based sampling. It would be interesting to investigate other registration approaches in this context, for example, the registration approach proposed by Garyfallidis et al. [7].

As an end note, it is also important to refer that FX segmentation results still do not allow for an accurate investigation of this structure anisotropy. There are still too many erroneous fibers present. Guevara et al. [10] and O'Donnell et al. [25] also found for their methods that the association type fibers, FX and CG for the used atlas labels, were the ones more difficult to segment.

5 Conclusions

Our study shows that is feasible to automatically segment WM structures in the neonatal brain, by using an atlas-based and fiber-wise processing of DTI data. We observe that our fiber-wise method, i.e., making use of global information, allows for the radiologist and neonatologist to have a better overview of the patient's WM anatomical structure and eventual abnormalities. An automatic segmentation also means less user-dependency and a less time-consuming analysis, thus allowing to study WM maturation in an easier and more objective way. Segmentation performance showed not to be influenced by presence of WM pathology in subjects, even when anatomical structures were missing due to severe WM deviations. Indicating that is a good segmentation approach to be used when pathology is present.

In order to improve further registration quality, we believe that future work should aim to increase atlas variability but probably even more important to probe similarity metrics between fibers that are less computational expensive but still adequate for neonatal tractography. In addition, it might be of interest to extend the atlas for representing all gestational ages, for allowing study of full-term neonates at risk of neurodevelopmental disorders.

Acknowledgements We are grateful to Floris Groenendaal, Linda de Vries, and Manon Benders, that we are granted the re-use of the fiber atlas that part of our group developed in a previous study together with the Department of Neonatology, Wilhelmina Children's Hospital/University Medical Center Utrecht. In addition, we thank Lauren O'Donnell for providing us with their fiber registration software.

References

1. Anjari, M., Srinivasan, L., Allsop, J., Hajnal, J., Rutherford, M., Edwards, A., Counsell, S.: Diffusion tensor imaging with tract-based spatial statistics reveals local white matter abnormalities in preterm infants. NeuroImage **35**(3), 1021–1027 (2007)
2. Basser, P.J., Mattiello, J., LeBihan, D.: MR diffusion tensor spectroscopy and imaging. Biophys. J. **66**(1), 254–267 (1994)
3. Dubois, J., Dehaene-Lambertz, G., Perrin, M., Mangin, J., Cointepas, Y., Duchesnay, E., Le Bihan, D., Hertz-Pannier, L.: Asynchrony of the early maturation of white matter bundles in healthy infants: quantitative landmarks revealed noninvasively by diffusion tensor imaging. Hum. Brain Mapp. **29**, 14–27 (2008)

4. Durrleman, S., Pennec, X., Trouvé, A., Ayache, N.: Statistical models of sets of curves and surfaces based on currents. Med. Image Anal. **13**(5), 793–808 (2009)
5. Frey, B., Dueck, D.: Clustering by passing messages between data points. Science **315**, 972 (2007)
6. Garyfallidis, E., Brett, M., Correia, M.M., Williams, G.B., Nimmo-Smith, I.: Quickbundles, a method for tractography simplification. Front. Neurosci. **6**, 175 (2012)
7. Garyfallidis, E., Ocegueda, O., Wassermann, D., Descoteaux, M.: Robust and efficient linear registration of white-matter fascicles in the space of streamlines. NeuroImage **117**, 124–140 (2015)
8. Geng, X., Gouttard, S., Sharma, A., Gu, H., Styner, M., Lin, W., Gerig, G., Gilmore, J.H.: Quantitative tract-based white matter development from birth to age 2 years. NeuroImage **61**(3), 524–557 (2012)
9. Glass, H., Bonifacio, S., Shimotake, T., Ferriero, D.: Neurocritical care for neonates. Pediatr. Neurol. - Curr. Treat. Opt. Neurol. **13**, 574–589 (2011)
10. Guevara, P., Duclap, D., Poupon, C., Marrakchi-Kacem, L., Fillard, P., Le Bihan, D., Mangin, J.: Automatic fiber bundle segmentation in massive tractography datasets using a multi-subject bundle atlas. NeuroImage **61**, 1083–1099 (2012)
11. Hua, K., Zhang, J., Wakana, S., Jiang, H., Li, X., Reich, D., Calabresi, P., Pekar, J., van Zijl, P., Mori, S.: Tract probability maps in stereotaxic spaces: analyses of white matter anatomy and tract-specific quantification. NeuroImage **39**, 336–347 (2008)
12. Huppi, P., Dubois, J.: Diffusion tensor imaging of brain development. In: Seminars in Fetal & Neonatal Medicine, vol. 11, pp. 489–497 (2006)
13. Jong, M., Verhoeven, M., Baar, A.: School outcome, cognitive functioning, and behavior problems in moderate and late preterm children and adults: a review. In: *Seminars in Fetal & Neonatal Medicine*, vol. 17, pp. 163–169 (2012)
14. Kooij, B.: MRI analysis and neurodevelopmental outcome in preterm infants. PhD thesis, Utrecht University (2011)
15. Labra, N., Guevara, P., Duclap, D., Houenou, J., Poupon, C., Mangin, J.F., Figueroa, M.: Fast automatic segmentation of white matter streamlines based on a multi-subject bundle atlas. Neuroinformatics **15**(1), 71–86 (2017). doi: 10.1007/s12021-016-9316-7
16. Latal, B.: Prediction of neurodevelopmental outcome after preterm birth. Ped. Neuro **40**, 413–419 (2009)
17. Leemans, A., Jones, D.: A new approach to fully automated fiber tract clustering using affinity propagation. In: *ISMRM*, p. 17 (2009)
18. Leemans, A., Sijbers, J., De Backer, S., Vandervliet, E., Parizel, P.: Multiscale white matter fiber tract coregistration: a new feature-based approach to align diffusion tensor data. Magn. Reson. Med. **55**, 1414–1423 (2006)
19. Maddah, M., Grimson, W.E.L., Warfield, S.K., Wells, W.M.: A unified framework for clustering and quantitative analysis of white matter fiber tracts. Med. Image Anal. **12**(2), 191–202 (2008)
20. Mayer, A., Zimmerman-Moreno, G., Shadmi, R., Batikoff, A., Greenspan, H.: A supervised framework for the registration and segmentation of white matter fiber tracts. IEEE Trans. Med. Imag. **30**, 1 (2011)
21. Moberts, B., Vilanova, A., van Wijk, J.J.: Evaluation of fiber clustering methods for diffusion tensor imaging. In: IEEE Visualization, pp. 65–72 (2005)
22. Mori, S., Zijl, P.: Fiber tracking: principles and strategies - a technical review. NMR Biomed. **15**, 468–480 (2002)
23. Mori, S., Wakana, S., Nagae-Poetscher, L., Zijl, P.: MRI Atlas of Human White Matter. Elsevier, Amsterdam (2005)
24. Neil, J., Miller, J., Mukherjee, P., Hüppi, P.S.: Diffusion tensor imaging of normal and injured developing human brain - a technical review. NMR Biomed. **15**, 543–552 (2002)
25. O'Donnell, L., Westin, C.: Automatic tractography segmentation using a high-dimensional white matter atlas. IEEE Trans. Med. Imag. **26**(11), 1562–1575 (2007)

26. O'Donnell, L., Wells, W., Golby, A., Westin, C.: Unbiased groupwise registration of white matter tractography. In: MICCAI, vol. III, pp. 123–130 (2012)
27. Olivetti, E., Nguyen, T.B., Garyfallidis, E.: The approximation of the dissimilarity projection. In: 2012 International Workshop on Pattern Recognition in NeuroImaging (PRNI), pp. 85–88. IEEE, New York (2012)
28. Powell, M.J.D.: Direct search algorithms for optimization calculations. Acta Numer. **7**, 287–336 (1998)
29. Pul, C., Buijs, J., Vilanova, A., Roos, G., Wijn, P.: Infants with perinatal hypoxic ischemia: feasibility of fiber tracking at birth and 3 months. Radiology **240**, 203–214 (2006)
30. Pul, C., Kooij, B., Vries, L., Benders, M., Vilanova, A., Groenendaal, F.: Quantitative fiber tracking in the corpus callosum and internal capsule reveals microstructural abnormalities in preterm infants at term-equivalent age. AJNR Am. J. Neuroradiol. **33**, 678–684 (2012)
31. Ratnarajah, N., Qiu, A.: Multi-label segmentation of white matter structures: application to neonatal brains. NeuroImage **102**, 913–922 (2014)
32. Rose, J., Vassar, R., Cahill-Rowley, K., Guzman, X., Stevenson, D., Barnea-Goraly, N.: Brain microstructural development at near-term age in very-low-birth-weight preterm infants: an atlas-based diffusion imaging study. NeuroImage **86**, 244–256 (2014)
33. Saigal, S., Doyle, L.: An overview of mortality and sequelae of preterm birth from infancy to adulthood. Lancet **371**(9608), 261–269 (2008)
34. Shankaran, S., Barnes, P., Hintz, S., Laptook, A., Zaterka-Baxter, K., McDonald, S., Ehrenkranz, R., Walsh, M., Tyson, J., Donovan, E., Goldberg, R., Bara, R., Das, A., Finer, N., Sanchez, P., Poindexter, B., Van Meurs, K., Carlo, W., Stoll, B., Duara, S., Guillet, R., Higgins, R.: Eunice kennedy shriver national institute of child health and human development neonatal research network.brain injury following trial of hypothermia for neonatal hypoxic-ischaemic encephalopathy. Arch. Dis. Child Fetal Neonatal Ed. **97**(6), F398–404 (2012)
35. Shannon, C.: A mathematical theory of communication. Bell Syst. Tech. J. **27**(3), 379–423 (1948)
36. Vilanova, A., Berenschot, G., van Pul, C.: DTI visualization with streamsurfaces and evenly-spaced volume seeding. In: Joint Eurographics - IEEE TCVG Symposium on Visualization, pp. 173–182 (2004)
37. Vincent, L.: Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. IEEE Trans. Image Process. **2**(2), 176–201 (1993)
38. Volpe, J.: Brain injury in premature infants: a complex amalgam of destructive and developmental disturbances. Lancet Neurol. **8**, 110–124 (2009)
39. Wakana, S., Jiang, H., Nagae-Poetscher, L., van Zijl, P., Mori, S.: Fiber tract-based atlas of human white matter anatomy. Radiology **230**, 77–87 (2004)
40. Wang, R., Benner, T., Wedeen, V.: Diffusion toolkit: a software package for diffusion imaging data processing and tractograph. In: ISMRM, vol. 15, p. 3720 (2007)
41. Wang, X., Grimson, W.E.L., Westin, C.-F.: Tractography segmentation using a hierarchical dirichlet processes mixture model. NeuroImage **54**(1), 290–302 (2011)
42. Wassermann, D., Bloyb, L., Kanterakisb, E., Vermab, R., Derichea, R.: Unsupervised white matter fiber clustering and tract probability map generation: applications of a gaussian process framework for white matter fibers. NeuroImage **15**(1), 228–241 (2010)
43. Westin, C.-F., Maier, S., Khidhir, B., Everett, P., Jolesz, F., Kikinis, R.: Image processing for diffusion tensor magnetic resonance imaging. In: MICCAI. Lecture Notes in Computer Science, vol. 1679, pp. 441–452. Springer, New York (1999)
44. Woodward, L., Anderson, P., Austin, N., Howard, K., Inder, T.: Neonatal mri to predict neurodevelopmental outcomes in preterm infants. New Engl. J. Med. **355**(7), 685–694 (2006)
45. Zhai, G., Lin, W., Wilber, K., Gerig, G., Gilmore, J.: Comparisons of regional white matter diffusion in healthy neonates and adults performed with a 3.0-t head-only mr imaging unit. Radiology **229**, 673–681 (2003)
46. Zvitia, O., Mayer, A., Shadmi, R., Miron, S., Greenspan, H.: Co-registration of white matter tractographies by adaptive-mean-shift and Gaussian mixture modeling. IEEE Trans. Med. Imag. **29**(1), 132–145 (2010)

Part V
Machine Learning Approaches

A Deep Learning Approach to Identifying Shock Locations in Turbulent Combustion Tensor Fields

Mathew Monfort, Timothy Luciani, Jonathan Komperda, Brian Ziebart, Farzad Mashayek, and G. Elisabeta Marai

Abstract We introduce a deep learning approach for the identification of shock locations in large scale tensor field datasets. Such datasets are typically generated by turbulent combustion simulations. In this proof of concept approach, we use deep learning to learn mappings from strain tensors to Schlieren images which serve as labels. The use of neural networks allows for the Schlieren values to be approximated more efficiently than calculating the values from the density gradient. In addition, we show that this approach can be used to predict the Schlieren values for both two-dimensional and three-dimensional tensor fields, potentially allowing for anomaly detection in tensor flows. Results on two shock example datasets show that this approach can assist in the extraction of features from reacting flow tensor fields.

1 Introduction

The design of efficient combustion systems requires an in-depth understanding of the underlying physics that occur within combustion chambers. The challenge is significantly escalated for supersonic combustion due to the presence of shocks and other waves. Considering that turbulent flows are inherently three-dimensional and shocks occur over very thin regions of the order of one mean free path, the presence of shocks introduces additional complexities to a flow that is characterized by a multitude of scales in time and space. Computational methods typically use artificial diffusion to smear the shocks such that they can be captured on a grid that is more coarse than the thickness of the shock [3]. Despite this approximation, the simulation of supersonic turbulent combustion is not only computationally demanding but also able to produce large quantities of data.

M. Monfort • T. Luciani • J. Komperda • B. Ziebart • F. Mashayek • G.E. Marai (✉)
University of Illinois at Chicago, Chicago, IL 60607, USA
e-mail: mathewmonfort@gmail.com; tlucia2@uic.edu; jonk@uic.edu; bziebart@uic.edu; mashayek@uic.edu; gmarai@uic.edu

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_16

As discussed in detail in our previous work [35], an important aspect of turbulence modeling and model validation involves analysis of the subgrid scale stress tensor or the stress tensor. Unfortunately, the scale of these datasets is very large; it is necessary to capture all the scales of turbulence, combustion, as well as the discontinuities. For example, Reynolds Averaged Navier-Stokes (RANS) simulations of the scramjet engine have used in excess of 33 million cells [15, 16, 46]. Meanwhile, Large eddy simulation (LES) of a simplified scramjet engine model have required upwards of 6 million cells [6], and preliminary LES simulations of a full scramjet at low Reynolds numbers have used 14 million cells [8]. It is expected that high-fidelity LES simulations of the full scramjet geometry at high Reynolds numbers will exceed 100 million cells. These datasets cannot be output frequently for visualization because of the computational cost of writing the files. When one considers that the computational cost of such a simulation can easily exceed 17,000 core-hours, it becomes necessary to limit disk-intensive operations to preserve core-hours [28].

Feature extraction can be a useful approach to reducing the size of the dataset to visualize. In this approach, a small subset of points are identified as features of interest and output for visualization. While multiple filtering techniques exist for flow shock feature extraction, some of these methods incorrectly identify regions of turbulence as shocks, and conversely some turbulence models incorrectly identify shocks as turbulence [17]. Most importantly, filtering techniques generally do not have the ability to identify abnormalities in the flow in the absence of expert knowledge input. As a result, it is necessary to investigate more flexible approaches to analyzing discontinuities in the flow that may correspond to shockwaves, including approaches which can identify normal shock behavior and abnormalities. Such anomaly detection tasks tend to require a statistical approach.

In this work we investigate the potential and limitations of deep learning [5, 29]—a machine learning technique based on learning representations of the data—with respect to shock feature extraction from strain tensor values. Deep learning has been successfully used in a variety of applications, including image analysis and speech recognition, and, beyond feature extraction, has the potential to capture abnormalities in the data. Here we take an exploratory first step in this direction by investigating the feasibility of using deep learning to retrieve shock locations. Future work will investigate the ability of deep learning with respect to anomaly detection.

2 Background and Related Work

2.1 Stress and Strain Tensors

A tensor is an extension of the concept of a scalar and a vector to higher orders. Scalars and vectors are 0-th and 1-st order tensors, respectively. In general, a k-th order tensor can be represented by a k-dimensional array, e.g. a second order tensor

is a two-dimensional array (a matrix). For example, while a stress *vector* is the force acting on a given unit surface, a stress *tensor* is defined as the components of stress vectors acting on each coordinate surface; thus stress can be described by a symmetric 2-nd order tensor.

The velocity stress and strain tensor fields are manifested in the transport of fluid momentum, which is a vector quantity governed by the following conservation equation:

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}, \qquad \text{for } i = 1, 2, 3 \tag{1}$$

where the Cartesian index notation is employed in which the index $i = 1, 2, 3$ represents spatial directions along the x, y, and z Cartesian coordinates, respectively; and the repeated index j implies summation over the coordinates. t is time, ρ is the fluid density, $\mathbf{u} \equiv [u_1, u_2, u_3]$ is the Eulerian fluid velocity, p is the pressure, and $\boldsymbol{\tau}$ is the stress tensor defined as:

$$\tau_{ij} = 2\mu \left(S_{ij} - \frac{1}{3}\delta_{ij} \frac{\partial u_k}{\partial x_k} \right) \tag{2}$$

where μ is the dynamic viscosity coefficient (a fluid-dependent parameter) and S is the velocity strain tensor defined as:

$$S_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{3}$$

As indicated by the definitions above, both the stress and strain tensor are two-dimensional, symmetric, positively-defined arrays. Density, along with the three velocity components and total energy, are the primary variables calculated in the code. All other information, including the stress tensor and velocity strain tensor, are secondary variables calculated from these primary variables.

2.2 Shock Feature Extraction

Most feature extraction techniques fall into one of three basic categories. The most widespread method uses feature attributes such as mass, centroid, volume, texture, or moment of inertia [10, 50–52]. A number of filtering techniques specifically for detecting and visualizing shocks waves have been developed, including using shock surface alignment to the pressure gradient vector [32], and using the density gradient in the direction of the velocity [33, 44]. A second approach to feature extraction uses isosurfacing in higher dimensions [24]. A third class of approaches uses various machine learning techniques to aid in feature tracking. Tzeng and Ma [58] utilize neural networks to learn which transfer functions are most appropriate in tracking

the features of interest. Ozer et al. [43] use a clustering algorithm to group features based on similarity measures. In our previous work [35], we introduced a large scale K-Means clustering approach to define and track regions of interest. Our approach is similar to these last category approaches in that we also utilize machine learning.

The approach we use to generate labels for the tensor data is based on the Schlieren filter [34]. The density-gradient of combustion datasets relates indirectly to the stress tensor through the conservation equation [34]. Such density-gradient descriptors can be used to generate flow visualizations in the style of Schlieren images [19], and have been shown to accurately reflect shock boundaries [34].

The use of the Schlieren computation is intended as an exploratory first step in investigating the ability of deep learning to identify shockwaves and other features in CFD datasets. While the computation of the Schlieren itself is not costly enough to justify a learning alternative, more accurate shock prediction methods are significantly more costly and difficult to pose numerically. The ultimate goal of the future work would be to employ a deep learning approach to directly pinpoint and differentiate different phenomena at a lower computational cost than the many sensors currently available in literature.

2.3 Deep Learning

In 1943 McCulloch and Pitts [38] introduced a set of simplified computational models of biological networks. These ideas were soon extended to include models of how these networks might learn including Hebbian learning [20], multilayer perceptrons [42] and eventually backpropagation [49]. However, the extensive computational complexity of training large networks [40] prohibitively limited their usefulness.

Deep architectures yield a greater expressive power than shallow networks since functions that can be compactly represented by an architecture of depth k, would require an exponential number of computational elements to be represented by an architecture with a depth of $k - 1$ [5]. This breadth for depth trade-off allows deep architectures to represent a wide family of functions with reduced complexity and improved generalization. Each succeeding layer of the network combines the features of the previous layer forming a higher level abstraction of the features in the preceding layer. This increasing level of abstraction from layer to layer allows deep networks to produce strong generalizations for highly varying functions. The difficulty lies in efficiently training the large number of parameters needed to form the network [40].

Convolutional networks are specific types of deep architectures inspired by the structure of the visual cortex [29, 36] which do not suffer from the typical convergence issues of other deep architectures [5]. The defining characteristic of convolutional networks is the use of local receptive fields with shared parameters. These fields are used to scan input features with a two dimensional structure and form feature maps that capture low-level (e.g., edges and corners) representations

of the input. This process is then repeated in each succeeding layer allowing for the formation of progressively higher-level abstractions. This two-dimensional representational power of convolutional networks has allowed them to dominate the field of computer vision [27, 55].

Volumetric convolutional networks [37, 39, 47] are an extension of convolutional surface-based architectures to input features with three dimensional structure. They have successfully been applied to the area of object recognition [37] and MRI segmentation [39]. The ability of these architectures to take advantage of three dimensional structure directly translates to the problem of inferring Schlieren features in combustible fluids.

2.4 Application Domain Background

Supersonic combustors, such as scramjet engines, are prime examples of flows where turbulence and combustion interact with shock waves [15, 46, 48]. However, the concurrent presence of shock waves and turbulence in the simulation of supersonic turbulent flow presents additional challenges compared to subsonic flows; the numerical methods designed to treat these properties must predict the presence and capture these features accurately [17]. The inability to accurately predict these features contributes to the many unresolved fundamental issues that surround supersonic combustors, such as the scramjet. Flames in the presence of shocks are known to become distorted, generate vorticity, and break up or stretch [25, 26]. Additionally, when turbulence is seen in the presence of a shock, there is an amplification of velocity fluctuations [8]. The inability to accurately predict these behaviors affects the remainder of the solution domain, thereby causing a failure to compare well against experiment.

While numerical simulations can provide an acceptable prediction of turbulence behavior, either through the use of large eddy simulation or direct numerical simulation, there is still a significant challenge in the simulation of flows involving flow discontinuities, such as shocks. Methods for numerically modeling supersonic turbulent combustion[4, 23] rely on the accurate prediction of the shock location. Current methods for shock capturing may rely on an artificial viscosity to dissipate the shock, to smear it across many solution points so that it may be captured [1–3]. However, incorrectly predicting the location of the shock may have unintended side effects, such as adding the artificial viscosity in the incorrect regions, thereby dissipating other regions of the flow. Research has been performed in developing sensors to accurately predict the shock location [3, 14, 17]. However, some of these methods incorrectly identify regions of turbulence as shocks, and conversely some turbulence models incorrectly identify shocks as turbulence [17]. As a result, it is necessary to investigate more flexible approaches to analyzing discontinuities in the flow that may correspond to shockwaves, including approaches which can identify normal shock behavior and abnormalities.

3 Methods

To study the applicability of deep learning to the feature identification problem, we trained convolutional neural networks to learn a mapping from strain tensors to Schlieren values [34] for each time-step in a turbulent flow. To accomplish this we form a regression network similar to an auto-encoder [9, 21] where instead of learning to replicate the strain tensors used as input, we learn to construct the associated Schlieren values for each time step. The strain tensor is calculated with Eq. (3) above [34]. Additionally, the Schlieren value for each pixel is derived with the following equation:

$$\text{Schlieren}(x, y, z) = \beta e^{-\frac{k|\nabla\rho|}{|\nabla\rho|_{\max}}}, \tag{4}$$

where x, y, and z are the position coordinates, β and k are rendering parameters set to 0.8 and 20 respectively, and Δp is the gradient of the density field.

We will examine this approach on two datasets, a three-dimensional Sod dataset and a two-dimensional blast dataset. The differing spatial dimensions of both of these domains requires the formation of two different network architectures. In this section we will describe the methods used to construct and train both of these networks.

3.1 Data Processing

3.1.1 Three-Dimensional Sod Dataset

The three-dimensional Sod problem is one form of a shock tube problem, which is frequently considered a benchmark test for shock capturing methods. It is also commonly used for testing compressibility terms in numerical codes due to its inclusion of spatial pressure variation [31]. The initial condition contains a driver and driven gas separated by a diaphragm in the center. When the diaphragm breaks, at time zero, a discontinuity forms and travels to the end of the tube. The final solution, which is available as an analytical solution, consists of rarefaction, contact, and shock waves [57].

For each time-step (of a total of 1,775 steps in our experiments) we discretize the position coordinates of the Sod dataset into a $804 \times 4 \times 4$ volume and for each position calculate the strain tensor S and the Schlieren using Eqs. (3) and (4).

We then represent each of the six unique values of the strain tensor as a different channel of a three dimensional image ($6 \times 804 \times 4 \times 4$) and the Schlieren value as a single channel image ($1 \times 804 \times 4 \times 4$). Representing the data in this form allows for us to use computer vision techniques such as volumetric convolutional networks to learn a mapping from the strain tensors to the Schlieren values.

With the data in this format we place every tenth time-step into a test set (177 total) and randomly separate the 90% remaining time-steps into a training set (1416 total) and 10% into an evaluation set (182 total) that is used to determine when the network training has converged.

3.1.2 Two-Dimensional Blast Dataset

The second problem considered is a two-dimensional explosion. The initial condition consists of a high-density, high-pressure region located inside of a circle in the center of the geometry and low-density, low-pressure in the remainder of the computational domain. The two regions are joined by a discontinuity, which travels outwards in time, forming shock, contact, and rarefaction waves. Comparison along the radial directions gives virtually identical results due to the problem's symmetry, and the resolution of discontinuities that travel in all directions is the same as that in the one-dimensional Sod problem [57]. The solution of this problem requires high resolution throughout the domain due to the sharp discontinuities traveling in all spatial directions.

For each time-step of a total of 158 steps we discretize the position coordinates into a 70×70 surface and for each position calculate the strain tensor S using Eq. (3) and the Schlieren value with Eq. (4).

We then represent each of the six unique values of the strain tensor as a different channel of an image ($6 \times 70 \times 70$) and the Schlieren value as a single channel image ($1 \times 70 \times 70$). As with the three-dimensional dataset we place every tenth time-step into a test set (15 total) and randomly separate the 90% remaining time-steps into a training set (133 total) and 10% into an evaluation set (ten total) that is used to determine when the network training has converged.

3.2 Network Architecture

For both the three-dimensional and the two-dimensional dataset we constructed an eight-layer all convolutional network [53] that is divided into two parts; a feature extractor that learns a function for condensing the input features into a low-dimensional feature vector, and an image constructor that learns a function that transforms the low-dimensional feature vector calculated in the previous part into a schlieren image. We construct the network in this way in order to improve generalization by forcing the network to learn a sparse feature representation with useful (general) features of the strain tensors. The ultimate structure is chosen by tuning the hyper-parameters on the evaluation set [7]. The main difference between the two networks is the use of volumetric convolutions for the three-dimensional dataset and regular convolutions for the two-dimensional dataset.

The feature extractor for both networks consist of four strided volumetric convolutional layers that reduce the input strain tensors into a 64 neuron feature vector. Strided convolutions incorporate regularization into the convolutional layers while improving the efficiency in network performance [54] when compared to standard max-pooling based sub-sampling. The image extractor consists of an inverse mapping, sometimes referred to as deconvolutional layers [41], with matching strides and kernel sizes as the feature extractor. The key difference is that the image extractor outputs a single channel image compared to the six channel input of the feature extractor. Figure 1 details the structure of both of the networks with a feature extractor reducing the input strain tensors to a low dimensional vector of 64 features and the image extractor building the Schlieren from the feature vector. Additionally, we use Exponentiated Linear Units (ELU) [11] as our activation function for both networks which leads to improved efficiency and performance while addressing the vanishing gradient problem in training deep networks.

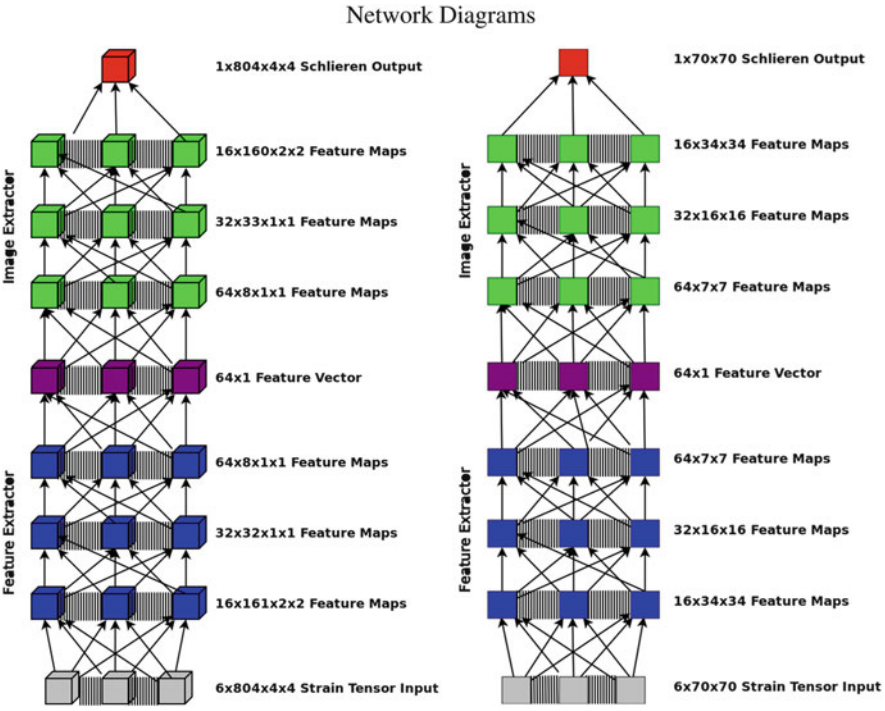


Fig. 1 Networks for three-dimensional Sod dataset (*left*) and the two-dimensional Blast dataset (*right*). Each network comprises two parts; a feature extractor that learns a function for condensing the input features into a low-dimensional feature vector, and an image constructor that learns a function that transforms the low-dimensional feature vector calculated in the previous part into a Schlieren image

3.3 Optimization

We train our network parameters to minimize the mean squared error between the predicted Schlieren values and the ground truth values and optimize the weights with an adaptive learning rate calculated using the adadelta optimization function [59]. To avoid large gradient updates that may be caused from learning regression values, we incorporate gradient clipping [45] to constrain the gradient norm to lie within a specific threshold (i.e., [0, 1]).

In order to improve the efficiency of our training routine we use spatial batch normalization [22] to normalize the feature maps generated after each convolutional layer. This ensures that the input distribution for each layer is consistent (zero mean and unit variance) which greatly improves learning performance.

Convergence of the network training is determined using a five-step average windowed delta loss on the evaluation set. When the average delta loss (mean squared error) on the evaluation set is positive, meaning the network is not improving its ability to construct the Schlieren, learning is stopped. This ensures that we do not over-train on the training set and preserve the network's ability to generalize.

3.4 Regularization

When training a system on a limited amount of data (or an unbalanced dataset) there is a risk of over-training on the training set and losing the ability to generalize to new instances. For this reason we incorporate spatial dropout [56] after each of the convolutional layers in the network. During training this randomly drops entire feature maps in forward propagation (by setting all values to 0). This forces the network to learn a sparsified representation of the feature vector (condensed low-dimensional feature representation) leading to improved generalization.

4 Results

We evaluate our approach on two datasets, a three-dimensional Sod dataset and a two-dimensional Blast dataset as described in Sect. 3.1. In this section we describe our results in training the networks and examine the features the networks learned.

It is important to note that while calculating the Schlieren for the three-dimensional Sod dataset via Eq. (4) takes an average of 550 ms per time step on an Intel Xeon E5-2697 2.6 GHz processor, our trained network generates the Schlieren in less than 107 ms on the same CPU processor (respectively less than 1 ms on a GeForce GTX 1080 GPU) running the Torch framework [12], an open source machine learning library and scientific computing framework that provides a range

of algorithms for deep machine learning. Torch uses the scripting language LuaJIT and an underlying C implementation. This increase in efficiency becomes extremely significant as datasets grow to include hundreds of thousands of time-steps. In total the three-dimensional network trained for about 35 s on the Sod dataset and the two-dimensional network trained for about 6 s on the Blast dataset before convergence. Nevertheless, the main strength of the deep learning approach lies in its potential for anomaly detection, which the Schlieren filter cannot do.

4.1 Training

Figure 2 shows the prediction (mean-squared) error on the training, evaluation, and test set for both datasets after each training epoch. We can clearly see that the networks perform very well on the training sets and converge quickly on the evaluation sets. The test error matches closely with the evaluation error indicating that the evaluation error is a strong representation of the network's ability to generalize to the test set. In fact we were able to achieve an average mean-squared error of 0.14 on the three-dimensional Sod test set and 0.12 on the two-dimensional Blast test set. These results suggest that an adequately trained network has the ability to detect anomalies in the tensor field by comparing the predictive error of the network to the true Schlieren values for each time step. A large error would signal a possible deviation from the regular tensor flow indicating anomalous behavior.

4.2 Learned Network Features

In this section we show examples of the features that the two-dimensional network learns for the Blast dataset. We restrict this section to the two-dimensional dataset

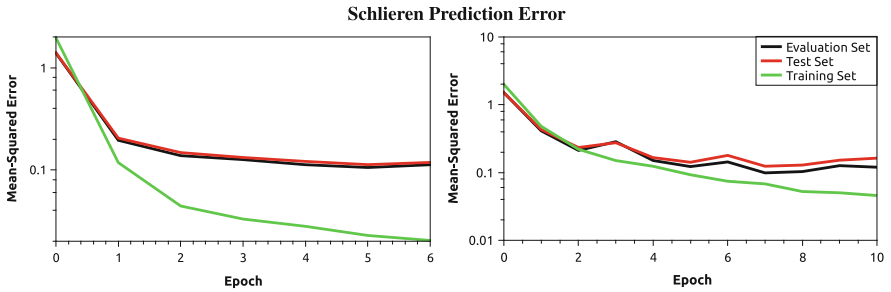


Fig. 2 The average error of the network generated Schlieren for both the three-dimensional Sod dataset (*left*) and the two-dimensional Blast dataset (*right*) after each training epoch. The Y axis represents the mean squared error of the Schlieren value and it is plotted in log-scale for clarity. A standard Schlieren carries units of density gradient (kg/m^4), however the flow simulation code uses a normalized function and as such in this case the Schlieren is unitless

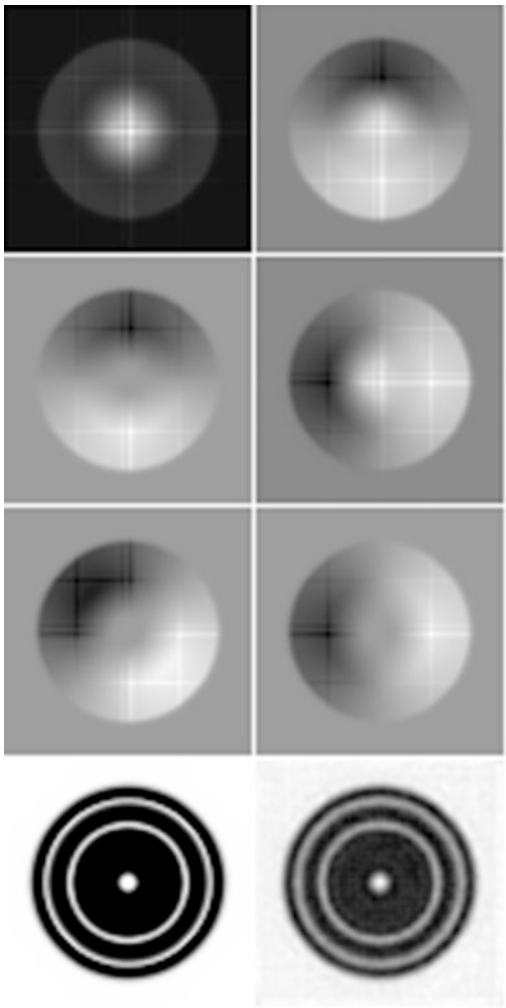


Fig. 3 The six channels of the strain tensor used as input for step 120 followed by the ground truth Schlieren (*bottom left*) and the Schlieren generated by the network (*bottom right*)

for the sake of visual clarity as viewing images of the low resolution volumetric features of the Sod dataset in two dimensions is not very informative. However, the conclusions from the two-dimensional dataset match those of the three-dimensional dataset in that the network learns strong feature representations of the input strain tensors allowing for accurate constructions of the associated three-dimensional Schlieren images.

As an example from the test set, Fig. 3 displays the six channel strain tensor (calculated via Eq. (3)) used as input into the network for time step 120 followed by the true Schlieren image (calculated via Eq. (4) and the Schlieren image generated by the network. We can see that the output of the network (bottom right) matches closely with the calculated Schlieren (bottom left) with a clear

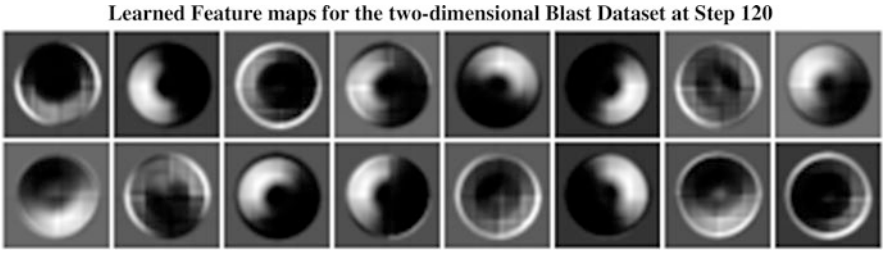


Fig. 4 Feature maps in the first layer of the feature extractor

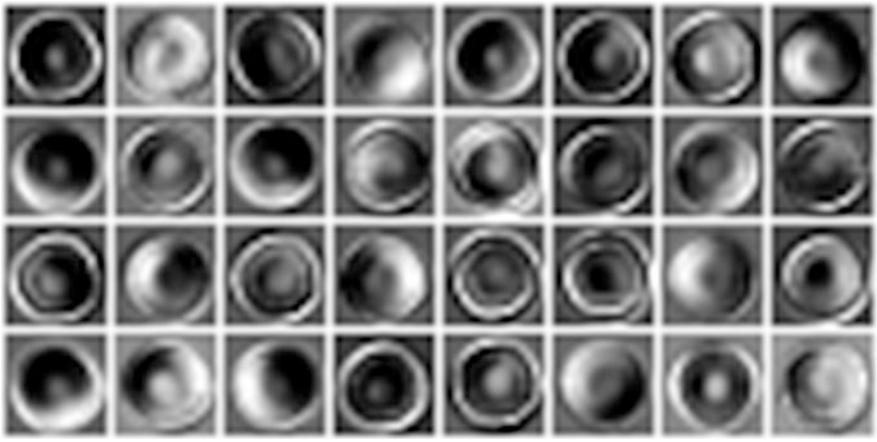


Fig. 5 Feature maps in the second layer of the feature extractor

separation between high and low-density regions. The fact that we can develop such a close approximation of the desired Schlieren for a time step in the test set is very impressive. Deep networks tend to require large amounts of data to generate strong results indicating that increasing the size of our training set will further improve our predictions.

Figures 4 and 5 show examples of feature maps in the first two layers of the feature extractor for time step 120 from the test set (we restrict our view to the first two layers due to the low resolution of feature maps in deeper layers). We observe that the network has learned the informative features of each unique strain tensor and combined them to form feature maps with the necessary information for constructing the Schlieren image. The specialization of these feature maps allows for the network to learn a general representation of the input data which is very important for accurately predicting the Schlieren values in unseen time steps. Additionally, these feature maps serve as a set of dictionary references with increasing abstraction (by layer) that allow the network to properly condense the input tensors into a general low-dimensional feature vector as shown in Fig. 1.

4.3 Output Visualization

The output of the network produces a reconstructed volume for every tenth time-step of both datasets (test cases), where each point corresponds to the floating-point Schlieren value at that location in the discretized grid. In our experiments, we evaluated 177 and 15 output Schlieren volumes for the three-dimensional and two-dimensional datasets, respectively.

To analyze the results, we visualize each volume as a two-dimensional pseudocolor image that encodes the Schlieren value between two diverging colors—red and black. Similar to grayscale photos, pseudocolor images map intensity to a color scale that ranges from the minimum and maximum value of a data sample [13].

We create three pseudocolor images for each reconstructed Schlieren volume to compare it against the input data and its ground truth. Figure 6 shows the analysis of the output corresponding to the three time-steps of the Blast dataset. Note how the distinct dark areas in the reconstructed images (right) match those of the ground truth (left). These areas signify the occurrence of large density gradients across data samples that correlate with potential shock locations. Figure 7 shows the similar ground truth and reconstruction for the three-dimensional dataset. In this figure, note the three distinct shock zones.

The noise in the Schlieren generated from the network in Fig. 6 is a symptom of the low amount of training data in this particular dataset. We trained on data from a single combustion run for both the three-dimensional and two-dimensional settings. Expanding the training data will likely improve results, as is the case in most deep learning applications. This paper is meant as a proof of concept that deep learning is a feasible tool for generating the Schlieren from strain tensors and had potential for identifying anomalies in the combustion fields. Additional image processing may create a clearer Schlieren, as would an increase in the amount of training data.

5 Discussion and Conclusion

As previously discussed, the goal of this project was to examine the potential of using deep learning on tensor field data generated by turbulent combustion simulations. First, we were interested in finding out whether a supervised approach can detect structures in the data, and whether these structures correlate with the regions of interest. Second, we wanted to examine whether this problem is computationally feasible. The answer to both questions is affirmative.

In summary, we found that the deep learning approach can effectively capture and construct shock features. The results indicate that deep networks have the ability to identify anomalies in the tensor flow. Furthermore, the efficiency of the machine learning algorithm exceeds that of calculating the Schlieren via Eq. (4). With a greater than 5× improvement on time complexity for the three-dimensional Sod dataset. The efficiency of the machine learning algorithm leads to a 500×

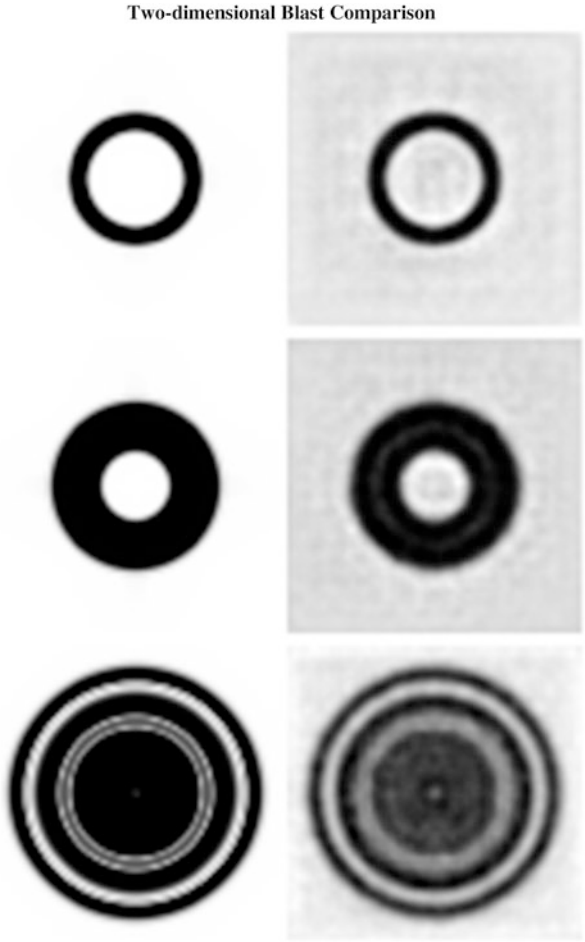


Fig. 6 The network comparison between the network ground truth (*left*) and the final reconstruction (*right*) for the 1st, 50th, and 150th time-step of the two-dimensional Blast dataset. The *dark areas* seen in the images indicate large density gradients corresponding to potential shock regions

improvement when running on the GPU, while optimized CPU to GPU throughput-computing transfers lead to only a 2.5× improvement on average [30]. A small deviation in the predicted values shows that deep learning has the potential to be an effective tool for efficiently generating visualizations of large tensor fields.

It is important to note that this work is exploratory and is meant to be a proof of concept for the use of deep networks in visualizing large tensor fields. While the results are strong, we trained and tested the networks on the same time sequences (with separate training and testing time steps). Predicting time steps in sequences that were not previously trained on may be a more difficult task, However, the results

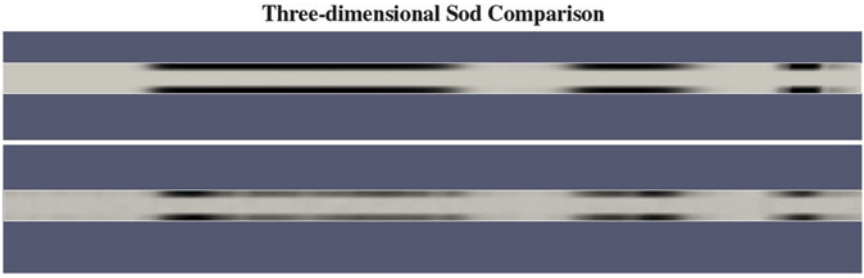


Fig. 7 The network comparison between the network ground truth (*top*) and the final reconstruction (*bottom*) for the 100th time-step of the three-dimensional Sod dataset. The *dark narrow bands* seen in the two images indicate large density gradients, and correspond to potential shock regions. Note the three distinct shock zones in the images

in this chapter were generated with a very limited dataset, 1775 and 158 time-steps for the Sod and Blast datasets respectively, and deep networks usually require very large amounts of data to be effectively trained [55]. This implies that while the predictive performance may drop with separate training and testing sequences, it may also improve with the inclusion of more data. Additionally, the datasets used in this chapter are sequential in nature lending to the notion that the results may improve with deep architectures that can take advantage of sequential features in their predictions such as recurrent neural networks [18]. Further exploration of this area is needed in order to form a decisive conclusion.

In conclusion, we have introduced a supervised machine learning approach for the segmentation of shocks in large scale tensor field datasets generated by computational turbulent combustion simulations. The approach employs a deep learning architecture based on volumetric convolutional networks. Our evaluation on two rich combustion datasets shows this approach can assist in the visual analysis of the combustion tensor field and that it is more effective than direct filtering calculations. Most importantly, the approach has potential for the detection of anomalies in the data.

Acknowledgements This work was partially supported by the National Science Foundation through award NSF CAREER IIS-1541277. We gratefully acknowledge the NSF Graduate Research Fellowship program for supporting Tim. We thank Adrian Maries and Shiwangi Singh for the initial literature search, and the Electronic Visualization Lab for their feedback and support. We further thank the Peyman Lab for the original motivation behind this line of work, and the Dagstuhl 16142 and 14082 seminars run by the Leibniz Center for Informatics for the many useful discussions regarding tensor field analysis and visualization.

References

1. Abbassi, H., Komperda, J., Mashayek, F., Jacobs, G.: Application of entropy viscosity method for supersonic flow simulation using discontinuous spectral element method. AIAA Paper 2012-1115 (2012)

2. Abbassi, H., Mashayek, F., Jacobs, G.: Entropy viscosity approach for compressible turbulent simulations using discontinuous spectral element method. AIAA Paper 2014-0947 (2014)
3. Abbassi, H., Mashayek, F., Jacobs, G.: Shock capturing with entropy-based artificial viscosity for staggered grid discontinuous spectral element method. Comput. Fluids **98**, 152–163 (2014)
4. Banaeizadeh, A., Li, Z., Jaberi, F.: Compressible scalar filtered mass density function model for high-speed turbulent flows. AIAA J. **49**(10), 2130–2143 (2011)
5. Bengio, Y., Lecun, Y.: Scaling Learning Algorithms Toward AI, pp. 321–359. MIT Press, Cambridge (2007)
6. Berglund, M., Fureby, C.: Les of supersonic combustion in a scramjet engine model. In: Proceedings of the Combustion Institute, vol. 31, pp. 2497–2504. The Combustion Institute, Pittsburgh, PA (2007)
7. Bergstra, J., Bengio, Y.: Random search for hyper-parameter optimization. J. Mach. Learn. Res. **13**, 281–305 (2012)
8. Bermejo-Moreno, I.: Subgrid-scale modeling of shock-turbulence interaction for large-eddy simulation. Annual Research Briefs, Center for Turbulence Research, Stanford University, Stanford, CA (2009)
9. Bourlard, H., Kamp, Y.: Auto-association by multilayer perceptrons and singular value decomposition. Biol. Cybernet. **59**(4), 291–294 (1988)
10. Caban, J., Joshi, A., Rheingans, P.: Texture-based feature tracking for effective time-varying data visualization. IEEE Trans. Vis. Comput. Graph. **13**(6), 1472–1479 (2007)
11. Clevert, D., Unterthiner, T., Hochreiter, S.: Fast and accurate deep network learning by exponential linear units (ELUs). CoRR abs/1511.07289 (2015)
12. Collobert, R., Kavukcuoglu, K., Farabet, C.: Torch7: A matlab-like environment for machine learning. In: BigLearn, NIPS Workshop (2011)
13. Dimitrov, L.I.: Pseudo-colored visualization of EEG-activities on the human cortex using MRI-based volume rendering and Delaunay interpolation. Medical Imaging, 460–469 (1995)
14. Ducros, F., Ferrand, V., Nicoud, F., Weber, C., Darracq, D., Gacherieu, C., Poinsot, T.: Large-eddy simulation of the shock/turbulence interaction. J. Comput. Phys. **152**(2), 517–549 (1999)
15. Edwards, J.R., Potturi, A., Fulton, J.A.: Large-eddy / reynolds-averaged navier-stokes simulations of scramjet combustor flow fields. AIAA Paper 2012–4262 (2012)
16. Fulton, J.A., Edwards, J.R., Hassan, H.A., McDaniel, J.C., Goyne, C.P., Rockwell, R.D., Cutler, A.D., Johansen, C.T., Danehy, P.M.: Large-eddy/reynolds-averaged Navier-Stokes simulations of reactive flow in dual-mode scramjet combustor. J. Propuls. Power **30**(3), 558–575 (2013)
17. Ghiasi, Z., Komperda, J., Li, D., Mashayek, F.: Simulation of supersonic turbulent non-reactive flow in ramp-cavity combustor using a discontinuous spectral element method. AIAA Paper 2016-0617 (2016)
18. Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., Schmidhuber, J.: A novel connectionist system for unconstrained handwriting recognition. IEEE Trans. Pattern Anal. Mach. Intell. **31**(5), 855–868 (2009)
19. Hadjadj, A., Kudryavtsev, A.: Computation and flow visualization in high-speed aerodynamics. J. Turbul. **6**, 16 (2005)
20. Hebb, D.O.: The Organization of Behavior. Wiley, New York (1949)
21. Hinton, G.E., Salakhutdinov, R.R.: Reducing the dimensionality of data with neural networks. Science **313**(5786), 504–507 (2006)
22. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. CoRR abs/1502.03167 (2015)
23. Irannejad, A., Jaberi, F., Komperda, J., Mashayek, F.: Large eddy simulation of supersonic turbulent combustion with FMDF. AIAA Paper, 1188, p. 2014 (2014)
24. Ji, G., Shen, H.-W., Wenger, R.: Volume tracking using higher dimensional isosurfacing. In: Proceedings of the 14th IEEE Visualization 2003 (VIS'03), Washington, DC, p. 28. IEEE Computer Society (2003)
25. Ju, Y., Shimano, A., Inoue, O.: Vorticity generation and flame distortion induced by shock flame interaction. In: Proceedings of the Combustion Institute, vol. 27, pp. 735–741. The Combustion Institute, Boulder, CO (1998)

26. Khokhlov, A.M., Oran, E.S., Thomas, G.O.: Numerical simulation of deflagration-to-detonation transition: the role of shock–flame interactions in turbulent flames. Combust. Flame **117**(1–2), 323–339 (1999)
27. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (2012)
28. Larsson, J., Vicquelin, R., Bermejo-Moreno, I.: Large eddy simulations of the HyShot II scramjet. Annual Research Briefs, Center for Turbulence Research, Stanford University, Stanford, CA (2011)
29. LeCun, Y., Bengio, Y.: Convolutional networks for images, speech, and time series. In: Arbib, M.A. (ed.) The Handbook of Brain Theory and Neural Networks, pp. 255–258. MIT Press, Cambridge, MA (1998)
30. Lee, V.W., Kim, C., Chhugani, J., Deisher, M., Kim, D., Nguyen, A.D., Satish, N., Smelyanskiy, M., Chennupaty, S., Hammarlund, P., et al.: Debunking the 100× GPU vs. CPU myth: an evaluation of throughput computing on CPU and GPU. ACM SIGARCH Comput. Archit. News **38**(3), 451–460 (2010)
31. Li, Z., Banaeizadeh, A., Rezaeiravesh, S., Jaberi, F.: Advanced modeling of high speed turbulent reacting flows. AIAA Paper 2012-116 (2012)
32. Lovely, D., Haimes, H.: Shock detection from computational fluid dynamics results. In: 14th Computational Fluid Dynamics Conference, Cambridge, MA, pp. 255–258. MIT Press, Cambridge (1999)
33. Ma, K.-L., Rosendale, J.V., Vermeer, W.: 3d shock wave visualization on unstructured grids. In: VVS, p. 87 (1996)
34. Marai, G.E., Luciani, T., Maries, A., Yilmaz, S.L., Nik, M.B.: Visual descriptors for dense tensor fields in computational turbulent combustion: a case study. In: Visualization and Data Analysis, pp. 1–11. Ingenta, Oxford (2016)
35. Maries, A., Luciani, T., Pisciuneri, P.H., Nik, M.B., Yilmaz, S.L., Givi, P., Marai, G.E.: A clustering method for identifying regions of interest in turbulent combustion tensor fields. In: Visualization and Processing of Higher Order Descriptors for Multi-Valued Data, pp. 323–338. Springer, Cham (2015)
36. Matsugu, M., Mori, K., Mitari, Y., Kaneda, Y.: Subject independent facial expression recognition with robust face detection using a convolutional neural network. Neural Netw. **16**(5–6), 555–559 (2003)
37. Maturana, D., Scherer, S.: Voxnet: a 3d convolutional neural network for real-time object recognition. In: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 922–928. IEEE, New York (2015)
38. McCulloch, W.S., Pitts, W.: A logical calculus of the ideas immanent in nervous activity. Bull. Math. Biophys. **5**(4), 115–133 (1943)
39. Milletari, F., Navab, N., Ahmadi, S.: V-net: fully convolutional neural networks for volumetric medical image segmentation. CoRR abs/1606.04797 (2016)
40. Mynski, M.L., Papert, S.A.: Perceptrons: An Introduction to Computational Geometry. MIT Press, Cambridge, MA (1969)
41. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. CoRR abs/1505.04366 (2015)
42. Orbach, J.: Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Arch. Gen. Psychiatr. **7**(3), 218–219 (1962)
43. Ozer, S., Wei, J., Silver, D., Martin, P.: Group dynamics in scientific visualization. in large data analysis and visualization (LDAV). In: 2012 IEEE Symposium on Large Data Analysis and Visualization (LDAV), October 2012, pp. 97–104
44. Pagendarm, H.-G., Walter, B.: Feature detection from vector quantities in a numerically simulated hypersonic flow field in combination with experimental flow visualization. In: Proceedings of the Conference on Visualization '94, VIS '94, pp. 117–123, Los Alamitos, CA. IEEE Computer Society Press (1994)
45. Pascanu, R., Mikolov, T., Bengio, Y.: Understanding the exploding gradient problem. CoRR abs/1211.5063 (2012)

46. Peterson, D.M., Hagenmaier, M., Carter, C.D., Tuttle, S.G.: Hybrid reynolds-averaged and large-eddy simulations of a supersonic cavity flameholder. AIAA Paper 2013–2483 (2013)
47. Qi, C.R., Su, H., Nießner, M., Dai, A., Yan, M., Guibas, L.J.: Volumetric and multi-view cnns for object classification on 3d data. CoRR abs/1604.03265 (2016)
48. Roy, C.J., Edwards, J.R.: Numerical simulation of a three-dimensional flame/shock wave interaction. AIAA J. **38**(5), 745–754 (2000)
49. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. In: Anderson, J.A., Rosenfeld, E. (eds.) Neurocomputing: Foundations of Research, pp. 696–699. MIT Press, Cambridge, MA, (1988)
50. Samtaney, R., Silver, D., Zabusky, N., Cao, J.: Visualizing features and tracking their evolution. IEEE Comput. **27**(7), 20–27 (1994)
51. Silver, D.: Object-oriented visualization. IEEE Comput. Graph. Appl. **15**(3), 54–62 (1)
52. Silver, D., Wang, X.: Tracking and visualizing turbulent 3d features. IEEE Trans. Vis. Comput. Graph. **3**(2), 129–141 (1997)
53. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.A.: Striving for simplicity: the all convolutional net. CoRR abs/1412.6806 (2014)
54. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.A.: Striving for simplicity: the all convolutional net. CoRR abs/1412.6806 (2014)
55. Srinivas, S., Sarvadevabhatla, R.K., Mopuri, K.R., Prabhu, N., Kruthiventi, S.S.S., Babu, R.V.: A taxonomy of deep convolutional neural nets for computer vision. Front. Robot. AI **2**, 36 (2016)
56. Tompson, J., Goroshin, R., Jain, A., LeCun, Y., Bregler, C.: Efficient object localization using convolutional networks. CoRR abs/1411.4280 (2014)
57. Toro, E.F.: Riemann Solvers and Numerical Methods for Fluid Dynamics, 3rd edn. Springer, Berlin (2009)
58. Tzeng, F.-Y., Ma, K.-L.: Intelligent feature extraction and tracking for visualizing large-scale 4d flow simulations. In: Proceedings of the 2005 ACM/IEEE Conference on Supercomputing, SC '05, Washington, DC, p. 6. IEEE Computer Society (2005)
59. Zeiler, M.D.: ADADELTA: an adaptive learning rate method. CoRR abs/1212.5701 (2012)

Reconstruction of Diffusion Anisotropies Using 3D Deep Convolutional Neural Networks in Diffusion Imaging

Simon Koppers, Matthias Friedrichs, and Dorit Merhof

Abstract The reconstruction of neural pathways is a challenging problem in case of crossing or kissing neuronal fibers. High angular resolution diffusion imaging models are required to identify multiple fiber orientations in a voxel. Disadvantage of those models is that they require a multitude of acquired gradient directions, otherwise these models become inaccurate. We present a new approach to derive the fiber orientation distribution function using a Deep Convolutional Neural Network, which remains stable, even if less gradient directions are acquired. In addition, the Convolutional Neural Network is able to improve the signal in a voxel by extracting useful information of surrounding neighboring voxels. Subsequently, the functionality of the network is evaluated using 100 different brain datasets from the Human Connectome Project.

1 Introduction

Diffusion-weighted MRI is able to provide subject specific information about the course and location of white matter tracts in the human brain. In cases of kissing or crossing fibers, which occur in 60–90% of human white matter [5], high angular resolution diffusion imaging (HARDI) is needed to better represent such complex structures. For this purpose many different gradient directions are typically acquired, which linearly increases the acquisition time, resulting in an acquisition time of several minutes up to hours. However, for clinical application it is necessary to reduce the number of required gradient directions to keep the acquisition time in a feasible range. As a downside, this reduction will result in a blurred reconstruction, if standard reconstruction applications are applied.

While Neural Networks have been known for decades, Deep Neural Networks have gained interest recently in the field of Diffusion Imaging [4, 6, 7]. It was shown that Deep Learning is able to stabilize state-of-the-art approaches, if only a few

S. Koppers (✉) • M. Friedrichs • D. Merhof
Institute of Imaging & Computer Vision, RWTH Aachen University, Aachen, Germany
e-mail: Simon.Koppers@lfb.rwth-aachen.de; Matthias.Friedrichs@rwth-aachen.de; Dorit.Merhof@lfb.rwth-aachen.de

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1_17

gradient directions are acquired. Nevertheless, none of these approaches is so far able to directly reconstruct the whole fiber orientation distribution function (fODF). Furthermore, the Diffusion Signal was so far reconstructed in a voxel-wise manner, excluding additional neighboring information.

In this work, a novel way to reconstruct a non-quantized fODF by including neighboring information is presented. For this purpose, we decompose the spherical fODF utilizing well-known Spherical Harmonics (SH), which are able to represent spherical signals in a general and complete way. The resulting SH coefficients are fitted utilizing a Neural Network regression approach. In addition, previously excluded neighboring information are included using Deep Convolutional Neural Networks (CNN), which are able to extract local neighboring information.

2 Methods

In order to represent an fODF, a model-free and non-sparse representation of a spherical signal is required. For this purpose, SH are utilized to represent the spherical signal, which are fitted utilizing a Deep Regression CNN. In the first part of this section, the material, which is used in this work, is described. The second part describes the utilized CNN.

2.1 Training Data and Labels

In order to evaluate the performance of our algorithm, a subset of 100 uncorrelated healthy brain scans from the Human Connectome Project is utilized. All scans are acquired with a 3T Siemens Connectome Skyra MRI scanner, a resolution of $1.25 \times 1.25 \times 1.25\text{mm}^2$ and 288 diffusion gradients, comprising 18 diffusion gradients with $b = 0\frac{\text{s}}{\text{mm}^2}$ and 3 shells comprising 90 diffusion gradients each, at $b = 1000\frac{\text{s}}{\text{mm}^2}$, $b = 2000\frac{\text{s}}{\text{mm}^2}$, $b = 3000\frac{\text{s}}{\text{mm}^2}$, respectively. Because of its high resolution ($145 \times 174 \times 145$ with 288 gradients directions each), the data is further reduced by selecting 5000 non-isotropic voxel neighborhoods ($3 \times 3 \times 3$) from each subject, resulting in 500,000 unique sets. In addition, only the third shell ($b = 3000\frac{\text{s}}{\text{mm}^2}$) is utilized.

The CNN is trained on 80 uncorrelated scans (400,000 samples), while validation is performed on the remaining 20 subjects (100,000 samples). To show the impact of a reduced number of gradients, three additional datasets are generated. For this purpose, the diffusion signal is equidistantly resampled for 15, 30 and 45 gradient directions utilizing the well-known SH.

2.1.1 Neighborhood

Due to the fact that additional information can be gained utilizing the signals' neighborhood [2], three different neighborhoods are compared. The first neighborhood

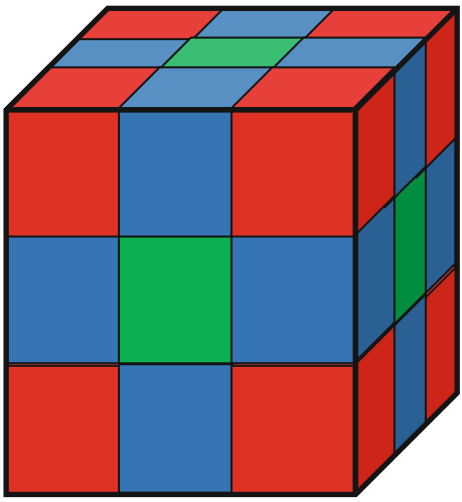


Fig. 1 3D voxel neighborhood, which is represented as a hierarchy of three types of neighborhoods (*green*; *green* + *blue*; *green* + *blue* + *red*)

consists out of six additional signals, which are marked in green in Fig. 1. The second neighborhood adds the blue-marked voxels to the first neighborhood resulting in 18 additional voxels. The biggest neighborhood includes 26 additional signals, which is represented by the surrounding cube (see Fig. 1).

2.1.2 Gold Standard

In order to generate a valid label for training and comparison, the well-known and state-of-the-art constrained spherical deconvolution [11, 12] (CSD) is utilized. The idea of CSD is that the diffusion signal can be decomposed, if its underlying single-fiber response function is known. Based on this response function, the diffusion signal can be deconvoluted in order to generate the fODF.

The CSD is applied on the raw signal as well as on its resampled versions containing 45, 30 and 15 gradient directions, respectively. To be comparable and stable for each set of gradient directions, the reconstruction order is set to 4 to be stable. Its output is a sampled fODF, which is described utilizing SH of order 8, resulting in 45 coefficients, which are used as label for training.

Since CSD is unable to directly define the resulting fiber direction, the Ball-and-Stick [10] (BS) model is utilized to calculate a relative ground truth fiber direction. It is a well-known model and one of the most accurate approaches to identify the correct fiber direction [8], if the correct number of fibers is known.

2.2 Deep Regression Convolutional Neural Network

Assuming the fODF to be sufficiently represented by SH coefficients, a regression approach is required to fit these coefficients, which is based on Deep Learning. In this work a Deep Regression CNN is utilized to fit the SH coefficients. CNNs are a special case of neural networks, in which the weights between neurons are combined in a matrix. This matrix can be interpreted as a convolutional kernel, which can be applied to 1D, 2D and also higher dimensional signals. In this work, additional information about the fODF are gained by combining 3D neighboring signals, which is why a 3D CNN is utilized.

In addition, each neural network contains an activation function. We choose the Rectified Linear Unit (ReLU), because of its good convergence behavior, with

$$Y = \max(0, x), \tag{1}$$

which is applied after each convolutional layer, with x as input and Y as output signal.

A combination of these two layers is further called a functional unit (FU), which is further specified by its convolutional kernel size, the number of kernels and its number of input channels. Due to the fact that every kernel leads to a new dataset, the number of input channels of the actual layer needs to be equivalent to the number of kernels of the previous layer. Furthermore, zero-padding with size p may be considered during convolution.

In most CNNs, several layers are stacked together, followed by a fully-connected layer that maps different features to the output. In the end, a cost function calculates a loss, which is utilized to train the network. For our CNN, a standard least square loss function is utilized, which is in general a good loss function for a regression problem. Training is performed using a stochastic gradient descend algorithm with error backpropagation.

2.3 The Net

The neural network is composed out of four FUs, while the last FU is fully connected to the output. The whole network is presented in Fig. 2. In order to be invariant to different gradient directions, the input and output signal are represented utilizing SH. Due to this fact, the number of input channels in the first layer and the number of kernels in the last layer are defined by the number of expected coefficients.

For this CNN, the input signal is represented using an SH order of 4, resulting in 15 input coefficients, in order to be more robust to noise. The output represents the fODF utilizing the SH coefficients corresponding to an order of 8.

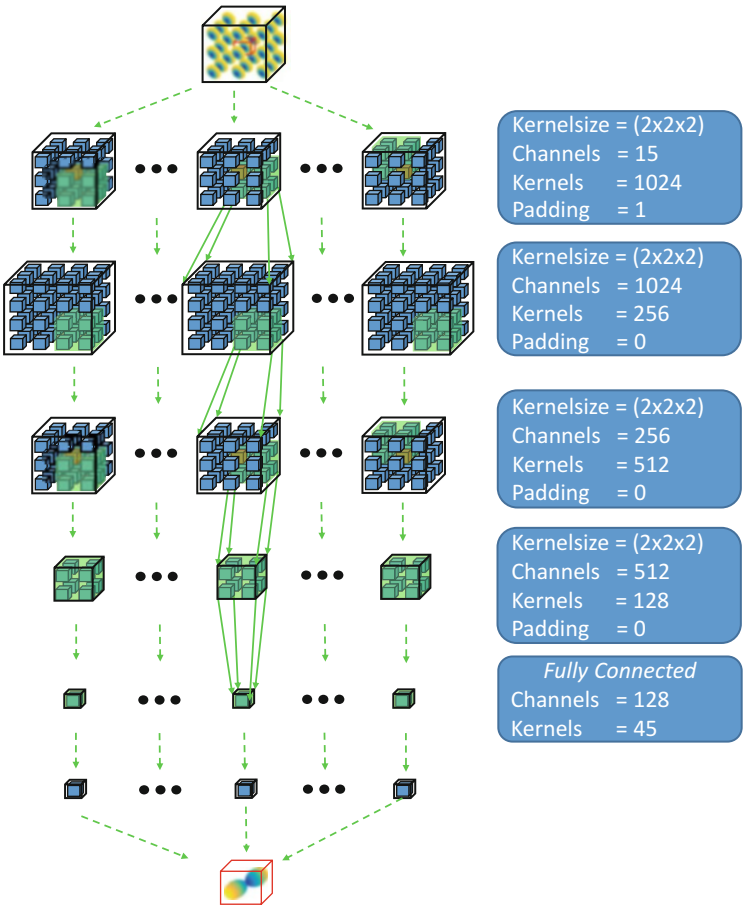


Fig. 2 The 3D CNN, which is utilized to reconstruct the fODF based on a diffusion signal and its neighborhood

3 Experimental Results

The CNN is implemented in Python based on the TensorFlow framework [1], while the CSD is implemented in Python using Dipy [3].

Training is performed using stochastic gradient descend with a learning rate of $\eta = 0.01$ and a batchsize of 100.

In order to validate the performance of the CNN, two different error quantities are evaluated. On the one hand, the surface deviation of the fODF is calculated, which is an important quantity for probabilistic tractography approaches and diffusion characteristics. It is evaluated using the normalized mean square error [6] with

$$\text{NMSE} = \frac{\|S_{\text{true}} - S_{\text{rec}}\|_2^2}{\|S_{\text{true}}\|_2^2}, \tag{2}$$

which is common error metric for comparing surfaces. S_{true} defines the original signal vector for each gradient direction, while S_{rec} is the reconstructed signal vector.

Furthermore, the angular accuracy is evaluated, which is particularly important for generating a deterministic tractography. In order to evaluate the angular accuracy, 250 single-fiber validation voxels, 250 validation voxels with two fibers and 250 three fiber validation voxels are manually selected, based on their relative ground truth fODF, while BS defines the relative ground truth fiber direction.

The mean angular error (MAE) is calculated based on

$$\text{MAE} = \frac{1}{k} \sum_{i}^{k} \alpha_i, \tag{3}$$

where k defines the number of fibers in a voxel and α_i the angular error of fiber i.

3.1 fODF Reconstruction

In order to evaluate the reconstruction quality, each fODF is relatively compared to its gold standard fODF based on 90 gradient directions. The resulting qualities for the CSD and the CNN are presented in Table 1, which contains the resulting NMSE. In addition, it should be noted that the CNN is completely retrained for each gradient and neighborhood scenario.

The smallest error, about 1.65%, occurs for a reconstruction based on 45 gradient directions utilizing the CSD, while the highest NMSE, about 32.81%, is also observed for the CSD, when only 15 gradient directions are used for reconstruction. In any other case, the CSD gets outperformed by the CNN. Taking the neighborhood into account, it can be seen that there is a tendency that a neighborhood results in a lower NMSE, if less gradient directions are available. The lowest NMSE is achieved utilizing 18 neighbor voxels (green + blue).

In addition to the NMSE, Fig. 3 presents a randomly chosen region out of an axial slice. It contains the fODF for each voxel, with a big fiber bundle running from the upper left to the right. Moreover, there is a crossing of two fiber bundles

Table 1 Impact of signal neighborhood on the surface deviation based on 15, 30 and 45 diffusion gradients with CNN as well as CSD

Method	Neighborhood	fODF deviation [%]		
		15 gradients	30 gradients	45 gradients
DL	Green + blue + red	13.19	9.72	6.64
	Green + blue	13.12	9.21	6.24
	Green	14.46	10.24	6.96
	None	18.89	10.53	5.89
CSD	–	32.81	11.21	1.65

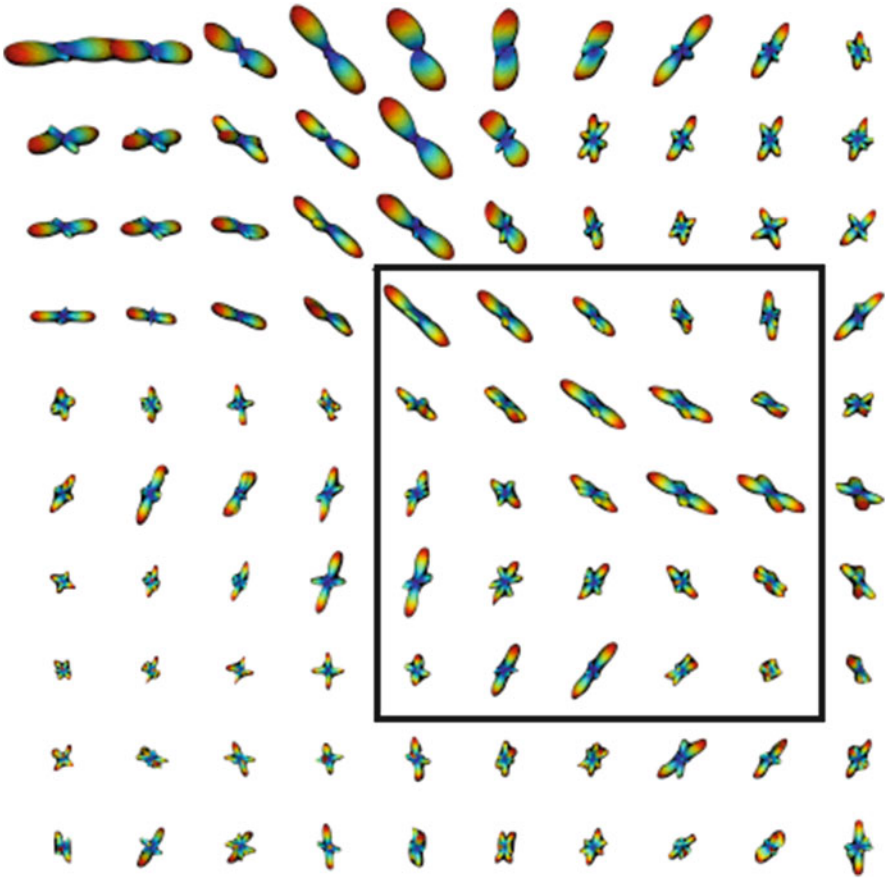


Fig. 3 Mapping of the gold standard fODF in the human brain with a marked region of interest

within the box region. This region of interest is further evaluated in Fig. 4 for 15 and 90 gradient directions utilizing the CSD and the CNN for reconstruction. For evaluation purposes, no neighborhood is included for a fair comparison. Figure 4b shows that the CNN is not able to identify weak fiber directions, which constitute only a small fraction to the remaining signal.

On the other hand, the fODF reconstructed by CSD gets worse as the number of gradient directions is reduced to 15. Due to this reduction of gradient directions, CSD is, in most cases, not able to identify fiber crossing, while the CNN is still able to identify crossings.

Furthermore, the fiber directions seem to diverge, which would lead to an increased angular error, if the number of gradient directions is reduced.

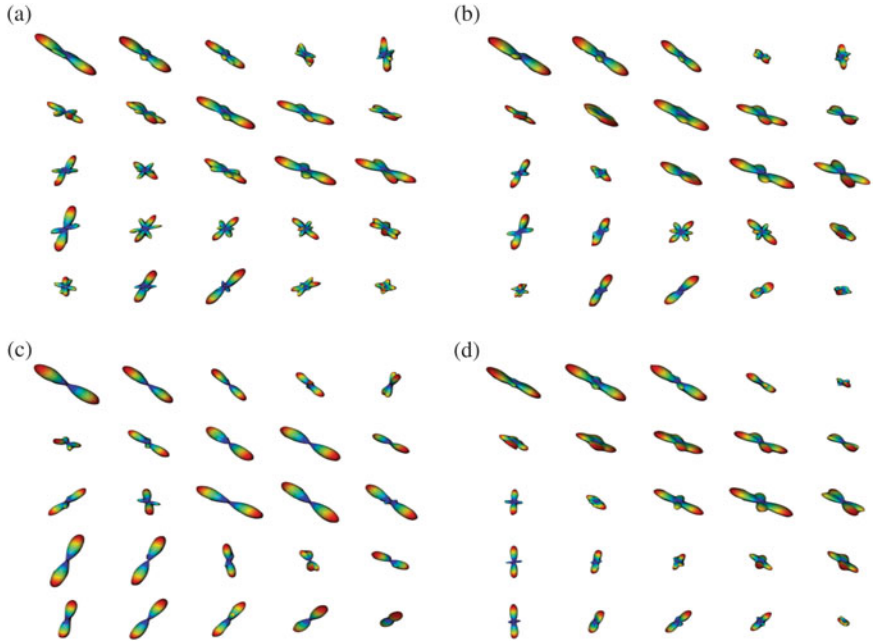


Fig. 4 Reconstruction of fODF in the human brain for various methods. Reconstruction is based on 15 and 90 equidistant diffusion gradients. (**a**) Reconstruction based on CSD and 90 acquired gradient directions (gold standard). (**b**) Reconstruction based on CNN and 90 acquired gradient directions. (**c**) Reconstruction based on CSD and 15 acquired gradient directions. (**d**) Reconstruction based on CNN and 15 acquired gradient directions

3.2 Angular Accuracy

In order to evaluate the angular accuracy, each fODF is further processed using a local maximum finder to define a specific fiber direction. The resulting mean angular errors (MAE) of the regular CSD in comparison to the CNN with different neighborhoods and 15, 30 and 45 diffusion gradients are provided in Tables 4, 3 and 2.

Each table includes the mean angular error for the single-fiber, two-fiber and three-fiber case. For 45 gradient directions (see Table 2) the smallest error is achieved utilizing CSD for single-fiber cases and three-fiber cases, while the CNN is slightly better for two-fiber cases. Considering Table 3 both algorithms achieve nearly the same results. If only 15 gradient directions are available, the CNN achieves the smallest angular error in each case, whereas the error of the CSD reconstruction is significantly increased.

Table 2 Impact of signal neighborhood on angular accuracy for 45 acquired diffusion gradients utilizing the CNN in comparison to CSD

Method	Neighborhood	Angular error [°]		
		1 Fiber	2 Fibers	3 Fibers
DL	Green + blue + red	1.86	3.18	12.64
	Green + blue	1.84	3.07	12.54
	Green	1.79	2.95	12.89
	None	1.73	2.76	12.31
CSD	None	1.65	3.02	6.06

The angular error represents the mean angular error for 1-Fiber, 2-Fiber and 3-Fiber voxels

Table 3 Impact of signal neighborhood on angular accuracy for 30 acquired diffusion gradients utilizing the CNN in comparison to CSD

Method	Neighborhood	Angular error [°]		
		1 Fiber	2 Fibers	3 Fibers
DL	Green + blue + red	1.98	4.87	16.20
	Green + blue	1.96	4.68	15.99
	Green	2.08	5.11	15.26
	None	1.91	5.76	15.74
CSD	None	1.90	5.59	14.47

The angular error represents the mean angular error for 1-Fiber, 2-Fiber and 3-Fiber voxels

Table 4 Impact of signal neighborhood on angular accuracy for 15 acquired diffusion gradients utilizing the CNN in comparison to CSD

Method	Neighborhood	Angular error [°]		
		1 Fiber	2 Fibers	3 Fibers
DL	Green + blue + red	2.86	7.10	16.58
	Green + blue	2.87	6.80	16.01
	Green	2.91	7.49	17.43
	None	3.43	10.87	18.06
CSD	None	3.63	14.55	21.88

The angular error represents the mean angular error for 1-Fiber, 2-Fiber and 3-Fiber voxels

In addition, taking the neighborhood into account, the same tendency as before can be observed. The angular error decreases for 15 and 30 gradient directions if a neighborhood is used to reconstruct the fODF, while it increases for 45 gradient directions. Once again, the 18-neighborhood (green + blue) achieves the best result.

4 Discussion

Considering the surface deviation, Tables 1, Figs. 3 and 4 show that the CNN results in a higher NMSE for 45 gradient directions than the CSD, while it achieves a lower NMSE for 30 and 15 gradient directions. In addition, the performance of the CNN stays stable in comparison to the performance of the CSD, which decreases and nearly doubles the resulting NMSE in comparison to the CNN. Furthermore,

this objective result is verified in Fig. 4d, which is visually very similar to the CSD label with 90 gradient directions in Fig. 4a. Moreover, Fig. 4c shows that the CSD approach is not able to correctly reconstruct crossings for 15 gradient directions.

Additionally, the performance of the CNN improves if neighboring voxels are included. Comparing the different neighborhoods shows that the second neighborhood (green + blue) achieves the best NMSE for 15 and 30 gradient directions, while any neighborhood results in a decreasing performance if 45 gradient directions are available. This coincides with the previously stated hypothesis, that additional information can be gained from neighboring voxels if only few measured gradient directions are available. In addition, it should be noted that the biggest neighborhood including the red voxels (see Fig. 1) decreases the performance in comparison to the second neighborhood (green + blue). This deterioration may be due to the fact that the distance between the main voxel and the included neighboring voxels increases, which results in a decreased correlation between the corresponding signals. In those cases, the CNN is not able to train the corresponding weights to zero, which results in a decreased performance because of included weighted noise. The same effect can be seen for more than 45 gradient directions.

The resulting mean angular error, presented in Tables 2, 3 and 4, shows an effect similar to the surface deviation. While the angular error for the first and second fiber is on the same level for the CNN as well as for the CSD, the CNN is not able to identify the third fiber even for 45 gradient directions, resulting in a high angular error. The same effect can be seen in Fig. 4b, which shows that the CNN is mostly unable to identify fibers with a small fraction compared to the remaining fibers in the voxel, even for 90 gradient directions. Nevertheless, the CNN outperforms the CSD for 30 and 15 gradient directions. Here, the CNN stays relatively stable, while the CSD becomes unstable, which may be due to the fact that it is only able to reconstruct the main fiber in a voxel.

As before for the surface deviation, the impact of neighboring signals is even more visible. The second neighborhood (green + blue) results in a lower angular error. Again, the gap between the CSD and the CNN increases as the number of gradients decreases. In addition, the performance of the biggest neighborhood still doesn't result in an improvement.

The fact, that the CNN is in most cases not able to identify the correct third fiber direction can be explained as follows:

- First, the chosen SH order may be too low, which would result in an fODF blur, while fibers with small inter-fiber angles can't be distinguished. On the other hand, an increasing SH order would result in a bigger output vector, which increases the complexity of the 3D CNN. Because of this, the 3D CNN may not be able to achieve a better performance than the empirically chosen 3D CNN.
- Secondly, the CNN is trained utilizing 80 different subjects for training. This results in a high variance, which results in a good generalization of the CNN, making it insensitive against head orientations and different diffusivities. However, this positive effect could turn into a disadvantage, because as the generalization of the CNN increases, it may get less sensitive to subtle changes of the

diffusion signal. To face this issue, [9] showed that a synthetically generated dataset can be utilized for training in order to generate a unique CNN for a specific subject. This would result in a net, which is not blurred due to high variance.

- The last point is that the chosen label is based on the CSD applied on real human data. A disadvantage of this method is that noise, which is inevitably contained in each acquisition, leads to a noisy label. This noise would especially affect weak signals, which mostly occur for signals of the second or third fiber. Again, a synthetic dataset could be utilized for training, in order to obtain a noise free label.

Since not all MRI scanners provide the possibility to measure high b-values, the CNN is also evaluated using the same dataset and a b-value of $b = 1000\frac{\text{s}}{\text{mm}^2}$. Utilizing this new dataset, the performance of the CNN as well as for the CSD decreases dramatically, making it unsuitable for clinical purpose. This may be, due to the reduced contrast of the diffusion signal within a voxel at a lower b-value.

5 Conclusion

The results of this work show that the proposed 3D CNN approach is able to reconstruct a quantization-free fODF and outperforms the CSD approach, if less than 45 gradient directions are available. In addition, we show that the CNN can be further improved by including additional information from neighboring voxels. The best performance is reached by including the 18-neighborhood (green + blue in Fig. 1). For 15 gradient directions, this neighborhood improves the surface deviation from 18.89% to 13.12%, while the angular error improves from 3.43° to 2.87° for a single fiber case, from 10.87° to 6.80° for a two fiber case and from 18.06° to 16.01° in case of three crossing fibers in a voxel. On the other hand, we showed that including a neighborhood may lead to a decreased performance, if 45 and more gradient directions are available. In this case, the acquired signals contain enough information to reconstruct the fODF, while neighboring signals only add noise. Considering the applicability in a clinical scenario, it has to be taken into account that the trained CNN cannot be applied to data acquired with another scanner or scanner protocol, e.g. with different b-value.

In future work, we will investigate the behavior across different scanner types and acquisition protocols. Moreover, we will investigate a multi-shell approach, containing multiple b-values, in order to collect more information about the correct fODF from each voxel. In addition, we will investigate the impact of noise on the reconstruction.

Acknowledgements This work was supported by the International Research Training Group (IRTG 2150) of the German Research Foundation (DFG).

Data were provided by the Human Connectome Project, WU-Minn Consortium (Principal Investigators: David Van Essen and Kamil Ugurbil; 1U54 MH091657) funded by the 16 NIH Institutes and Centers that support the NIH Blueprint for Neuroscience Research; and by the McDonnell Center for Systems Neuroscience at Washington University.

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X.: TensorFlow: Large-scale machine learning on heterogeneous systems (2015). http://tensorflow.org/. Software available from tensorflow.org
2. Alexander, D.C., Zikic, D., Zhang, J., Zhang, H., Criminisi, A.: Image Quality Transfer via Random Forest Regression: Applications in Diffusion MRI, pp. 225–232. Springer International Publishing, Cham (2014)
3. Garyfallidis, E., Brett, M., Amirbekian, B., Rokem, A., Van Der Walt, S., Descoteaux, M., Nimmo-Smith, I.: Dipy, a library for the analysis of diffusion MRI data. Front. Neuroinform. **8**(8), 1–8 (2014)
4. Golkov, V., Dosovitskiy, A., Sämann, P., Sperl, J., Sprenger, T., Czisch, M., Menzel, M., Gómez, P., Haase, A., Brox, T., Cremers, D.: q-Space deep learning for twelve-fold shorter and model-free diffusion MRI scans. In: MICCAI, pp. 37–44. Springer International Publishing, Cham (2015)
5. Jeurissen, B., Leemans, A., Tournier, J.D., Jones, D.K., Sijbers, J.: Investigating the prevalence of complex fiber configurations in white matter tissue with diffusion magnetic resonance imaging. Hum. Brain. Mapp. **34**(11), 2747–2766 (2013)
6. Koppers, S., Haarburger, C., Merhof, D.: Diffusion MRI signal augmentation - from single shell to multi shell with deep learning. In: MICCAI Workshop on Computational Diffusion MRI. Springer International Publishing, Cham (2016)
7. Koppers, S., Merhof, D.: Direct estimation of fiber orientations using deep learning in diffusion imaging. In: MICCAI Workshop on Machine Learning in Medical Imaging. Springer International Publishing, Cham (2016)
8. Koppers, S., Merhof, D.: Qualitative Comparison of Reconstruction Algorithms for Diffusion Imaging, chap. 4, pp. 51–67. Der Andere Verlag, Uelvesbüll (2016)
9. Schultz, T.: Learning a reliable estimate of the number of fiber directions in diffusion MRI. In: Ayache, N., Delingette, H., Golland, P., Mori, K. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012: 15th International Conference, Nice, October 1–5, 2012, Proceedings, Part III, pp. 493–500. Springer, Berlin, Heidelberg (2012)
10. Schultz, T., Westin, C.F., Kindlmann, G.: Multi-diffusion-tensor fitting via spherical deconvolution: a unifying framework. In: Jiang, T., Navab, N., Pluim, J.P.W., Viergever, M.A. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2010: 13th International Conference, Beijing, September 20–24, 2010, Proceedings, Part I, pp. 674–681. Springer, Berlin, Heidelberg (2010)
11. Tournier, J.D., Calamante, F., Connelly, A.: Robust determination of the fibre orientation distribution in diffusion MRI: non-negativity constrained super-resolved spherical deconvolution. NeuroImage **35**(4), 1459–1472 (2007). doi:10.1016/j.neuroimage.2007.02.016
12. Tournier, J.D., Calamante, F., Gadian, D.G., Connelly, A.: Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. NeuroImage **23**(3), 1176–1185 (2004). doi:10.1016/j.neuroimage.2004.07.037

Index

© Springer International Publishing AG 2017

T. Schultz et al. (eds.), *Modeling, Analysis, and Visualization of Anisotropy*,
Mathematics and Visualization, DOI 10.1007/978-3-319-61358-1