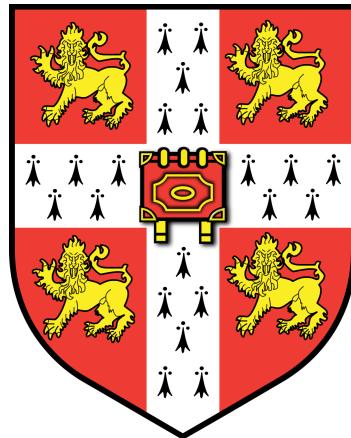


TOWARDS AN ACCURATE BRAIN TRACTOGRAPHY



ELEFTHERIOS GARYFALLIDIS

Wolfson College
University of Cambridge
United Kingdom

May 2012

A dissertation submitted to the University of Cambridge for the degree
of Doctor of Philosophy

Supervisors

DR. IAN NIMMO-SMITH

Medical Research Council
Cognition and Brain Sciences Unit
Cambridge, UK

DR. GUY B. WILLIAMS

Wolfson Brain Imaging Centre
Department of Clinical Neurosciences
University of Cambridge
Cambridge, UK

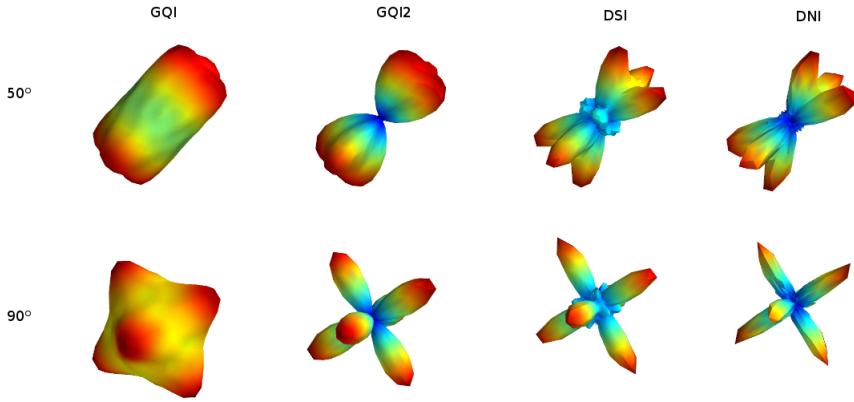


Figure 2.1: Showing the ODFs from two randomly oriented simulated 3-fibre crossings at 50° (top) and 90° angles between each pair of fibres using different Cartesian lattice q-space reconstruction methods.

2.3 Other methods

Pickalov et al. [76] proposed a new method for reconstructing the diffusion propagator by applying an iterative inverse Radon transform on measurements along many radial lines; computing 1D tomographic projections to reconstruct the 3D EAP. This technique measures DW images along a few radial lines of q-space but still requires hundreds of samples to reliably recover the EAP. Currently, to reconstruct the EAP, the state-of-the-art model-free techniques apart from diffusion spectrum imaging are hybrid diffusion imaging (HYDI) [77] and multiple q-shell diffusion propagator imaging (mq-DPI) [78]. HYDI acquires the signal values on five concentric spherical q-space shells, then interpolates onto a cubic grid and applies the standard Fourier transform in the same way as DSI. In mq-DPI the EAP is calculated by solving Laplace's equation for the diffusion signal using a real and symmetric modified spherical harmonic basis. The EAP can be found analytically by the inversion of a linear system using Laplace-Beltrami regularization. In addition, Exact Q-ball imaging (EQBI) [75] provides a different method to calculate the ODF analytically using multiple spherical q-space shells. Similarly, Aganj et al. [79], proposed an analytical solution for the multi-shell case by incorporating a mono-exponential or bi-exponential model (CSA-ODF). Another method for finding a directional distribution on the sphere was proposed by Özarslan et al. [22] called the diffusion orientation transform (DOT). This method calculates a different statistic $P(r_0\hat{u})$, the probability of finding a particle initially at the origin at the point $r_0\hat{u}$, using spherical

harmonics. Not surprisingly there is a relationship connecting CSA with DOT which is

$$\psi_{CSA}(\hat{\mathbf{u}}) = \int_0^\infty DOT(r\hat{\mathbf{u}})r^2 dr \quad (2.17)$$

Jansons et al. [80] proposed a different function on the sphere than the ODFs described above, to be used on data sets acquired on a single spherical q-space shell. They called this spherical function persistent angular structure (PAS). This method has very good angular resolution. It uses the principle of maximum entropy however, it is rather slow as it nonlinear fitting is used in order to identify many parameters. PAS is a statistic on the sphere defined as $PAS(\hat{\mathbf{u}}) = \exp(\lambda_0 + \sum_{j=1}^N \lambda_j \cos(\mathbf{q}_j \cdot k\hat{\mathbf{u}}))$ where λ are the unknown parameters, k is constant and N is the number of DWIs. The relationship $\int PAS(\hat{\mathbf{u}}) \exp(i\mathbf{q}_j \cdot k\hat{\mathbf{u}}) d\hat{\mathbf{u}} = E(\mathbf{q}_j)$ provides the bridge between PAS and the diffusion signal ($E(\mathbf{q})$).

The first publication of using spherical harmonic expansions with diffusivity profiles, which are now quite common in the literature, was by Alexander et al. [81]. Q-ball imaging was introduced by Tuch [32] and a new ODF defined as $\psi(\hat{\mathbf{u}}) = \frac{1}{Z} \int_0^\infty P(r\hat{\mathbf{u}}) dr$ where Z is a normalization constant. It was later provided for Q-Ball imaging a fast and analytical solution using spherical harmonics (SH) and Laplace-Beltrami regularization [82]. Tournier et al. [83], [84] introduced a spherical deconvolution method where first the SH coefficients were estimated, then single fiber ODFs were used as a deconvolution kernel estimated from the real data. Then, the sharper fODF (fiber orientation distribution function) was obtained by a simple linear transformation [85]. Other deconvolution approaches were proposed in [86] and [87].

On Tensor related methods we have the classical Single Tensor [67], Sticks and Ball[35], Multi-Tensor [88], [89] and Higher Rank Tensors [70], [71]. In addition there are also model based methods which try to calculate non-Gaussian properties, for example the Kurtosis Tensor [23], [90] which is used in Diffusion Kurtosis Imaging (DKI).

Finally, new model-based methods are emerging which are trying to calculate statistics like the axonal thickness distribution from dMRI data sets. These are usually based on model free and restricted components; CHARMED [91], [92], AxCaliber [93] and the orientation invariant ActiveAx [94] are some well known methods of this type. Q-space Imaging

(QSI) can be used to identify distributions of axon-diameter too [95].

2.4 Diffusion Nabla Imaging

A new method for the calculation of the real ODF is proposed here. This is based on the theoretical work done by Aganj et al. [96] and Canales-Rodriguez et al. [16] using two important theorems from Fourier Analysis

1. The Fourier transform of $P(\mathbf{r})r^2 = -\nabla^2 E(\mathbf{q})$ where ∇^2 is the Laplacian operator (for proof see section A.3).
2. For a symmetric function $E : \mathbb{R}^3 \rightarrow \mathbb{R}$ and for the arbitrary unit vector $\hat{\mathbf{u}}$ we have $\int_0^\infty E(r\hat{\mathbf{u}})dr = \frac{1}{8\pi^2} \int \int_{\hat{\mathbf{u}}^\perp} E(q) q dq d\phi$ where $\hat{\mathbf{u}}^\perp$ is the plane perpendicular to $\hat{\mathbf{u}}$ (for proof see section A.4).

From Eq. 2.7 we see that the integration is over $P(r\hat{\mathbf{u}})r^2$, therefore we can write

$$\psi_{DNI}(\hat{\mathbf{u}}) = -\frac{1}{8\pi^2} \int_{\hat{\mathbf{u}}^\perp} \int_0^\infty \nabla^2 E(q) q dq d\phi \quad (2.18)$$

where ϕ is the angular rotation component operating on the plane perpendicular to $\hat{\mathbf{u}}$, ∇^2 is the Laplacian operator and $E(q) = S(q)/S_0$ is the normalized diffusion signal. Eq. 2.18 has the advantage that no Fourier transform is necessary. We need however, a way to calculate the Laplacian of the signal. This can be analytically derived for a spherical grid [96] and we propose here that it can be directly calculated in a cubic grid using the standard 3D discrete Laplacian filter which is given by the 3D kernel defined by the following $3 \times 3 \times 3$ array:

$$\left[\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 1 & -6 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right].$$

This is a filter commonly used for image processing. From now on when we use the Laplacian operator in order to measure the directionality of the diffusion signal we will call this reconstruction method Diffusion Nabla Imaging as nabla-squared (∇^2) is the symbol for the Laplacian operator. In Fig. 2.1 we present the ODFs from two randomly oriented simulated 3-fibre crossings at 50° and 90° angles between each other using different grid based reconstruction methods. The parameters used here are for DSI:

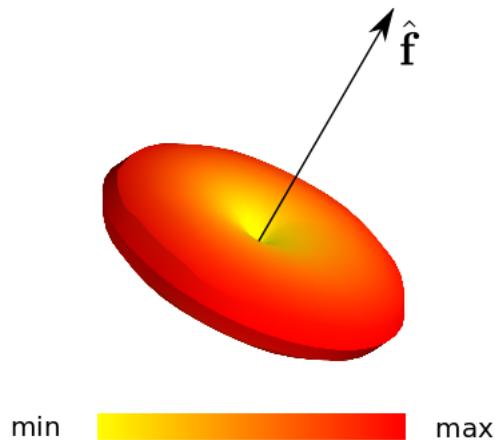


Figure 2.2: The diffusion signal has the beautiful property that it is minimum along the direction of a fibre with unit direction \hat{f} and maximum along the equator defined by the plane perpendicular to that fibre direction. This property is the basic inspiration behind the EIT. In this picture the 3D surface plot of a simulated signal for a spherical grid acquisition with b-value 2,000 is shown using a yellow-red colourmap.

radial sampling range 2.1 – 6 with 0.2 and Hanning filter width 36, GQI: $\lambda=1.2$, GQI2: $\lambda = 3$ and DNI: radial sampling 0 – 5 with 0.2 steps. All methods used the same reconstruction sphere with 642 vertices and 1,280 faces.

2.5 Equatorial Inversion Transform

We propose an important theoretical construction called the Equatorial Inversion Transform (EIT) which creates a general formulation for the interpretation of the directionality of the diffusion signal. This idea is founded on two general properties of the diffusion signal: (a) If we visualize the diffusion signal for a single fibre for all gradient directions we see the generated shape to be lowest towards the direction of the fibre and highest on the plane perpendicular to that direction (see Fig. 2.2). (b) The diffusion signal is additive i.e. $S(\hat{f}_1) + S(\hat{f}_2) = S(\hat{f}_1 + \hat{f}_2)$, where \hat{f}_1, \hat{f}_2 are the unit directions of the fibres. In simple terms the signal of 2-fibre crossing can be decomposed linearly to the signals of the two fibres that create the crossing. The same holds for any number of fibres in a crossing.

These are two very important geometric properties of the signal that we can try to exploit to its limit by calculating equatorial integrals in order to identify the directionality of the signal.

Apart from the visual confirmation, further supporting evidence that equatorial integration is crucial for derivation of directionality can be seen in Eq. 2.18 where an equatorial integral creates a connection between the real ODF and the signal. The Funk-Radon Transform (FRT) used by [32] is another example where equatorial integration is employed using the reconstruction sphere. We will see next that DNI and FRT are just a subset of the EIT.

With EIT the most important goal is to try to identify the orientational variation in the signal in the most accurate way by generating a spherical density. However, it is possible to calculate additionally the classical ODF as defined by Wedeen et al. [73].

The EIT shown in Eq. 2.19 consists of an integration along the equator and along radial lines. A function F of the signal is multiplied by a radial weighting function O . This construction is a generalization of the previous ODFs and it can support successfully many different function families for F and O which can all more or less accurately identify the directional distribution of the signal. More precisely the EIT is defined as

$$\psi_{EIT}(\hat{\mathbf{u}}) = \int_{\hat{\mathbf{u}}^\perp} \int_0^\infty F(E(q))O(q)dq d\phi \quad (2.19)$$

where F could be for example any of the following functions

$$F(E(q)) = \begin{cases} E(q) & (I) \\ -\nabla^2(E(q)) & (II) \\ \nabla^4(E(q)) & (III) \\ \dots \end{cases} \quad (2.20)$$

and O could be for example any of the following functions

$$O(q) = \begin{cases} 1 & (0) \\ q & (1) \\ q^2 & (2) \\ \dots \end{cases} \quad (2.21)$$

In Tab. 2.1 we see that by choosing different functions for F or O we can generate both old and new distribution functions on the sphere. With

F	O	Name	Comment
$-\nabla^2(E(q))$	q	DNI≡EITL	calculates the real ODF without the complications of the FFT
$\nabla^4(E(q))$	q	EITL2	high resolution at low angles
$E(q)$	q	EITS	impressive resolution without any preprocessing of the signal
$E(q)$	1	'QBI'-like	similar to the Funk Radon Transform

Table 2.1: The Equatorial inversion transform (EIT) can be used to explain many other reconstructions algorithms.

$F(E(q)) = -\nabla^2(E(q))$ and $O(q) = q$ we can generate ψ_{DNI} which is theoretically identical to the DSI real ODF(ψ_{DSI}). If $F(E(q)) = E(q)$ and $O(q) = 1$ then this is similar to the Funk Radon Transform (used in Q-ball imaging) but applied to multiple spherical shells. However, we can also try to use different functions like $F(E(q)) = -\nabla^4(E(q))$ and $O(q) = q$ which can potentially increase the amount of directional information beyond that of the standard ODFs. Before starting investigating the realms of EIT we will first give a short overview of other methods commonly found in the literature.

2.6 Implementation

2.6.1 Standard EIT

Eq. 2.18 and 2.19 can be implemented in a standard way by evaluating the 3D signal on the grid multiple times for every direction \hat{u} as shown in Fig. 2.3A. This suggests that if for example we use a reconstruction sphere of 642 vertices and the signal is centered inside a cubic grid of size $17 \times 17 \times 17$ where the radial integration (q) takes place in 30 steps and the equatorial (ϕ) in 63 steps, then we need to interpolate $642 \times 30 \times 30 \simeq 1.2$ million times on the cubic grid. For this reason we invented Fast EIT, a new method that needs an order of magnitude less evaluations.

In this document whenever we use the prefix 's' in front of a method it means it was calculated with the standard EIT algorithm. For example if standard EIT is used for DNI we will write sDNI or sEITL. ('L' stands for 'Laplacian' or 'Nabla'). Of course, sDNI and sEITL are equivalent.

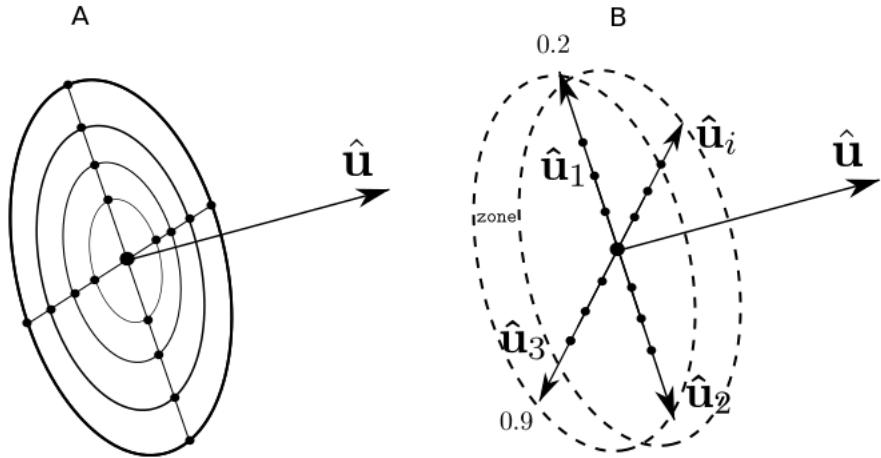


Figure 2.3: A: Standard EIT. B: Fast EIT. Fast EIT is an order of magnitude faster than standard EIT. The key idea here is that we can reduce computations by storing the sum of the radial integrals for every vertex in the reconstruction sphere and then we can also precompute the indices of the vertices that are near the equator of every vertex (inside an equatorial zone).

2.6.2 Fast EIT

A much faster algorithm than the standard EIT is described here. The main idea is that we can store the sum of the radial integrals for every vertex in the reconstruction sphere so we can then precompute the indices of the vertices that are near the equator of every vertex (inside an equatorial zone). This becomes clear in Fig. 2.3B. Following these calculations, the spherical distribution function can be approximated with much less operations. The full algorithm is given in Alg. 1. The input is the vertices $\hat{\mathbf{u}}_i$ of the reconstruction sphere and the normalized signal E . Then, for every point of the reconstruction sphere $\hat{\mathbf{u}}_i$, we save the indices of the vertices j of $\hat{\mathbf{u}}_j$, which are inside an equatorial zone, in list J_i . The width of the equatorial zone z is a constant set empirically to 5° . If a very highly dense reconstruction sphere is used with more than 642 vertices, which is the one we used, then the zone can be smaller. That can potentially increase the angular resolution of the method.

At the next stage we calculate sums along every radius on the direction of $\hat{\mathbf{u}}_i$ in the following way: $\mathbb{B}(\hat{\mathbf{u}}_i) = \sum_{k=0}^n F(E(q_k \hat{\mathbf{u}}_i)) O(q_k \hat{\mathbf{u}}_i)$ and obtain the final EIT ODF as the average of the sums in the equator $\psi_{EIT}(\hat{\mathbf{u}}_i) = \frac{1}{N_i} \sum_{j \in J_i} \mathbb{B}(\hat{\mathbf{u}}_j)$ where F is evaluated with trilinear for example interpolation on the lattice and N_i is the number of indices in J_i .

Algorithm 1 Fast Equatorial Inversion Transform

Input $U = \{\hat{u}_1 \dots \hat{u}_m\}$, E**Output** ψ_{EIT} **For** \hat{u}_i in U **Do**

$$J_i = \{j : |\arccos(\hat{u}_i \cdot \hat{u}_j)| \leq z\}$$

$$\mathbb{B}(\hat{u}_i) = \sum_{k=0}^n F(E(q_k \hat{u}_i)) O(q_k \hat{u}_i)$$

where $F(E(q_k \hat{u}_i))$ is interpolated on the lattice.**EndFor****For** \hat{u}_i in U **Do**

$$\psi_{EIT}(\hat{u}_i) = \frac{1}{N_i} \sum_{j \in J_i} \mathbb{B}(\hat{u}_j)$$

where N_i is the number of indices in J_i .**EndFor**

In section 2.9.1 the standard EITL (sEITL) is compared with fast EITL. In Fig. 2.11 it is shown that the fast EIT has very similar results with the standard EIT. From now on whenever we write EIT we refer to the fast version.

2.7 Peak Finding

After we have generated the ODFs we need to find the peaks (local maxima) from which we can easily approximate the direction of the fibres. Peak finding is not trivial if there are many local maxima in the ODFs or the ODFs are noisy. Here we present an algorithm (see Alg. 2) which reduces the amount of small local variations and returns a number of sorted peaks and their indices in the reconstruction sphere. The input of this algorithm is ψ (ODF) and the faces of a symmetric on the z-axis evenly distributed sphere (see Fig. 2.4C).

We have used a triangulation of the unit sphere (which we refer to simply as 'sphere') obtained by triangular subdivision of a regular icosahedron. It is symmetric over the z-axis, i.e. for each vertex (x, y, z) there is a corresponding vertex $(x, y, -z)$ in the opposite hemisphere. The same sphere was used in [72] for GQI reconstructions. Every face (triangle) corresponds to a list of the 3 indices of the 3 vertices on the sphere. The idea here is that we can travel from face to face and nullify all points on a face which are lower than the higher value of the face. At the end only local maxima will survive the procedure. The algorithm is presented in detail in Alg. 2.

Algorithm 2 Peak Finding with a Symmetric Ordered Sphere

Input ODF ψ , faces Φ
Output peaks P and indices I

```
For face  $\Phi_i$  in  $\Phi$  Do
     $f_0, f_1, f_2 = \Phi_i$ 
     $d_0, d_1, d_2 = \psi[f_0], \psi[f_1], \psi[f_2]$ 
    If  $d_0 \geq d_1$  and  $d_2$  Do
         $P[f_1] = P[f_2] = 0$ 
        continue
    If  $d_1 \geq d_0$  and  $d_2$  Do
         $P[f_0] = P[f_2] = 0$ 
        continue
    If  $d_2 \geq d_0$  and  $d_1$  Do
         $P[f_0] = P[f_1] = 0$ 
        continue
    EndIf
EndFor
```

The sphere we use is of course discrete and therefore it adds some constraints on the angular resolution (worst case $\pm 4.96^\circ$) of the peaks found from the ODF. In addition, the proposed Peak Finding algorithm can reduce slightly more the angular resolution. For example, in Fig. 2.4A, B we show that if point **a** was a local maxima then only points **b** could be alternative local maxima for **a** but none of the unlabeled points could be a second peak. Nevertheless, we found Alg. 2 to be extremely useful and fast. The same algorithm was used also in [72] but it was not documented as such.

2.8 Spherical Angular Smoothing

All current non-parametric dMRI reconstruction algorithms use some type of “smoothing” to reduce the effect of noise in the real data sets. DSI uses a Hanning filter and then avoids sampling from low values in \mathbf{r} -space. In GQI, smoothing is controlled from a scalar parameter; the diffusion sampling length and in spherical harmonic inversion methods [97], [96] the amount of smoothing is controlled by using only a number of the first components of the SH series.

All these approaches smooth and calculate the ODFs simultaneously. Our approach differs in that we propose that the ODF is first calculated and

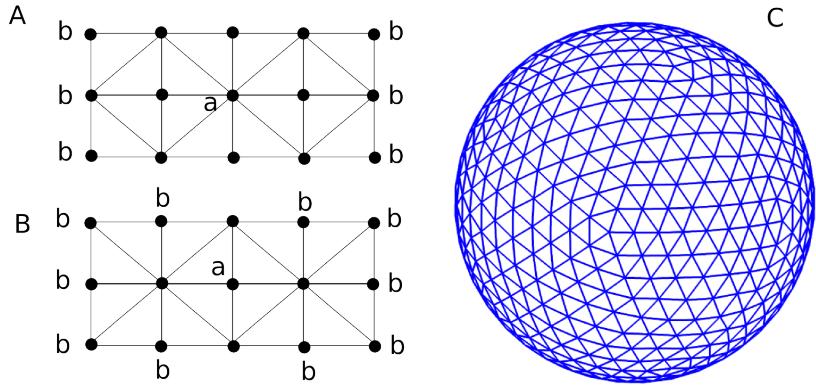


Figure 2.4: A and B: Each point **a** is a local maximum for all its neighboring faces, then only at **b** are other possible local maxima. This simple illustration shows that the triangulation of the sphere is important for the determination of closed peaks and that peaks which belong to the same triangle cannot be determined. C: the sphere used for ODF reconstructions consisting of 642 vertices and 1,280 faces produced by subdivisions of the icosahedron.

then smoothed. For example, using the operator shown below in matrix form

$$W = \exp\left(\frac{U \cdot U^T}{\sigma}\right)$$

where U is the an $N \times 3$ matrix holding the N points of the ODF reconstruction sphere and σ is a smoothing parameter acting like the variance. At the next step we can smooth any $ODF(\psi)$ creating a new $ODF(\psi')$ in the following way

$$\psi' = \psi \cdot \frac{W}{\sum_j W_j} \quad (2.22)$$

where j denotes row indexing, $\sum_j W_j$ acts as a normalization for the angular weighting W , ψ is the initial ODF and ψ' is the smoothed ODF. The advantage of this method is that it is more comprehensive and direct. It also uses information from all directions simultaneously. Similar operators can be constructed that weight more lower or higher peaks. The operator shown here weighs more peaks that are closer in angular distance. In Fig. 2.5 we see the effect of this equation on a simulated triple-fibre crossing; distorted with Gaussian noise with SNR 20 and reconstructed as an EITL density function. The simulation was used using a Sticks and Ball model with diffusivity value 0.0015 and S0=100.

We can easily observe that when we increase the smoothing factor σ ,

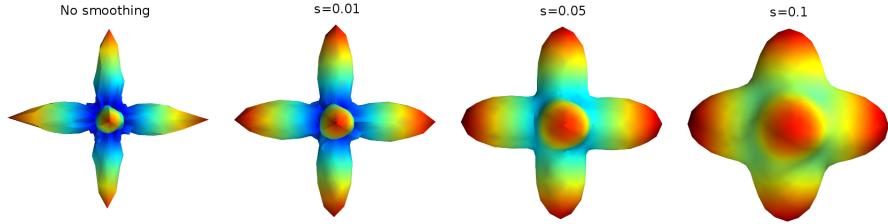


Figure 2.5: An example of spherical angular Gaussian smoothing applied with different smoothing factors on the ODF of a triple-fibre crossing on the left.

small noisy peaks, as seen in the center of the unsmoothed spherical function, can easily be removed. However, with too much smoothing even the strongest peaks can lose their definition. This spherical operator can help to set the trade-off between noise and signal and it can also simplify the peak finding process, i.e. finding the underlying primary fibre directions as this problem is much easier on smooth surfaces.

Finally, decoupling the smoothing from the reconstruction step gives an important advantage: reducing the effect of the noise to our data more strongly and independently. Many spherical operators can be added as plugins independently of the reconstruction phase, and these can work with any function on the sphere (see Eq. 2.22).

2.9 Comparisons and Results

Validation of reconstruction and tractography algorithms is not straightforward due to the lack of relevant gold standards. Simulated voxels and software phantoms are a useful way to overcome this difficulty and test new methods. Following the simulation results, we show results with real human data sets.

2.9.1 Multi-fibre Simulations

For single voxel simulations we used the model proposed in Behrens et al. [35]; the multi-compartment model also known as Sticks and Ball which simulates the diffusion signal as

$$S_i = S_0 \left((1 - \sum_{j=1}^N f_j) \exp(-b_i d) + \sum_{j=1}^N f_j \exp(-b_i d \cos(\theta_{ij})^2) \right) \quad (2.23)$$

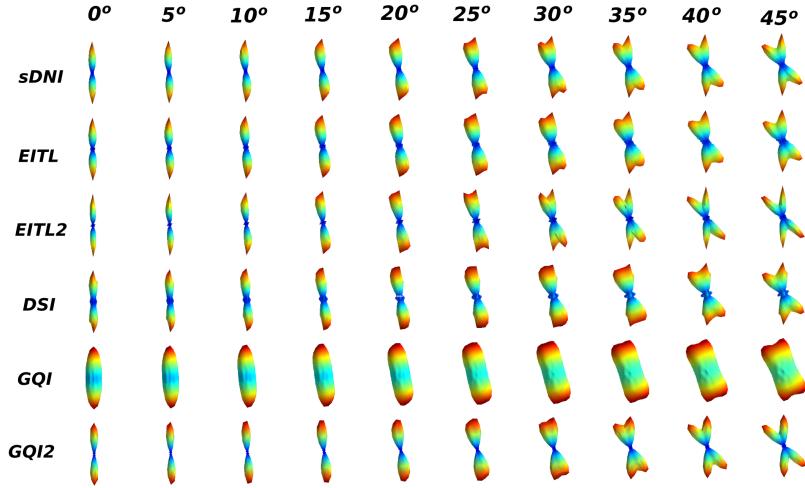


Figure 2.6: Visualizing ODFs created from different reconstruction methods sDNI (sEITL), EITL, EITL2, DSI, GQI and GQI2. We can see that standard DNI (sDNI), EITL and EITL2 can resolve the correct angular fibre directions at lower angles than the other methods. For example see column at angle of 25° .

where θ_{ij} is the angle between gradient direction $\hat{\mathbf{g}}_i$ and fibre (stick) unit direction $\hat{\mathbf{u}}_j$. The amount of representation for every fibre is given by f and d is the diffusivity value for the entire model. A Multi Tensor [89] approach was also created for software phantoms using the formula

$$S_i = S_0 \sum_{j=1}^N \exp(-b\hat{\mathbf{g}}^T D_j \hat{\mathbf{g}}) \quad (2.24)$$

where D_j is the diffusion tensor for every fibre j .

In Fig. 2.6 we present the outcome of an experiment of two crossing fibres using different reconstruction methods: sDNI (sEITL), EITL, EITL2, DSI, GQI and GQI2. These are based on simulations of 2-fibre crossings from 0° to 90° using Eq. 2.23 with diffusivity value of $1.5 \times 10^{-3} \text{ mm}^2/\text{sec}$ and 257 b-values with maximum b-value 11,000. However, all these methods will perform accurately beyond 50° therefore in this figure we present only the lower angles. We can see that sDNI, EITL and EITL2 performed better than the other methods. Especially EITL2 was able to resolve crossing at 25° which is lower than the accuracy resolved in the current state of the art methods. In order to confirm this fascinating result we created

Known	Measured	AS
$(1, 0, 0), (0, 1, 0)$	$(0, 0, 1)$	0
$(1, 0, 0), (0, 1, 0)$	$(0, 1, 0)$	1
$(1, 0, 0), (0, 1, 0)$	$(0, \sqrt{2}/2, \sqrt{2}/2)$	$\sqrt{2}/2$
$(1, 0, 0), (0, 1, 0), (0, 0, 1)$	$(1, 0, 0), (0, 0, 1)$	2

Table 2.2: Examples of angular similarity (AS) behaviour with simple unit vector sets.

a more general experiment with many iterations (of 2-fibre and 3-fibre crossings) which is presented below.

A comparison metric is needed in order to evaluate the new and old reconstruction methods discussed in this chapter. The standard procedure is to calculate the similarity between the measured and simulated ground truth data sets. We want to calculate the angular precision of the ODFs from simulations derived from Eq. 2.23. We define a new similarity metric called Angular Similarity (AS) which computes the cosine distance of the best match between the set of measured fibre orientations and the known set of simulated fibres. This metric will be used to compare 2-fibre and 3-fibre crossings. AS is 0 when there is no match i.e. angular distance is 0, 1 when one fibre is matched (0°), 2 when two fibres are matched and 3 when three fibres are matched. In table 2.2 we show a few examples of AS behaviour with simple unit vector sets.

If our ground truth set consists of $g = [(1, 0, 0), (0, 1, 0)] = [g_0, g_1]$ and the measured set consists of $m = [(0, 0, 1)]$ then AS=0. If the measured set was $m = [(0, \sqrt{2}/2, \sqrt{2}/2)] = [m_0]$ then AS is $\sqrt{2}/2$. This is because according to the AS definition we have $AS(g, m) = \max(|g_0 \cdot m_0|, |g_1 \cdot m_0|)$. Which is equal to $\sqrt{2}/2$.

If $g = [(1, 0, 0), (0, 1, 0)] = [g_0, g_1]$ and $m = g$ then $AS(g, m) = \max(|g_0 \cdot m_0| + |g_1 \cdot m_1|, |g_0 \cdot m_1| + |g_1 \cdot m_0|) = 2$. We created an experiment where we set two fibres at an increasing angle of 2.5° from 0° to 90° and then rotate them uniformly around 200 random axes. This operation produces 7,400 simulated ODFs and the results are shown in Fig. 2.7 and Fig. 2.8 with different signal to noise ratio. For these simulations noise was normally distributed. What we see in the figures is the average angular similarity where the average is calculated from the 200 random orientations for the same angle.

We can easily observe in Fig. 2.7 that EITL2 can resolve more accurately

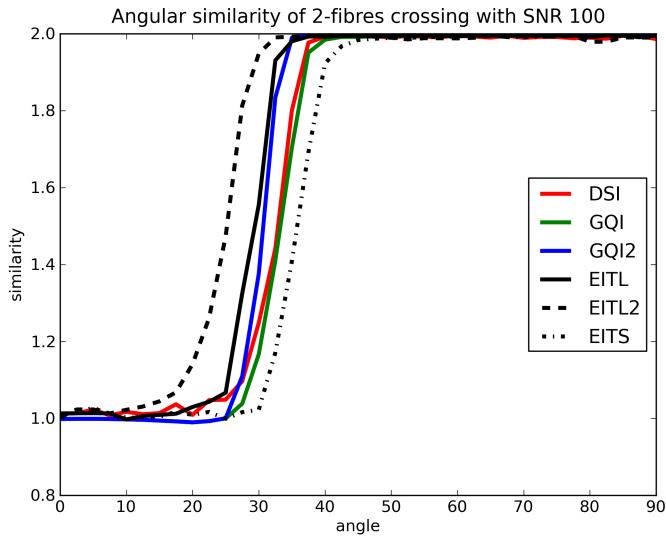


Figure 2.7: Average angular similarity of 2-fibre crossings with SNR 100

fibre crossings at low angles and continues to perform well even at higher angles $> 50^\circ$. EITL performs better than DSI, GQI, GQI2 and EITS at low angles and very well at high angles as well. GQI2 performs better than DSI, GQI, and ETS. It is also impressive that EITS can have such a good performance although it is such a simple operation. In summary we see from the graphs that EITL2 > EITL > GQI2 > DSI > GQI > EITS where $>$ means higher average angular similarity. The same pattern takes place even when we increase the noise level see for example Fig. 2.8. We will see next that the same pattern appears even with 3-fibre crossings and high levels of noise.

We also measured the accuracy in 3-fibre crossings. In this experiment the 3-fibres will always have the same angular distance between each other. That distance will increase from 0° to 90° with steps of 2.3° on average and all 3 fibres will be reoriented 200 times. That gave 8,000 simulated crossings.

The results of the 3-fibre crossings shown in Fig. 2.9 and Fig. 2.10 were very similar to those of the 2-fibre crossings; EITL2 performed better at low angles, showing slightly reduced performance at high angles. EITL performed better with low angles than the rest of the methods, having also high accuracy on larger angles.

These summarization plots give strong evidence that both DNI (EITL) and in general EIT can be used to accurately generate spherical distri-

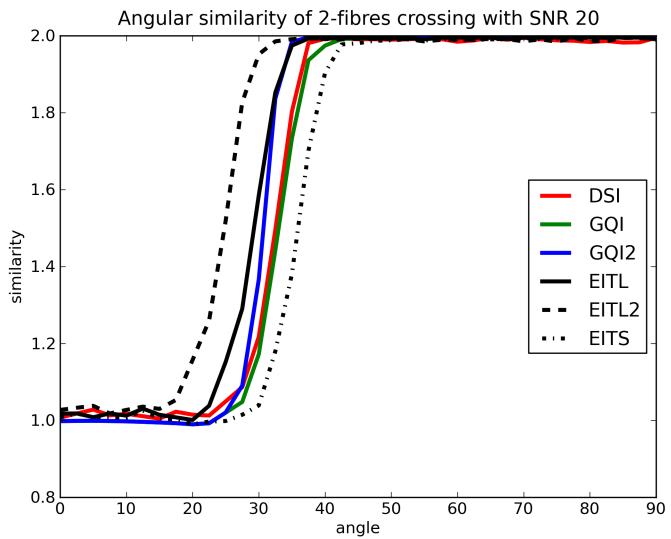


Figure 2.8: Average angular similarity of 2-fibre crossings with SNR 20.

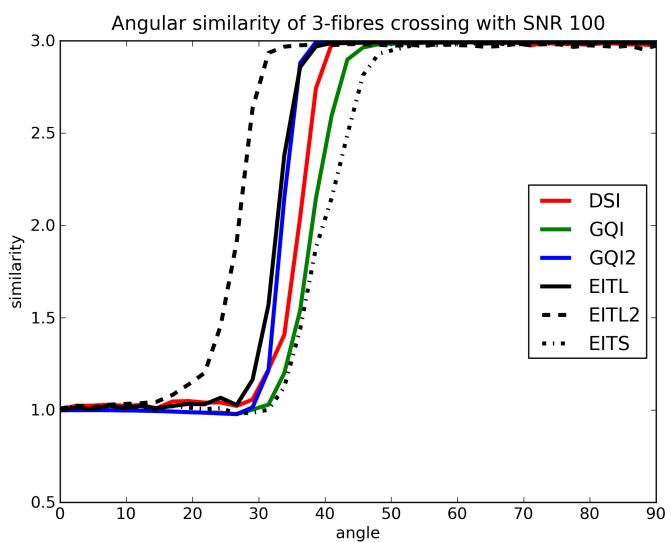


Figure 2.9: Average angular similarity of 3-fibre crossings with SNR 100.

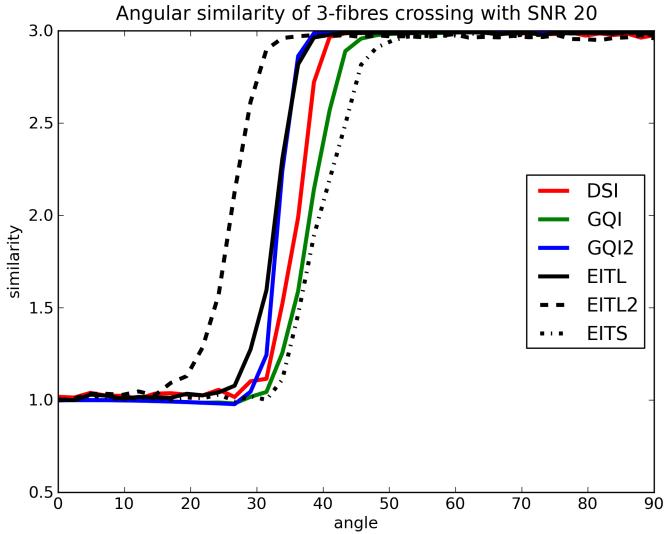


Figure 2.10: Average angular similarity of 3-fibre crossings with SNR 20.

bution functions for the determination of the directional information of the diffusion signal. They performed equivalently or better than the current state-of-the-art grid-based reconstruction methods i.e DSI and GQI. The determination of the fibre directions was not affected considerably by noise.

Furthermore, we can also see that GQI2 can do better than DSI, GQI and that EITS gives results that are very similar to GQI. The parameters used for these simulations were DSI: radial sampling 2.1 – 6, hanning filter width: 36, GQI: $\lambda=1.2$, GQI2: $\lambda = 3$, and EITS, EITL, EITL2 were all calculated with the standard options zonal width ($z = 5^\circ$), grid size $17 \times 17 \times 17$, radial sampling 0 – 5 with 0.1 steps and no further post-processing or smoothing was used. All methods were using the same reconstruction sphere with 642 vertices and 1,280 faces.

In these tests, EIT and fast EIT produced very similar results. For example, a simple test for the 3-fibre case as seen in Fig. 2.11 shows that there is close agreement between the two methods i.e. their results are nearly equivalent. We can therefore conclude that the fast EIT is an acceptable approximation of the standard EIT.

2.9.2 Software Phantoms

A software phantom generation tool was developed which can simulate the diffusion weighted signal for one or more fibres represented by differ-

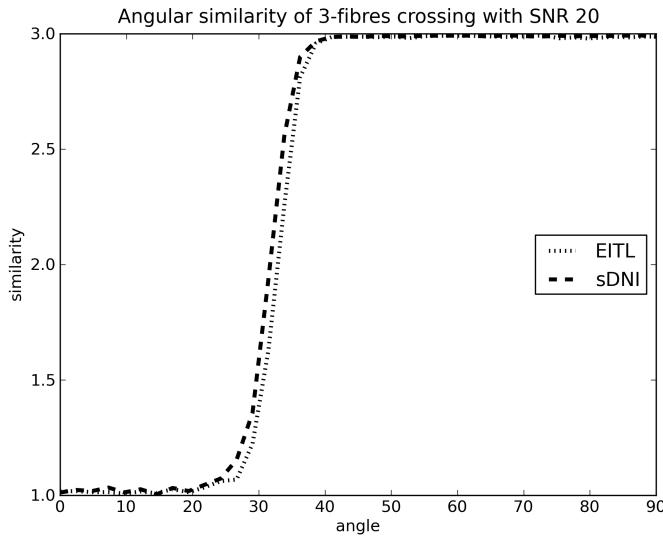


Figure 2.11: This diagram shows that when we compute EITL with the fast or standard method the results are nearly equivalent. The mean angular similarity for the case of 3-fibres crossings is very similar when using standard DNI or fast DNI (EITL).

ent discrete 3D orbital functions. This work is an extension of the phantom developed by Correia et al. [98] which supported only paths with analytically calculated derivatives.

The idea here is that we first create any orbital function $f(t) : \mathbb{R} \rightarrow \mathbb{R}^3$ and calculate numerically its derivatives at small steps Δt . We can then scale it and centre it so that it fits in an image volume of the desired size. We expect that many segments of the discrete function f will fall into every voxel in the volume and that more curved parts of f will have higher representation in the voxel than less curved parts. For every segment, we can find the main direction of the orbit $\mathbf{v} = \frac{f(t+1) - f(t)}{\Delta t}$ and calculate the rotation matrix \mathbf{R} that rotates $\hat{\mathbf{x}} = (1, 0, 0)$ to \mathbf{v} . Then, the signal for each element of the fibre for a given b-value b and a given gradient sampling direction $\hat{\mathbf{g}}$, is given by the following Single Tensor formula

$$\Delta S = S_0 \exp(-b\hat{\mathbf{g}}^T \mathbf{R} \Lambda \mathbf{R}^T \hat{\mathbf{g}}) \quad (2.25)$$

where S_0 is the unattenuated signal of the fibre, and the diffusion tensor is given by

$$\boldsymbol{\Lambda} = \begin{pmatrix} \lambda_{\parallel} & 0 & 0 \\ 0 & \lambda_{\perp} & 0 \\ 0 & 0 & \lambda_{\perp} \end{pmatrix} \quad (2.26)$$

Therefore, the total signal of the voxel for one gradient direction is given by the summations of all the contributions of the K elements in the voxel

$$S_{vox} = \sum_{i=1}^K \Delta S_i \quad (2.27)$$

In addition, we can generate simulations of more than one fibre by generating a single volume for every orbit and then add them all together to create complex configurations in the final volume. This is acceptable, under the assumption that the diffusion is Gaussian in all compartments, because the diffusion signal is additive i.e. the signal of a crossing of two fibres is equal to the sum of the signals of the individual fibres. In this way, we can simulate phantoms with Multi Tensor based diffusion signals as that described in Eq. 2.24. We can increase the thickness of the fibres using a typical smoothing kernel or duplicate the fibres radially. At the end we can add different levels of noise e.g. Rician or Gaussian noise with a prespecified SNR.

The method we use to create these software phantoms offers the opportunity to simulate partial volume effects. If partial volume effects are not desired, we need to normalize by dividing by the number of fibre elements for each voxel. In Fig. 2.12 we can see the volume renderings of two different phantoms created with the method described here. This function is implemented in module `dipy.sims.phantom`.

2.9.3 Results with software phantoms

With the purpose of comparing and visualizing the differences between the reconstruction methods described in this chapter a phantom of two crossing bundles was created. The bundles are crossing at an angle of 90° . The phantom was generated using the method described in the previous section. Here we describe the basic steps: (a) We first represented the first bundle as a discrete straight path starting from point $(-1, -1, 0)$ and ending at point $(1, 1, 0)$ with using 1,000 time steps. (b) We scaled, centred

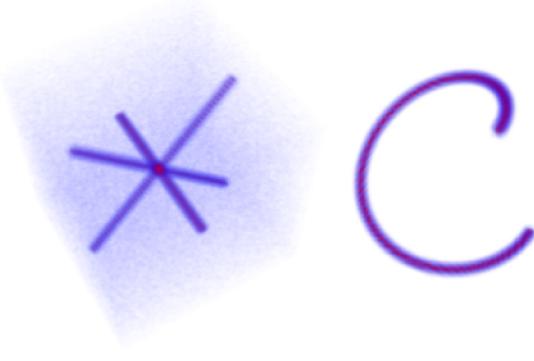


Figure 2.12: Volume renderings of the unattenuated signals of two digital phantoms. On the left 3 fibres intersect on regular angles with Rician noise of SNR=20. On the right a helicoidal fibre is shown clear of noise. For both phantoms $S_0 = 100$ and prolate tensors with eigenvalues $\lambda_{\parallel} = 1.4 \cdot 10^{-3} m^2/sec$ and $\lambda_{\perp} = .35 \cdot 10^{-3} m^2/sec$ were used.

and radially expanded this path so that it fits a volume of size $64 \times 64 \times 64$. This volume corresponds to the diffusion volume without any weighting. (c) We then applied the weightings for all the following volumes corresponding to non-zero b-values. (d) We replicated the same procedure for the other bundle which initially started as an orbit from position $(-1, 1, 0)$ and ended at position $(1, -1, 0)$. (e) We added the two volumes together to create an 'x' shape (see Fig. 2.13, 2.15). (f) We added Rician noise with SNR=5. As in this chapter we concentrate on Cartesian Lattice Q-space acquisitions we generated b-vectors and b-values using a keyhole Cartesian sampling grid [99] with 515 q-vectors. The maximum b-value was 11,538 and the minimum was 0. Two sets of simulation experiments were performed each using a Tensor of different shapes.

In the first experiment shown in Fig. 2.13, 2.14 we used a more anisotropic prolate tensor for the simulation with eigenvalues $\lambda_{\parallel} = 1.4 \times 10^{-3} mm^2/sec$ and $\lambda_{\perp} = 0.1 \times 10^{-3} mm^2/sec$. In the second experiment shown in Fig. 2.15 and Fig. 2.16 we used a much less anisotropic prolate tensor with $\lambda_{\parallel} = 1.7 \times 10^{-3} mm^2/sec$ and $\lambda_{\perp} = 0.3 \times 10^{-3} mm^2/sec$. The values of λ are based in [16]. It is well known that noise has higher effect on less anisotropic areas. We can see this effect by comparing the overlapped FAs of these two figures (2.13, 2.15). We can also see that all six methods (DSI, GQI, GQI2, EITL, EITL2, EITS) can resolve correctly the fibre directions by looking at their spherical distribution functions using a standard

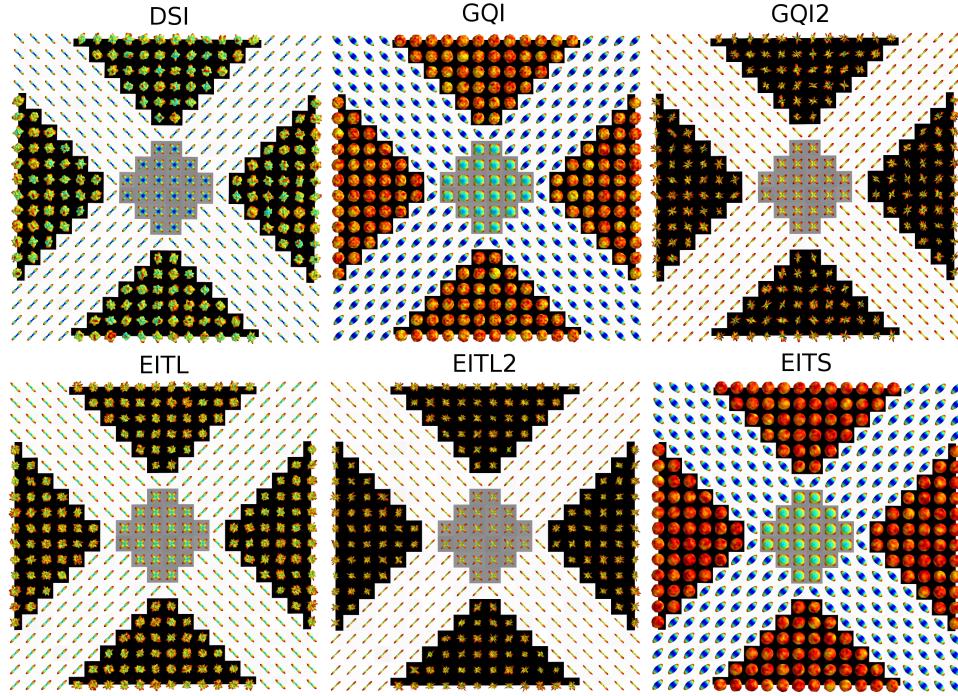


Figure 2.13: Results with an 'x' shape software phantom. Every single tensor compartment had the following eigenvalues $\lambda_{\parallel} = 1.4 \times 10^{-3} \text{ mm}^2/\text{sec}$ and $\lambda_{\perp} = 0.1 \times 10^{-3} \text{ mm}^2/\text{sec}$. Rician noise was added with SNR = 5. GQI is very similar to EITS, GQI2 is very similar to EITL and DSI is very similar to EITL. In Fig. 2.14 the regions at the centers of the phantoms are depicted in higher resolution.

colour map. For visualization purposes all ODFs are shown in relative size as they have been scaled so that their maximum values correspond to 1.

Furthermore, we can easily observe that GQI is mostly similar to EITS, GQI2 is very similar to EITL and DSI is mostly similar to EITL. The fact that DSI ODFs are very similar to those of EITL ODFs is to be expected as the two methods create theoretically the same real ODFs. Remarkably, EITL can create these ODFs without using the Fourier Transform neither using any filter or thresholds in r-space which are necessary in DSI.

Fig. 2.13, 2.15 show that all the different grid-based reconstruction methods can reconstruct correctly the underlying fibre directions even when noise is present. However, we can see that when tensors are less anisotropic the noise has a stronger effect in the resulting spherical distributions. We can also see that GQI & EITS are less sharp than DSI & EITL and these are less sharp than GQI2 & EITL2. Also DSI, GQI2, EITL, EITL2 have much

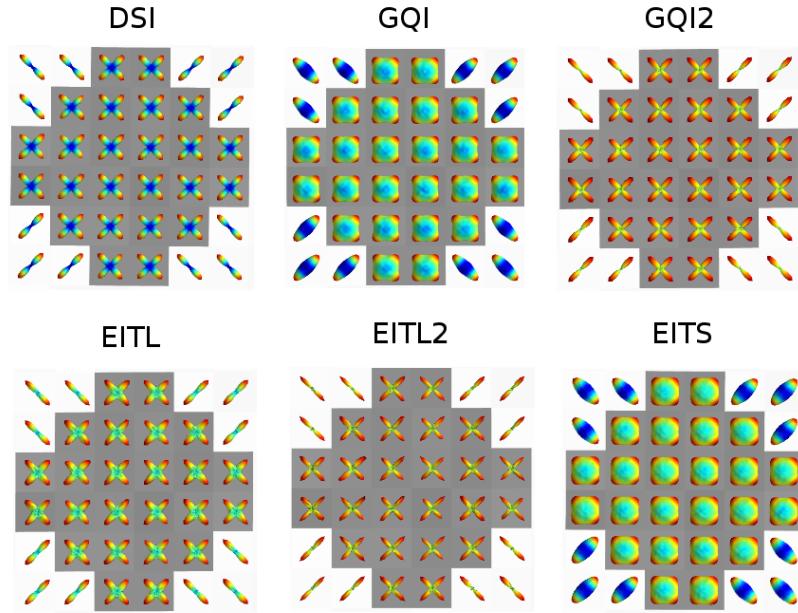


Figure 2.14: Same as Fig. 2.13 showing in higher resolution the spherical distributions in the centers of the phantoms.

lower minima than GQI and EITS.

In the EIT-based reconstruction results shown in Fig. 2.14 and Fig. 2.16 we do not use any amount of smoothing as used in DSI (through hanning filter), GQI, GQI2 (through sampling length) and it is extraordinary that we still obtain such well defined distributions. If we want to apply some weighting/smoothing/denoising in EIT-based methods that is simply possible through the spherical angular smoothing approach described in section 2.8.

The parameters used for these simulations were for DSI: radial sampling 2.1 – 6, hanning filter width: 36 , GQI: $\lambda=1.2$, GQI2: $\lambda = 3$, and EITS, EITL, EITL2 were all calculated with the standard options ($z = \pm 5$) and no further post-processing or smoothing was used. All methods were using the same reconstruction sphere with 642 vertices and 1,280 faces.

2.9.4 Results with humans

We want to compare reconstruction methods on Cartesian grid-based acquisitions first with data sets which are rich on directions and commonly used for DSI processing. For this purpose we used a data set which was available online at cmtk.org from the Diffusion Group at Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland. So, this data set was

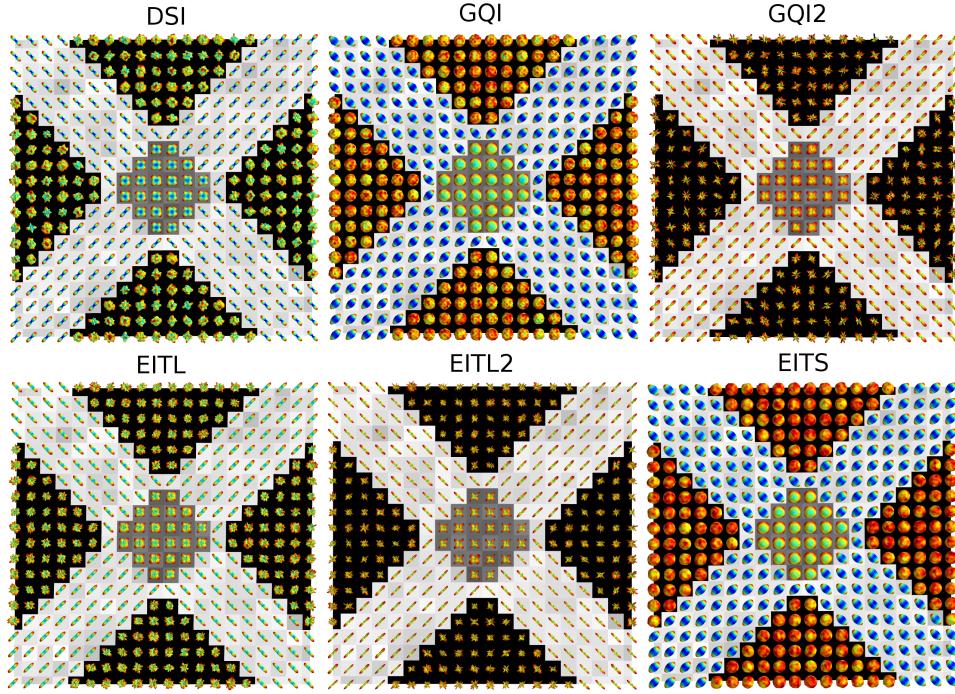


Figure 2.15: Showing the spherical distribution functions (DSI, GQI, GQI2, EITL, EITL2, EITS) of a software phantom generated by two bundles where each bundle contains single tensors along the direction of the phantom. On the crossing area, a dual Tensor effect in every voxel is observed. Every single Tensor compartment had the following eigenvalues $\lambda_{\parallel} = 1.7 \times 10^{-3} \text{ mm}^2/\text{sec}$ and $\lambda_{\perp} = 0.3 \times 10^{-3} \text{ mm}^2/\text{sec}$. Rician noise was added with SNR=5. We also visualize simultaneously the FA for this slice. We can see that in the crossing area (gray background) the FA values drop considerably. However, the ODFs represent precisely the crossing.

obtained from a 3T scanner (TIM Trio, Siemens) with a 32 channels head coil. The field of view was $210 \times 210 \text{ mm}^2$, matrix size 96×96 , and slice thickness 3 mm. 44 slices were acquired and the voxel resolution was $2.2 \times 2.2 \times 3.0 \text{ mm}^3$. A 258-point half grid acquisition scheme with a maximum b-value of 8011 s/mm^2 also known as DSI515 [100] was used. The total acquisition time was 34 min with TR=8200 ms and TE=165 ms.

The parameters used for these simulations were for DSI: radial sampling 2.1 – 6, Hanning filter width: 36, GQI: $\lambda=1.2$, GQI2: $\lambda = 3$, and for EITS, EITL, EITL2 were all calculated using the standard options for zonal width ($z = 5^\circ$) and spherical angular smoothing ($s = 0.05$). All methods were using the same reconstruction sphere 642 vertices and 1,280 faces. The results of this experiment are shown on top of an FA slice of a healthy human in Fig. 2.17 and in higher resolution in Fig. 2.18. It is observed

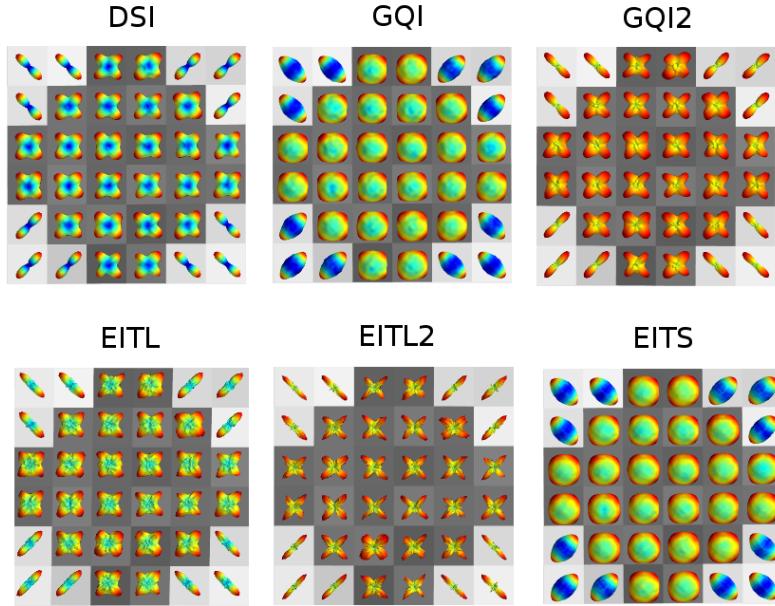


Figure 2.16: A zoomed version of Fig. 2.15 showing the spherical distributions in the centers of the phantoms at higher resolution.

that EITL, EITL2 and EITS can be used for reconstructing these data sets as their results appear very similar to the results given by DSI, GQI and GQI2. We can also easily see that EITL and EITL2 are relatively sharp which can be of an advantage for the purpose of recovering correctly the underlying real fibre directions.

We also tested our results with another human brain data set generated at a 3T scanner (TIM Trio, Siemens) at the Medical Research Council Cognition and Brain Sciences Unit, Cambridge, UK. We used Siemens advanced diffusion work-in-progress sequence, and STEAM [101, 15] as the diffusion preparation method. The field of view was $240 \times 240 \text{ mm}^2$, matrix size 96×96 , and slice thickness 2.5 mm (no gap). 55 slices were acquired to achieve full brain coverage, and the voxel resolution was $2.5 \times 2.5 \times 2.5 \text{ mm}^3$. In this experiment a smaller number of gradient vectors were used. A 102-point half grid acquisition with a maximum b-value of $4,000 \text{ s/mm}^2$ was used. The total acquisition time was only 14 min 21 s with TR=8, 200 ms and TE=69 ms.

In Fig. 2.19 a slice is shown where different parts of white matter are visible with the FA background image. We can clearly see structures like the Corpus Callosum (CC) and Cortical-Spinal Tract (CST) and Centrum Semiovale areas. The ODFs of EITL are shown superimposed on the FA.

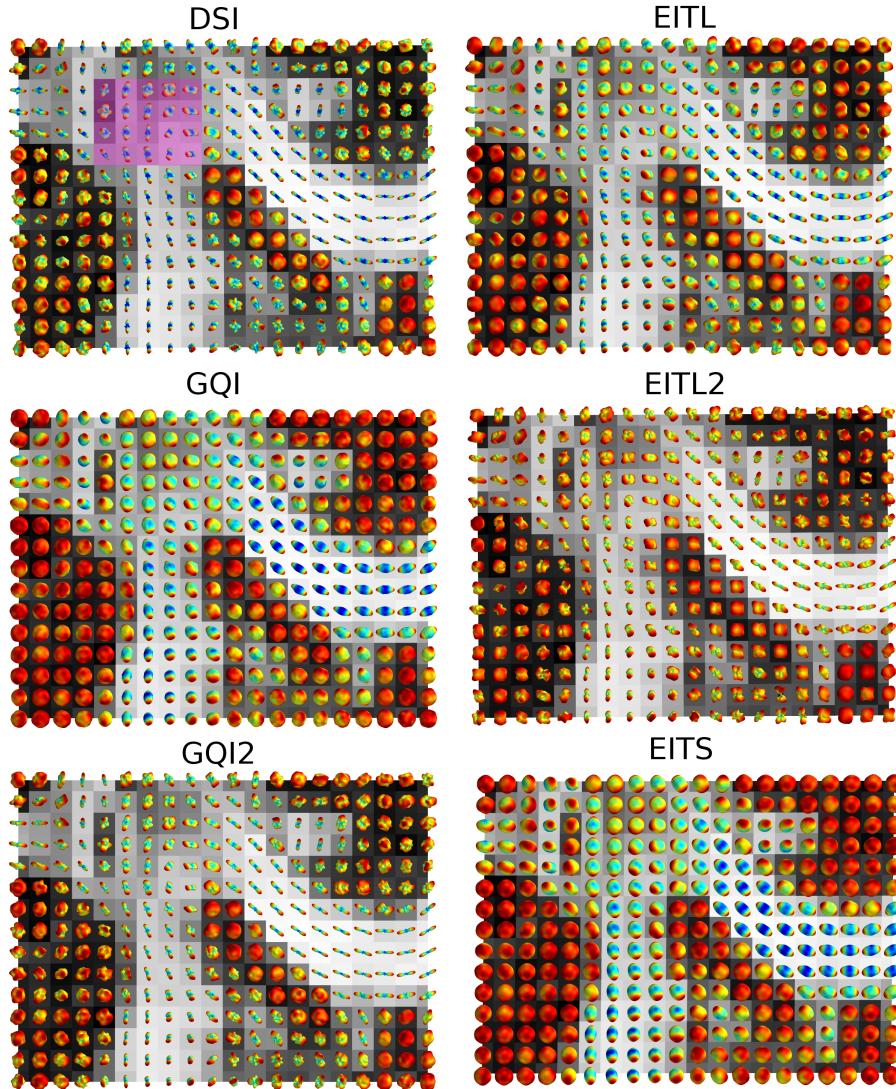


Figure 2.17: Showing the same slice of a human brain reconstructed with 6 different Cartesian grid q-space based methods. The ODFs are visualized on top of the FA slice. A clearer presentation of a region near the left upper corner (with purple shading) is given in Fig. 2.18 for all the 6 methods.

The parameters used for EITL were: a standard zonal width $z = 5^\circ$ and spherical angular smoothing $s = 0.05$ with the same reconstruction sphere (642 vertices, 1,280 faces) as before.

For illustration purposes the upper part of Fig. 2.19 is depicted again in Fig. 2.20, and the region with purple shading from Fig. 2.20 is given at an even higher resolution in Fig. 2.21. We used Mayavi [102], a Python visualization library based on VTK to make the visualizations shown in the figures of this section.

Although less directions were used in this acquisition scheme we ob-

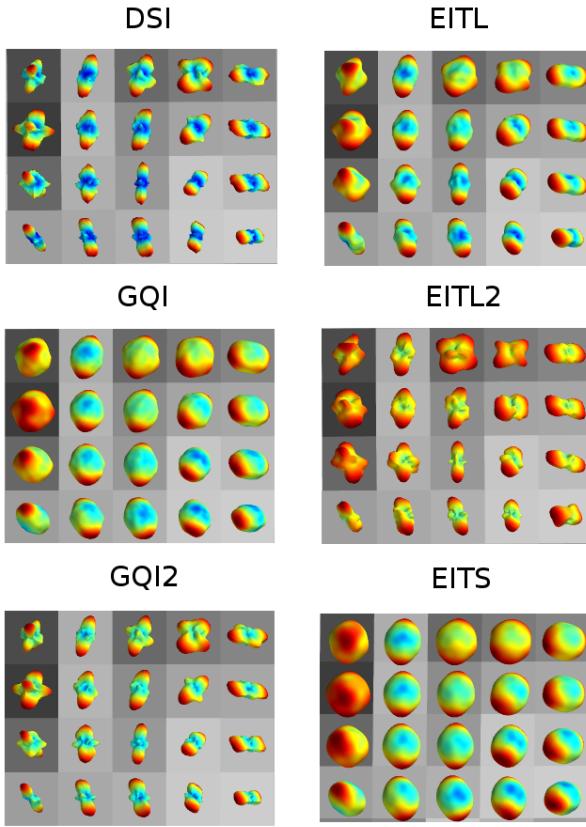


Figure 2.18: The upper-left corners (purple shading region) of the panels of Fig. 2.17 are shown here at higher resolution. These data sets belong to a real human. In contrast with the results shown in simulations (see Fig. 2.14) we applied spherical angular smoothing with $s = 0.05$ for EITL, EITL2 and EITS in order to remove small noisy spikes in the distributions. In agreement with the results of Fig. 2.14, EITS is very similar to GQI. The difference between DSI, GQI2 and EITL, EITL2 is smaller as a result of the application of angular weighting.

tain a similarly accurate depiction of the underlying white matter structure in comparison with that of 258 directions. This gives great hope that we can use grid-based reconstruction methods with half-grid sequences with 100 gradient directions. This was also shown by [74] and [72] who used similar number of directions.

In all the figures with real data sets we can see single fibres as those usually found at the center of CC, and 2 or 3-fibre crossings in the intersection areas of CC with the CST and other bundles.

We can see for example in Fig. 2.21 that the effect of spherical angular smoothing can help alleviate the noise effects and focus our concentration on depicting the major directions which are also of highest concern.

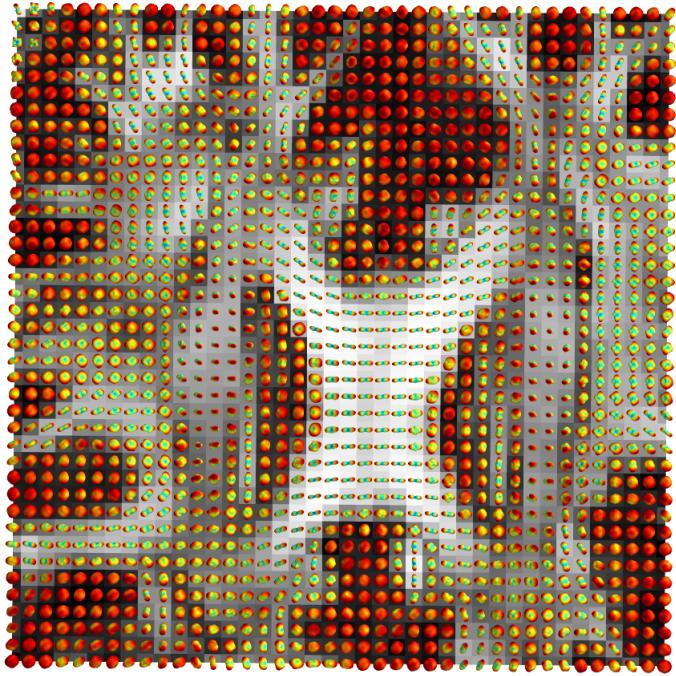


Figure 2.19: EITL ODFs rendered on top of FA of a human brain data set. A small 102-point half grid acquisition with a maximum b-value of $4,000 \text{ s/mm}^2$ was used. Fig. 2.20 and 2.21 are zoomed versions of the same figure. We can see clearly single fibres on the CC and CST areas but also crossing fibres at the Centrum Semiovale and at the areas where big bundles cross. Also the non-white matter areas are evidently more isotropic.

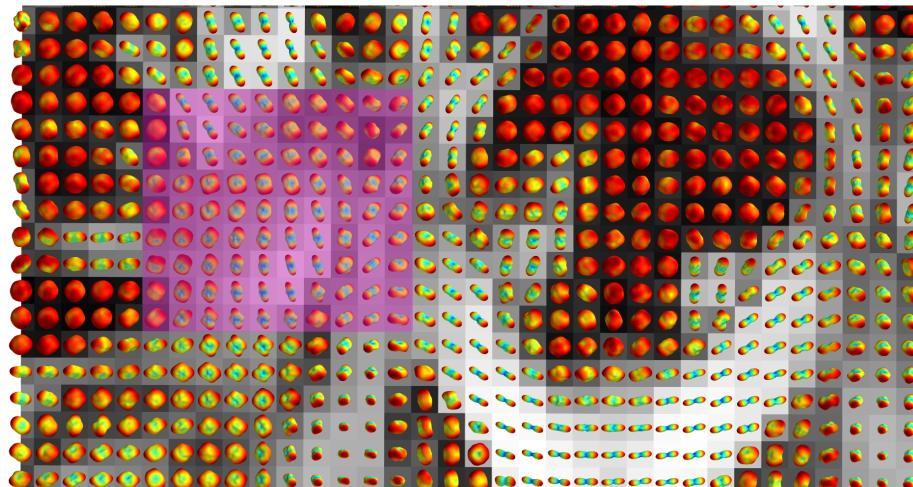


Figure 2.20: The upper part of Fig. 2.19 is shown here at higher resolution. The purple shaded part is given in higher resolution in Fig. 2.21

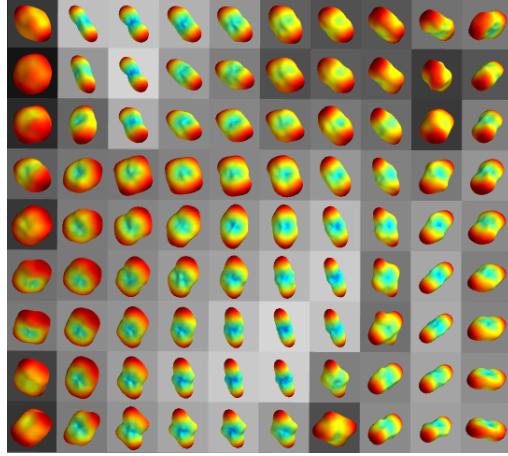


Figure 2.21: EITL ODFs of 1-fibre, 2-fibre and 3-fibre crossings from a real human data set of 101 applied weighted diffusion volume and 1 without weighting (b_0). This picture is a zoomed version of the purple shaded area shown in Fig. 2.20.

2.10 Anisotropy metrics

Until this moment we discussed about density functions on the sphere as a way to represent complex fibre directionality in the voxel. These density functions are represented as multidimensional vectors containing 200 or more dimensions in each voxel and it can be cumbersome to use them directly for subject comparisons or visualization purposes. For this purpose most people use simple scalar summarizing metrics e.g. Tensor-based FA, MD or ODF-based such as the Generalized FA (GFA) [68]. In this section we will show that a similar scalar function like FA can be constructed non-parametrically. We call this NPA which stands for non-parametric anisotropy. We will also start experimenting with metrics that have more than one scalar value and can represent more accurately the directionality in each voxel that is lost with FA, GFA and MD. We will investigate and explain here the realms and robustness of Quantitative Anisotropy which was first introduced by Yeh et al. [72].

2.10.1 Non-parametric Anisotropy

Local voxelwise measures such as fractional anisotropy (FA), apparent diffusivity coefficient (ADC), or mean diffusivity (MD) have been extensively adopted in clinical and applied research practice based on diffusion weighted MR imaging (dMRI). This underlines the need for valid and reliable measures which can indicate the degree of local organisation

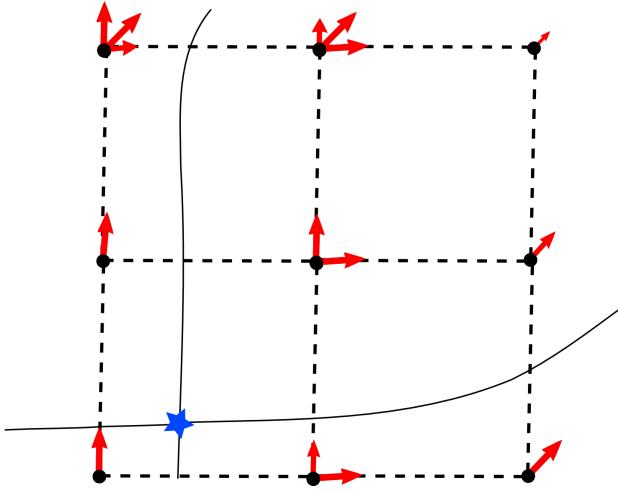


Figure 3.1: In every voxel centre (black dot) there are one or more vectors. These vectors represent peaks where their length is equal to their anisotropy value and the direction is equal to the direction of the peak e.g. calculated from a given ODF. EuDX can track multiple peaks starting from a single seed point (star) if their anisotropy values are higher than a threshold. In that way we can track from the same seed towards different directions and support tracking in crossing areas as it is shown here.

where $i \in [1, 3]$ denotes the current peak. The peak can be characterised by two things: (a) the anisotropy value α_i and (b) the unit direction of the peak \mathbf{u}_i . The concept of tracking with the combination of multiple peaks is presented in Fig. 3.1. In order to reduce storing space, the vector \mathbf{u} can be replaced by an index to the closest vertex of an evenly distributed and dense unit sphere. For generality and simplicity we require this indexing process even for the case of the Single Tensor where the peak direction is only one. Alternatively, we can see this process as a strategy which always maps any representation of the voxel on the sphere.

The EuDX algorithm can be described further in the following way: $\mathcal{A}(\mathbf{u}_i)$ is estimated at every point of the volume. This represents a composite vector field where every point contains the peak directions superimposed to the anisotropy values. We create an empty list of tracks $T = \emptyset$ and then we generate random or prespecified seed points. In more detail: we select a seed point \mathbf{p}_0 and start propagating. We need to remember that the propagation direction can go forward and backward, or better said towards one direction (orthograde) and its opposite direction (retrograde).

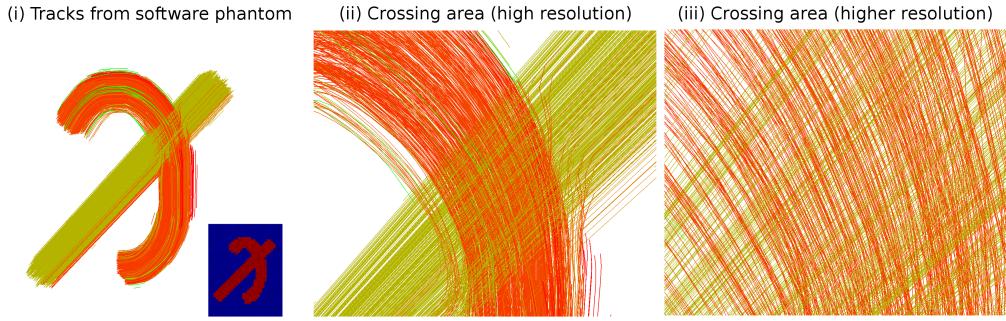


Figure 3.2: (i) Tractography generated by EuDX with DSI as the reconstruction method on a software phantom containing two intersecting bundles. A slice from the b0 volume of the phantom is also shown at the lower right corner. (ii) and (iii) In higher resolution we can see that tracks traveling from the two different bundles cross unimpeded in the intersection area of the two simulated bundles.

for these type of data. In contrast, QBI is not suitable because it assumes a spherical grid in q-space (see section 1.5). For both experiments we used a high SNR of 100 as the main goal was to validate the algorithm on good conditions. We will discuss later the validation process with human data sets where SNR is naturally lower.

In the first experiment (see Fig. 3.2) we generated 200,000 uniformly sampled random seeds in the entire 3D volume of the phantom. We used DSI reconstruction with standard parameters: q-space grid size 16, hanning filter width 32, and radial integration range from 2.1 to 6 at steps of 0.2. As we presented earlier, EuDX expects as input the peaks and the directions of the peaks. For every voxel we used DSI to create the corresponding ODF (sampled on an evenly distributed sphere of 642 vertices and 1,280 faces) and from the ODF we used the peak finding function introduced in section 2.7. We further removed the peaks which had values less than 70% of the highest peak and normalized the rest so that the maximum peak equals to 1. From now on we will call this output function from the ODF; PK (PeaK anisotropy). PK is different from QA because we do not remove the isotropic component neither we normalize with the overall maximum ODF value as it is common practice with QA (see section 2.10.2). We also calculated FA and zeroed the peaks with FA values lower than 0.2. In that way we ensured that there will be no tracking in the background area of the phantom.

For EuDX we used parameters $\mathcal{A}_{thr} = 0.2$, $\Delta s = 0.5$, $\theta_{thr} = 60^\circ$ and

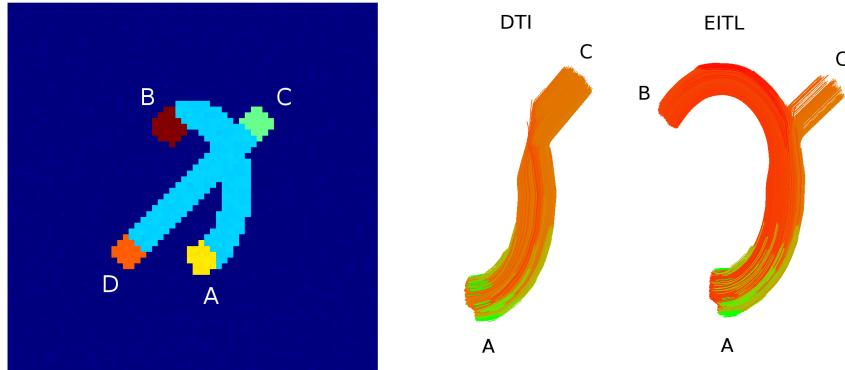


Figure 3.3: On the left panel we see the b_0 volume of the software phantom with 4 ROIs; A, B, C and D presented with different colours. 2,000 seeds were generated inside each ROI and we then measured the amount of tracks which reached the other regions. On the right panel we see that when the DTI (Single Tensor) was used for the reconstruction; the tracking started from A reached only C. However, when EITL was used the tracks reached both areas B and C as it was expected.

$TW = 0.5$. The results are shown in Fig. 3.2. The tracks are colour-coded with their orientation; this is defined as the unit vector connecting the first with the last point of each track. Because we generated random seeds that went everywhere in the phantom some of them had to fall in the crossing area. In accordance with what we discussed in the previous section we know that if a seed falls in a crossing region, EuDX will propagate towards both directions of the crossing if and only if multiple peaks are supported from the underlying reconstruction method. We can confirm this by observing the result at Fig. 3.2. We can see that the crossing area is well represented and that the propagation was successful. We can also observe that not all tracks travel the entire distance from end to end. Otherwise we would expect to see all tracks of the entire elliptic part with one colour and the tracks of the straight orbit all with a different colour because they have different orientation. It seems that this observation had something to do with the discrete nature of the phantom especially near the bundles' boundary areas and of course the functionality of EuDX. For this purpose, we created a different experiment to evaluate this finding.

In the second experiment we generated 2,000 seeds inside specific regions of the bundles (see Fig. 3.3). More precisely, the end-point areas of the phantom denoted with A, B, C and D. The goal was to count the percentage of tracks which reached any of the other end-point areas. We tried this with 4 different reconstruction methods: Single Tensor, EITL,

DTI	A	B	C	D	\emptyset	EITL	A	B	C	D	\emptyset	
	A	—	0%	76.4%	0%	23.6%	A	—	63.4%	8.7%	0%	27.9%
	B	0%	—	0%	59.8%	40.2%	B	65.6%	—	0%	5.6%	28.8%
	C	79.9%	0%	—	0%	20.1%	C	14.5%	0%	—	76.8%	8.7%
	D	0%	56.5%	0%	—	43.5%	D	0%	0.5%	95.8%	—	3.7%
DSI	A	B	C	D	\emptyset	GQI	A	B	C	D	\emptyset	
	A	—	65.3%	9.6%	0.0%	25.1%	A	—	57.8%	8.7%	0.0%	33.5%
	B	72.6%	—	0.0%	5.5%	21.9%	B	67.2%	—	0.0%	10.7%	22.1%
	C	14%	0.0%	—	79.9%	6.1%	C	37.5%	0.0%	—	55.5%	7%
	D	0.0%	10.6%	84.8%	—	4.6%	D	0.0%	22.3%	72.3%	—	5.4%

Table 3.1: Every sub-table shows the percentage of tracks which started from areas A, B, C or D (rows) and reached the other areas A, B, C, D or \emptyset (columns) using EuDX with input from different reconstruction methods DTI, EITL, DSI, GQI. The column \emptyset symbolizes the number of tracks which did not reach any of the A, B, C, D areas. For example, by looking only the first row of each sub-table we easily observe that the crossing area was well represented by EITL, DSI and GQI but not from DTI as it was expected because the Single Tensor cannot resolve crossings. We can also observe by comparing all the rows that in the more curve branch AB of the phantom fewer tracks reached their target than in the diagonal branch CD.

then MaxSED= 5.2%. The same value can be used as a conservative MaxSED in comparisons involving Tab. 3.2 for which the effective n is greater than 2,000. Differences greater than MaxSED are highly significant ($p < 0.0001$).

When DTI is compared with any of EITL, DSI and GQI, each of the connections $A \rightarrow \emptyset$, $B \rightarrow \emptyset$, $C \rightarrow \emptyset$, $D \rightarrow \emptyset$ is significantly more frequent with DTI. Similarly $A \rightarrow B$, and $B \rightarrow A$ are less frequent with DTI, as are $C \rightarrow D$, and $D \rightarrow C$. By contrast $A \rightarrow C$, and $C \rightarrow A$ are more frequent with DTI, as are $B \rightarrow D$, and $D \rightarrow B$.

Between EITL and DSI we note that for tracks starting in B , $B \rightarrow \emptyset$ more frequently with EITL whereas $B \rightarrow A$ less frequently. For tracks starting in D , $D \rightarrow C$ more frequently with EITL than DSI , and $D \rightarrow B$ less frequently with EITL then DSI.

Between EITL and GQI, for tracks starting in A , $A \rightarrow \emptyset$ less often with EITL than with GQI, whereas $A \rightarrow B$ and $A \rightarrow C$ occur more often. However $B \rightarrow \emptyset$ more often for EITL than GQI. By contrast $C \rightarrow D$ occurs more frequently, and $C \rightarrow A$ less frequently. Also $D \rightarrow B$ occurs less.

Finally, comparing DSI with GQI, $A \rightarrow \emptyset$ less often with DSI whereas

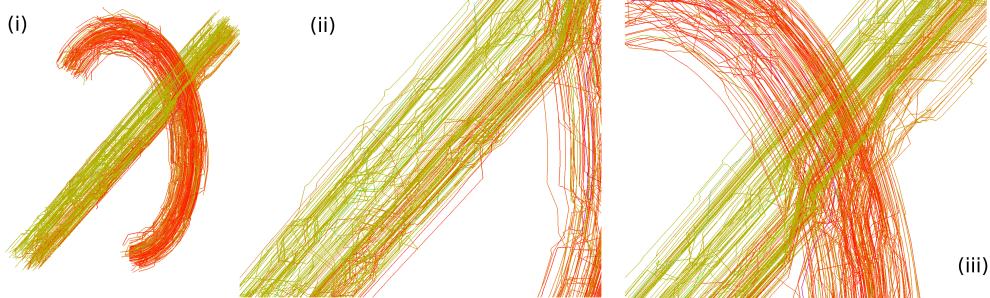


Figure 3.4: Probabilistic tractography performed on the software phantom shown in Fig. 3.2. We use Pico and a Two Tensor model (PMT) to generate the tracks. (i) 1 iteration of the Pico (showing 321 tracks). (ii) A detail of the left bundle. Some of the tracks are diverting on a zig-zag fashion although the phantom at these voxels has only one main direction. (iii) Focus on the crossing region of the phantom. Many of the PMT tracks are successfully propagating towards both pathways.

which will reach the crossing region will follow the branch to which they are closer to in terms of spatial and angular distances. This is possible because of the usage of trilinear interpolation.

Independently from the deterministic tractography described previously we repeated the same experiments with probabilistic tracks generated from the Camino Diffusion MRI Toolkit³. We will call these PMT tracks where PMT stands for Probabilistic MultiTensor Tractography. In order to create the PMT tracks we fitted a Two Tensor model in the data with both Tensors enforced to be prolate. For the fitting method we used the standard non-linear fit provided in Camino with positivity constraints. For this purpose we used the command `modelfit` with options `-model cylcyl nldt_pos`. The probabilistic tractography was generated using Pico PDFs sampled from a Watson distribution (`picopdfs -pdf -watson`). In order to generate the streamlines we used the Camino command `track` which took as input the PDFs from before and a seed file with the mask.

As previously, we performed two experiments; one where the seeds can be anywhere in the simulated bundles and a second where the seeds were constrained in the pre-specified ROIs A, B, C or D (see Fig. 3.3). The results of the first experiment can be seen in Fig. 3.4.

For illustration purposes we see in Fig. 3.4(i) the result with 321 tracks of 1 iteration of the Camino `track` command. In Fig. 3.4(ii) we see a detail of the left bundle. It is obvious that some of the tracks are diverting vigor-

³cmic.cs.ucl.ac.uk/camino

PMT100	A	B	C	D	\emptyset	PMT5000	A	B	C	D	\emptyset
A	–	22%	45.2%	0.1%	32.6%	A	–	22.6%	44.%	0.2%	33.2%
B	40.8%	–	0.6%	11.1%	47.5%	B	40.9%	–	0.7%	9.3%	49.1%
C	28.9%	0.4%	–	51%	19.7%	C	28.2%	0.3%	–	50.2%	21.3%
D	0.1%	3.8%	85.5%	–	10.6%	D	0.3%	4.2%	85.2%	–	10.3%

Table 3.2: In comparison with Tab. 3.1 we observe that PMT performed worse than EuDX with DSI, GQI or EIT. This becomes evident by observing the \emptyset columns of the corresponding tables where EuDX had a lower percentage of tracks which did not reach any of the other end-points. The difference between PMT100 and PMT5000 is in the number of iterations which was 100 and 5,000 respectively. Increasing the number of iterations did not increase considerably the performance of PMT.

ously on a zig-zag fashion although the phantom at these voxels has only one main direction. In Fig. 3.4(iii) we zoom in the crossing region of the phantom where we observe that many of the PMT tracks are successfully propagating towards both pathways through the crossing region.

The results from the second experiment where the seeding takes place only in specific regions of the phantom are summarized in Tab. 3.2. The distinction between PMT100 and PMT5000 is in the number of iterations run from Camino track command. The default value recommended is 5,000 however for this experiment we do not see a significant difference between 100 and 5,000 iterations. In comparison with Tab. 3.1 we observe that PMT performed worse than EuDX with any of DSI, GQI or EIT. This becomes obvious when comparing the \emptyset columns of the corresponding tables where we see that EuDX had a significantly lower percentage of tracks that did not reach any of the other end-points. Further detailed comparison of EITL with PMT shows that the two-way connections $A \rightarrow B$ & $B \rightarrow A$, and $C \rightarrow D$ & $D \rightarrow C$ are stronger with EITL. By contrast $A \rightarrow C$ & $C \rightarrow A$ are stronger with PMT.

This is an important outcome as probabilistic approaches are usually supposed to perform better than deterministic [35], [113]. In this study we have not investigated if PMT performs poorly because of the Two Tensor fit or because of the probabilistic orientation sampling. However, we plan to investigate this further in the future.

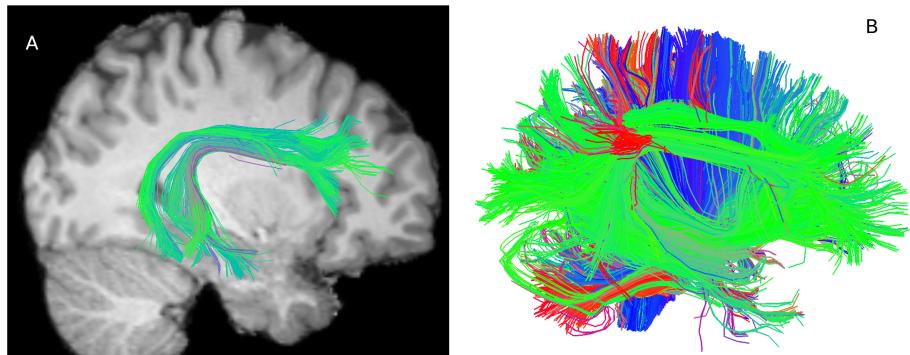


Figure 3.5: A: right arcuate fasciculus generated by EuDX and segmented by expert neuroanatomists. The tracks are in MNI coordinates and we visualize simultaneously the T1 slice ($X=29$) for the same subject. B: the sagittal view of the whole brain tractography of the same subject is shown. For visualization purposes we are depicting only tracks of length from 120 mm to 150 mm.

2.5 mm (no gap). 55 slices were acquired to achieve full brain coverage, and the voxel resolution was $2.5 \times 2.5 \times 2.5 \text{ mm}^3$. A 102-point half grid acquisition with a maximum b-value of $4,000 \text{ s/mm}^2$ was used. The total acquisition time was only 14 min 21 s with TR=8, 200 ms and TE=69 ms.

The tractographies were generated in diffusion native space and then linearly registered in MNI space. For that reason FA volumes were generated from the same data sets using Tensor fitting with weighted least squares after skull stripping with FSL bet. These FA volumes were again in diffusion native space, therefore we used FSL flirt to align them in MNI space. For this purpose a standard FA atlas FMRIB58 from the FSL toolbox was used as the reference image. We then applied the affine transformation matrix from the previous step to the initial tractography to align it to MNI space.

In order to help the neuroanatomists to guide themselves with the segmentation of bundles, we linearly registered structural MRIs (T1 MPRAGE) images from the same subjects to the standard template MNI152. Therefore, the tractography and the T1 image were registered in the same space. This was very useful for the neuroanatomists because they could find known bundles from the regions they connect in the cortex which are better visible in the T1 image rather than in the FA image.

In section 3.3 we discussed with simulation experiments that EuDX was able to propagate correctly in crossing areas (see Fig. 3.2) by gener-

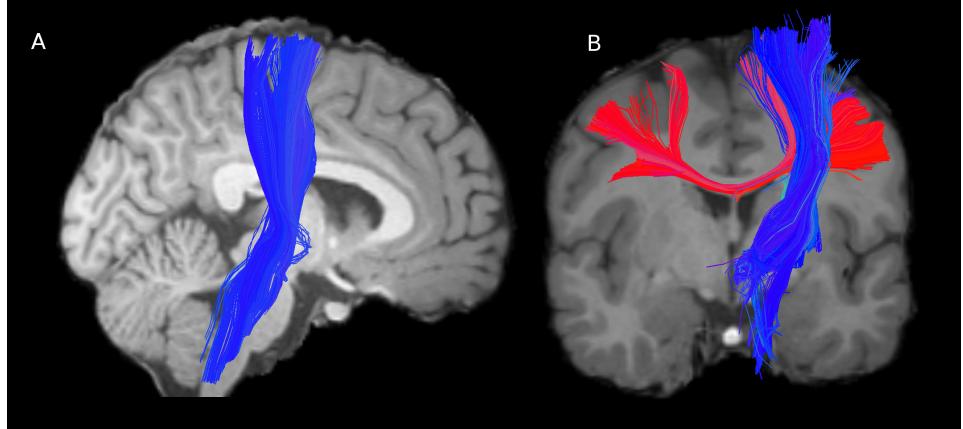


Figure 3.6: A: EuDX tracks from the CST segmented by our expert neuroanatomists. The tracks are linearly registered in MNI standard space and visible is also the T1 slice ($Y=5$) from the same subject. B: The intersection of the BCC with the right CST. This is a confirmation that EuDX can propagate successfully in crossing areas of human brain data. The T1 slice ($Y=-1$) from the same subject is also visible.

ating tracks towards both directions of the crossings. We wanted to confirm if tracking in crossing areas was also robust with human data sets where the noise artefacts are less predictable. This was indeed confirmed by looking at the intersection of two well known bundles: the Body of Corpus Callosum (BCC) and the Corticospinal Tract (CST). In Fig. 3.6B we can see that a part from the BCC shown with red is intersecting the CST bundle shown with blue without being diverted from the CST. If EuDX was not able to propagate in crossing areas then the tracks from BCC would stop in the intersection area or divert towards the direction of the CST.

In the next experiment we visually compared specific bundles across different healthy subjects. We concentrated at a pair of bundles not so often studied in literature: the Cingulum bundles; across 12 healthy subjects (20 – 40 years old). The Cingulum (CG) is an association fibre tract that runs within the Cingulate Gyrus along its entire length. It collects axons from the Cingulate Gyrus that travel immediately dorsal to the Corpus Callosum (CC) and along the ventral face of the Hippocampus, forming a large C-shape trajectory. It carries afferent connections from the Cingulate Gyrus to the entorhinal cortex. Because of its narrow tubular shape, it is often difficult to reveal its entire length by a single data set [11]. CG has the characteristic that both left and right CGs travel in parallel and are very close to each other. Often many tractography algorithms do not rep-

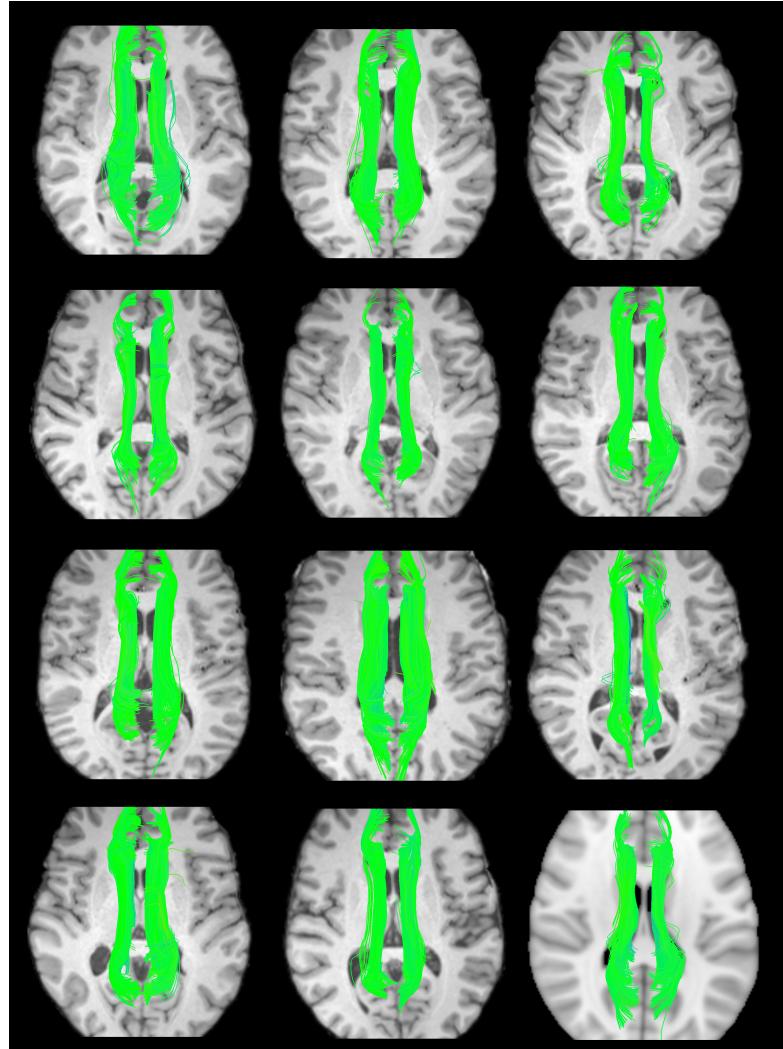


Figure 3.7: The left and right Cingulum (CG) bundles of 12 healthy subjects are presented here. Left and right CGs were selected in MNI space from entire EuDX tractographies of 1 million seeds and GQI as the reconstruction input. We also show the T1 image slice Z=10 for every subject except from last subject whose T1 was not available. For that subject we use the standard MNI template.

resent the Cingulum bundle very well because it is very close to the CC. However, using EuDX we see that both left and right CG bundles were consistent across all 12 subjects (see Fig. 3.7). The cingulum was selected using the interactive tool referenced previously from the entire tractography. The reconstruction parameters and EuDX parameters were the same as before with the exception of the number of initial seeds which was 1 million. Furthermore, the tractographies of all subjects were registered together with the structural images in MNI space. The only exception was with the last subject 12 whose T1 image was not available and it was visualized together with the MNI standard template (MNI152) (see bottom-right corner of Fig. 3.7).

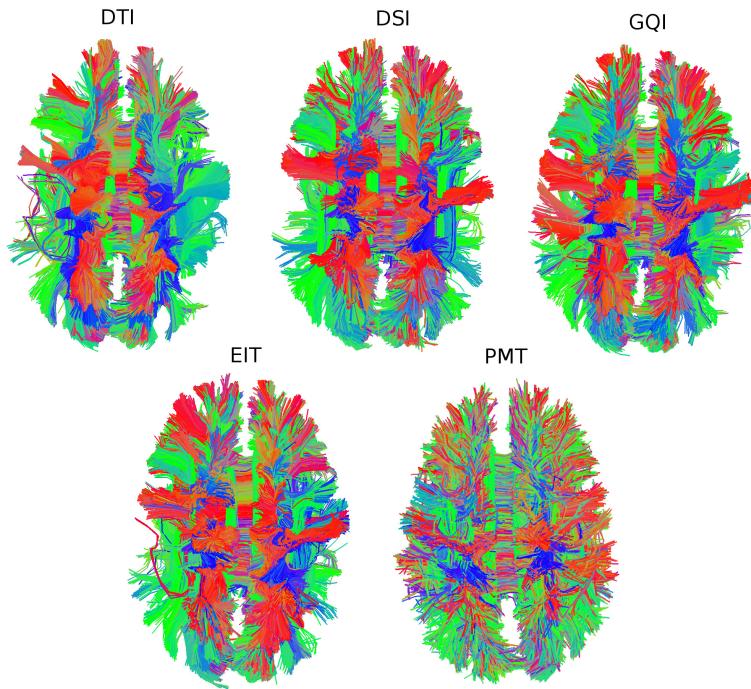


Figure 3.8: 5 tractographies from the same data sets of the same subject. The 3 top and the bottom left are created using our proposed deterministic approach (EuDX). The bottom right is created using Probabilistic tractography (PMT) .

In Fig. 3.8 we show results generated from a single subject. The data had size $96 \times 96 \times 55 \times 102$ and were acquired as described previously in this section. We first removed the scalp using bet and secondly created an FA image for this data. We masked out all voxels with $FA < 0.2$ in order to make sure that seeding will take place in areas of high anisotropy. We

4.2 Track distances and preprocessing

For clarity we first give brief details of various metrics for distances between tracks as they are integral to an understanding of the track clustering literature. Numerous distance metrics between two trajectories have been proposed in the literature, such as in [158], [159], [160] with the most common being the Hausdorff distance found in [161] and many other studies. We mainly use a very simple symmetric distance proposed in [162] and [163] which we call Minimum average Direct-Flip MDF(s_A, s_B) distance between track s_A and track s_b (see Eq. 4.1). This distance can be applied only when both tracks have the same number of points. Therefore, we assume that an initial downsampling of tracks has been implemented, where all segments on a track have the same length, and all tracks have the same number of segments. Under that assumption MDF is defined as:

$$\begin{aligned} \text{MDF}(s_A, s_B) &= \min(d_{\text{direct}}, d_{\text{flipped}}), \text{ where} \\ d_{\text{direct}}(s_A, s_B) &= \frac{1}{K} \sum_{i=1}^K \|\mathbf{x}_i^A - \mathbf{x}_i^B\|_2 \text{ and} \\ d_{\text{flipped}}(s_A, s_B) &= \frac{1}{K} \sum_{i=1}^K \|\mathbf{x}_i^A - \mathbf{x}_{K-i}^B\|_2 \end{aligned} \quad (4.1)$$

where K is the number of points \mathbf{x}_i on the two tracks A and B .

In some cases it is still valid to use a family of Hausdorff distances which for simplicity we denote as MAM distances – short for Minimum, or Maximum, or Mean, Average Minimum distance (MAM). We mostly use the Mean version of this family, (see Eq. 4.5) but the others are potentially useful as they can weight different properties of the tracks. These distances are slower to compute than MDF but they can work with different number of segments on tracks; a property that is useful for some ap-

plications. The equations below show the formulation of these distances:

$$d_{\text{avg}}(s_A, s_B) = \frac{1}{K_A} \sum_{i=1}^{K_A} d(x_i^A, s_B),$$

$$d_{\min}(s_A, s_B) = \min_{j=1, \dots, K_B} d(x_j^A, s_B), \text{ and} \quad (4.2)$$

$$d_{\max}(s_A, s_B) = \max_{j=1, \dots, K_B} d(x_j^A, s_B) \text{ where} \quad (4.3)$$

$$d(\mathbf{x}, s_B) = \min_{j=1, \dots, K_B} \|\mathbf{x} - \mathbf{x}_j^B\|_2.$$

$$\text{MAM}_{\min}(s_A, s_B) = \min(d_{\text{avg}}(s_A, s_B), d_{\text{avg}}(s_B, s_A)) \quad (4.4)$$

$$\text{MAM}_{\max}(s_A, s_B) = \max(d_{\text{avg}}(s_A, s_B), d_{\text{avg}}(s_B, s_A))$$

$$\text{MAM}_{\text{avg}}(s_A, s_B) = (d_{\text{avg}}(s_A, s_B) + d_{\text{avg}}(s_B, s_A))/2 \quad (4.5)$$

where the number of points K_A and K_B on the two tracks are not necessarily the same. For the same threshold value MAM_{\min} , MAM_{\max} and MAM_{avg} will give different results. For example, MAM_{\min} will bring together more short tracks with long tracks than MAM_{\max} and MAM_{avg} will have an in between effect. Finally, other distances than the average minimum based on the minimum (see Eq. 4.2) or maximum distance (see Eq. 4.3) can be used. However, we have not investigated them in this thesis.

The main advantages of the MDF distance (see Eq. 4.1), are that it is fast to compute, it takes account of track direction issues through consideration of both direct and flipped tracks, and that it is easy to understand how it will behave, from the simplest case of parallel equi-length tracks to the most complicated of very divergent tracks. Another advantage is that it will separate short tracks from long tracks; a track A that is half the length of track B will be relatively poorly matched on MDF to B. We will see later in this chapter that this helps to find broken or erroneous tracks. An asset of having tracks with the same number of points is that we can easily do pairwise calculations on them; for example add two or more tracks together to create a new average track. We will see in the next section that track addition is a key property of our clustering algorithm. Some care should be taken into consideration with the number of points allowed in a track (track downsampling). We always keep the endpoints intact and then downsample in equidistant segments. This means that short tracks will have the same number of points as long tracks. There-

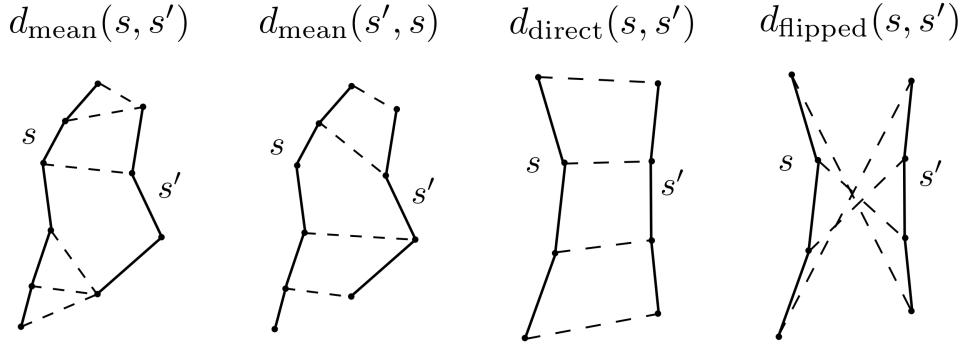


Figure 4.1: Distances used in this work. The main distance used is minimum average direct flip (MDF) distance $MDF = \min(d_{\text{direct}}, d_{\text{flipped}})$ which is a symmetric distance that can deal with the track bi-directionality problem and works on tracks which have the same number of points. Another distance is the mean average distance which is again symmetric but does not need for the tracks to have the same number of points $MAM_{\text{avg}} = (d_{\text{avg}}(s_A, s_B) + d_{\text{avg}}(s_B, s_A))/2$. The components of both distances are shown; with solid lines we draw the tracks, and then with dashed lines we connect the pairs of points of the two tracks whose distances contribute to the overall metrics.

fore, the curvature from the long tracks will be lost relative to the short tracks i.e. the short tracks will have higher resolution. We found empirically that this is not an important issue and that for clustering purposes even downsampling to only 3 points in total could be useful [162]. Depending on the application less or more points can be used.

4.3 Related Work

During the last 10 years there have been numerous efforts from many researchers to address the unsupervised and supervised learning problems of brain tractography. As far as we know all these methods suffer from low efficiency, however they provide many useful ideas which we describe in this section.

Tractography clustering algorithms are rarely compared in the literature. Nonetheless, Moberts et al. [164] are an exception. They evaluated different popular hierarchical clustering methods including a less common one, shared nearest neighbor (SNN), against a gold standard segmentation by physicians. The authors concluded that single-link cluster-

why QB has on average linear time complexity derives from the structure of the cluster node: we only save the sum of current tracks h in the cluster and the sum is cumulative; moreover there is no recalculation of clusters, the tracks are passed through only once and a track is assigned to one cluster only.

Algorithm 5 QuickBundles

Input tracks $T = \{s_1, \dots, s_i, \dots, s_N\}$, threshold θ
Output clustering $C = \{c_1, \dots, c_k, \dots, c_M\}$ where cluster $c = (I, h, N)$

```

 $c_1 \leftarrow ([1], s_0, 1)$ 
 $C \leftarrow \{c_1\}$  # the first track becomes the first cluster
 $M \leftarrow 1$  # the total number of clusters is 1
For  $i = 2$  to  $N$  Do # all tracks
     $t \leftarrow T_i$ 
     $\text{alld} \leftarrow \text{infinity}(M)$  # distance buffer
     $\text{flip} \leftarrow \text{zeros}(M)$  # flipping check buffer
    For  $k = 1$  to  $M$  Do # all clusters
         $v \leftarrow C_k.h / C_k.n$ 
         $d \leftarrow d_{\text{direct}}(t, v)$ 
         $f \leftarrow d_{\text{flipped}}(t, v)$ 
        If  $f < d$  Then
             $d \leftarrow f$ 
             $\text{flip}_k \leftarrow 1$ 
             $\text{alld}_k \leftarrow d$ 
        EndIf
         $m \leftarrow \min(\text{alld})$ 
         $l \leftarrow \text{argmin}(\text{alld})$ 
        If  $m < \theta$  Then # append in current cluster
            If  $\text{flip}_l = 1$  Then
                 $C_l.h \leftarrow C_l.h + \text{reverse}(t)$ 
            Else
                 $C_l.h \leftarrow t$ 
                 $C_l.n \leftarrow C_l.n + 1$ 
                append( $C_l.I, i$ )
            Else # create new cluster
                 $c_{M+1} \leftarrow ([i], t, 1)$ 
                append( $C, c_{M+1}$ )
                 $M \leftarrow M + 1$ 
            EndIf
        EndFor
    
```

QB creates an online list of cluster nodes. The cluster node is defined as $c = (I, h, n)$ where I is the list of the integer indices of the tracks in that

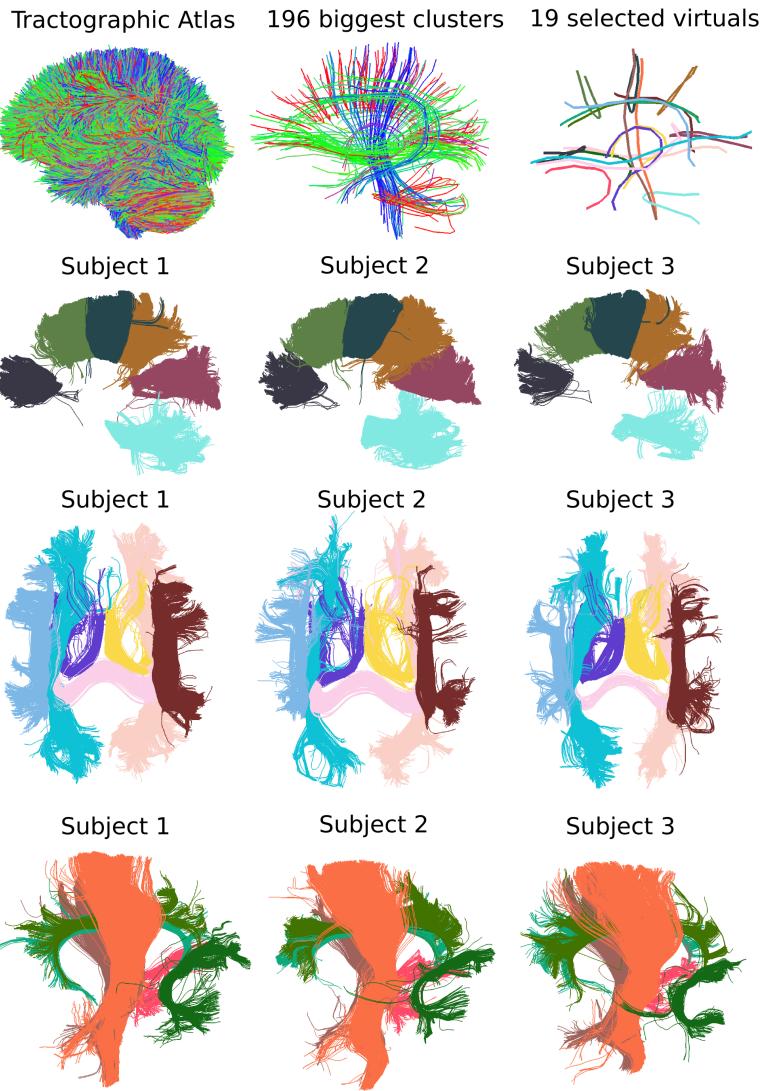


Figure 4.10: A novel way to do comparisons between subjects. Correspondence between different subjects (last 3 rows) and a few landmarks picked from the tractographic atlas generated by merging QB clusterings of 10 subjects (top row). The fact, there is such a level of agreement and continuity on the last 3 rows from such a few skeletal tracks offers a great prospect for implementing new robust ways of statistical comparisons using tractographic data sets.

computers. A naive solution would be to use sparse matrices to approximate the distance matrix; however tractographies are densely packed and produce very dense distance matrices. Therefore, this is not a viable solution. The straightforward solution to this problem is to use QB in order to first segment in small clusters and then use the representatives (i.e. exemplar or virtual tracks) of these clusters with other higher complexity

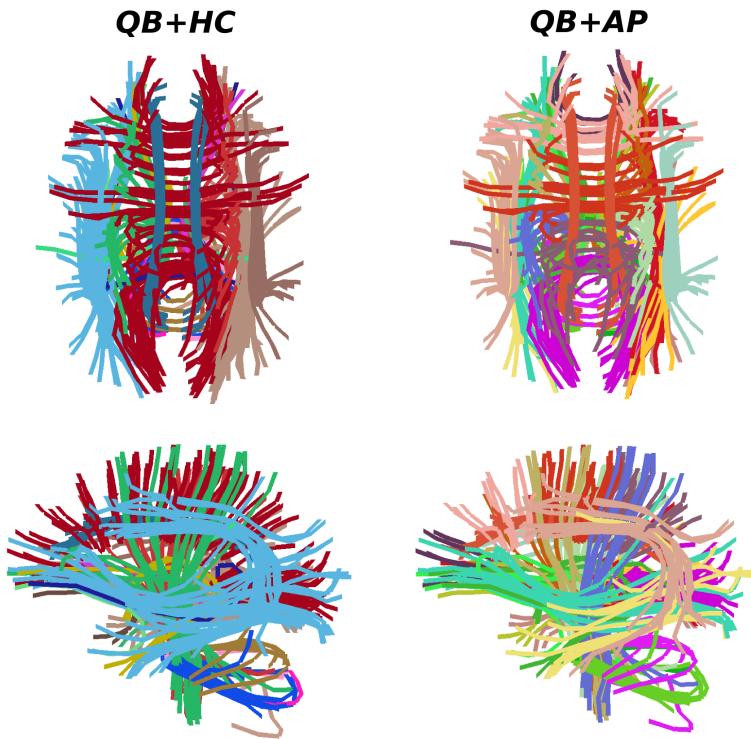


Figure 4.11: Two examples where QB output is used to cluster an entire set of 10 tractographies together and then the result is given as input to hierarchical clustering (HC) using single linkage on the left and to affinity propagation (AP) on the right. Colours encode cluster labels. On the left side we see 19 clusters and on the right 23. QB facilitates significantly the operation of the other two algorithms which would not be able to cluster the entire data sets on current computers. Pay attention at the top left panel where QB+HC have managed to cluster the entire CC as one bundle.

4.8.4 Exemplars vs ROIs vs Masks

Medical practitioners and neuroanatomists often argue that when they use multiple spherical or rectangular masks to select some bundles many tracks are thrown away because they are small and the mask operations cannot get hold of them. Our method provides a solution to this problem as it can identify broken or smaller bundles inside other bigger bundles which are otherwise very difficult or even sometimes impossible to identify visually or with the use of masks. Our method attacks this problem and suggests a very efficient and robust solution which sets the limit for unsupervised clustering of tractographies and facilitates tractography exploration and interpretation. One can now use exemplar tracks as access



Figure 4.12: Two tractographies from different subjects before (left) and after rigid registration (right) using our method.

In Fig. 4.12 we see the result of this algorithm applied to two tractographies – represented with their exemplar tracks – depicted with orange and purple. We can see in the left panel that the orange tractography is misaligned with respect to the purple one, and in the right panel we see their improved alignment after applying our algorithm.

Metric. SMD is proposed here for registration of trajectory data sets, but one could equally use mutual information [221] or the correlation ratio [222] for registration of volumetric data sets. Nonetheless, the advantage of SMD is that it comes from robust landmarks generated by QB which bring together local and global components. Initially, it was not clear if we should use SMD or just the sum of all distances $SD = \sum_{i,j} D(i,j)$. Therefore, we performed an experiment to validate the smoothness and convexity of these two cost functions. We plotted both functions under a single-axis translation or a single-angle rotation of the same tractography as show in Fig. 4.13. From these two diagrams we can see, that although for translations only the SD was entirely convex, with rotations, the SD had stronger local minima which is not a good property for registration. Furthermore, the SMD had steeper gradients towards the global minimum which is a positive indicator for faster convergence.

Experiments. The first large scale experiment took place using the same tractography of a single individual copied and transformed 1,000 times with range of all three angles from -45° to 45° and range of all

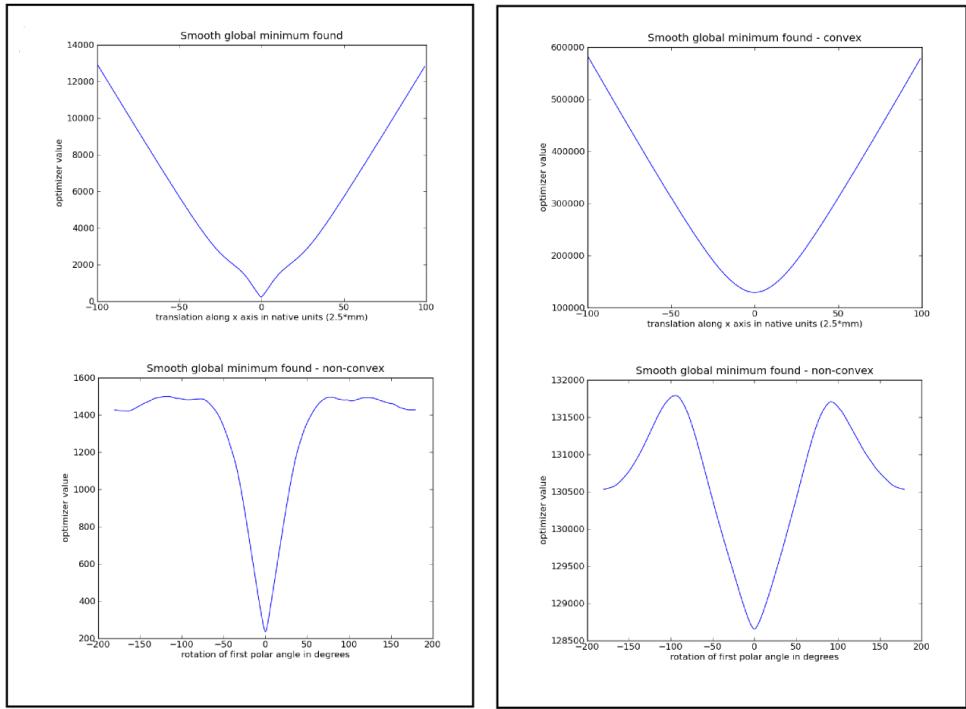


Figure 4.13: Left: The metric *SMD* that we chose to optimize for two copies of the same tractography with the second copy translated (above) and rotated (below). This metric appears to be smooth with a single global minimum and is only slightly non-convex with small local minima. Right: Another possible candidate metric was the *SD*. Although more convex on translations it had stronger local minima with rotations.

x, y, z translations from -113 to 113 mm. Then we registered all transformed tractographies to the static one and calculated all pairwise MDF distances storing them in a square matrix D . We would expect that if the registration was correct then the sum of all diagonal elements of D would be close to 0. This was confirmed with both cost functions used SD and SMD getting close to zero 99.8% of the time; however, SMD was always closer to perfect alignment than SD, having precision of more than 7 decimals. Consequently we chose SMD as a better cost function for direct tractography registration.

We used GQI-based tractographies from 10 subjects and we registered all combinations of pairs $\binom{10}{2} = 45$. Comparing different tractographies is not a trivial problem however, we can use the bundle adjacency (BA) metric explained in section 4.6.4. We are happy to report the mean initial BA was $34.8\% \pm 8.0\%$ and the mean final BA after applying our direct registration method was $48.1\% \pm 6.1\%$. This was a statistically highly sig-

nificant improvement ($t_{\text{paired}}(44) = 11.2$, $p \leq 10^{-13}$). We are planning in the future to compare this registration method against other standard methods which are common in the literature.

4.10 Bundle Quality Control

In many parts of this document we did not consider short tracks. That is perfectly valid because (a) the longer tracks are more likely to be used as useful landmarks when comparing or registering different subjects because it is more likely for them to exist in most subjects, (b) removing short tracks facilitates the usage of distance based clustering (no need for manually setting the distance threshold) and interaction with the tractography, (c) someone would first want to see the overall representation of the tractography and go to the details later. Nonetheless, after having clustered the longer tracks there are many ways to assign the smaller bundles to their closest longer bundles. For this purpose, we recommend the use of different distance from MDF for example the minimum version of MAM referred to as MAM_{\min} (see Eq. 4.4).

Some simple strategies for clustering short fibres are discussed. The first is for unsupervised clustering and the second one is for supervised learning.

1. Cluster the long tracks using QB with distance threshold at 10 mm and then cluster the short tracks (<100 mm) to a lower threshold and assign them to their closest long track bundle from the first clustering using the MAM_{\min} distance.

2. Read the tractography of a single subject, use a tractographic atlas as the one created in section 4.8.2 and pick one or more representative tracks from that atlas. Then, find the closest tracks from the subject to that selected tracks using MDF. Cluster the closest tracks found from the previous step and for each one of these new skeletons find the closest tracks using MAM_{\min} distance. We should now have an amalgamation of shorter and longer fibres in one cluster.

An example of this second strategy is shown in Fig. 4.14. First we selected a single track from Arcuate Fasciculus. Next, we brought all tracks closer than 15 mm using the MDF distance. Then, we cluster the last tracks to 23 virtuals using QB with $\theta = 6.25$ mm. Finally, we bring all tracks with 6 mm (MAM_{\min} distance) from the entire tractography. Using this simple

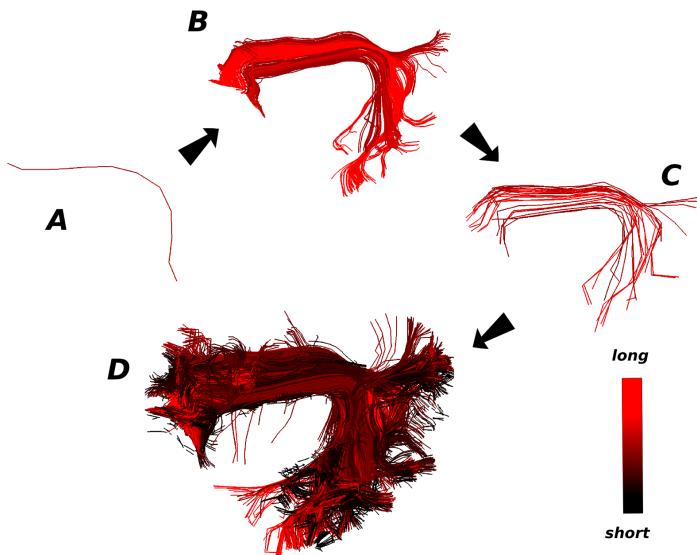


Figure 4.14: A simple and vigorous strategy for handling short and long tracks together by picking a track of interest from one of our atlases. Colourmap encodes track length. A: one selected atlas track, B: 245 subject tracks closer than 15 mm (MDF distance), C: B tracks clustered in 23 virtuals, D: 3,421 tracks closer than 6 mm (MAM distance) from the representatives of B are shown. A great number of short tracks have been brought together along with the tracks in B. In that way we managed to bring together an entire bundle consisting both of long and short fibres by just selecting one track.

strategy we were able to bring together from the entire data set and with minimum effort a bundle that consists of many shorter and longer tracks.

4.11 Discussion and conclusion

In this chapter we presented a novel and powerful algorithm – QuickBundles (QB). This algorithm provides simplifications to the old problem of white matter anatomy packing which has recently attracted much scientific attention; it can also be used for any trajectory clustering problem and it is recommended when large data sets are involved. QB can be used with all types of diffusion MRI tractographies which generate streamlines (e.g. probabilistic or deterministic) and it is independent of the reconstruction model.

In common with mainstream clustering algorithms such as k-means, k-centers and expectation maximization (EM), QB is not a global clustering method. It can give different results under different initial conditions

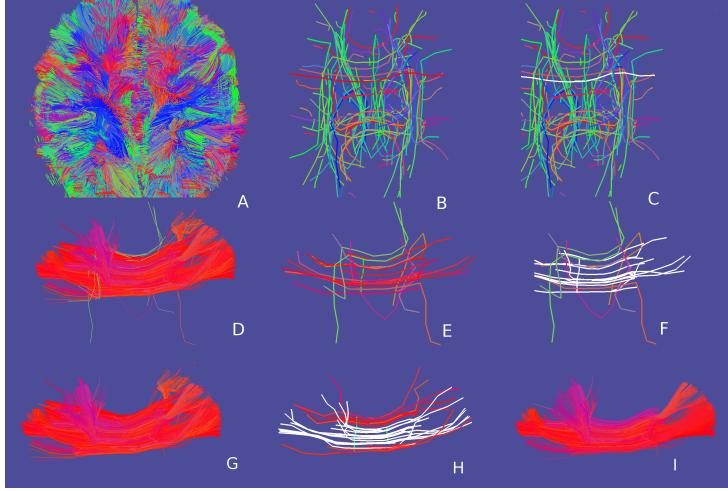


Figure 5.1: An example of how a medical practitioner can use our visualization software to select fibre bundles of their interest.

with defects in white matter architecture caused by disrupted connectivity [228].

Interactive Labeling. We are developing a scientific visualization tool that solves the problem of interacting with tractographies by creating real-time simplifications of the underlying anatomical bundle structures. The process that we propose works recursively: starting from a small number of clusters of streamlines the user decides which clusters to explore. Exploring a cluster means that the application re-clusters its content at a finer grained level in real-time. Of course these representative tracks are provided by QuickBundles which can cluster thousands of tracks in milliseconds.

Our approach starts by providing a first simplified version (see Fig. 5.1B) of the initial full tractography (see Fig. 5.1A). After visually inspecting the simplified tractography (see Fig. 5.1B) the practitioner can interactively select one or more representative tracks (see Fig. 5.1C, white track). When one or more representative tracks are selected the practitioner can see the content of the related clusters (see Fig. 5.1D). In order to explore the detailed structure of the selection the user may ask to re-cluster the selected BOIs into smaller clusters (see Fig. 5.1E). In that way one can further refine his previous selection. After selecting one or more of the small clusters through their representatives (see Fig. 5.1F, white tracks) the user can repeat the visual inspection step (Fig. 5.1G), and the re-clustering step (Fig. 5.1H) as required in order to unveil the local structures (Fig. 5.1I)

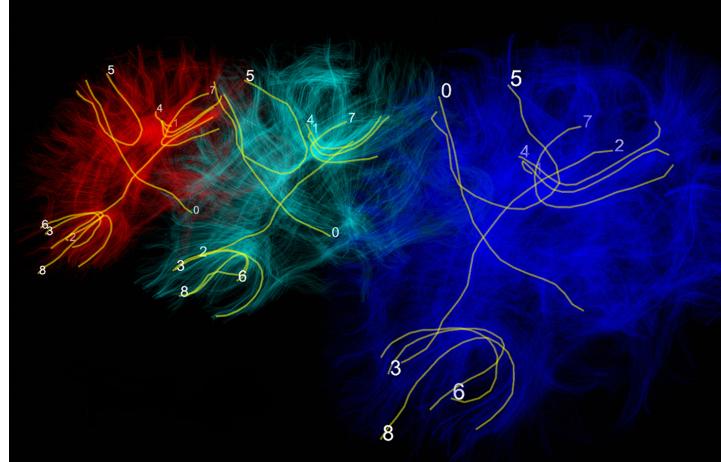


Figure 5.2: Direct track correspondence between different subjects.

which are interesting to their work.

Shape Correspondence. Using multi-brain visualization we were able to have a first investigation of shape correspondence [229] of tracks between different subjects (see Fig. 5.2). We would like to augment this first implementation from track to bundle correspondence. In other words a medical practitioner could select a representative track or a bundle in one subject and see in real-time the corresponding bundles in the other subjects. We imagine this as an amalgamation of the tools presented in Fig. 5.2 and 5.1 which are developed in DIPY and FOS.

Microstructure. Our work up till now has focused on estimating the structure of white matter in the brain from standard diffusion MRI acquisitions. However, these acquisitions provide voxel sizes of about 2 mm. Axon fibres, which are about one micron in diameter, are much smaller than single voxels. Nevertheless, diffusion MRI can provide information on the distribution of microstructural features, such as the fibre orientation, fibre diameter or density, within each voxel. We would also like to extend our work in that area of microanatomy tracking which has recently shown very interesting results [230], [231]. This high resolution domain require model-fitting of many parameters. It would be interesting to investigate if EIT or other non-parametric ideas could help alleviate this problem.

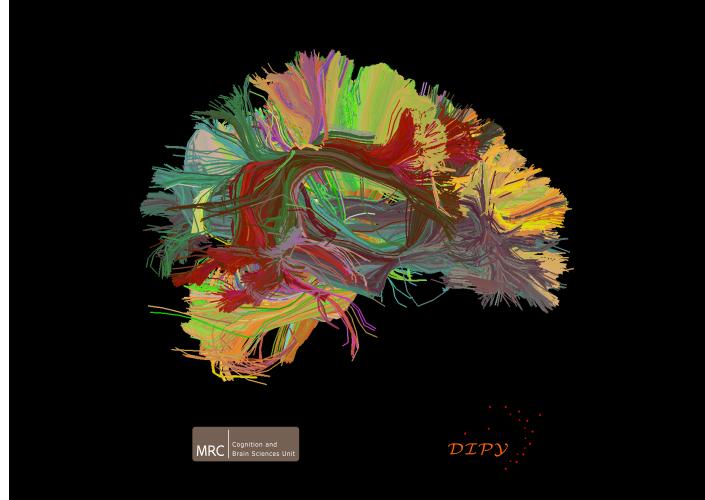


Figure 5.3: Every participant received a picture of his tractography as a gift for their help to our experiments.

5.4 New frontiers

White matter fibre crossings – from the voxel level to the tractography and bundle level – have been a major motivation for this thesis. In March 2012, Wedeen et al. [232] published in Science Magazine a fascinating work that reinforces the importance of these topics. They showed that they can identify in the tractographies of the brains of humans and other animals (in vivo and in vitro) fibre bundles which are in agreement with confocal microscopy and other staining techniques. The authors clarified that a grid-like structure is prevalent in the brain i.e. fibre bundles crossing in more areas than would previously have been expected. Furthermore, they also showed that the bundles curve more vigorously than previously understood. We believe that in this thesis we have significantly enhanced and extended the techniques that were used to establish these groundbreaking results, and have created a framework for them to be applied by the neuroscience community.

In bringing this thesis to a close we would like to thank the participants who took part in our imaging studies. To honour them we created special pictures of their tractographies like the one shown in Fig. 5.3 which were subsequently presented to them.