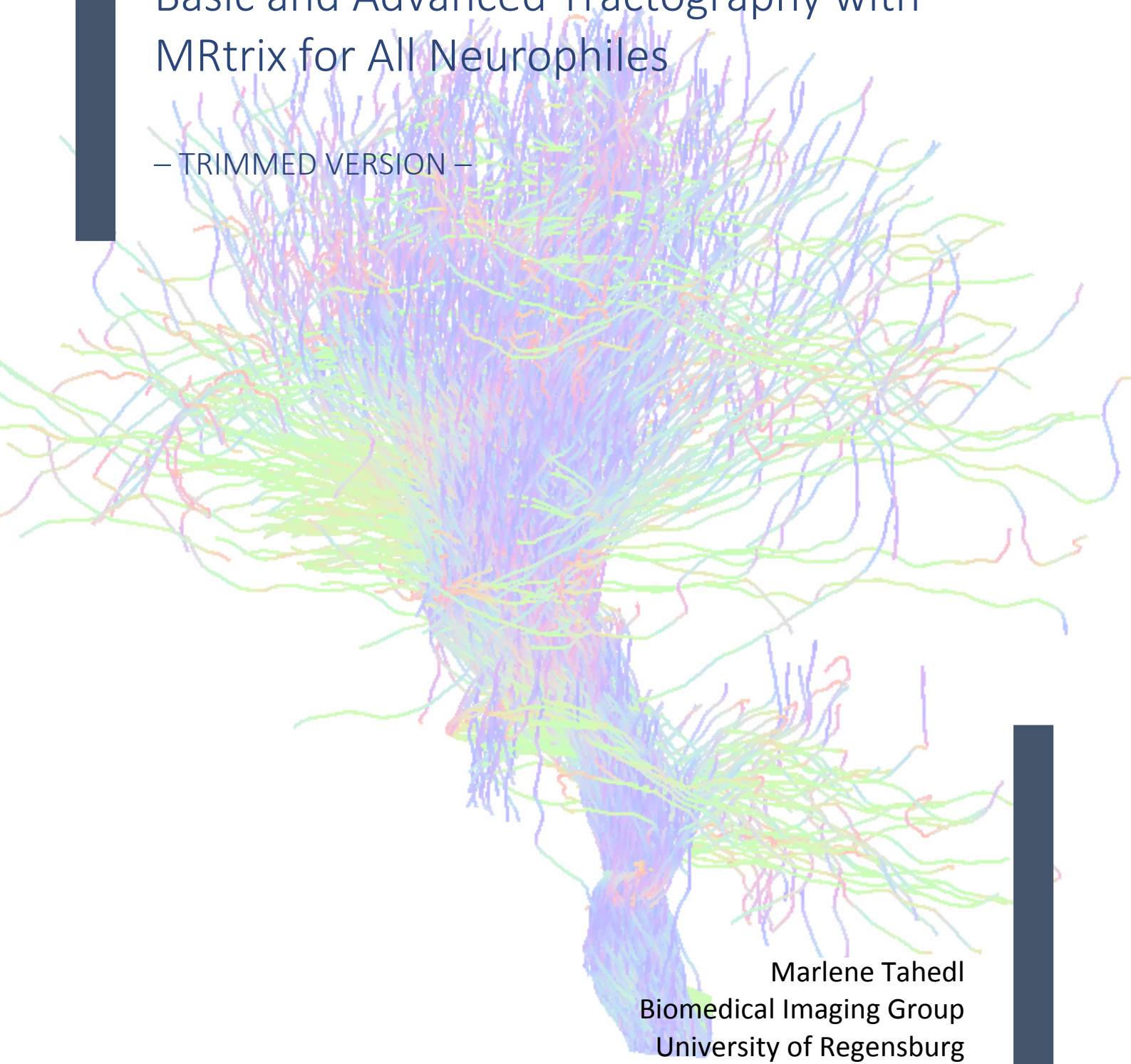
Beyond the tensor model

BATMAN

Basic and Advanced Tractography with MRtrix for All Neurophiles

– TRIMMED VERSION –



Marlene Tahedl
Biomedical Imaging Group
University of Regensburg

1 General tutorial info

1.1 About the tutorial

Welcome to *BATMAN* – Basic and Advanced Tractography Made Accessible with MRtrix – the trimmed version! If you want to learn about the latest approaches in diffusion weighted imaging (DWI), this is where you want to be ☺. Most students and researchers that have been working with DWI and/or tractography will likely have worked with diffusion tensor imaging (DTI) (Basser et al., 1994) to estimate how nerve fibers are oriented within the brain. However, the diffusion tensor model, on which DTI is based on, does not perform well in brain regions containing crossing or kissing (i.e. tangentially touching) fibers. The reason for this is that the tensor model approaches fiber orientation with an ellipsoid shape. Lately, with the introduction of Constrained Spherical Deconvolution (CSD), an alternative to the tensor model (Tournier et al., 2004, 2007), fiber tracking can now yield accurate results even in regions of crossing fibers (Farquharson et al., 2013; Tournier et al., 2008).
CSD can be performed with the open-access software MRtrix, and in this tutorial, we show how you can get to some really nice tractography results with MRtrix, starting from raw data. We also provide you with some test data to complete the tutorial. Besides CSD, we will also introduce you to some other algorithms, implemented in MRtrix, which will further improve tractography. Note that since this is only a trimmed version of the original tutorial, and we will not go into much detail behind the theory of these algorithms. Instead, we will focus on how to get good tractography done quickly. However, if you want to learn more about each of the analysis steps, we recommend to take a look at the full-length tutorial as well. You can find that version in the complementary tutorial data (see below).

1.2 About the tutorial data

1.2.1 Tutorial data structure

The tutorial data comes with the following directory structure:

```
BATMAN
├── DWI
│   ├── b0_PA
│   ├── b1000_AP
│   ├── b2000_AP
│   └── b3000_AP
├── Supplementary_Files
└── T1
```

The DWI and the T1 directories contain raw dicom data, the `Supplementary_Files/` directory contains all files that will be created throughout this tutorial as a backup for you! Additionally, you will find the extended version of this tutorial there.

1.2.2 Data acquisition

The data were acquired using a 3 Tesla Siemens Prisma 3T MRI system using a multi-shell acquisition scheme based on previous recommendations (Jeurissen et al., 2014; Tournier et al., 2013).
Additionally, a high-resolution MPRAGE anatomical image data set was acquired. The imaging parameters of that dataset encompassed: FoV 256 x 256 mm, matrix size 256 x 256, 160 slices, 1 mm isotropic resolution, TE/TR =3.67/1910 ms, flip angle 9°, 5 min acquisition scheme.

1.2.3 Subject information and data sharing policy

The tutorial data was acquired at the University of Regensburg in the year 2018. The subject was the author of this document, and I am happy to share this data freely with anyone interested in learning about advanced tractography with this tutorial. The data may be used for all analyses related to this tutorial. If you wish to use or share the data for purposes beyond this tutorial, please contact me.

1.3 Software requirements/more useful hints

In order to do this tutorial, **the following software must be installed** on your machine:

- **MRtrix, version 3.0_RC**
- **FSL**

Optionally and recommended, but not absolutely necessary to complete this tutorial, are

- **FreeSurfer**
- **ANTS**

2 Getting started: Preprocessing*

2.1 Preparatory steps

We recommend you define a project directory in which all analyses of this tutorial are carried out. This can be done by opening a terminal and typing, e.g.

```
tutorialDir=/Users/Marlene/MRI/projects/BATMAN
```

→ *Remember to adjust the path to wherever in your system you want to do this tutorial ;-)*

Now change into the `DWI/` directory of the tutorial data, e.g. by typing

```
cd $tutorialDir/DWI
```

and concatenate all b-images into MRtrix's-specific data format, the so-called `.mif` image ("**M**Rtrix **i**mage **f**ormat"; you can find more details on that data format here):

```
mrcat b1000_AP/ b2000_AP/ b3000_AP/ dwi_raw.mif
```

2.2 Denoising

- Purpose: Estimate the spatially varying noise map
- Main reference(s): Veraart et al., 2016b, 2016a

```
dwidenoise dwi_raw.mif dwi_den.mif –noise noise.mif
```

2.3 Unringing

- Purpose: remove Gibb's ringing artefacts
- Main reference(s): Kellner et al., 2016

```
mrdegibbs dwi_den.mif dwi_den_unr.mif –axes 0,1
```

→ *The "axes" option must be adjusted to your dataset: With this option, you inform the algorithm of the plane in which you acquired your data: –axes 0,1 means you acquired axial slices; -axes 0,2 refers to coronal slices and –axes 1,2 to sagittal slices!*

* Notice that since this is only the trimmed version of the tutorial, we do not provide further details on these algorithms, nor are we showing pictures of how these preprocessing steps change the results. If you are interested in that, we recommend you check out the original extended version of the tutorial, which can be found in the `Supplementary_Files/` directory of the tutorial data.

2.4 Motion and distortion correction

- Purpose: The purpose of this step should be pretty self-explanatory ;)
- Main reference(s):
  - EPI-distortion correction: Holland et al., 2010 (suggest using a pair of b0s in in phase encoding (PE) and reversed PE correction)
  - B0-field inhomogeneity correction: Andersson et al., 2003; Smith et al., 2004 (FSL's *topup* tool is called by MRtrix's preprocessing tool *dwipreproc*)
  - Eddy-current and movement distortion correction: Andersson and Sotiropoulos, 2016 (FSL's *eddy* tool is called by MRtrix's preprocessing tool *dwipreproc*)

For EPI distortion correction, we will use a pair of b0 images (Holland et al., 2010): One which was acquired in phase encoded (PE) direction (i.e. in the same direction as the non-b0 images) and one in the reversed PE direction. For each PE direction, we acquired several images and will average them to get cleaner results:

```
dwiextract dwi_den_unr.mif - -bzero | mrmath – mean mean_b0_AP.mif
–axis 3
```

→ *"-" denotes that the output will be piped as an input into the next command, which follows after "|". Like this, no additional output files are created!*
→ *"`-axis 3`" denotes that the mean image will be calculated along the third axis*

Now calculate the mean image of the b0s in the reversed phase-encoded direction (in our case PA):

```
mrconvert b0_PA/ - | mrmath – mean mean_b0_PA.mif –axis 3
```

Further preprocessing requires to concatenate the two mean b0-images into one file:

```
mrcat mean_b0_AP.mif mean_b0_PA.mif –axis 3 b0_pair.mif
```

Now we are ready to run motion and distortion correction:

```
dwipreproc dwi_den_unr.mif dwi_den_unr_preproc.mif –pe_dir AP –rpe_pair
–se_epi b0_pair.mif –eddy_options " --slm=linear"
```

→ *This command will take several hours to compute. You will best run this overnight. If you don't want to wait so long for the purpose of this tutorial, you can find the output of this command in the tutorial data, in the subfolder `Supplementary_Files/`. There, you'll find the output of the above command, `dwi_den_unr_preproc.mif`. Just copy this file into your `DWI/` folder and continue with the next step.*

2.5 Bias field correction

- Purpose: Improve brain mask estimation
- Main reference: Tustison et al., 2010

You can improve downstream brain mask estimation with bias field correction, preferably used with ANTS (which must be installed on your system to complete this step):

```
dwibiascorrect –ants dwi_den_unr_preproc.mif
dwi_den_unr_preproc_unbiased.mif –bias bias.mif
```

2.6 Brain mask estimation

- Purpose: Create a binary mask of the brain. Downstream analyses will be performed within that mask to improve biological plausibility of streamlines and reduce computation time.

```
dwi2mask dwi_den_unr_preproc_unbiased.mif
mask_den_unr_preproc_unb.mif
```

→ *Note that we compute the mask on the upsampled DW-image!*

→ *Remember to check the brain mask in MRview. You can do this as outlined above (biasfield correction)*

3 Fiber orientation distribution

3.1 Response function estimation

- Purpose: Estimate different response functions for the three different tissue types: white matter (WM), gray matter (GM), and cerebrospinal fluid (CSF).
- Main reference: Dhollander et al., 2016

Now we are finished with data preprocessing and can start to create streamlines. To get there, we first need to estimate the orientation of the fiber(s) in each voxel. We will use CSD to do that (Tournier et al., 2004, 2007). The first step is to estimate a so-called "response function" (RF) – a kernel for deconvolution. Latest studies have shown that estimating different response functions for different tissue types (white and gray matter, CSF) improves tractography. This technique is known as multi-shell multi-tissue CSD (MSMT; Jeurissen et al., 2014), and is best performed with DW data with diverse b-values.

```
dwi2response dhollander dwi_den_unr_preproc_unbiased.mif wm.txt
gm.txt csf.txt
```

→ *Note that the outputs of this command are order-dependent: The first output will always be the response function of white matter (`wm.txt`), the second of the gray matter (`gm.txt`) and the third of CSF (`csf.txt`)!*

3.2 Estimation of Fiber Orientation Distributions (FOD)

- Purpose: In every voxel, estimate the orientation of all fibers crossing that voxel.
- Main reference(s): Tournier et al., 2004, 2007

Having the different response functions for the different tissue-types at hand, we can finally estimate the orientation distribution of the fibers in each voxel (FOD):

```
dwi2fod msmt_csd dwi_den_unr_preproc_unbiased.mif –mask
mask_den_unr_preproc_unb.mif wm.txt wmfod.mif gm.txt gmfod.mif csf.txt
csffod.mif
```

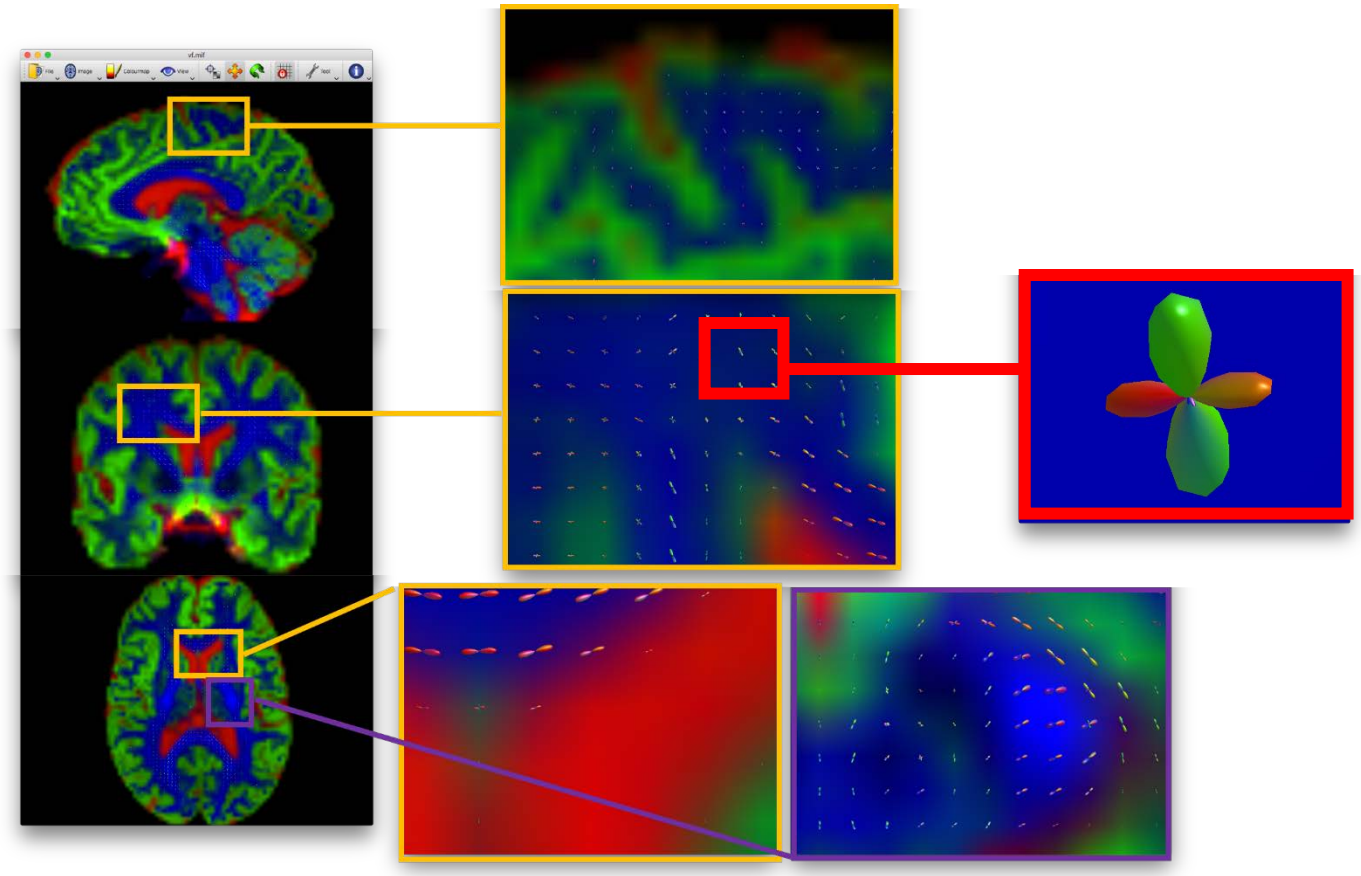


Figure 1. Quality check of Fiber Orientation Distribution (FOD) estimation. Red insert shows the FOD of a single voxel. Orange boxes show regions which contain crossing fibers by anatomy, and are therefore especially fit for quality check of FOD. Also, note that FOD is almost only performed within the white matter boundaries (blue areas). CSF (red areas) and gray matter (green areas) are free from FODs, which is consistent with anatomy! This is a result of multi-shell multi-tissue constrained spherical deconvolution!

You can check the results of FOD, for example with the following command:

```
mrconvert –coord 3 0 wmfod.mif - | mrcat csffod.mif gmfod.mif – vf.mif
mrview vf.mif –odf.load_sh wmfod.mif
```

Figure 1 shows some results of this process. In that figure, we marked some locations which are suited for a quality check (see the highlighted areas of Figure 1).

3.3 Intensity Normalization

- Purpose: Correct for global intensity differences (especially important when performing group studies!)

```
mtnormalise wmfod.mif wmfod_norm.mif gmfod.mif gmfod_norm.mif csffod.mif
csffod_norm.mif –mask mask_den_unr_preproc_unb.mif
```

4 Creating a whole-brain tractogram

4.1 Preparing Anatomically Constrained Tractography (ACT)

- Purpose: Increase the biological plausibility of downstream streamline creation.
- Main reference(s): Smith et al., 2012

4.1.1 Preparing a mask for streamline termination

Before creating streamlines, we recommend to include Anatomically Constrained Tractography (ACT) into the pipeline (Smith et al., 2012): The idea of ACT is to identify such streamlines which end at biologically implausible regions (e.g. the CSF) and reject those streamlines. ACT requires a highresolution T1-weighted image, which then needs to be segmented into different tissue types. With a few steps, you can make the raw T1-data fit to be used for ACT:

```
mrconvert ../T1/ T1_raw.mif
5ttgen fsl T1_raw.mif 5tt_nocoreg.mif
```

This image is in a different space as the DWI and must therefore be registered. This can be done, for example, with the following set of commands:

```
dwiextract dwi_den_unr_preproc_unbiased.mif - -bzero | mrmath – mean
mean_b0_preprocessed.mif –axis 3

mrconvert mean_b0_preprocessed.mif mean_b0_preprocessed.nii.gz
mrconvert 5tt_nocoreg.mif 5tt_nocoreg.nii.gz

flirt –in mean_b0_preprocessed.nii.gz –ref 5tt_nocoreg.nii.gz –interp
nearestneighbour –dof 6 –omat diff2struct_fsl.mat

transformconvert diff2struct_fsl.mat mean_b0_preprocessed.nii.gz
5tt_nocoreg.nii.gz flirt_import diff2struct_mrtrix.txt

mrtransform 5tt_nocoreg.mif –linear diff2struct_mrtrix.txt –inverse
5tt_coreg.mif
```

4.1.2 Preparing a mask of streamline seeding

Now we will create a mask to define where our streamlines should *start*! Again, from anatomy, we know that the gray-matter/white-matter-boundary should be a reasonable starting point for that. In MRtrix, it is very easy to create a mask of the gray-matter/white-matter-boundary, which we can later use as a mask for streamline seeding. Type:

```
5tt2gmwmi 5tt_coreg.mif gmwmSeed_coreg.mif
```

.

→ Note that since we are creating the seed mask from the co-registered `5tt`-image, the resultant seed mask will be co-registered as well.

4.2 Creating streamlines

Ready for the big moment? Now we have everything set for an (almost) perfect streamline creation! The relevant command for that in MRtrix is `tckgen`. You have to specify how many streamlines you want to create, and we recommend to do at least 10 million. Also, don't forget to specify the ACT option, which we prepared above:

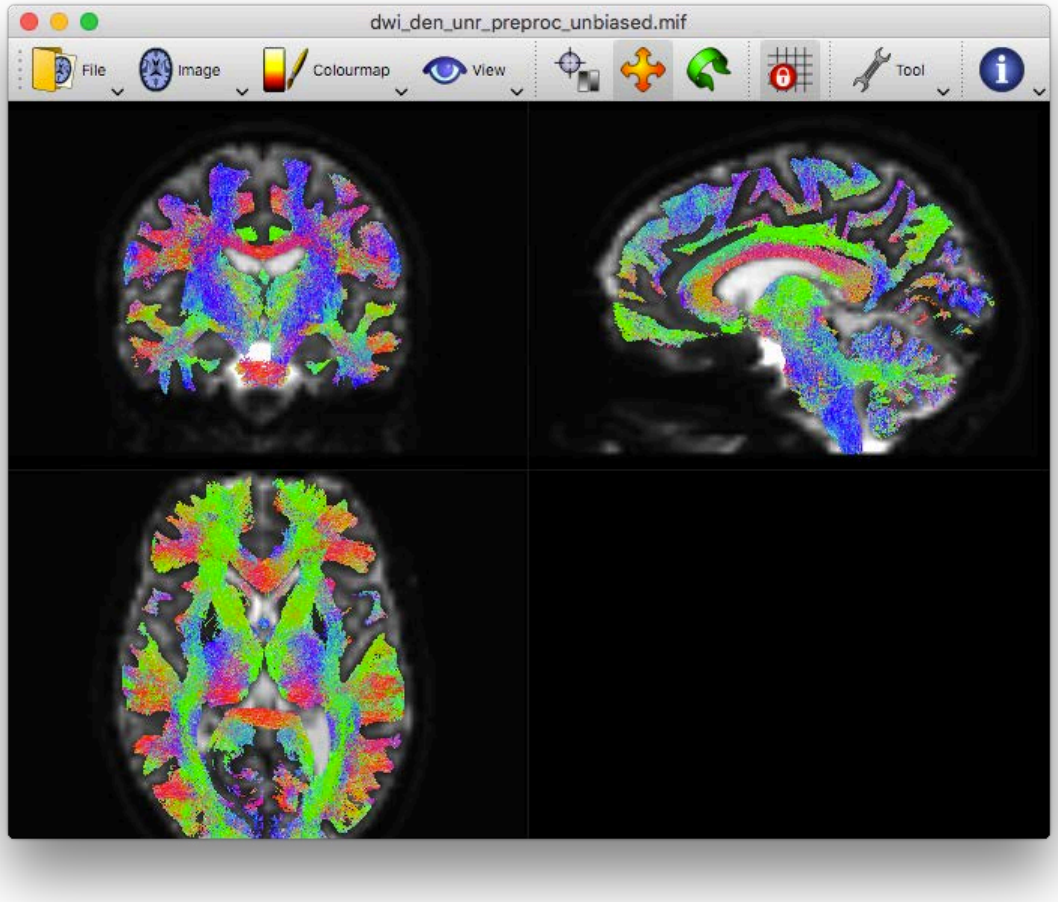


Figure 2. Results of streamline creation, displayed as an overlay on the DW image. The tracks are color-coded (i.e. red indicates streamlines from right to left, green from anterior to posterior, blue from superior to inferior. Note that the streamlines terminate reliably at the gray-matter/white-matter-boundary. This is a result of ACT.

```
tckgen –act 5tt_coreg.mif –backtrack –seed_gmwmi gmwmSeed_coreg.mif
–select 10000000 wmfod_norm.mif tracks_10mio.tck
```

→ *This step will take a few hours to complete. If you don't have that much time to wait, copy the resulting file from the `Supplementary_Files/` directory!*
→ *Note that in MRtrix, files with streamlines have the ending `.tck` (for "tracks")*

Once the product is finished you should view the results. This is best done with a subset of 10 million tracks rather than with all to save RAM. You can extract a random set of tracks with

```
tckedit tracks_10mio.tck –number 200k smallerTracks_200k.tck
```

→ *Note that MRtrix understands "k" to be 1000*

and view them with

```
mrview dwi_den_unr_preproc_unbiased.mif –tractography.load
smallerTracks_200k.tck
```

.

What you see should look like Figure 2.

4.3 Reducing the number of streamlines

- Purpose: Filtering the tractograms to reduce CSD-based bias in overestimation of longer tracks compared to shorter tracks; reducing the number of streamlines
- Main reference: Smith et al., 2013

In our track-file, the density of long tracks is still overestimated compared to short tracks, which is a by-product of CSD. We can correct for that bias with an algorithm called *spherical-deconvolution informed filtering of tracks* (SIFT, Smith et al. 2013). Type

```
tcksift –act 5tt_coreg.mif –term_number 1000000 tracks_10mio.tck
wmfod_norm.mif sift_1mio.tck
```

→ *Again, this command takes several hours to complete. Run it overnight, or copy the resultant file (`sift_1mio.tck`) from the `Supplementary_Files/` directory of the tutorial data!*

4.4 Region-of-interest filtering of tractograms

If you want to select tracks pass a certain region of interest (ROI) only, you can do this with either providing a binary mask of that ROI or by providing coordinates of your ROI. We will demonstrate here how to filter the corticospinal tract (CST) with coordinates. First, identify a ROI through which the CST passes. To do that, load the filtered track-file on the DW image and find a suited spot (Figure 3A):

```
mrview dwi_den_unr_preproc_unbiased.mif –tractography.load
smallerSIFT_200k.tck &
```

→ *To select a ROI, it's good to see all 3 planes of the volume. Therefore, choose the "Ortho view" option from the "View" toolbar.*

A B

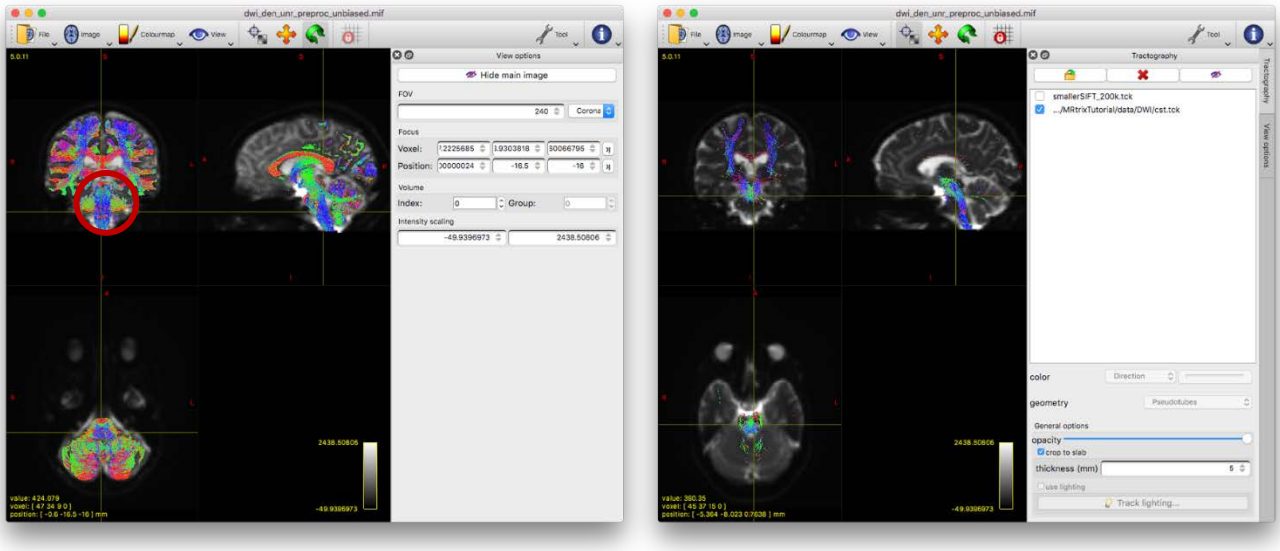


Figure 3. Region-of-interest-based (ROI) filtering of the whole-brain tractogram. (A) The first step is to find the coordinates in real-world (i.e. millimeter space) of the track-file which you want to filter on. The coordinates can be found at the lower left corner, where it says "Position". We selected a ROI right before the corticospinal tract divides into a right and a left branch (red circle). (B) After `tckedit` is completed, you can manually load the newly created track-file via the "Tractography" option from the "Tool" menu. Click on the folder-icon to load other track-files, and/or (dis)enable currently loaded track-files by (un)ticking them.

Scrolling around the volume, we can see that the position [–0.6, –16.5, –16.0] (**real-word space**, not voxel space!) is be a good ROI for filtering for the CST (Figure 3A). We can now do the coordinate-based ROI-filtering with

```
tckedit –include -0.6,-16.5,-16.0,3 sift_1mio.tck cst.tck
```

→ *The coordinates (x,y,z) have to be provided as a comma-separated list without spaces with four values: the fourth value indicates the radius of the sphere your ROI should have (in millimeters).*

View the results:

```
mrview dwi_den_unr_preproc_unbiased.mif –tractography.load cst.tck
```

What you see should look similar to Figure 3B (shown in "Ortho view" selected from the "View" menu). To see the track in 3D, choose "Volume render" from the "View" menu. Play around with the "Clip planes" option from the "Tool" menu (> "View options") to get a good view of the CST, e.g. like in Figure 4A. Alternatively, hit the M key to hide the main image completely (Figure 4B).

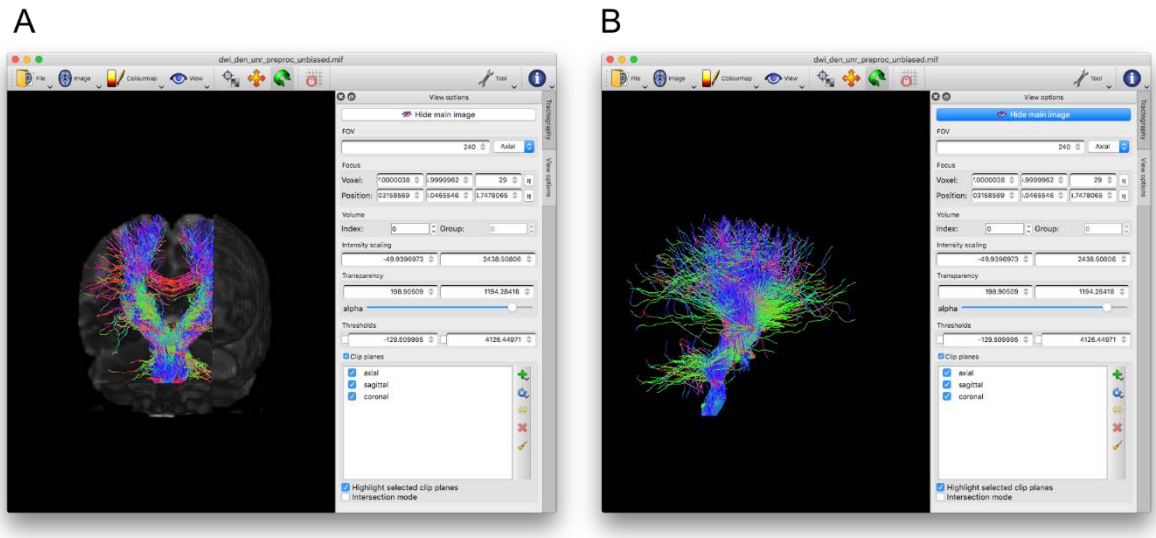


Figure 4. The volume render mode. To get there, first select the "Volume render" option from the "View" menu. Then, select the "View options" from the "Tool" menu to see the GUI which you see on the right menu in (A) and (B). In the lower half, note the option "Clip planes". Add some planes: In (A), we selected one axial, one sagittal and one coronal slice. To change the relative position of each slice, mark them on the menu and push arrows up/down/right/left, until you are happy with what you see. If you want the main image to disappear completely, hit the "M" key (B). If you hit "M" again, the main image will reappear. Note that (B) shows the same track-file as (A), but from a lateral view.

5 Connectome construction

This part of the tutorial deals with how to create and visualize an atlas-based SC matrix in MRtrix. We will demonstrate this with the recently published atlas *Human Connectome Project Multi-Modal Parcellation 1.0* ("HCP MMP 1.0") (Glasser et al., 2016a).

5.1 Preparing an atlas for structural connectivity analysis

- Purpose: Obtain a volumetric atlas-based parcellation image, co-registered to diffusion space for downstream structural connectivity (SC) matrix generation
- Main reference: Glasser et al., 2016a (for the atlas used here for SC generation)

To construct an atlas-based SC matrix in MRtrix, you need a volumetric atlas-based parcellation image which needs to fulfill certain criteria. For the sake of this tutorial, we prepared such an image for you, and you will find it in the `Supplementary_Files/` directory. It's called `hcpmmp1_parcels_coreg.mif`. If you want a detailed description of the necessary steps to get there, you will find the instructions in the appendix of the extended version of this tutorial.

5.2 Matrix generation

- Purpose: Gain quantitative information on how strongly each atlas region is connected to all others; represent it in matrix format

Once you have the volumetric atlas-based parcellation image, it's very easy to obtain an SC matrix:

```
tck2connectome –symmetric –zero_diagonal -scale_invnodevol
sift_1mio.tck hcpmmp1_parcels_coreg.mif hcpmmp1.csv –out_assignment
assignments_hcpmmp1.csv
```

→ We like to use the two options "`-symmetric`" and "`-zero_diagonal`", which symmetrize the resulting SC matrix (if not specified, the lower triangular will be left blank) and set the diagonal to zero. We find that this makes subsequent manipulation of the matrix easier in external software, e.g. MATLAB. The option `–scale_invnodevol` scales the track counts according to the volume of each node.
→ We will need the output of the "`-out_assignment`" option in the next section. That option is not necessary for SC matrix generation, but for the moment just specify it and check out the next section to see what it is and what it is good for!

This will only take a few seconds to complete. The resulting file is a `.csv` file, which you can use for further analyses with your preferred software. Figure 5A and B show two exemplary visualizations of that matrix in MATLAB (The Mathworks, Natic USA), using different thresholds. Figures 5C and D show the distribution of the element values of the matrix.

5.3 Selecting connections of interest

We will now look at how to select streamlines corresponding specific brain regions. We will look at two different strategies: a) Selecting all streamlines that connect two regions and b) selecting all streamlines that emerge from a region of interest (ROI). For both strategies, the function `connectome2tck` can be used.

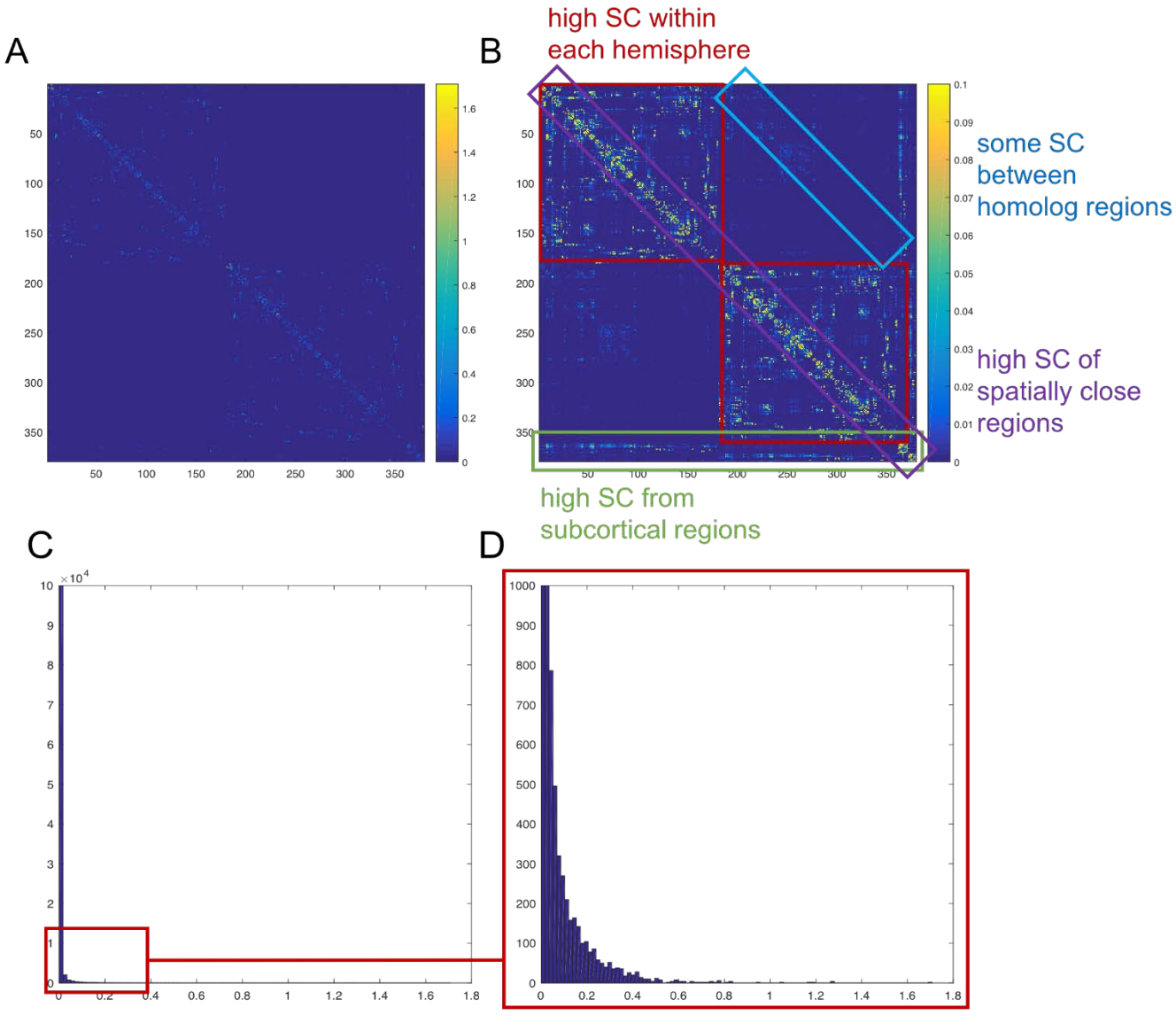


Figure 5. Structural connectivity (SC) matrix and its properties. (A) shows the SC matrix with the colorbar ranging from 0 to the highest element value. (B) shows the same matrix with a reduced colorbar. Now, more distinct features of the matrix are recognizable, such as increased intra- and interhemispheric SC, high SC of subcortical regions and high SC between spatially close regions (colored boxes). (C) and (D) show histograms of all matrix elements' distribution. The distribution follows a power law with many zero-elements (C). (D) shows the same histogram with a reduced y-axis to enhance the distribution of the non-zero elements.

5.3.1 Extracting streamlines between atlas regions

We will demonstrate how to extract streamlines between atlas regions with the left and right motor cortices. First, we need to identify the integers those regions have assigned in the volumetric atlas-based parcellation image. This information is stored in the color lookup table, which you will find in the `Supplemenatry_Files/` directory (called "`hcpmmp1_ordered.txt`"). The names for the atlas regions are not always self-explanatory, but together with the information of the supplementary information of the original publication of the article (Glasser et al., 2016b), we find that the core regions of the motor cortex are simply called "L_4" and "R_4", respectively for the hemispheres. In the color lookup table, we find that those regions have the integers 8 and 188 assigned to them.

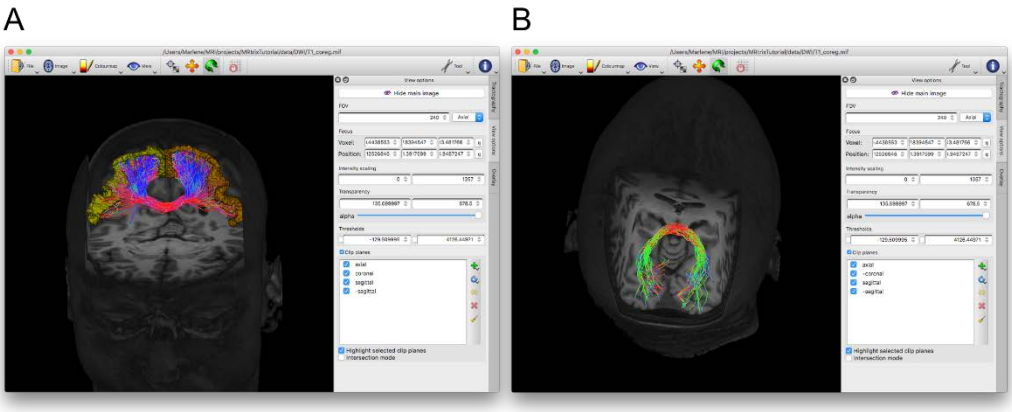


Figure 6. Selecting streamlines between two atlas regions. (A) shows all streamlines connecting homolog regions of the motor cortex. Additionally, the analyzed atlas regions are plotted as an overlay in yellow (right motor cortex) and orange (left motor cortex). (B) shows the streamlines connecting the left and right V1.

This is all information we need and we can do the ROI-tracking by typing:

```
connectome2tck –nodes 8,188 –exclusive sift_1mio.tck
assignments_hcpmmp1.csv moto
```

→ With the option `-exclusive`, you specify that you only want to select tracks between the two regions
→ Note that for this command, we need a file that specifies the starting and ending atlas regions of each streamline. Here, we pass this information with `assignments_hcpmmp1.csv`, which we created earlier.
→ The last argument specifies the prefix that the resultant file will have. Since we specified "moto", our resultant file will be called "moto8-188.tck"

Now look at your results (Figure 6A):

```
mrview dwi_den_unr_preproc_unbiased.mif –tractography.load moto8-188.tck
```

and select the "Volume render" mode. Choose the "Clip planes" option as discussed above and try to make your image look like Figure 6A. Quite a nice track, isn't it? ☺ Figure 6B shows the same analysis with V1 of the visual cortex. If you want to, you can try to do this now yourself!

5.3.2 Extracting streamlines emerging from a region of interest

If you wish to visualize all streamlines that emerge *from* a ROI, the procedure is similar. First, decide which region(s) to analyze. For this example, we decided to choose the thalamus. We can analyze the left and the right thalamus at the same time. In the color lookup table, you will find that the corresponding indices are 362 (left thalamus) and 372 (right thalamus). Type

```
connectome2tck –nodes 362,372 sift_1mio.tck assignments_hcpmmp1.csv
–files per_node thalamus
```

→ Note that unlike when selecting pathways between two regions, here we must not use the option –exclusive! Only then we will get all streamlines emerging from our ROI!
→ With the `–files per_node` option, we specify that we want our resulting files should be one file for each region/node that we analyze, which includes all streamlines emerging from those regions. If we do not specify this, we will get an individual file for each other region of ROI connects to, which in the case of the thalamus are hundreds of files!

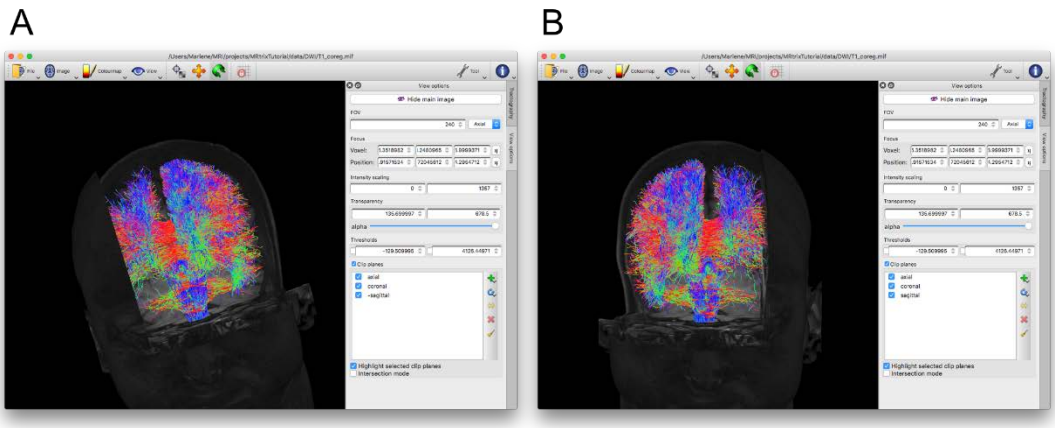


Figure 7. Selecting all streamlines that emerge from a region of interest. (A) shows all streamlines emerging from the left thalamus, (B) shows all streamlines emerging from the left thalamus. Note that although both thalami connect strongly to many cortical region, the connection density is higher on the ipsilateral hemisphere respectively!

Now look at the results. Start with the left thalamus:

```
mrview T1_coreg.mif –tractography.load thalamus362.tck
```

and switch to the "Volume render" mode and play around with the "Clip plane" option to see something like Figure 7A. The thalamus connects to many cortical regions, but the connection density on the ipsilateral side is even stronger. Now, load the result from the right thalamus via the "Tractography" option from the "Tool" menu. The respective file is called `thalamus372.tck`. Remember to unselect the track-file corresponding to the left thalamus (simply untick the file `thalamus362.tck`). Figure 7B shows what you should see: Like the left thalamus, also the right thalamus connects to most other cortical regions, with higher density on the ipsilateral side – in this case the right hemisphere!

6 Connectome visualization tool

6.1 Getting familiar with the tool

- Purpose: Visualize the SC matrix as nodes and edges using MRtrix's built-in connectome visualization tool
- Main reference(s): e.g. Rubinov and Sporns, 2010 (on brain network analyses)

For visualization of the SC matrix, MRtrix provides its own tool, called the connectome visualization tool. You can initialize the tool with the following command:

```
mrview hcpmmp1_parcels_coreg.mif –connectome.init
hcpmmp1_parcels_coreg.mif –connectome.load hcpmmp1.csv
```

→ You have to specify the parcellation image twice, otherwise MRview is confused and won't load anything, or crash. As a result, the parcellation image will be loaded as a 2D file into MRview. However, the connectome visualization tool is designed for three dimensions, therefore it doesn't make much sense to display the 2D-parcellation. Just hit "M" to hide that image and continue!

What you now see is a 3D-representation of the SC matrix (Figure 8A)! The little spheres represent the nodes. The geometric positioning of the nodes corresponds to the center of each atlas region on a 3D volume. The lines represent the edges, and their color indicates the connection strength. So basically, all information from the SC matrix is there. However, sometimes all information is too much, and in our case, the connectome looks messy. So now it's all about selecting which sets of information to represent, and making it prettier ☺! To do this, select the "Connectome" option from the "Tool" menu. A GUI appears, on which you can change the properties of both node and edge streamline visualization.

In terms of node visualization, there is another option called "Vector file". With this option, you can provide a vector file with as many entries as you have atlas regions. The data format must be logical, which means that those atlas regions which you want to hide are assigned "0" and those regions which you want to display get a "1". Note that the *n*th entry of that vector file corresponds to the *n*th atlas region of the parcellation image, defined in the (ordered) color lookup table. To test this option, you can now either create such a vector file yourself (e.g. in MATLAB) and load it in, but we also provide you with such a file in the `Supplementary_Files/`. It's called `randvec.csv`, and is a random selection of nodes, so don't be surprised that what you see may look scattered.

When you're done, focus on the GUI's section "Edge visualization". At first, we will get rid of all streamlines with low connection strength. Remember from Figure 5C and D that there are very many connections with very low connection strength. You can make such connections disappear by manipulating the "Threshold": Again, from Figure 5C and D we can see that a connection strength of 0.1 should reduce the number of visible edges substantially. Also, try out different values! Note that as you increase the threshold, you will also see nodes to which no edges connect. If you want to hide them and only see nodes with an edge, you can do that by changing the "Node visibility" to "Degree>1". Check out the "Edge smoothing" option as well, if you want a thicker representation of the edges. Does your visualization look similar to what you see in Figure 8D? Play around with the other options as well! When you're done, move on to the next section to learn about how to visualize nodes and edges more "anatomically" than just spheres and lines!

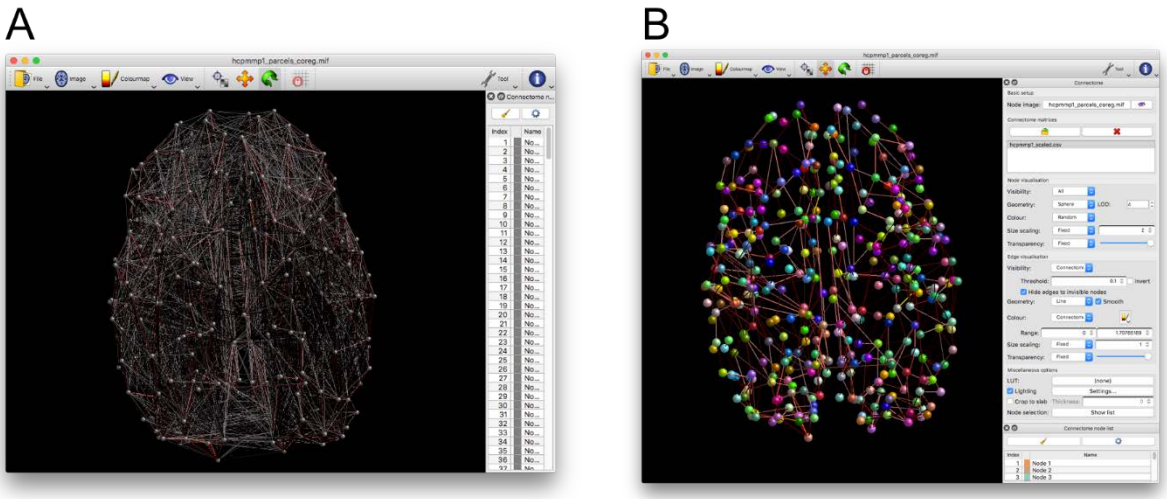


Figure 8. First steps in using the connectome visualization tool. After initializing the tool, you will see something like in (A). After increasing the threshold for edge visibility and smoothing the edges, as well as choosing a different ("random") color for the nodes and increasing their scaling factor to 2, it already looks clearer and prettier (B).

6.2 Node and edge geometry: Meshes and streamlines

We will now demonstrate how to make the nodes and streamlines look more anatomically plausible then just spheres and lines. Take a look at Figure 9 to see what we are heading at. You can get the **nodes** look like that by creating a mesh file from your atlas-based parcellation image. And this is how that's done:

```
label2mesh hcpmmp1_parcels_coreg.mif hcpmmp1_mesh.obj
```

→ *Currently, MRtrix only supports the* `.obj` *-file format for meshes!*

Now switch back to MRview. There, change the node "Geometry" to "Mesh". Select the `hcpmmp1_mesh.obj` image and there you go.

To get the **edges** look like Figure 9, we can use the `connectome2tck` command again:

```
connectome2tck sift_1mio.tck assignments_hcpmmp1.csv exemplar –files
single –exemplars hcpmmp1_parcels_coreg.mif
```

→ *"exemplar" is the prefix name for your output file, such that the resulting file will be called "*`exemplar.tck`*"*
→ *With "-files single" you specify that your output should be merged into one file. This is necessary for the downstream visualization in MRview.*

You can use the new file as follows: In the "Edge visualization" section, choose "Streamline" from the "Geometry" menu. Then, select the newly-created exemplars file (`exemplar.tck`). If you still have selected "All" for "Node visibility", you will probably see no change even after the file is loaded. If that is the case, get rid of some nodes (e.g. by switching back to the `randvec.csv` file). Figure 9 shows an example of what your visualization could look like. Once you have done this, also try changing the "Edge geometry" to "Streamtubes"!

6.3 Manipulating the visualization to match a research question

We finish the tutorial with a demonstration of how to analyze a simple research question with MRtrix. Together, we will try to answer the question of which structural connections are the strongest in our brain (at least – in this tutorial – in my brain ☺). Think about this question for a moment: What does it mean and which structural connections do you think are the strongest in the brain and why?

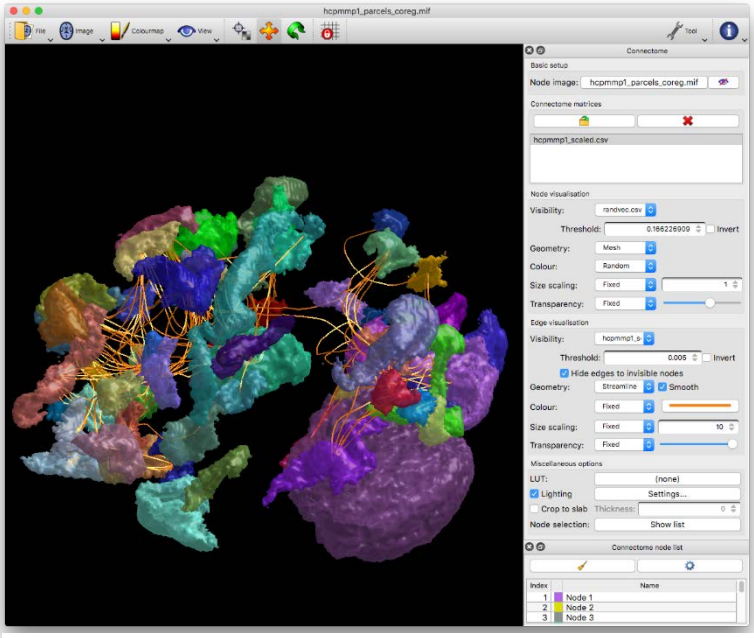


Figure 9. Making node and edge visualization more anatomical with the connectome visualization tool. Work through the tutorial to find out how to get there!

When you are set, think about how to operationalize our research question: We want to know which connections are the strongest, which means nothing else than we want to filter out those connections which have the highest values in our SC matrix (Figure 5A and B). Therefore, all we need to do is adjust the threshold of our "Edge visibility". Additionally, we want our "Node visibility" to match the edge visibility, i.e. we only want to see those nodes to which edges connect. Therefore, we change the "Node visibility" to "Degree>=1". So now start to change your threshold. Increase the threshold slowly to note the effects better, for example by steps of 0.1, since the highest matrix element value is roughly 1.7, so work your way towards that number. In Figure 10B, we show the edge visibility thresholded by 1.0. Now, very few edges and nodes are left, including regions on the visual cortices, as well as some regions around the motor cortices. Additionally, a few regions of the inferior frontal cortex survive this threshold, but only on the left hemisphere. Well, which main functions are located at the inferior frontal cortex of the left hemisphere only (at least in right-handed people, like myself)? Of course, language! So, there you have the best-connected regions of the brain: regions corresponding to vision, motor function, as well as language.
Now we want to see the corresponding edges as well. Decrease the node transparency (Figure 10C). Now you see the edges with the highest connectivity strengths, those edges to the very right of the connectivity distribution (Figure 5D). Obviously, the remaining edges are very short. More importantly, they connect nodes *within* rather than *between* brain lobes. Now we can answer our initial question completely: The (or rather: my ☺) brain's strongest connections are those between subregions of the visual, motor and language areas. What was your guess?

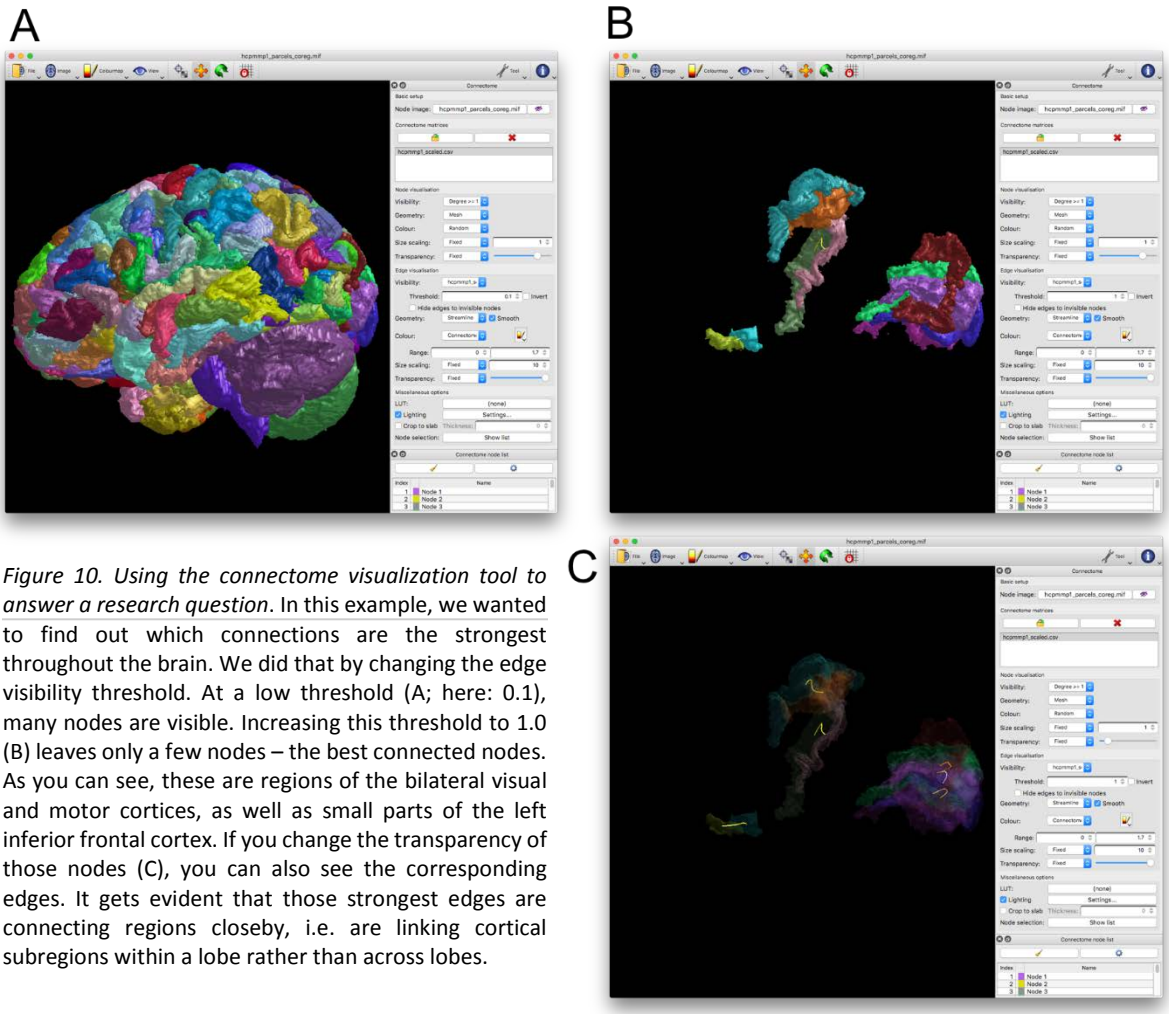


Figure 10. Using the connectome visualization tool to answer a research question. In this example, we wanted to find out which connections are the strongest throughout the brain. We did that by changing the edge visibility threshold. At a low threshold (A; here: 0.1), many nodes are visible. Increasing this threshold to 1.0 (B) leaves only a few nodes – the best connected nodes. As you can see, these are regions of the bilateral visual and motor cortices, as well as small parts of the left inferior frontal cortex. If you change the transparency of those nodes (C), you can also see the corresponding edges. It gets evident that those strongest edges are connecting regions closeby, i.e. are linking cortical subregions within a lobe rather than across lobes.

7 List of references

Andersson, J.L.R., Skare, S., Ashburner, J., 2003. How to correct susceptibility distortions in spin-echo echo-planar images: Application to diffusion tensor imaging. Neuroimage 20, 870–888. https://doi.org/10.1016/S1053-8119(03)00336-7

Andersson, J.L.R., Sotiropoulos, S.N., 2016. An integrated approach to correction for off-resonance effects and subject movement in diffusion MR imaging. Neuroimage 125, 1063–1078. https://doi.org/10.1016/j.neuroimage.2015.10.019

Basser, P.J., Mattiello, J., LeBihan, D., 1994. MR diffusion tensor spectroscopy and imaging. Biophys. J. 66, 259–67. https://doi.org/10.1016/S0006-3495(94)80775-1

Bihan, D. Le, Poupon, C., Amadon, A., 2006. Artifacts and Pitfalls in Diffusion MRI. J. Magn. Reson. Imaging 24, 478–488. https://doi.org/10.1002/jmri.20683

Caruyer, E., Lenglet, C., Sapiro, G., Deriche, R., 2013. Design of multishell sampling schemes with uniform coverage in diffusion MRI. Magn. Reson. Med. 69, 1534–40. https://doi.org/10.1002/mrm.24736

Dhollander, T., Connelly, A., 2016. Generating a T1-like contrast using 3-tissue constrained spherical deconvolution results from single-shell (or multi-shell) diffusion MR data. Conf. ISMRM Work. Break. Barriers Diffus. MRI.

Dhollander, T., Ra, D., Connelly, A., 2018. Accuracy of response function estimation algorithms for 3-tissue spherical deconvolution of diverse quality diffusion MRI data Accuracy of response function estimation algorithms for 3-tissue spherical deconvolution of diverse quality di usion MRI data 3–6.

Dhollander, T., Raffelt, D., Connelly, A., 2016. Unsupervised 3-tissue response function estimation from single-shell or multi-shell diffusion MR data without a co-registered T1 image, in: Conference: ISMRM Workshop on Breaking the Barriers of Diffusion MRI. Lisbon, Portugal, p. 5 pp.

Dietrich, O., Raya, J.G., Reeder, S.B., Ingrisch, M., Reiser, M.F., Schoenberg, S.O., 2008. Influence of multichannel combination, parallel imaging and other reconstruction techniques on MRI noise characteristics. Magn. Reson. Imaging 26, 754–762. https://doi.org/10.1016/j.mri.2008.02.001

Farquharson, S., Tournier, J.D., Calamante, F., Fabinyi, G., Schneider-Kolsky, M., Jackson, G.D., Connelly, A., 2013. White matter fiber tractography: why we need to move beyond DTI. J. Neurosurg. 118, 1367–1377.

Glasser, M.F., Coalson, T.S., Robinson, E.C., Hacker, C.D., Harwell, J., Yacoub, E., Ugurbil, K., Andersson, J., Beckmann, C.F., Jenkinson, M., Smith, S.M., Van Essen, D.C., 2016a. A multi-modal parcellation of human cerebral cortex. Nature 536, 171–178. https://doi.org/10.1038/nature18933

Glasser, M.F., Coalson, T.S., Robinson, E.C., Hacker, C.D., Harwell, J., Yacoub, E., Ugurbil, K., Andersson, J., Beckmann, C.F., Jenkinson, M., Smith, S.M., Van Essen, D.C., 2016b. A multi-modal parcellation of human cerebral cortex. Nature 536, 171–178. https://doi.org/10.1038/nature18933

Holland, D., Kuperman, J.M., Dale, A.M., 2010. Efficient correction of inhomogeneous static magnetic field-induced distortion in Echo Planar Imaging. Neuroimage 50, 175–183. https://doi.org/10.1016/j.neuroimage.2009.11.044

Jeurissen, B., Leemans, A., Tournier, J.D., Jones, D.K., Sijbers, J., 2013. Investigating the prevalence of complex fiber configurations in white matter tissue with diffusion magnetic resonance imaging. Hum. Brain Mapp. 34, 2747–2766. https://doi.org/10.1002/hbm.22099

Jeurissen, B., Tournier, J.D., Dhollander, T., Connelly, A., Sijbers, J., 2014. Multi-tissue

constrained spherical deconvolution for improved analysis of multi-shell diffusion MRI data. Neuroimage 103, 411–426. https://doi.org/10.1016/j.neuroimage.2014.07.061

Kellner, E., Dhital, B., Kiselev, V.G., Reisert, M., 2016. Gibbs-ringing artifact removal based on local subvoxel-shifts. Magn. Reson. Med. 76, 1574–1581. https://doi.org/10.1002/mrm.26054

Mori, S., Tournier, J.D., 2013. Introduction to Diffusion Tensor Imaging, 1st edition. Academic Press, Oxford.

Rubinov, M., Sporns, O., 2010. Complex network measures of brain connectivity: uses and interpretations. Neuroimage 52, 1059–69. https://doi.org/10.1016/j.neuroimage.2009.10.003

Schenck, J.F., 1996. The role of magnetic susceptibility in magnetic resonance imaging: MRI magnetic compatibility of the first and second kinds. Med. Phys. 23, 815–50. https://doi.org/10.1118/1.597854

Smith, R.E., Tournier, J., Calamante, F., Connelly, A., 2013. NeuroImage SIFT : Spherical-deconvolution informed fi ltering of tractograms. Neuroimage 67, 298–312. https://doi.org/https://doi.org/10.1016/j.neuroimage.2012.11.049

Smith, R.E., Tournier, J.D., Calamante, F., Connelly, A., 2012. Anatomically-constrained tractography: Improved diffusion MRI streamlines tractography through effective use of anatomical information. Neuroimage 62, 1924–1938. https://doi.org/10.1016/j.neuroimage.2012.06.005

Smith, S.M., Jenkinson, M., Woolrich, M.W., Beckmann, C.F., Behrens, T.E.J., Johansen-Berg, H., Bannister, P.R., De Luca, M., Drobnjak, I., Flitney, D.E., Niazy, R.K., Saunders, J., Vickers, J., Zhang, Y., De Stefano, N., Brady, J.M., Matthews, P.M., 2004. Advances in functional and structural MR image analysis and implementation as FSL. Neuroimage 23 Suppl 1, S208-19. https://doi.org/10.1016/j.neuroimage.2004.07.051

Tournier, J.-D., Calamante, F., Gadian, D.G., Connelly, A., 2004. Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. Neuroimage 23, 1176–85. https://doi.org/10.1016/j.neuroimage.2004.07.037

Tournier, J.-D., Mori, S., Leemans, A., 2011. Diffusion tensor imaging and beyond. Magn. Reson. Med. 65, 1532–1556. https://doi.org/10.1002/mrm.22924

Tournier, J., Yeh, C., Calamante, F., Cho, K., 2008. Resolving crossing fibres using constrained spherical deconvolution: Validation using diffusion-weighted imaging phantom data. Neuroimage 42, 617–625. https://doi.org/10.1016/j.neuroimage.2008.05.002

Tournier, J.D., Calamante, F., Connelly, A., 2013. Determination of the appropriate b value and number of gradient directions for high-angular-resolution diffusion-weighted imaging. NMR Biomed. 26, 1775–1786. https://doi.org/10.1002/nbm.3017

Tournier, J.D., Calamante, F., Connelly, A., 2007. Robust determination of the fibre orientation distribution in diffusion MRI: Non-negativity constrained super-resolved spherical deconvolution. Neuroimage 35, 1459–1472. https://doi.org/10.1016/j.neuroimage.2007.02.016

Tuch, D.S., 2004. Q-ball imaging. Magn. Reson. Med. 52, 1358–72. https://doi.org/10.1002/mrm.20279

Tustison, N.J., Avants, B.B., Cook, P.A., Zheng, Y., Egan, A., Yushkevich, P.A., Gee, J.C., 2010. N4ITK: improved N3 bias correction. IEEE Trans. Med. Imaging 29, 1310–20. https://doi.org/10.1109/TMI.2010.2046908

Veraart, J., Fieremans, E., Novikov, D.S., 2016a. Diffusion MRI noise mapping using random matrix theory. Magn. Reson. Med. 76, 1582–1593. https://doi.org/10.1002/mrm.26059

Veraart, J., Novikov, D.S., Christiaens, D., Ades-Aron, B., Sijbers, J., Fieremans, E., 2016b. Denoising of diffusion MRI using random matrix theory. Neuroimage 142, 394–406. https://doi.org/10.1016/j.neuroimage.2016.08.016

Wedeen, V.J., Wang, R.P., Schmahmann, J.D., Benner, T., Tseng, W.Y.I., Dai, G., Pandya, D.N., Hagmann, P., D'Arceuil, H., de Crespigny, A.J., 2008. Diffusion spectrum magnetic resonance imaging (DSI) tractography of crossing fibers. Neuroimage 41, 1267–1277. https://doi.org/10.1016/j.neuroimage.2008.03.036