ExploreDTI

Manual created by Alexander Leemans

Most recent version of *ExploreDTI* is: v4.8.6.

This manual is showcasing the very basic tools of *ExploreDTI*. It is far from complete, but it's a start! Any feedback is always welcome...

Q&A and discussion forum: https://groups.google.com/group/e_dti.
Notifications and updates: https://twitter.com/ExploreDTI.

ExploreDTI was released at the ISMRM 2009 meeting. The abstract can be found here. The reference is:

Leemans A, Jeurissen B, Sijbers J, Jones DK. "ExploreDTI: A graphical toolbox for processing, analyzing, and visualizing diffusion MR data". Proceedings of the 17th Scientific Meeting, International Society for Magnetic Resonance in Medicine, Honolulu, USA, p. 3537, 2009.

CONTENT

Contact info

Alexander Leemans
PROVIDI Lab
Image Sciences Institute
University Medical Center Utrecht
Heidelberglaan 100
Utrecht - 3584 CX
The Netherlands
http://www.PROVIDI-Lab.org
http://twitter.com/Alex_Leemans

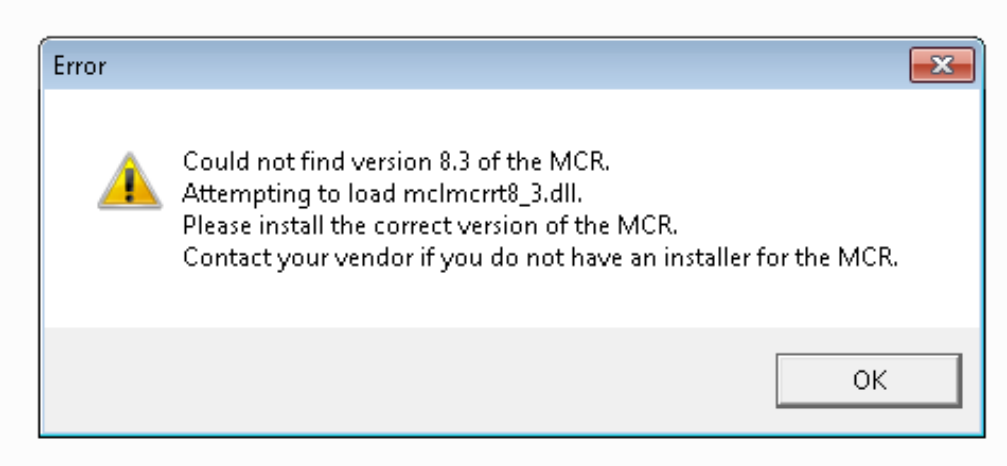
1. Installing / starting *ExploreDTI*

Standalone versions of *ExploreDTI* are available for Windows, Linux & Mac (only for 64 bit OS). With the standalone executable, there is no need have Matlab installed.

If you *do* have Matlab (R2010a or later) installed, there is also a P-code version of *ExploreDTI* that runs natively in Matlab and, therefore, in principal at least, can run on any platform (32 bit and 64 bit OS). At this stage, source code of *ExploreDTI* is not released.

It is recommended to install *ExploreDTI* on a computer with high-performance[1] specs: (i) multi-core CPU; (ii) RAM > 16 GB; and (iii) dedicated graphics cards / GPU's.

If you already tried to run the executable 'MainExploreDTI' before having read this part of the manual, you may have encountered the following error message (or something similar – MCR version may be different):



Reading section **B)** below will clarify things...

A) Using *ExploreDTI* natively in Matlab

Put the 'Source' folder and the 'MainExploreDTI.p' file in your 'Matlab path' and type 'MainExploreDTI' in the command prompt (and press 'enter')... that should do it!

B) Installing & starting the standalone of *ExploreDTI* (no Matlab needed)…

Make sure your computer is up-to-date re java (e.g., http://www.java.com), graphics card drivers (e.g., for Intel graphics cards, http://downloadcenter.intel.com), OpenGL libs (e.g., http://www.opengl.org/), etc. Otherwise, you might get error messages or just weird bugs. *Perhaps only perform these updates, if need be*.

The standalone version of *ExploreDTI* is currently being developed in Matlab 8.3.0.532 (R2014a). The executable, "MainExploreDTI", only works AFTER you installed the (free!) Matlab-Common-Runtime (MCR) libs of that Matlab version – you can download the one for your OS of interest here (~ 600 MB):
http://www.ExploreDTI.com/matlab/64/MCR_R2014a_win64_installer.zip (Windows – 64 bit)
http://www.ExploreDTI.com/matlab/64/MCR_R2014a_glnxa64_installer.zip (Linux – 64 bit)
http://www.ExploreDTI.com/matlab/64/MCR_R2014a_maci64_installer.zip (Mac – 64 bit)

To install the MCR libs file, simply follow the instructions. Note that MCR libraries from other Matlab versions are probably not compatible with these standalone versions of *ExploreDTI*. Again, note that you do NOT need a Matlab license to run *ExploreDTI*! For Linux/Mac, if you choose an installation directory for the MCR that is different from the default one, you need to use this path in the "run_MainExploreDTI.sh" as well.

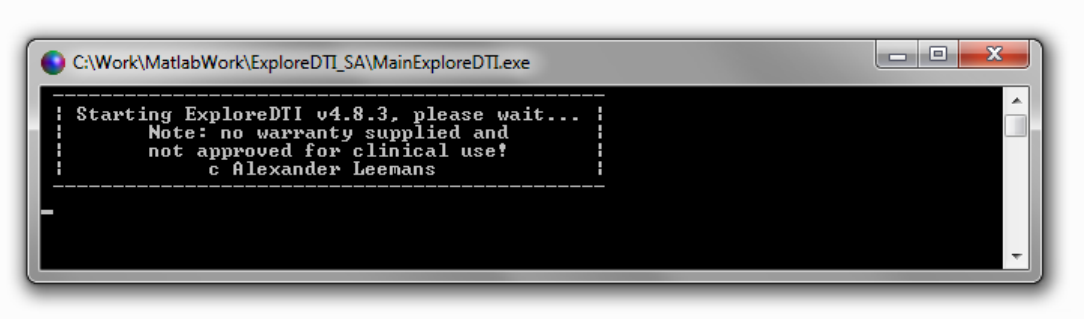
[1] High-performance is defined here as: *"Not getting annoyed by slow graphics rendering or insufficient RAM"*.

C) Running *ExploreDTI* …

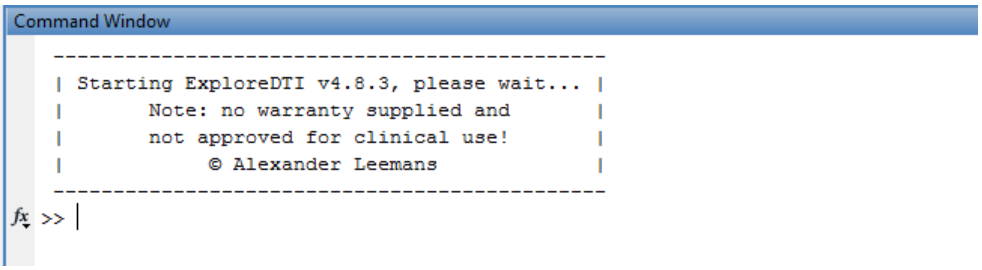
E.g. for Windows, after installing MCR libs – and perhaps the typically required Windows reboot – put the executable file 'MainExploreDTI.exe' in a folder, such as “C:\ExploreDTI". Then, double-click the executable 'MainExploreDTI.exe' (the actual *ExploreDTI* program!). This might take several seconds (or even a minute or two when executed for the first time), as it loads the required Matlab libs.

If things go well, for Windows, the screens below should appear – or at least something similar depending on the exact version of *ExploreDTI*:

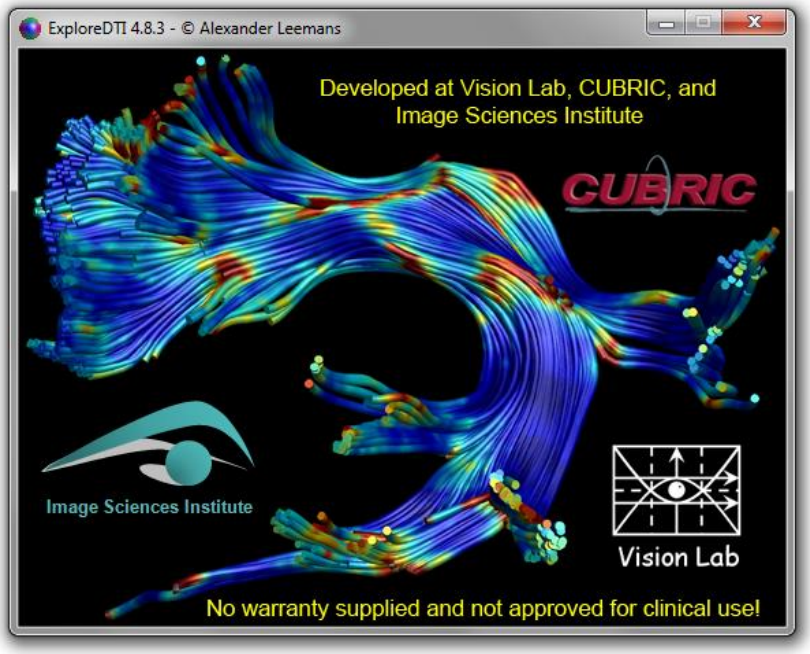
1) The command prompt appears first (read only): it will provide information about the progress of certain calculations and reports ‘errors/bugs/crashes’ (btw, feedback of the latter is useful). If you get errors at this stage, you might need to look at the ‘install’ sections above (or just consult the forum…).



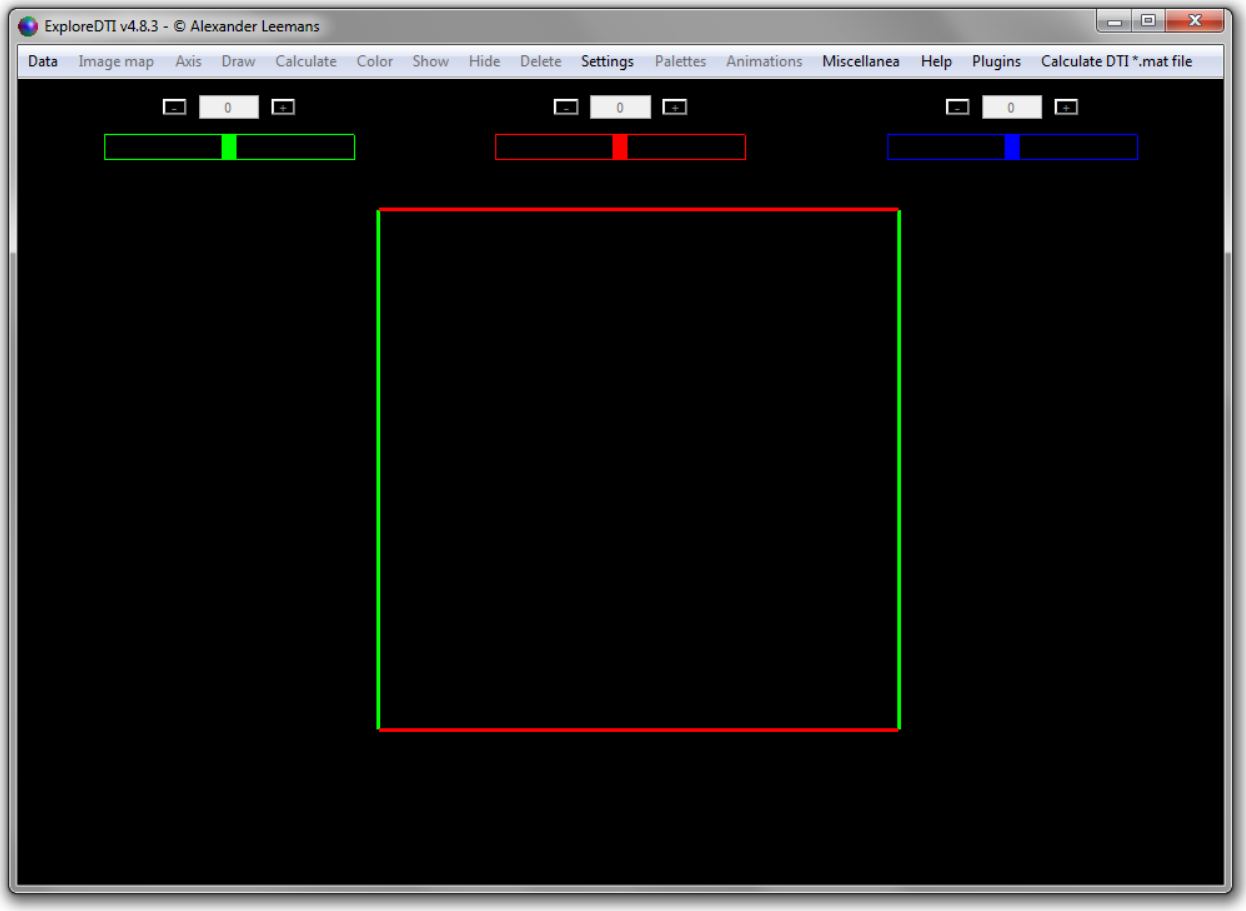
If you are running *ExploreDTI* in Matlab, it will appear in the Command Window:



2) Then, a ‘splash-window’ should appear. If so, things are looking good!



3) Finally, the actual *ExploreDTI* program should start up. If you see the window below, you are ready to go!



Most menu items are disabled at start-up to prevent you from accidentally crashing *ExploreDTI*. Exceptions are 'Data', 'Settings', 'Miscellanea', 'Plugins', 'Help', and 'Calculate DTI *.mat file' (if you run *ExploreDTI* in Matlab, the other menu items are not relevant, as they are built-in by Matlab itself). Note that menu item names may change in future releases. Also, the version number (here, v4.8.3) might be different.

If you are eager to try things out, download the example DTI *.mat file and load it via the 'Data' menu item: http://www.exploredti.com/exampledataset.htm. Tip: immediately after loading, the menu item 'Palettes' becomes available. Useful ones to start messing about are the 'Axis' and 'Draw' palettes.

If you want to use your own data, see the next section.

2. Converting your DWI's to an "*ExploreDTI* file" (*.mat)

This section describes how you can convert your raw diffusion data to the format of *ExploreDTI*. The end-result of this conversion step is a DTI *.mat file that contains pre-processed data, such as the diffusion tensor, eigenvalues, eigenvectors, mean diffusivity, fractional anisotropy, the DWIs, B-matrix, gradient directions, etc.

Background

Most likely, you conducted a diffusion MRI experiment (DTI, DKI, HARDI, DSI, …), which has resulted in a folder (perhaps with sub-folders) of (un-)sorted dicom files. This type of format is not really practical (typically, each 2D image slice has its own file), so it usually converted to more manageable formats (where all slices/volumes are put in a single file), such as *.par/*.rec (Philips-specific), *.nii(.gz) (NIFTI), *.hdr/*.img (analyze), etc.

As an example, consider a typical diffusion MRI acquisition with 30 diffusion weighted images (DWI's) and 3 non-DWI's, i.e., the "b = 0 s/mm^2" (or b0 in short) images, with each 3D image consisting of 60 slices. Compare the speed of loading 1980 (= [30 + 3] x 60) separate dicom images vs. a single 4D *.nii file… it *does* make a difference! Mainly due to this efficiency issue, I opted for *.nii as the standard input file format to start processing with *ExploreDTI*.

All DWIs (including the b0 images) are assumed to be included in the 4D *.nii file *with the b0 images at the beginning*. If the b0 images are not at the beginning, there is a plugin 'Plugins' → 'Shuffle/select 3D volume(s) in 4D *.nii file(s)' that you can use reorder the b0 images to the beginning. Alternatively, you can use the 'Sort DWI *.nii file(s) wrt b-values' plugin assuming you have a b-matrix *.txt file (see below).

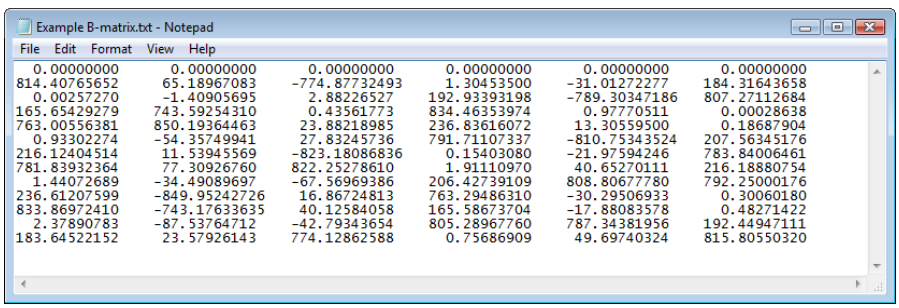
Converting data (dicom/hdr-img/2dseq/par-rec) to NIFTI (*.nii)

In my opinion, a great tool to convert dicom files to NIFTI files is Chris Rorden's "dcm2nii" tool: http://www.cabiatl.com/mricro/mricron/dcm2nii.html. I have included this tool and MRIcron itself (allows one to view *.nii files), version 15 October 2008, as plugins into *ExploreDTI* (see menu item 'Plugins' → 'Convert …' → 'Dicom folder(s) to *.nii files'). Similar for the other data formats (img/hdr, par/rec, '2dseq'), there are conversion tools to get the 4D *.nii data. In any case, the end result should be a 4D *.nii file. Important – as always – check the *.nii file for left/right flipping!

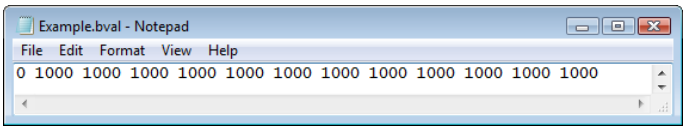
If you are familiar with Matlab, you may want to use the NIFTI tools developed by Jimmy Shen: http://www.mathworks.nl/matlabcentral/fileexchange/8797. This might be useful when you do not have dicom files, but some other format. Read the data into Matlab and then export them to a 4D *.nii file with these tools.

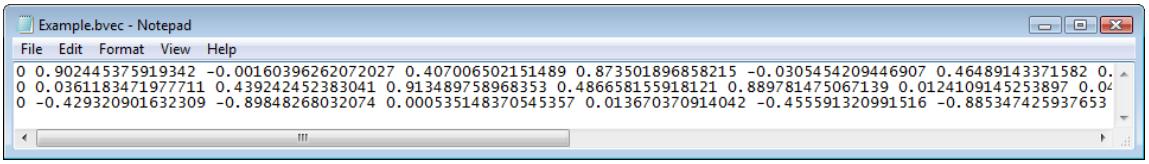
The B-matrix

To convert your *.nii data to an *ExploreDTI* 'DTI *.mat file', you also need the B-matrix (*or* the gradient directions and the b-value). I will spare you the details on how to derive the B-matrix – an example (B-matrix with 1 b0 and a b-value of 1000 s/mm^2) is given below (left), derived from the 12 gradient directions, shown on the right.



The nice thing about “dcm2nii” (see previous section) is that it can produce “*.bval” and “*.bvec” files (see below).

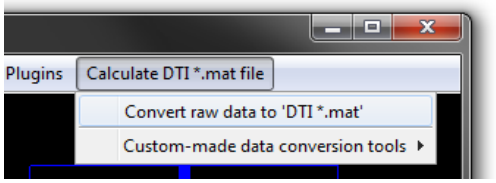




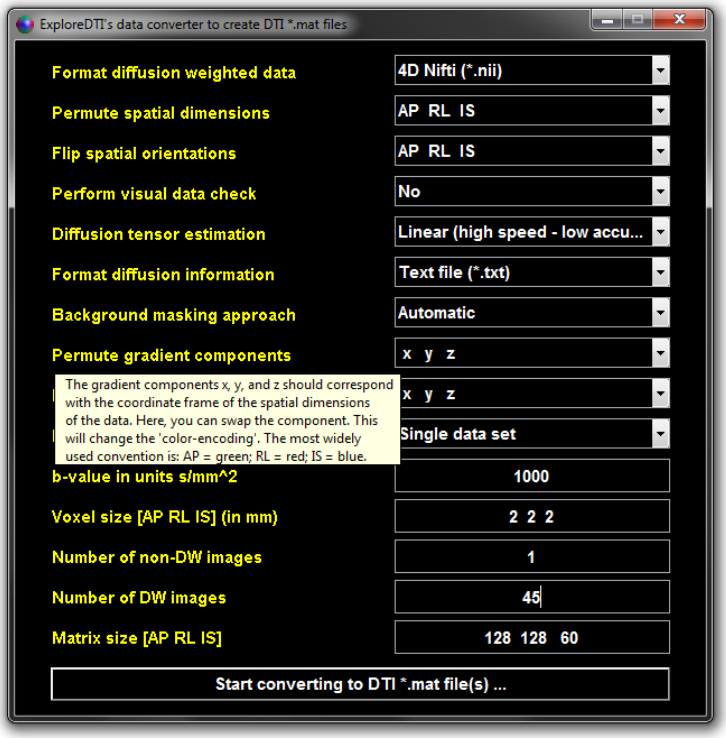
From these “*.bval” and “*.bvec” files, the B-matrix *.txt file can be calculated (see menu item ‘Plugins’ → ‘Convert …’ → ‘*.bval/*.bvec to B-matrix *.txt file(s)’).

Making the *.mat file for *ExploreDTI*

OK – now the last step! By now, you should have a 4D *.nii file and its corresponding B-matrix (or gradient) *.txt file (as described/shown above). Now you can use the 'Convert raw data to DTI *.mat' tool (see below) to create the DTI *.mat file, which can then be loaded into *ExploreDTI*. You should *NOT* try to run the ‘Custom-made data conversion tools’ from this same menu item, as they are *custom-made* and, therefore, will *NOT* be suited for your data - crashes & incorrectly converted data are not excluded then!



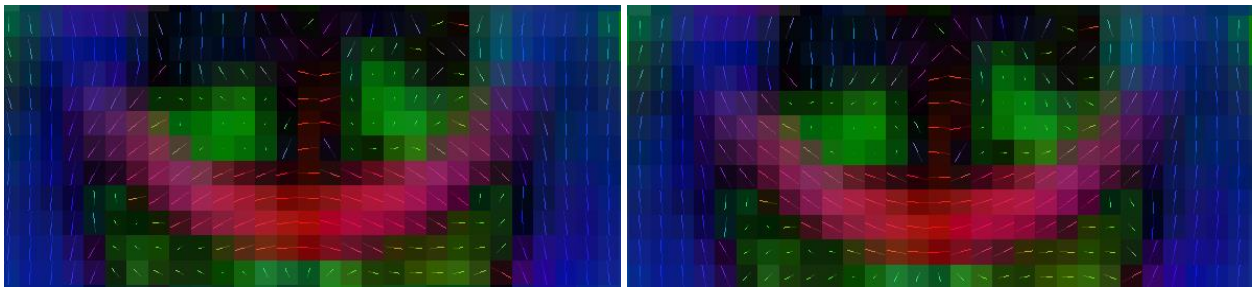
The screen below pops up with detailed info given when you go ‘mouse-over’ on the yellow text:



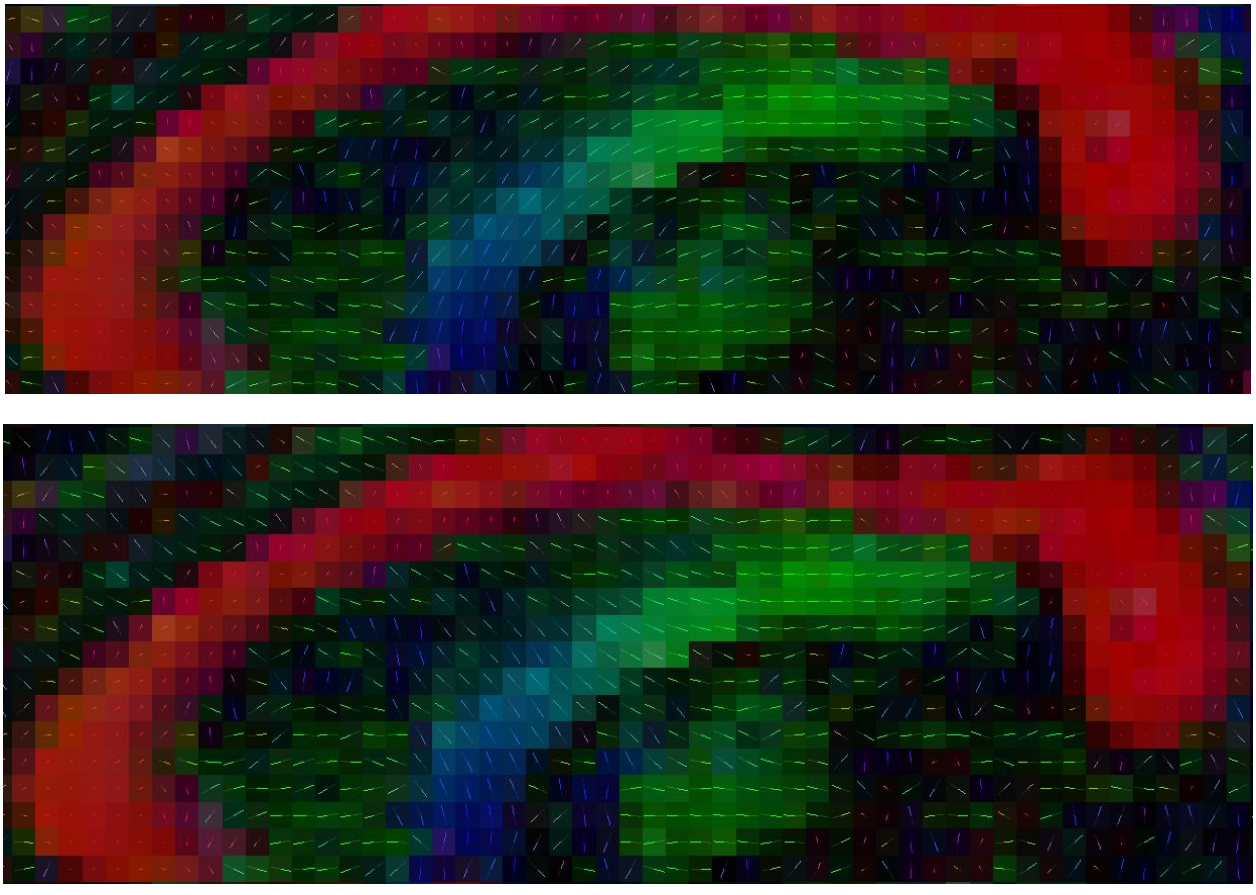
One thing you should definitely check is: *MAKE SURE LEFT-RIGTH ORIENTATION IS NOT FLIPPED*! Btw, I use the radiological convention, i.e. left and right in the image represent right and left in the data set, respectively, that is, if you look at it in a 2D axial view with default settings. If you are not sure, just 'rotate' the image, so you can unambiguously 'see' the 3D nature of the data/object. My advice: always attach a marker on the right side of the subject (e.g., a vitamin pill) – you will never switch sides ever again ;-)! And secondly, **use the plugin "Flip/permute dimensions in 3D/4D *.nii file(s)" to make all your nifti files – not just the DWIs, but also any other data such as T1, T2, etc. – "ExploreDTI compatible"** to begin with… this will avoid unexpected axis flips (& permutations) in any following analyses.

After converting to *.mat, check whether you used the widely used color-convention for your data (left-right: red, top-bottom: blue, and front-back: green). So, if need be, change 'Permute gradient components' from 'x y z' to, for instance, 'y x z' to interchange red and green)!

Furthermore, you might need to change the sign of your gradient components, which is obviously very relevant for tractography (to do this, change 'Flip sign of gradient components'). For example, compare the images below ('Draw ROI' → 'Draw Glyphs') – do you see the difference (the z-component was flipped: 'x y z' vs. 'x y -z')!?



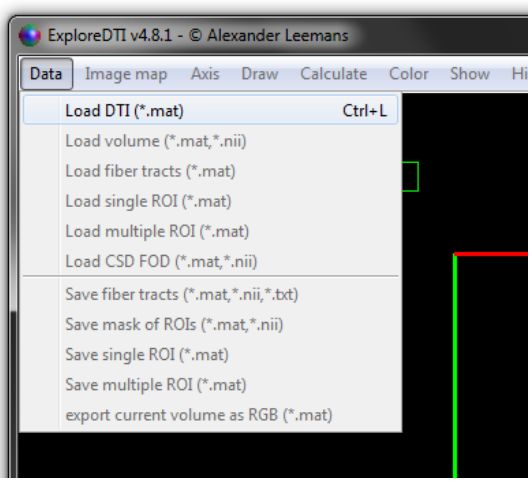
Perform this check for the other views as well – there may be multiple components that need to be changed simultaneously (see below for another example: the x-component was flipped: 'x y z' vs. '-x y z')!



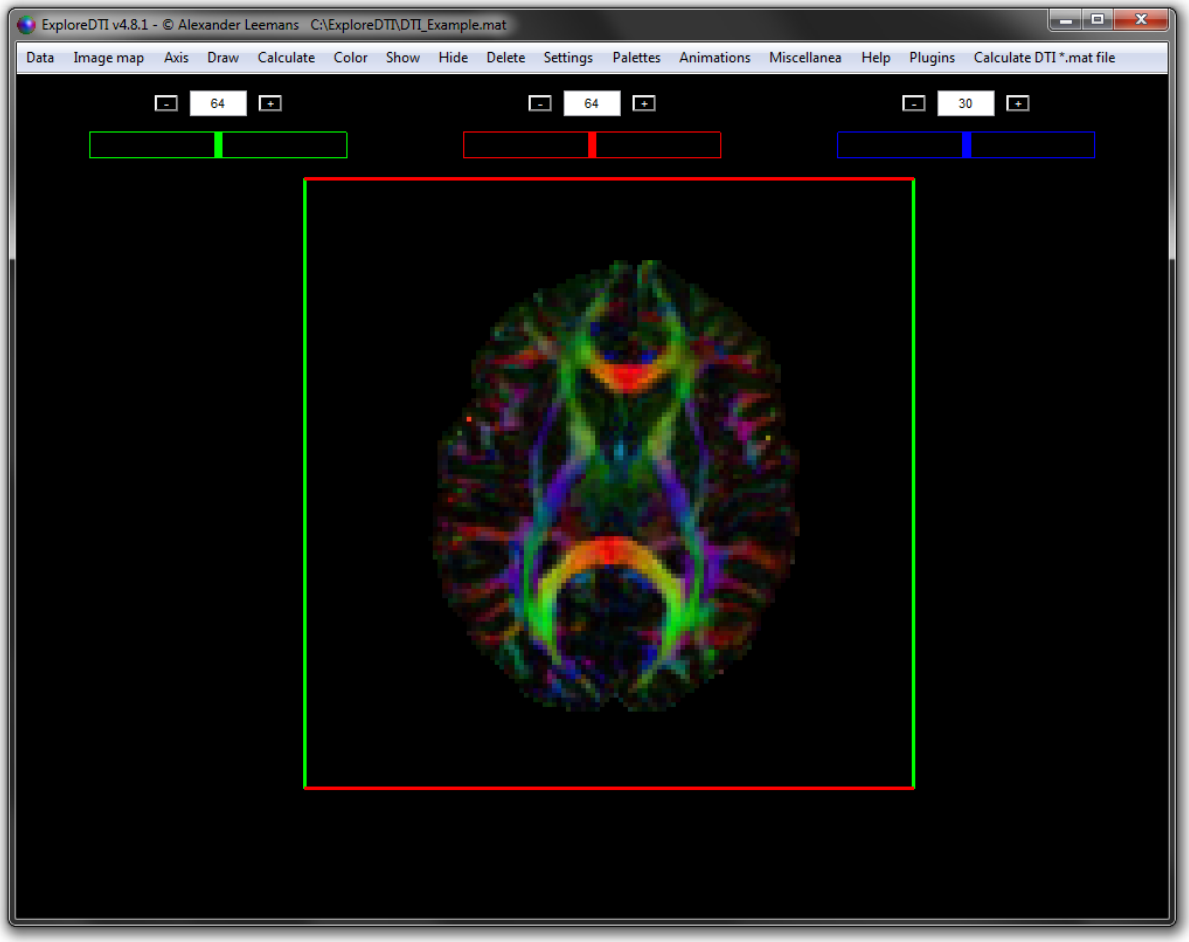
If you have problems converting your raw data, get help from the forum (upload your data). Otherwise, there are only a limited number of combinations possible... good luck!

3. Loading a DTI data set

Since *ExploreDTI* was developed originally only for DTI (hence the name – the other diffusion approaches came later), the only type of data you can load at start-up is a 'DTI *.mat file' (Matlab format). This *.mat file contains a set of variables with a fixed nomenclature (otherwise, *ExploreDTI* will not load the data). An example data set can be found on www.ExploreDTI.com.



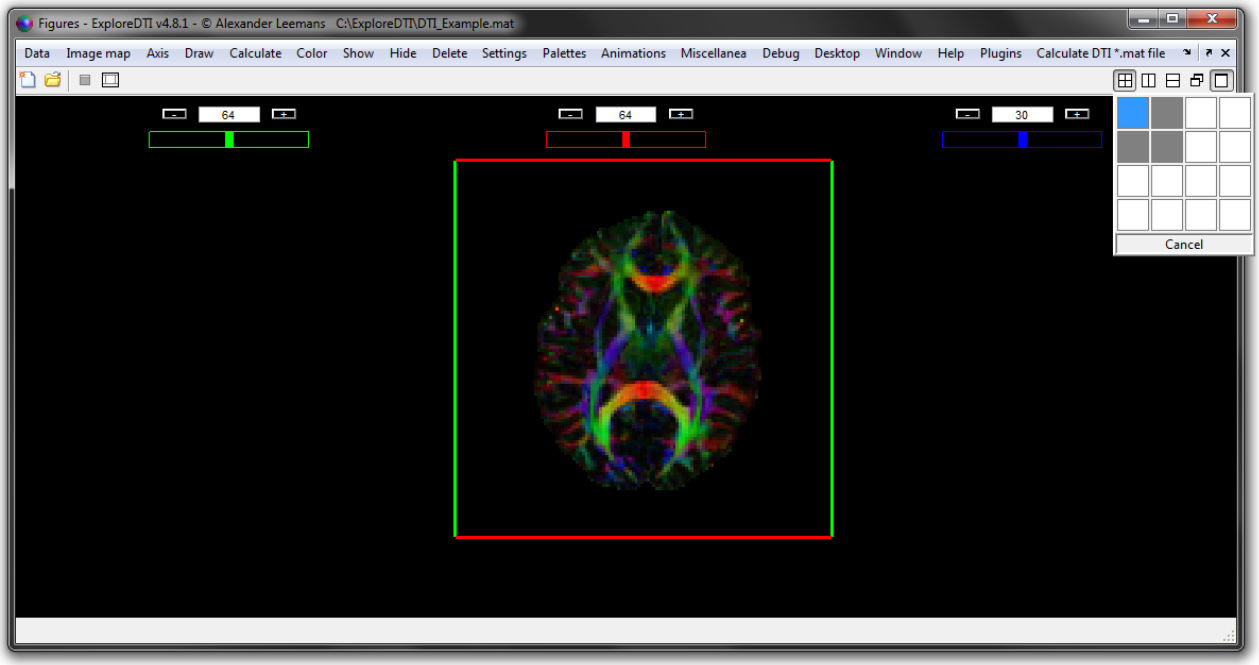
If you have loaded a DTI *.mat file correctly, you should be able to see the following screen (with the menu items enabled now):

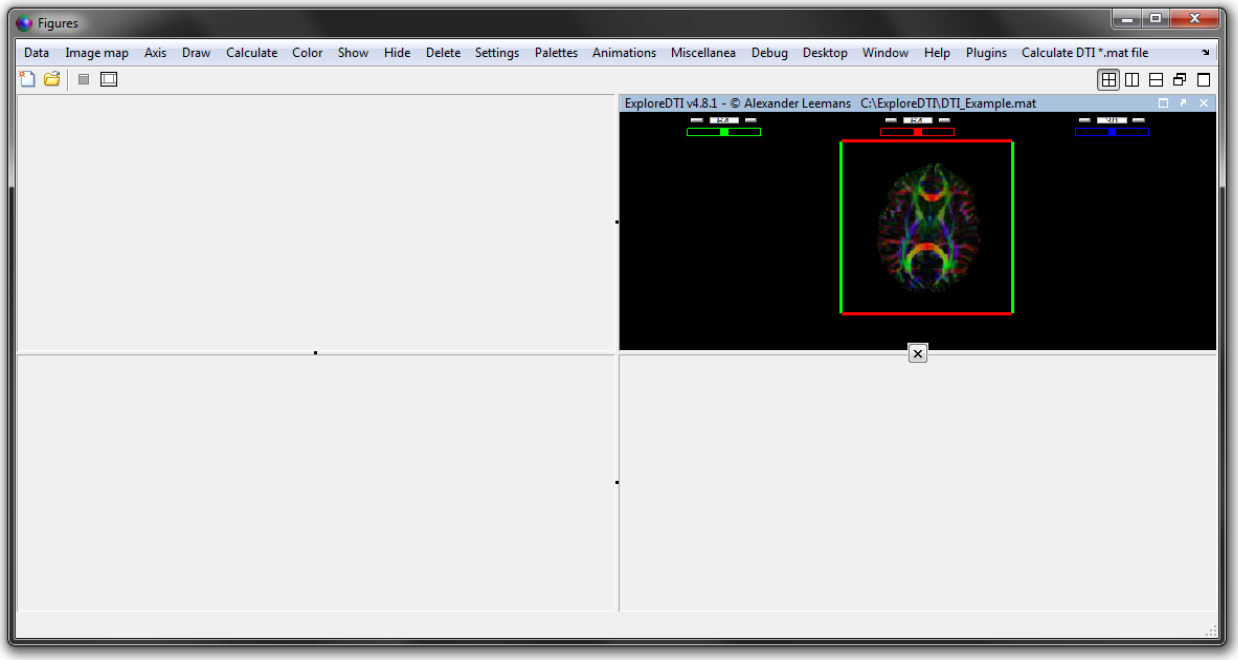


Notice that the figure name contains the full path (here: 'C:\ExploreDTI\DTI_Example.mat') of the DTI data set that is currently loaded (if you do not see anything, adjust masking or OpenGL setting: see Q&A forum).

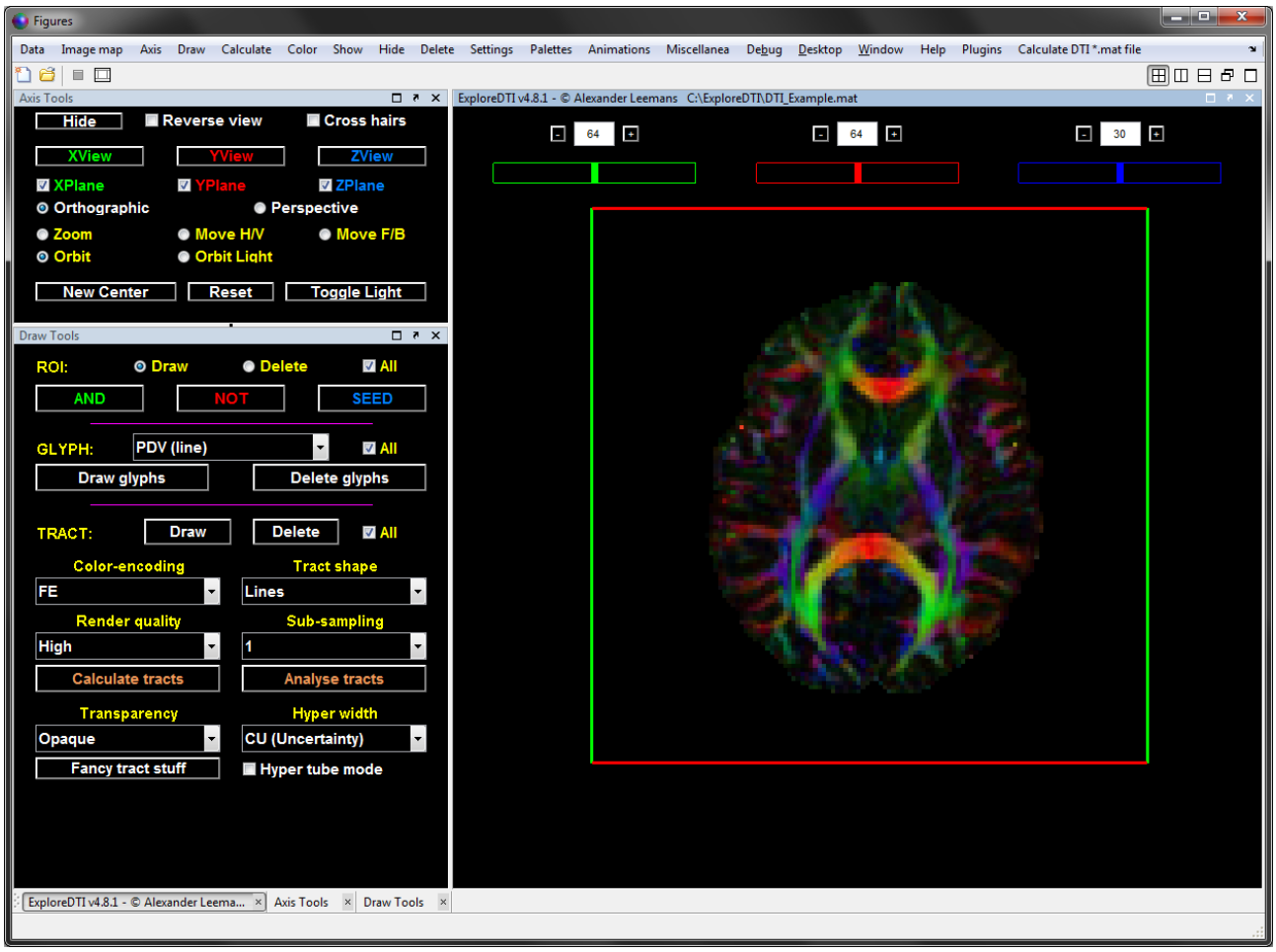
4. Palettes and multiple windows

Using the tools in *ExploreDTI* from the menu bar items can be a pain (e.g., when computationally expensive renderings are shown in the main axis object, the 'menu item refreshing' becomes extremely slow). To circumvent this issue, 'palettes' can be prompted containing a group of commands that can be accessed directly. When running *ExploreDTI* in Matlab (using the p-code version), these 'palettes' are 'docked' automatically in the figure window and can be placed in any way you want. The screenshots below will clarify this. I suggest to use these 'palettes' as much as possible – you will appreciate the benefits. As an example, if you are running *ExploreDTI* in Matlab, do the following (after loading a data set):



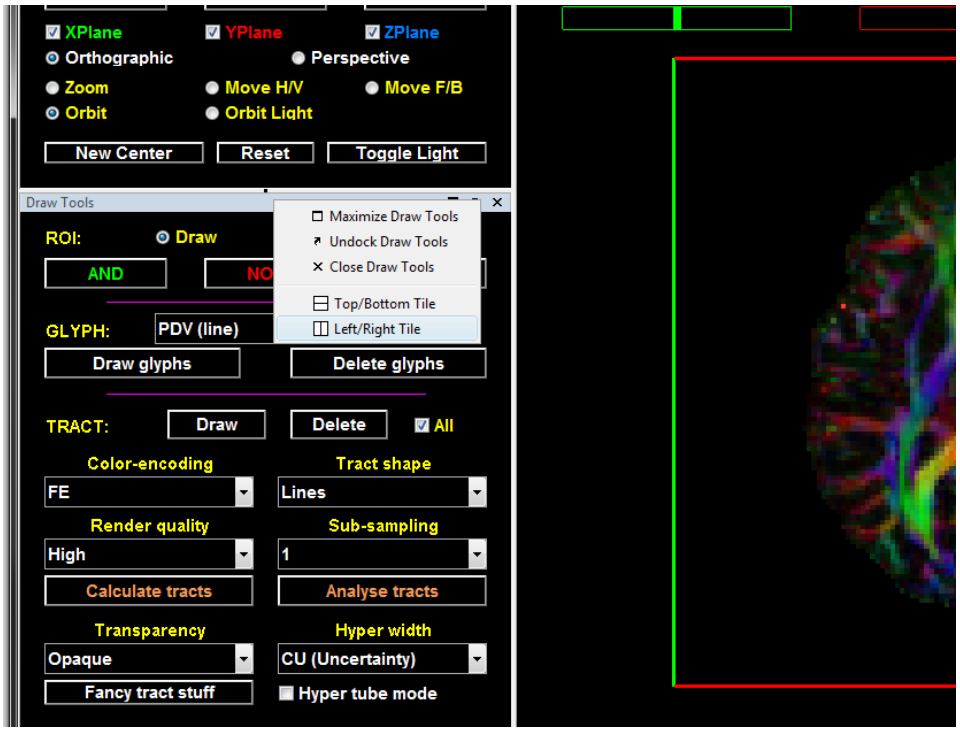


Then click 'Palettes' → 'Axis' and 'Draw', and then delete (click the 'cross') the empty window, so three windows remain (see below).

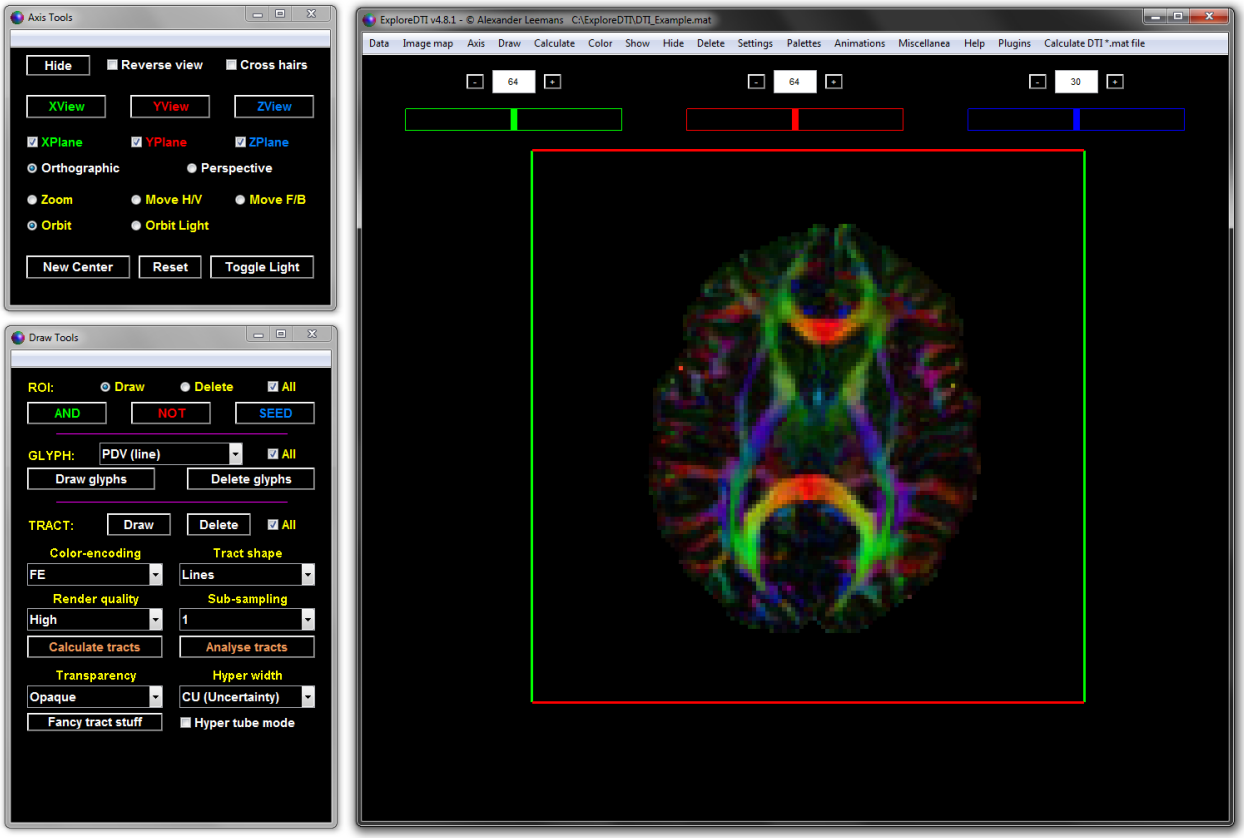


You can continue to do this with other ‘palettes’ as well - they will appear in the menu bar at the bottom (or left/top/…) in a new ‘tab’ – click such a ‘tab’ to make it active and drag it to a field of your choice.

Right click on a window header if you want to further subdivide the windows (and incorporate other ‘palettes’):



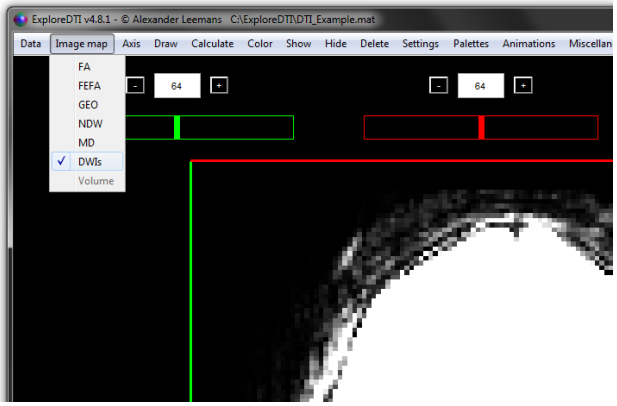
For the standalone versions, however, these ‘docking’ tools are not available anymore (it did work for releases R2009a and older – Matlab is going to look into it to see whether they will make it available again). As a result, all palettes are separate windows, which cannot be docked (see below)...

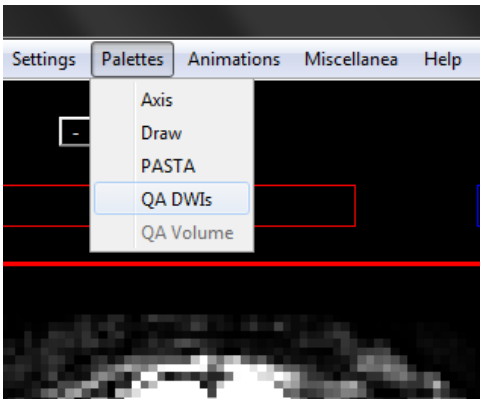


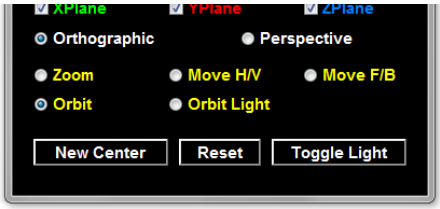
You will notice that the ‘Main Window’ (the one that contains the menu items) is ‘set active’ after any ‘click’ on any of the ‘palettes’ (is only useful when running it in Matlab using the docking tools). Just use a configuration like the one above where the windows do not overlap – otherwise, it can be confusing.

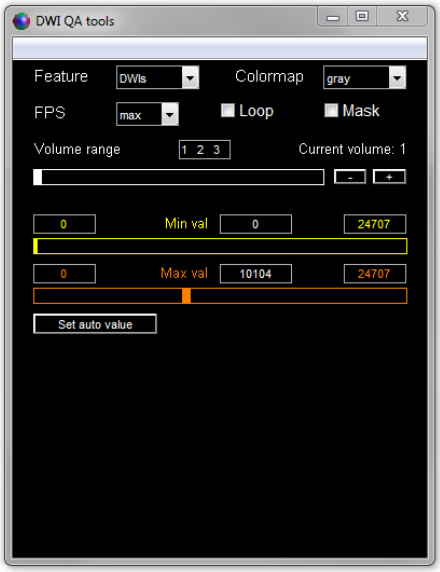
5. Data quality assessment

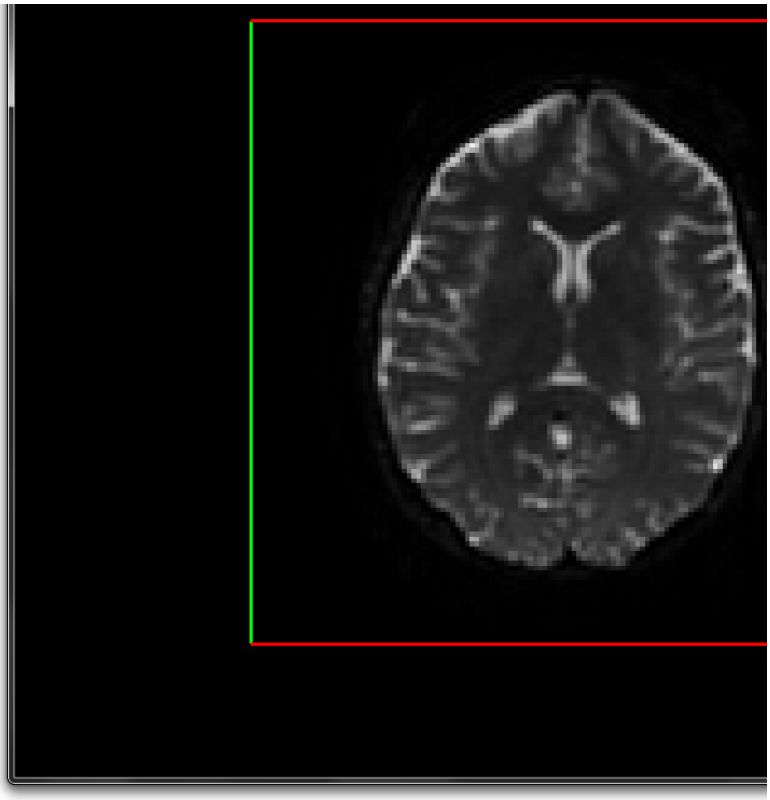
Quality assessment of DW images can be very labour-intensive. One approach to look at your 4D data is by 'looping' it. Do the following: 1) Load a DTI data set; 2) Set 'Image map' to DWIs. The resulting display might look weird, since the intensity range is by default set to [0 1000] – it represents the (first) non-DW image; 3) Now, 'Palettes' in the menu bar will allow you to open the 'QA DWIs' tools; 4) Click 'Set auto value' to set the intensity range for that image.





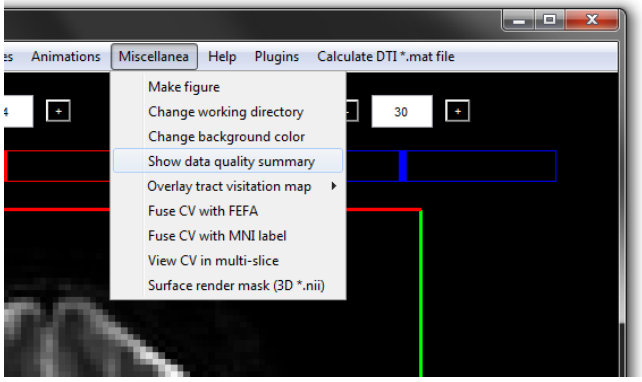


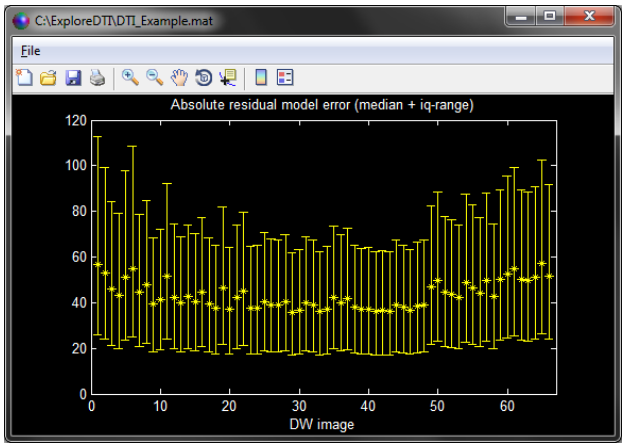


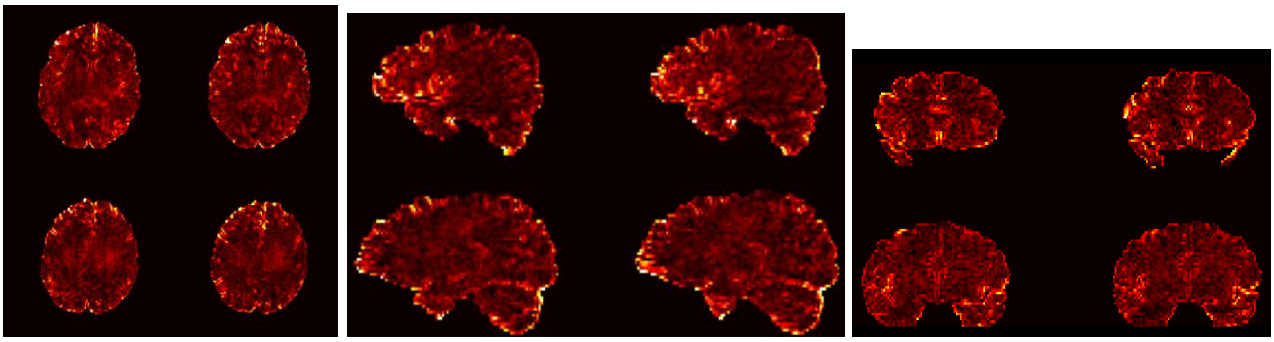


You can do several things now (see below): (1) use the image intensity sliders ('Min val' and 'Max val') to adjust the intensity range. (2) Activate 'loop'. Make sure there are multiple numbers indicated in the 'Volume range', for instance '23:50' (i.e., [23 24 25 … 48 49 50]) or '23 50' (notice the difference!?). Note that during the looping, you cannot use the image intensity sliders ('Min val' and 'Max val'). (3) select the other 'Feature': 'Residuals' (i.e., the absolute residual values to the tensor fit).

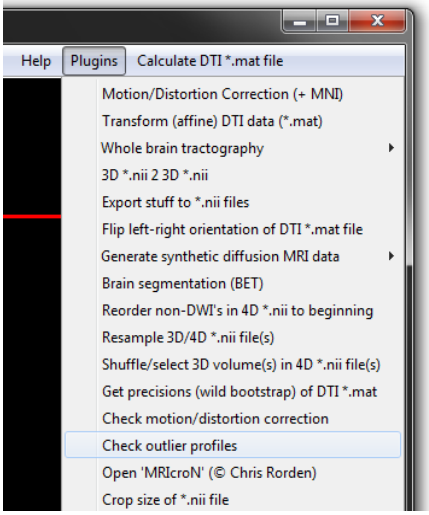
Another quick semi-qualitative check is to look at the average residuals per DWI or across the DWIs for each voxel ('Show data quality summary' – see below):

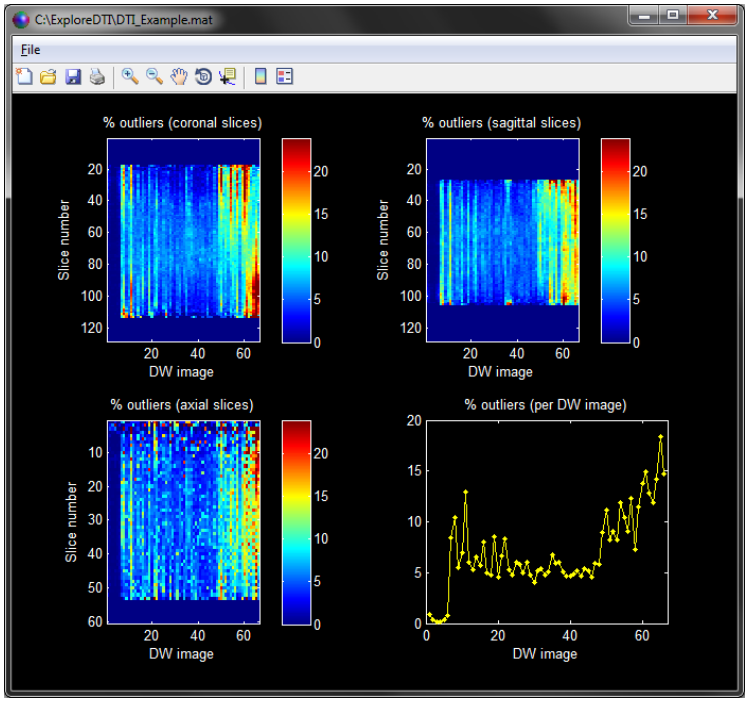






A last quick way to check data quality is by inspecting the outlier profiles:



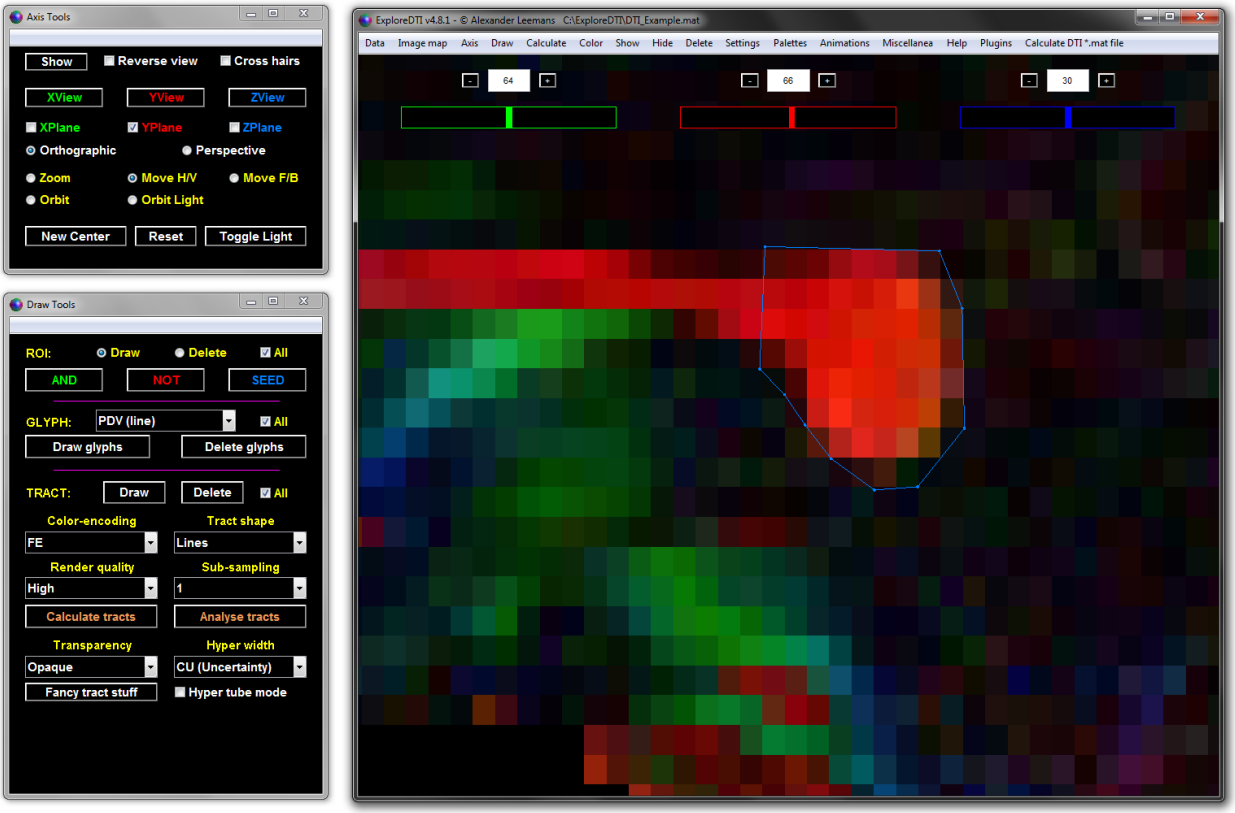


These tools are just useful from a qualitative point of view (e.g., to give feedback to the MR physicist in the event of artifacts). In practice, the robust diffusion (& kurtosis!) tensor estimation approach (i.e., 'REKINDLE' – the default approach in *ExploreDTI* during correction for subject motion, eddy current distortions, ect.) will have taken this into account!

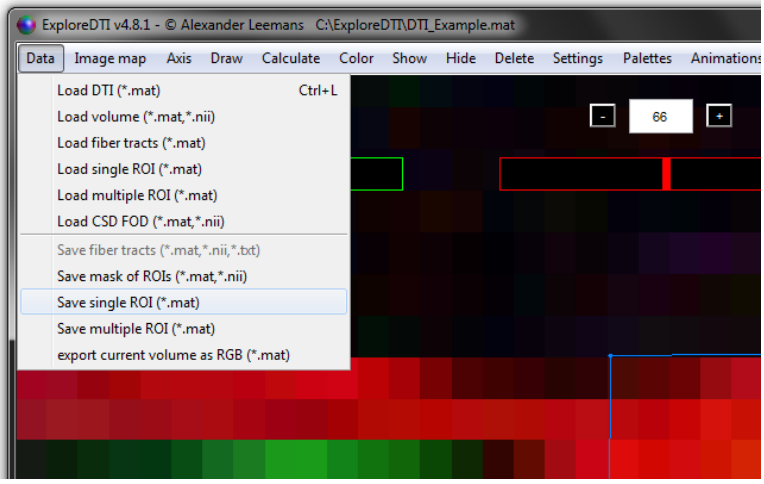
6. Drawing ROIs

Drawing an ROI requires you to 'uncheck' the 'image planes' on which you are *not* going to draw this ROI (it will give you an annoying warning pop-up, otherwise).

To actually draw an ROI (see below), first click the 'SEED' (or 'AND' / 'NOT') button and then 'left-click' each time to generate a point (notice that I have used the 'Zoom' and 'Move H/V' buttons to enlarge the slice to the region of interest). 'Right-click' to close the ROI (don't do anything else during this procedure or ExploreDTI will crash).

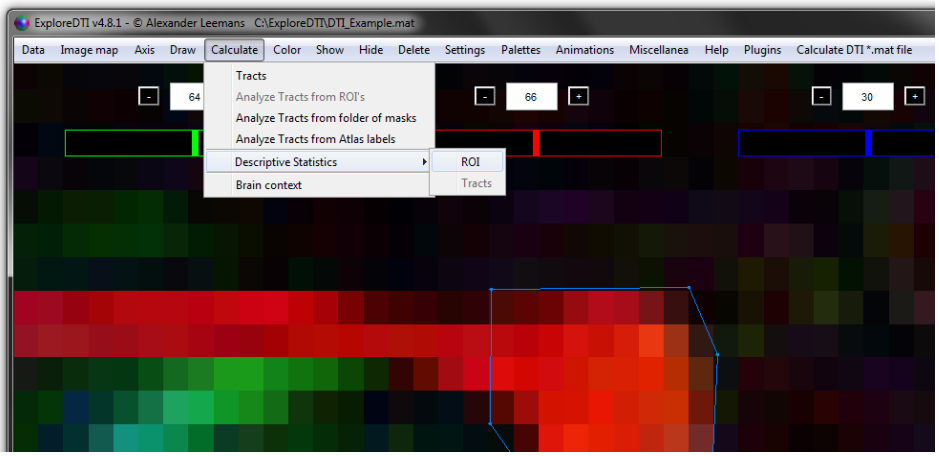


Note that you can save, load, hide, show, and delete single/multiple ROIs. See, for instance, below (menu item 'Data' → 'Save single ROI (*.mat)'):

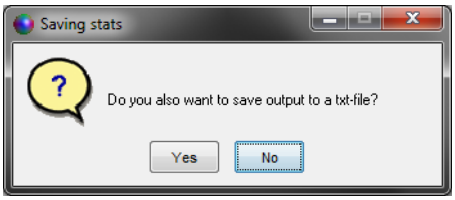


7. Quantitative ROI analysis

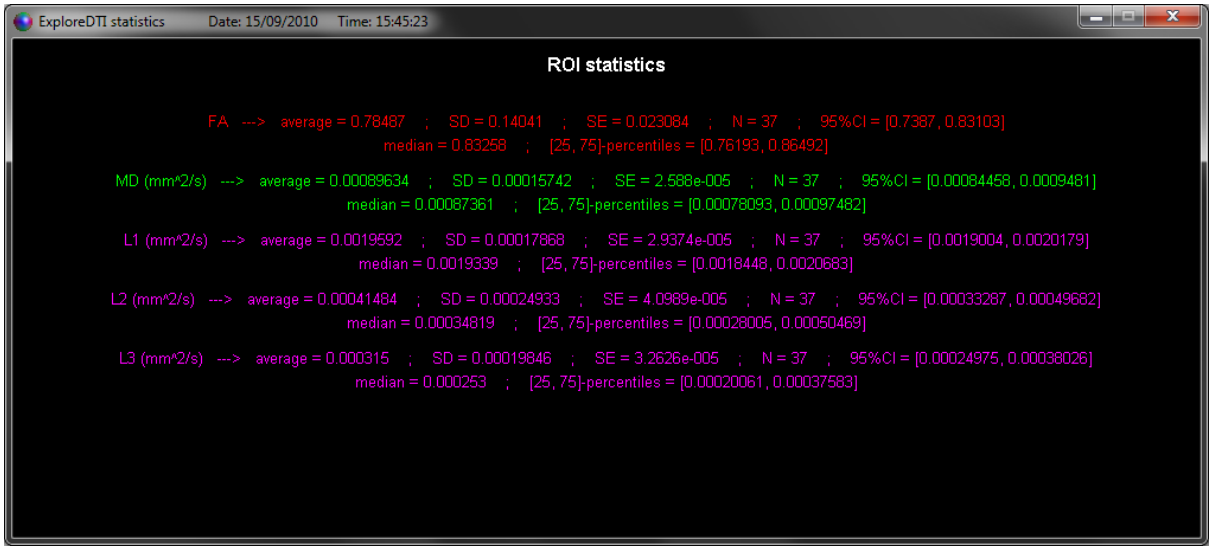
Once you have drawn one or more 2D ROIs (see example below), you can extract DTI related parameters, such as the FA, MD, and the eigenvalues, which may be useful for quantitative analyses.



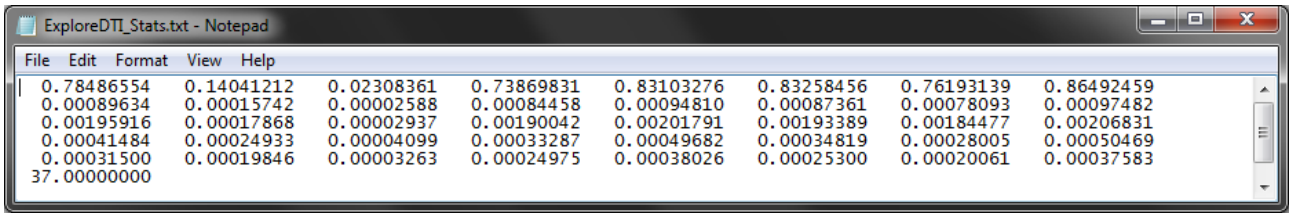
After you click 'Calculate' → 'ROI' (see above), a popup appears (see below) asking you to export the statistics to a simple text file (so afterwards, you can import the numbers into SPSS, Excel, or any other program you want to use for further statistical analyses…).



Finally, a popup window appears with the relevant numbers in it (see below).



The corresponding text file output looks as follows (I hope you understand – from comparing with the image above – what each number represents ;-):



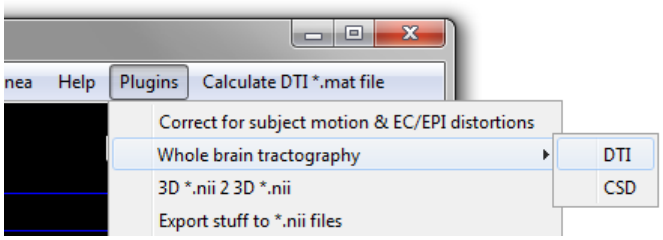
8. Tractography

DTI (*Basser et al, MRM 44:625-632, 2000*) and CSD (*Jeurissen B. et al, HBM 32:461-479, 2011*) based tractography algorithms have been implemented. By default, the CSD approach implemented in *ExploreDTI* uses recursive calibration of the response function (*Tax C.M.W. et al, NeuroImage 86:67-80, 2014*).

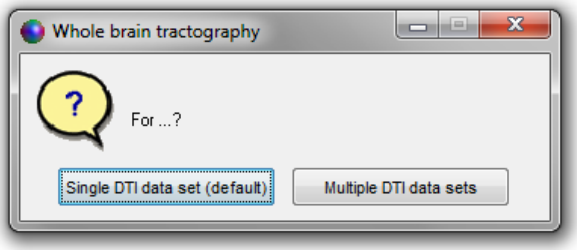
Although one could draw a seed region and then perform tractography from that seed region, I would suggest to do 'whole brain tractography' instead and then use the 'analyze' functionality to extract fiber tract pathways from ROIs, as defined by 'SEEDs' (equivalent to 'OR' gate then), 'ANDs', and 'NOTs'. This reduces the user-bias of SEED ROI definition.

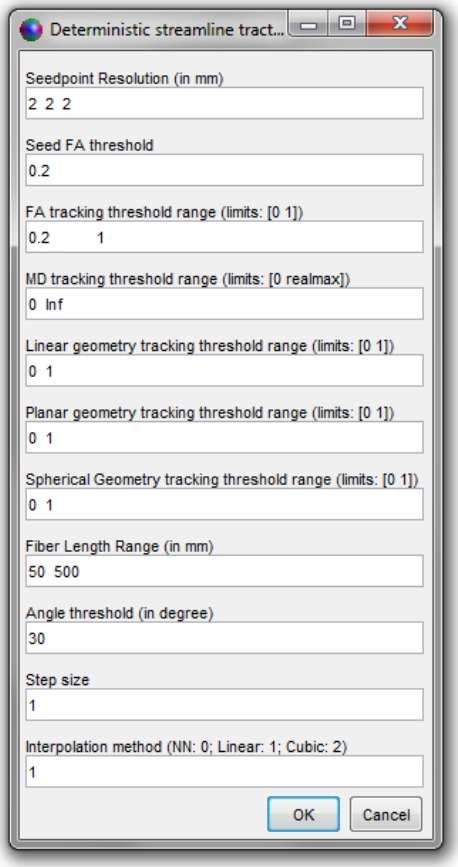
The best way to explain the above in detail is by giving an example: let us extract the 'arcuate fasciculus' (or at least, the reconstructed pathways that *may* represent that WM bundle) and calculate the average FA, MD, etc. along that pathway.

A) Perform 'whole brain tractography' (that is, using a deterministic streamline approach, based on the DTI model). No need to load a DTI *.mat-file yet – just start *ExploreDTI* and go to 'Plugins' (see below):

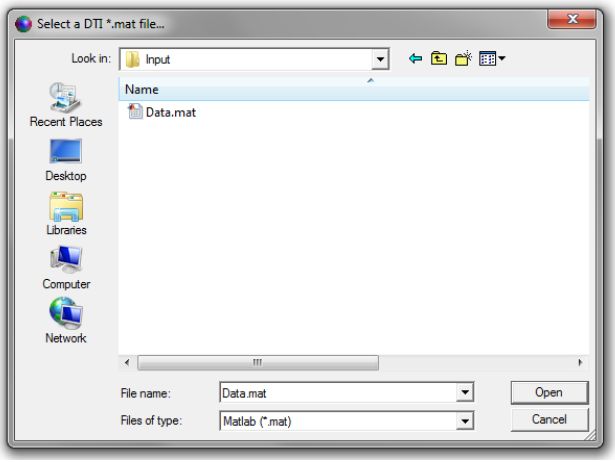


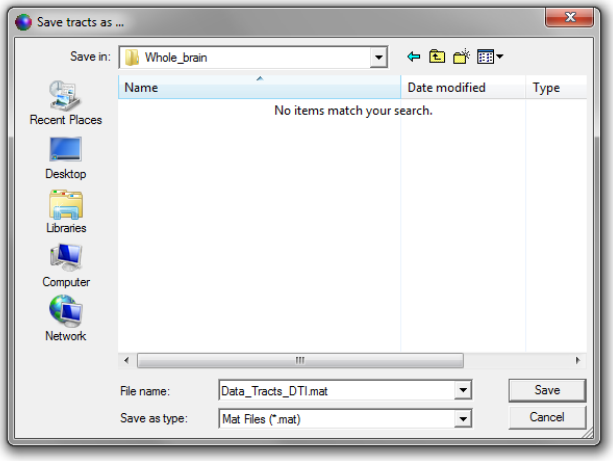
You can do this for multiple data sets as well (see below, by choosing a folder of DTI *.mat files), but for now, let us choose a single DTI data set.



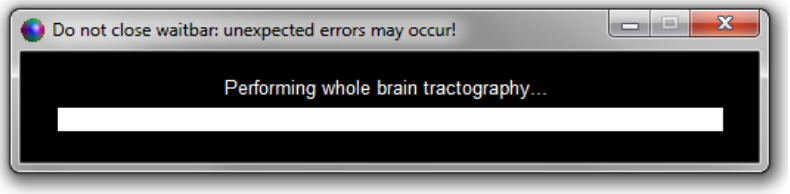


Select a DTI *.mat file and save the tracts to a *.mat file, as shown below:



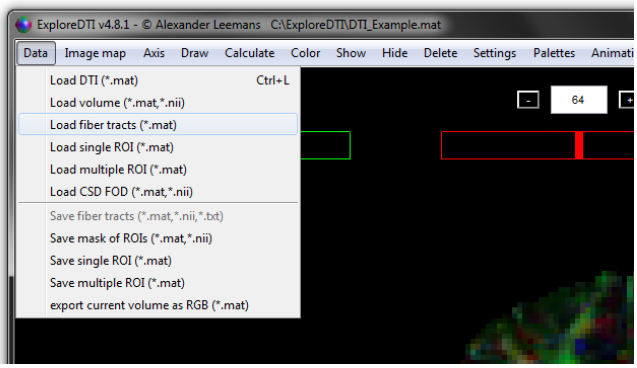


Then the calculations start:



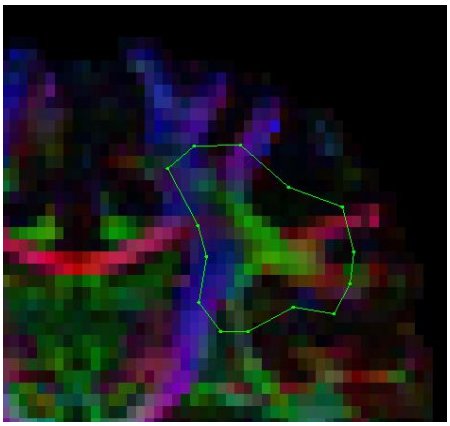
Depending on your computer 'specs' it will take around 1 to 10 minutes (that is, for the default values above).

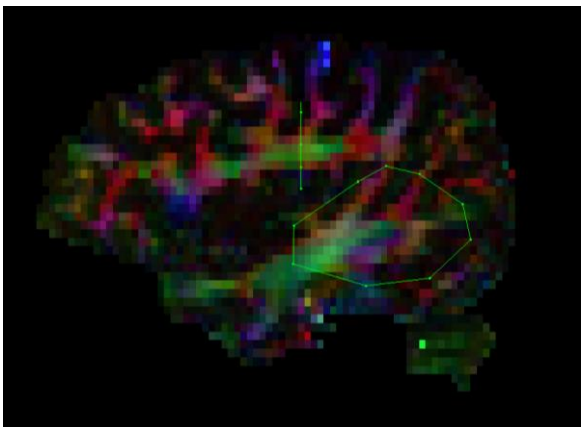
B) Load the tracts (after you have loaded the corresponding DTI *.mat file).

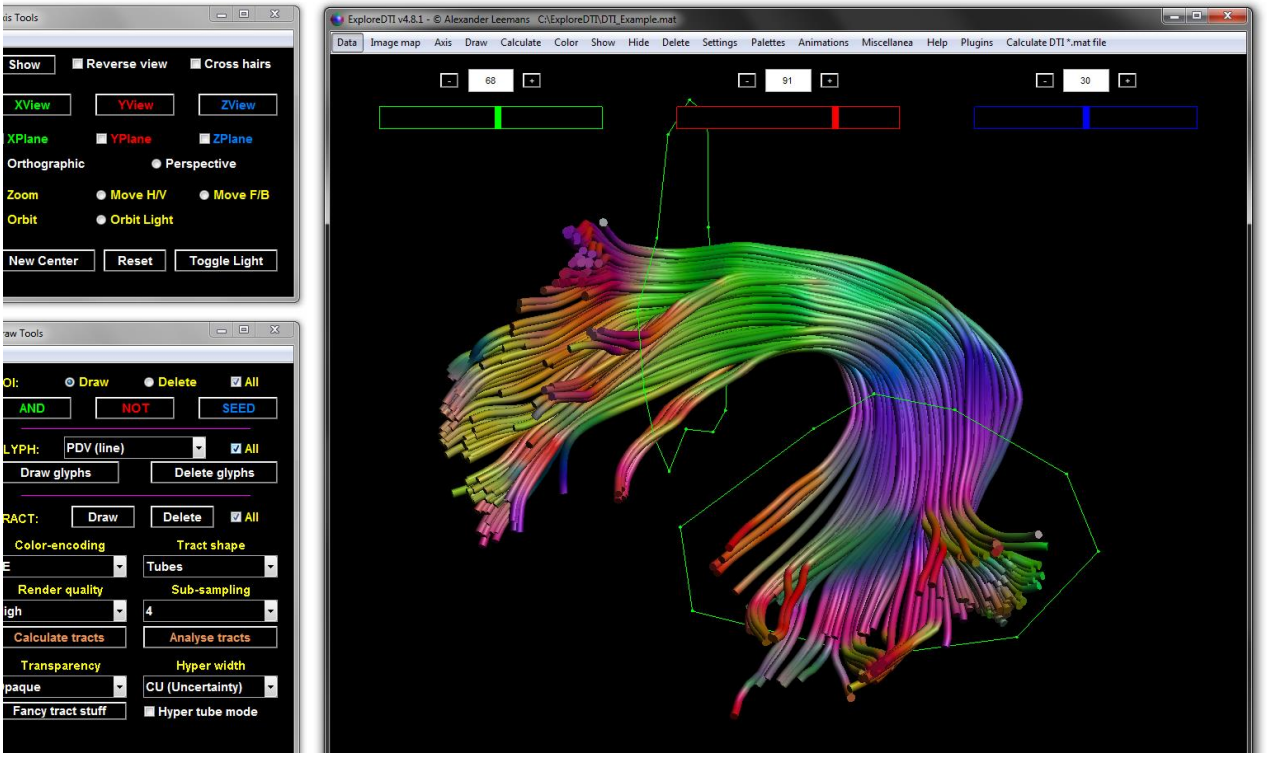


Notice that this eats away your memory (3 GB is no exception – depending mainly on step size and seed resolution)…

C) Placing ROIs to select a structure of interest. Draw 'AND' gates that define regions that would intersect the "arcuate fasciculus". You might need to draw 'NOT' gates to omit fiber tracts that are not of interest. Then, click 'Analyse tracts' and click 'Draw'. In this example, I used 'Tubes' as 'Tract representations' and clicked 'Toggle Light' to give a 3D feel to it. To increase render speed, I've set 'Sub-sampling' to '4'.



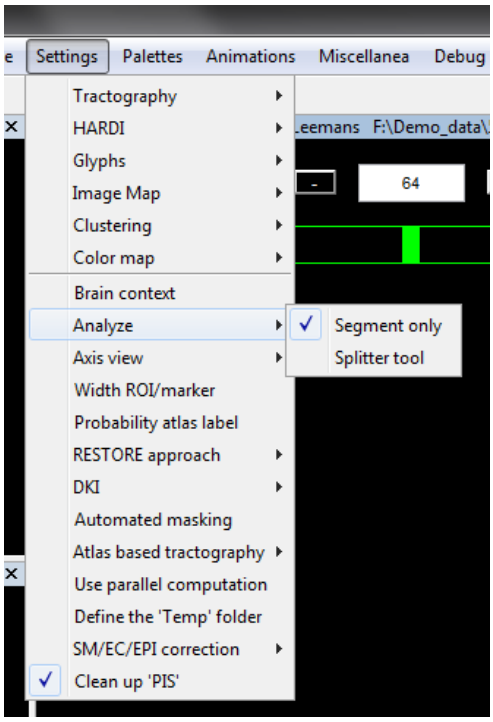


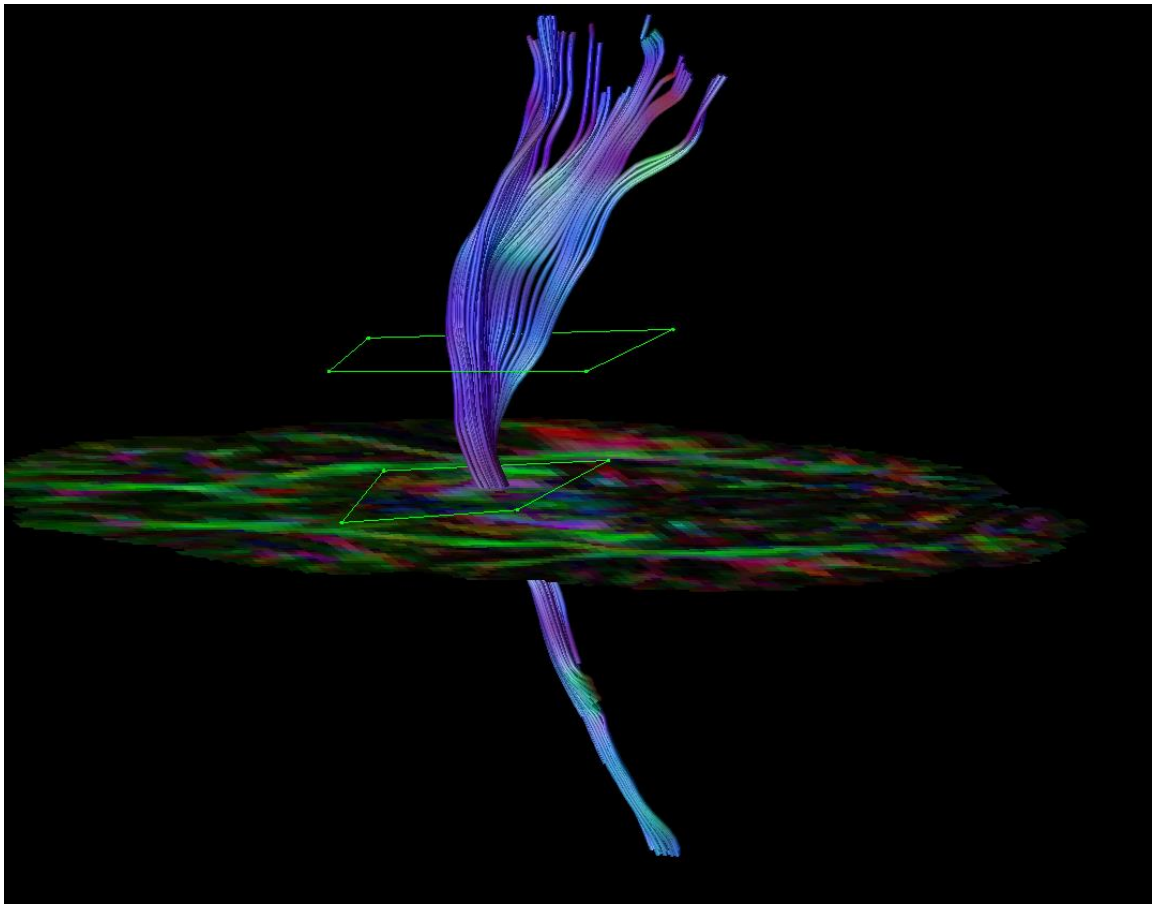


Similar to the ROI analysis, you can calculate the stats of fiber tract structures (note that you can also save, load, hide, etc. tracts). I *recommend* saving the end-result as a 'tracts *.mat file', as there is a plugin tool that can convert an entire set of such tract *.mat files to a *.txt file (see 'Plugins' → 'Convert' → 'Info of tract *.mat file(s) to *.txt'), which contains the corresponding summary statistics.

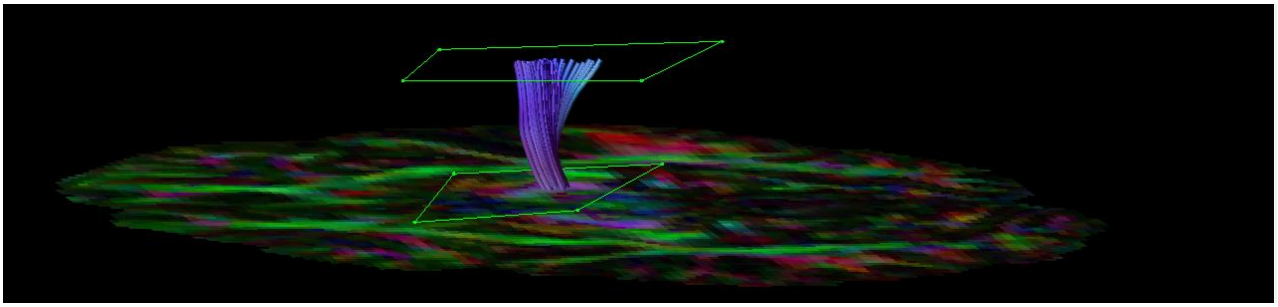
9. Tract segment analysis

Assume you want to analyse a particular segment of a fiber tract bundle, such as shown below. Here, for instance, we are only interested in a particular segment of these “corticospinal” tracts as outlined by the two AND ROIs. To extract this part of the tracts, go to settings first and check 'segment only' (see below). When checked, clicking 'analyse' will generate a pop-up to save the tract segment (uncheck it afterwards!).

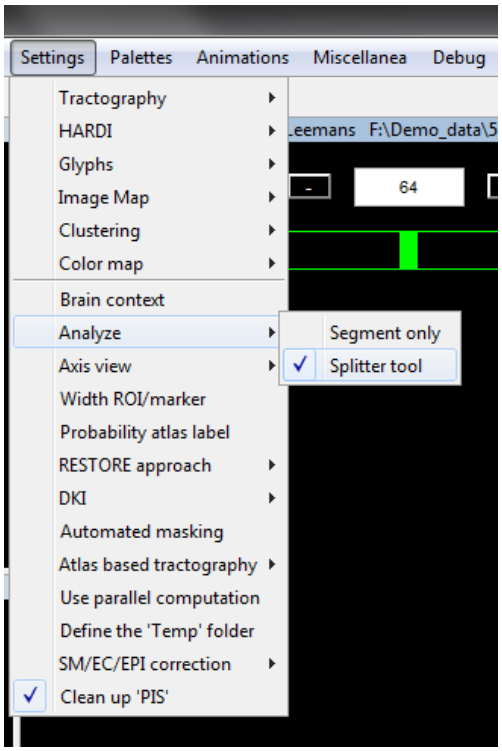


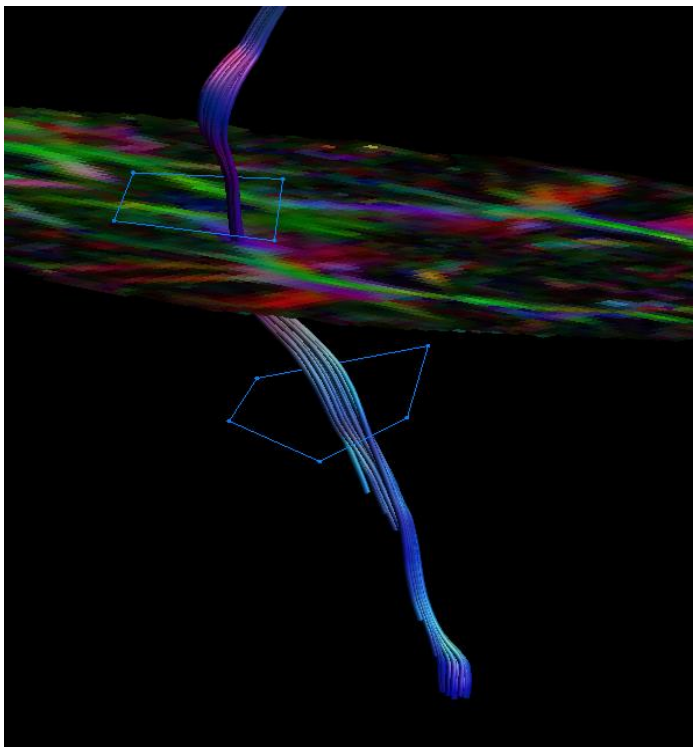


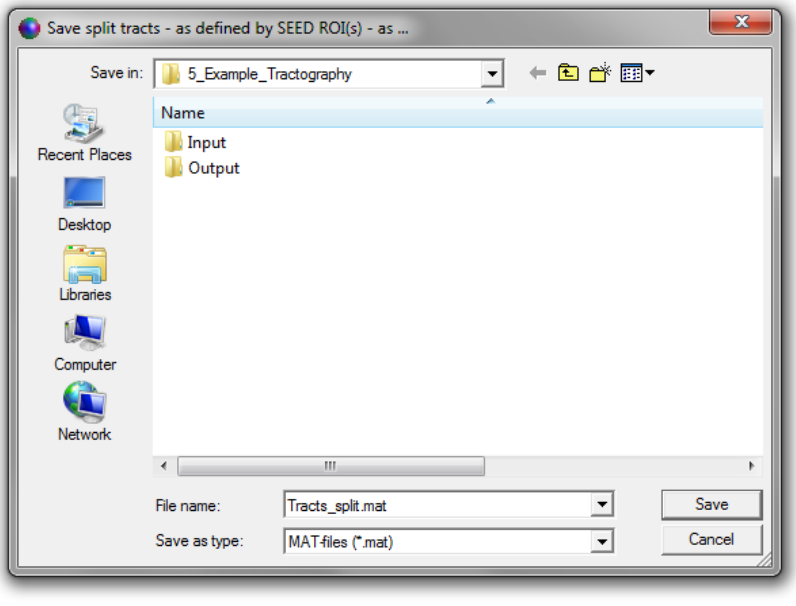
The segmented tracts can then be loaded into *ExploreDTI* (see below):



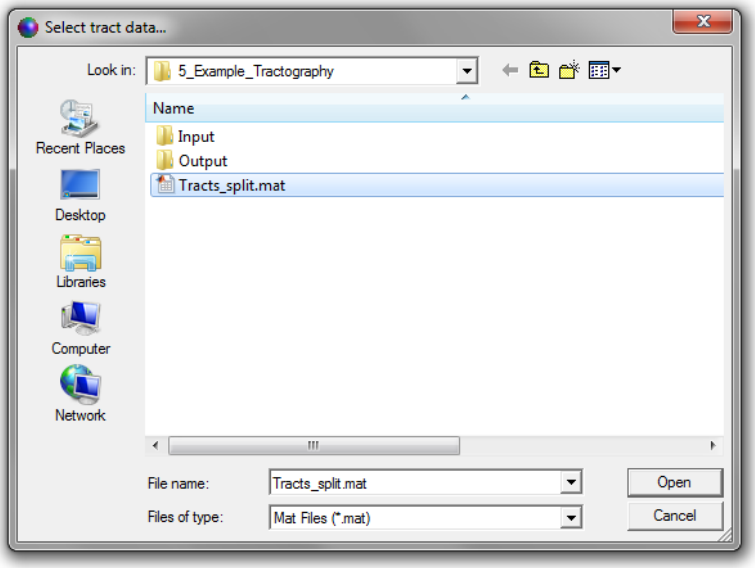
From this segment, measures of interest (FA, MD, etc.) can be calculated as outlined before. A more general approach to select 'tract segments' of interest is the 'splitter' tool, which uses the SEED (= OR) gates to 'split' the tracts at those positions. The tracts that have been split are then saved. Once reloaded, use 'AND'/'SEED' gates to select the segments of interest:



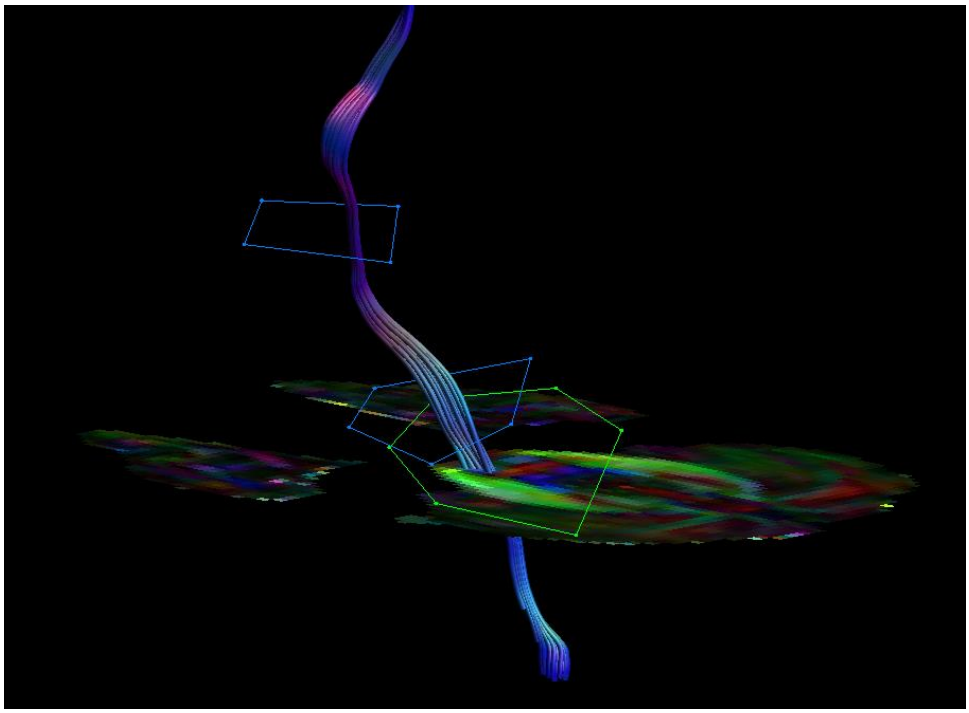




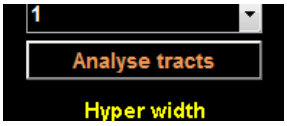
After saving the ‘split’ tracts, load them:

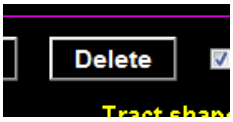


Draw an ‘AND’ (green ROI) to select the segment of interest:

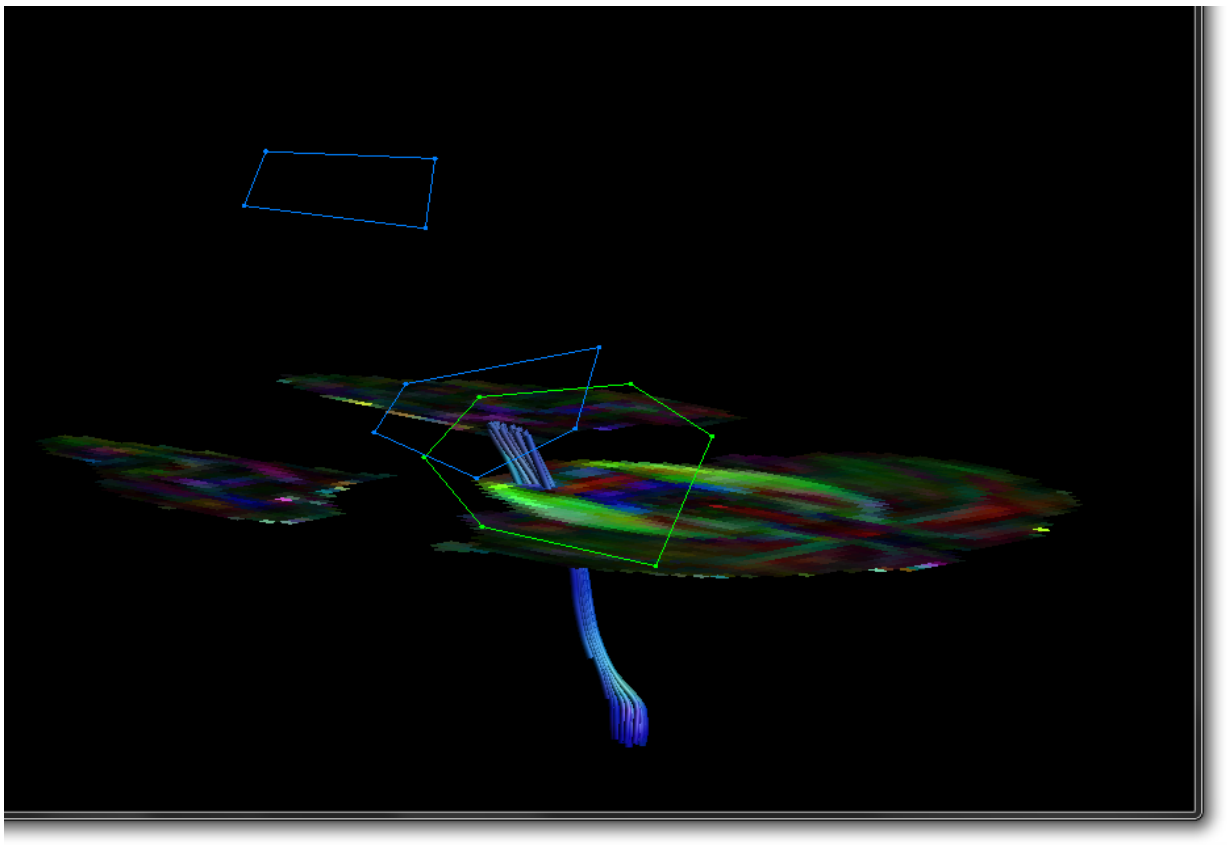


Uncheck ‘splitter tool’, then click ‘Analyse tracts’ and delete the previous ‘display’:





Finally, click ‘Draw’ tracts – only the selected segment will remain:

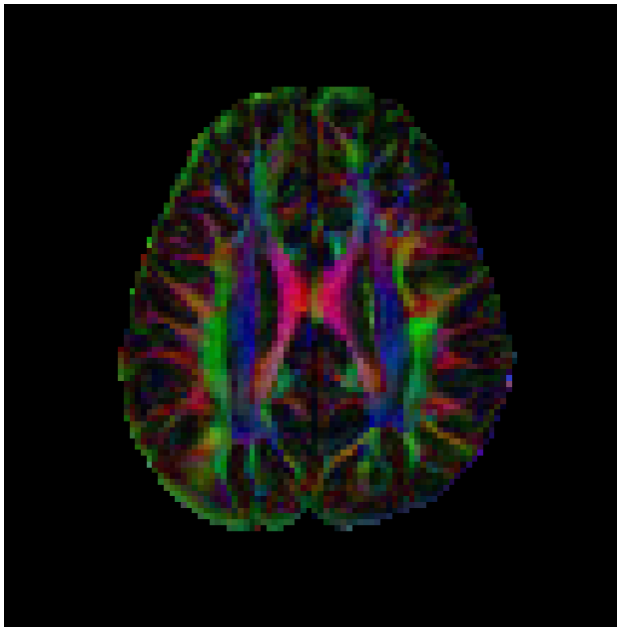


10. Correcting for subject motion & eddy current induced geometric distortions

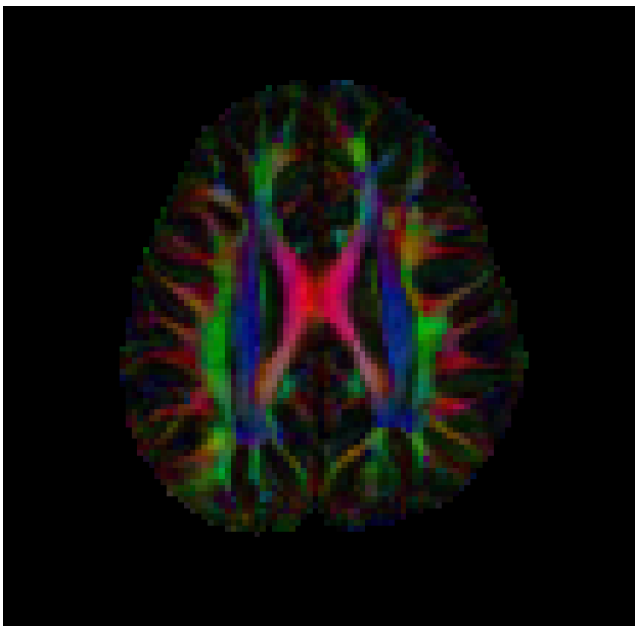
The relevant reference for this tool is “Leemans A. and Jones D.K., Magnetic Resonance in Medicine 61(6):1336-1349, 2009 (http://www.ncbi.nlm.nih.gov/pubmed/19319973). The tool to correct for subject motion & eddy current induced geometric distortions can be found in: ‘Plugins’ → ‘Correct for subject motion & EC/EPI distortions’. See the menu item ‘Settings’ → ‘SM/EC/EPI correction’ for more details. I also recommend using multi-core support for this tool (that is, if you have this Matlab toolbox installed): see ‘Use parallel computation’ in the ‘Settings’. By default, the EPI correction step is omitted (see Section 11).

The output is again a DTI *.mat file, but then with the (default) extension ‘_MD_C_native.mat’. See the before/after examples below:

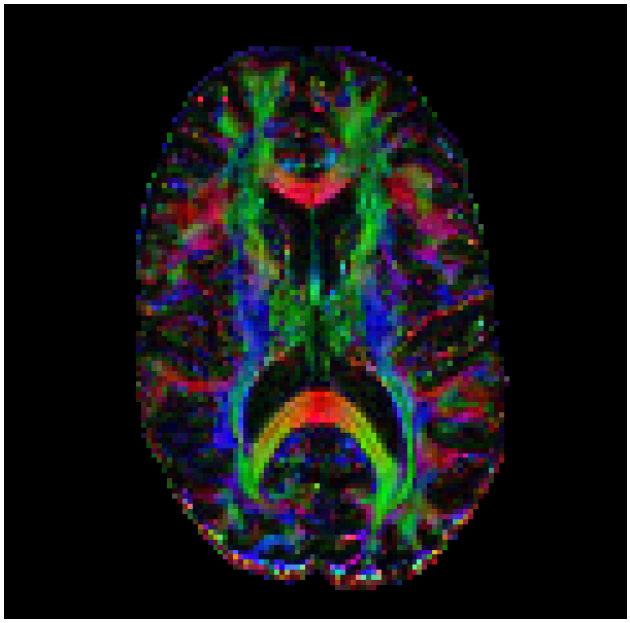
Before correction



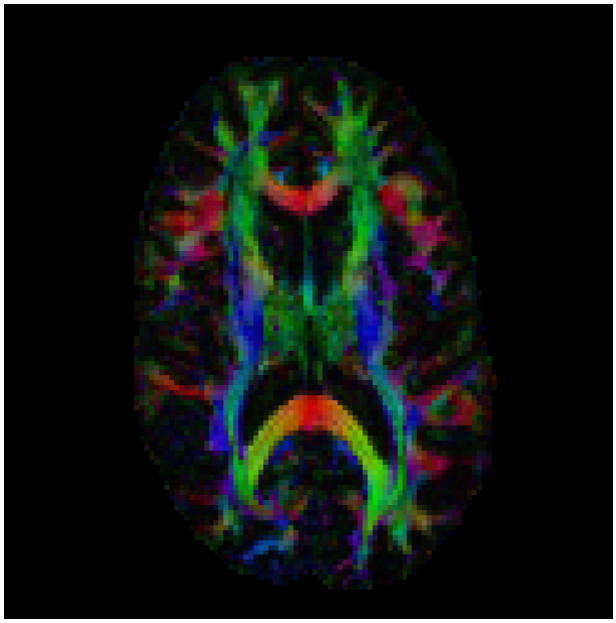
After correction



Before correction

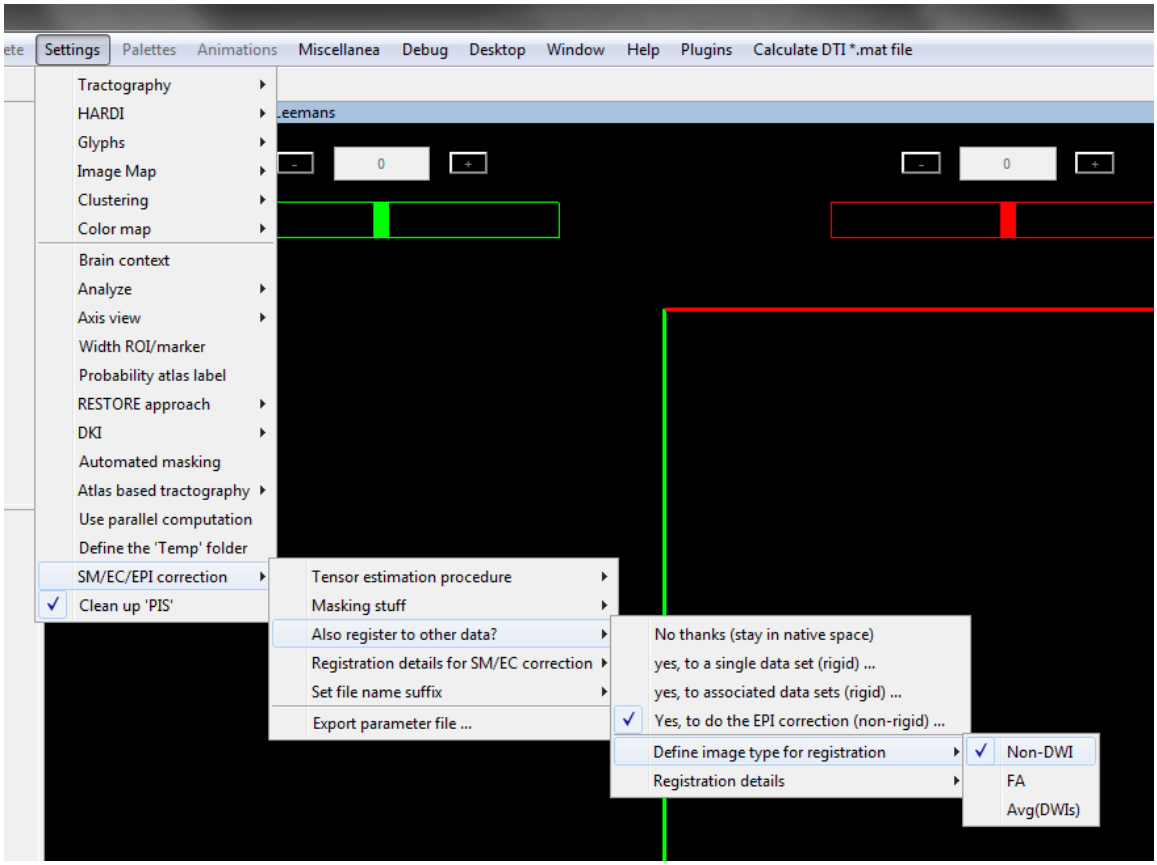


After correction

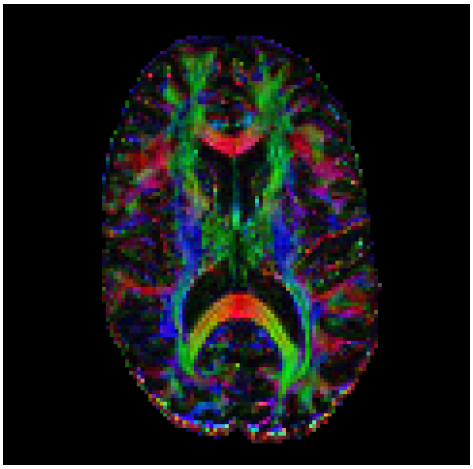


11. Correcting for EPI/susceptibility distortions

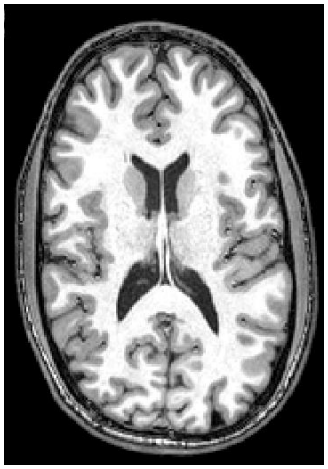
There are several ways to (try to) correct for these deformations. The approach I implemented is based on the work of Irfanoglu et al (http://www.ncbi.nlm.nih.gov/pubmed/22401760) and is integrated with the "subject motion / eddy current distortions" correction tool. In this way, all corrections (SM/EC/EPI) can be performed in one interpolation step (to minimize blurring effects). In summary, if you have an 'undistorted' modality (T1,T2,…), you can use it to unwarp the deformations that are present in your diffusion data:



After 'checking' → 'Yes, to do the EPI correction (non-rigid)', a popup will appear asking you to fill in the suffix of the file that will be used to do the correction. Example: if the DTI file is called "Data_set_M.mat" (below, left), then name the T1 file "Data_set_M_T1.nii" (below, right). For a T2, I suggest to use 'Non-DWI' as the choice for 'Define image type for registration', for a T1, you could also try 'Avg(DWIs)' or 'FA'. Masking the T1 (or T2 etc.) data ('zero' value as background) increases the performance (note: scalp may still be present as shown below).

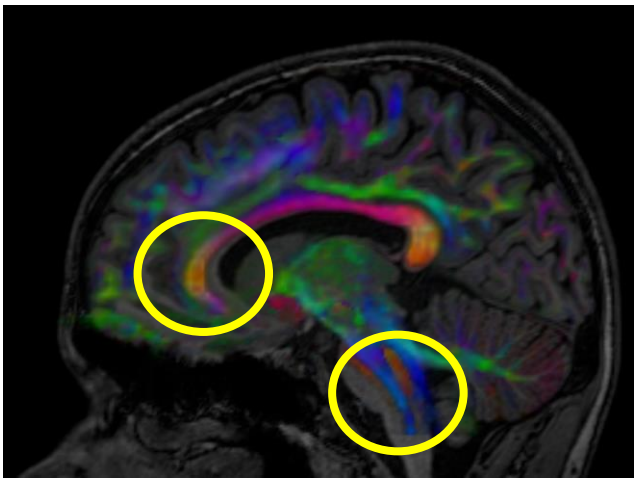


"Data_set_M.mat" (uncorrected)

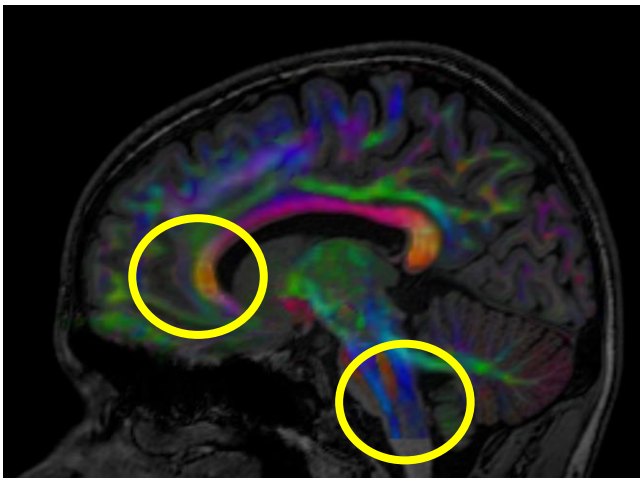


T1 data set: "Data_set_M_T1.nii"

Important to know: the diffusion data will be resampled to the space of the 'undistorted' modality… you may want to adjust the field-of-view and/or resolution to keep the size of the resulting 'corrected' diffusion data manageable. To actually 'see' how the tool performed, you may want to compare this corrected result with the 'non-corrected' data. You can do this, by checking one of the 'rigid' counterparts: e.g., 'yes, to a single data set (rigid)' instead – then the output will just resample the data to the undistorted (T1) data space:

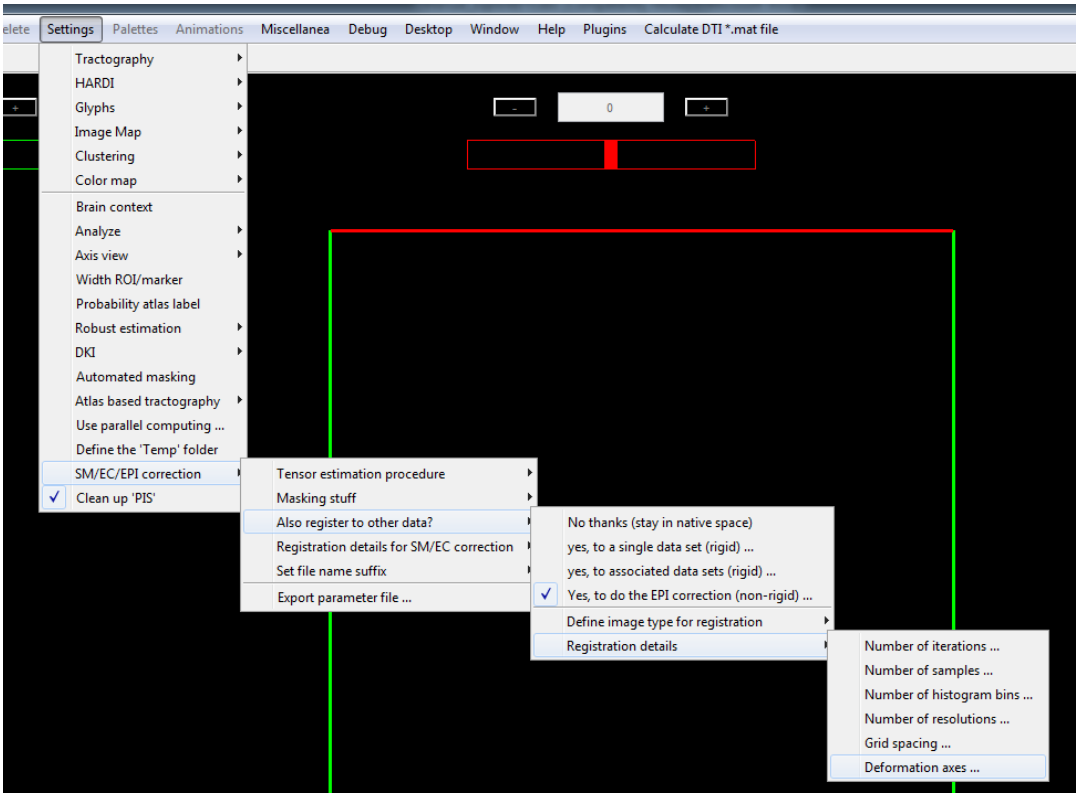


Uncorrected (rigid) for EPI distortions



Corrected (non-rigid) for EPI distortions

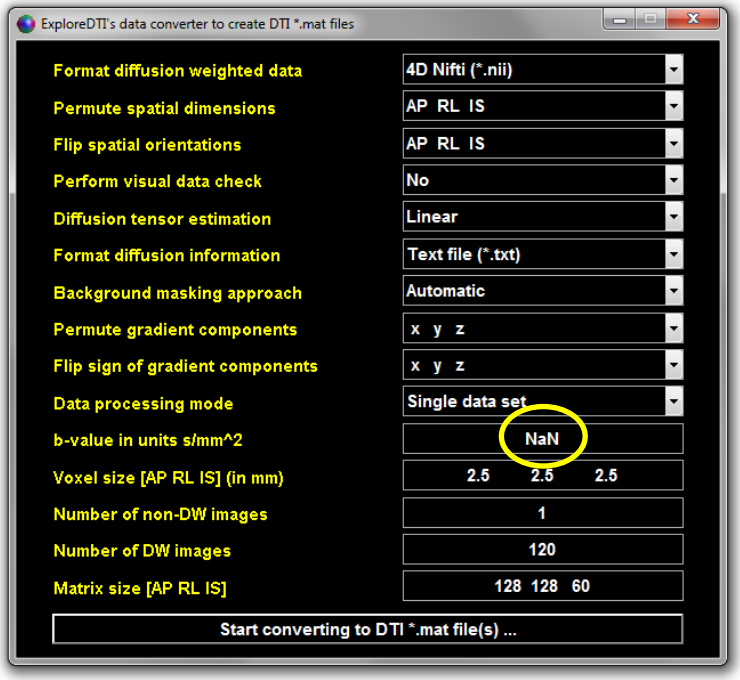
By default, the non-linear deformations are allowed along any orientation. In practice, however, correction will likely improve if the registration is constrained to model deformations only along the phase encoding direction. To do this (assume: A-P orientation), change "Deformation axes" to [1 0 0].



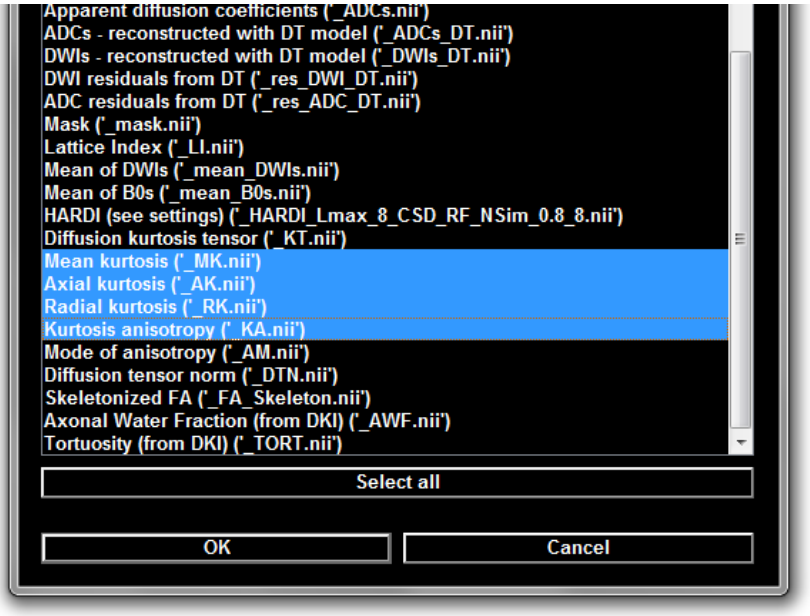
And make sure you first **use the plugin "Flip/permute dimensions in 3D/4D *.nii file(s)" to make your nifti files (T1, T2, etc.) "ExploreDTI compatible"**… this will avoid unexpected axis flips (& permutations) in the resulting files.

12. Diffusion Kurtosis Imaging (DKI)

To calculate the kurtosis parameters, just set ‘NaN’ as the b-value when converting your *.nii DWIs and *.txt B-matrix to a *.mat file:



You can do everything (correction steps, DTI based tractography, etc.) as before. The most common kurtosis metrics (MK, RK, AK, KA) can be exported from the ‘Export stuff to *.nii files’ tool in the menu item ‘Plugins’:

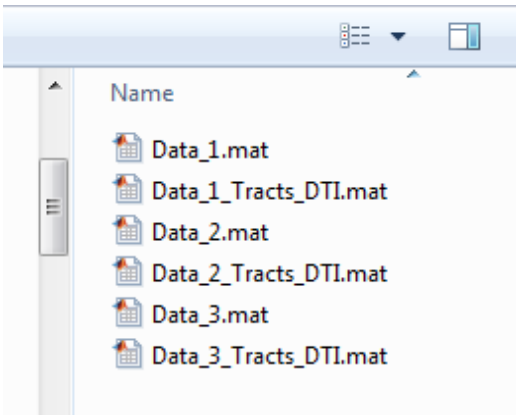


The REKINDLE (**R**obust **E**xtraction of **K**urtosis **IND**ices with **L**inear **E**stimation) approach (default setting while correcting for subject motion, eddy current induced distortions and/or EPI deformations) is preferred for computing these kurtosis metrics (Tax C.M.W. et al, Magnetic Resonance in Medicine 73:794-808, 2015).

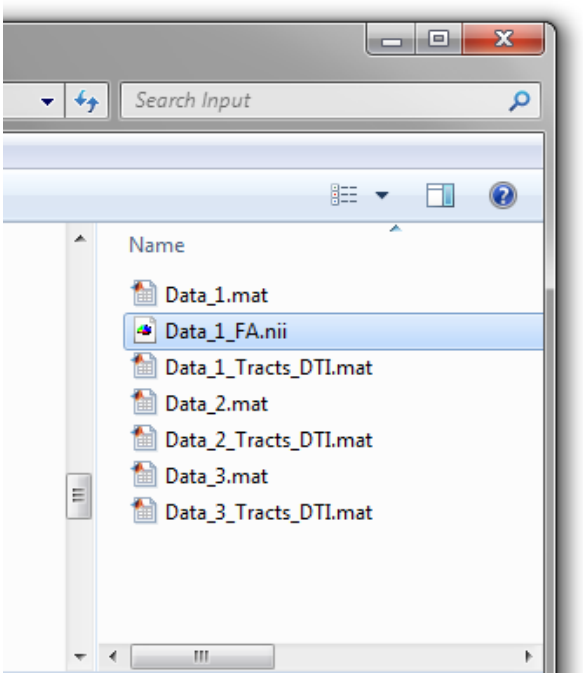
13. Automated “atlas based” tractography

This tool is based on the work presented in Lebel et al. (http://www.ncbi.nlm.nih.gov/pubmed/18295509). In summary, it works as follows. First, define a set of ROIs (AND/NOT gates) that would extract your tract of interest (based on an atlas/template or on a representative single subject). Using this information (the ‘template’ and these regions-of-interest – ROIs), the tract of interest is automatically reconstructed for all your data sets. Below, an example is given on how to do this for pathways of the right uncinate fasciculus (R-UNC).

The input of this tool is a set of DTI *.mat files and their corresponding ‘whole-brain’ tract *.mat files. In this example, the data sets were all in native space and the tracts were reconstructed with DTI. As you can see below, there are just 3 subjects (FYI – the computation time is roughly 5 to 20 min per data set).



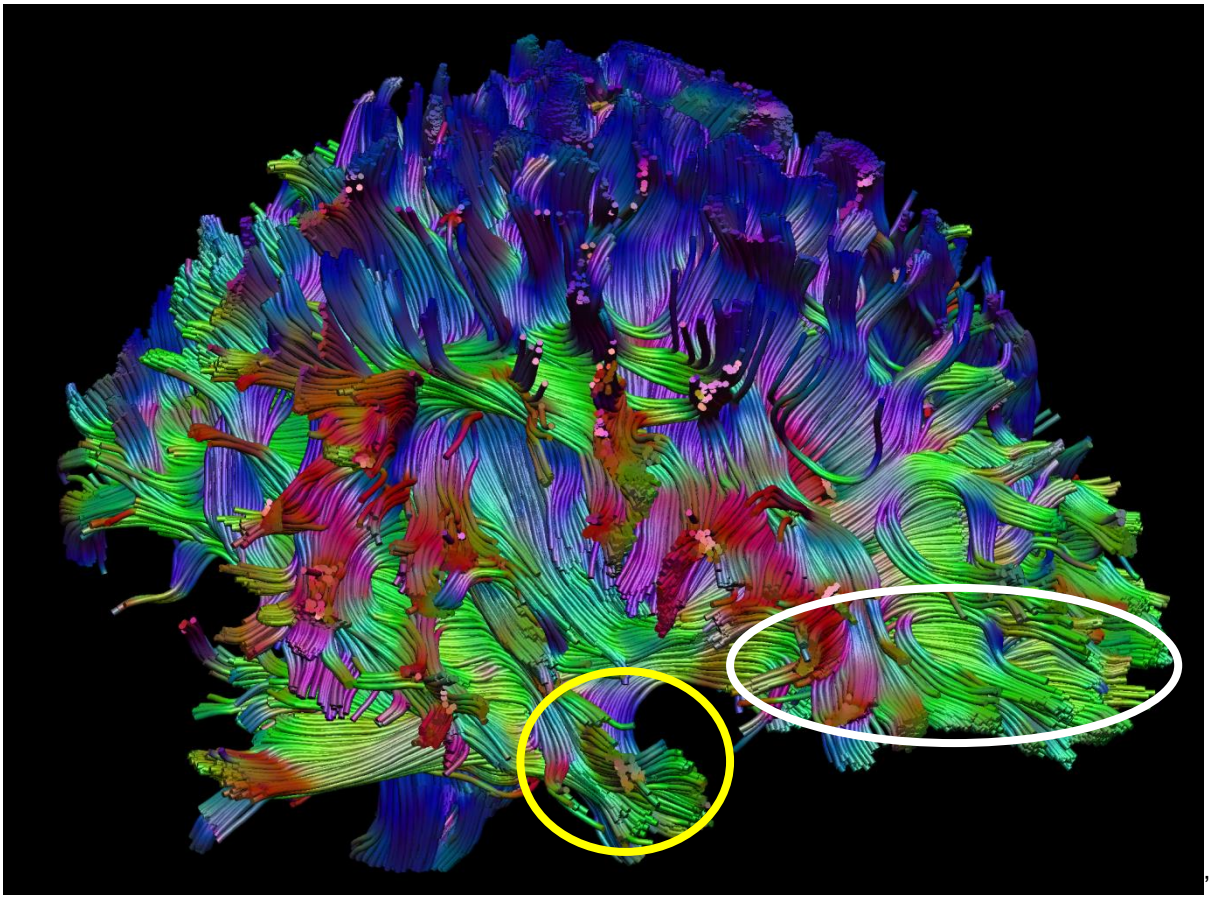
I could use a nice FA template (there are many!), but in this example, I will create a ‘pseudo FA atlas template’ myself from “Data_1”. So simply export the FA from ‘Data_1.mat’ via ‘Plugins’ → ‘Export stuff to *.nii files’: select ‘Single data set’, then select the ‘FA’, and save it somewhere (e.g., in the same folder that contains all the *.mat files):



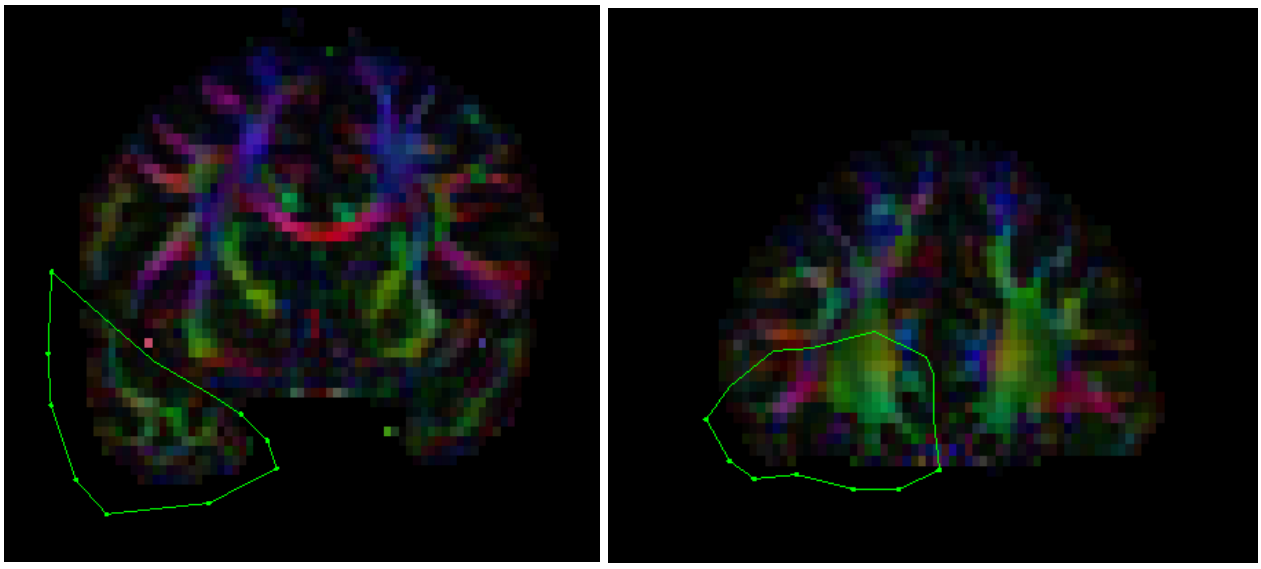
Now, we are going to define the ROIs that will be used to reconstruct the pathways of the R-UNC. This is the step that requires our ‘prior knowledge’ (aka ‘anatomical expertise’): *“The uncinate fasciculus is a hook-shaped bundle that links the forward portions of the temporal lobe with the inferior frontal gyrus and the lower surfaces of the frontal lobe. It does this by arising lateral to the amygdala and hippocampus in the temporal lobe curving in an upward pathway behind the external capsule inward of the insular cortex and continuing up into the posterior part of the orbital gyrus.”* – that is, according to the Wiki page! The idea now is to define a configuration of AND/NOT ROIs that would select only these pathways and would leave out all other tract

pathways of the brain. So let us create such ROIs for our 'template', i.e., 'Data_1.mat'. First, load the DTI file: 'Data_1.mat' and then the tracts file: 'Data_1_Tracts_DTI.mat'. If you were to click 'Draw tracts', you would see all of them (see below – but just do NOT do this – rendering will take forever!). I only do this to show the location of the R-UNC pathways in the context of the rest of the brain. The R-UNC connects the temporal (yellow) with the 'inferior frontal' (white) region.

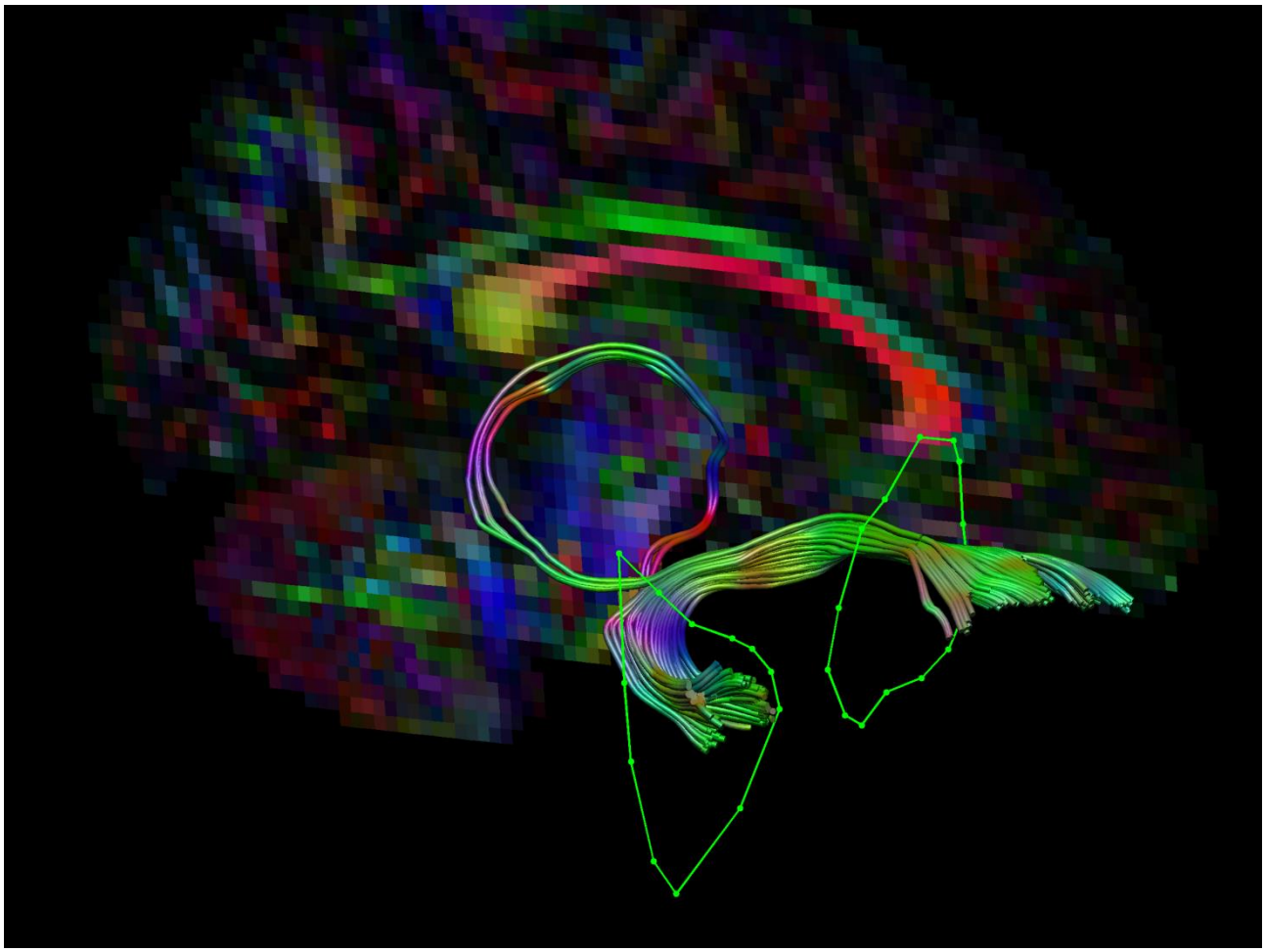
Data_1_Tracts_DTI.mat



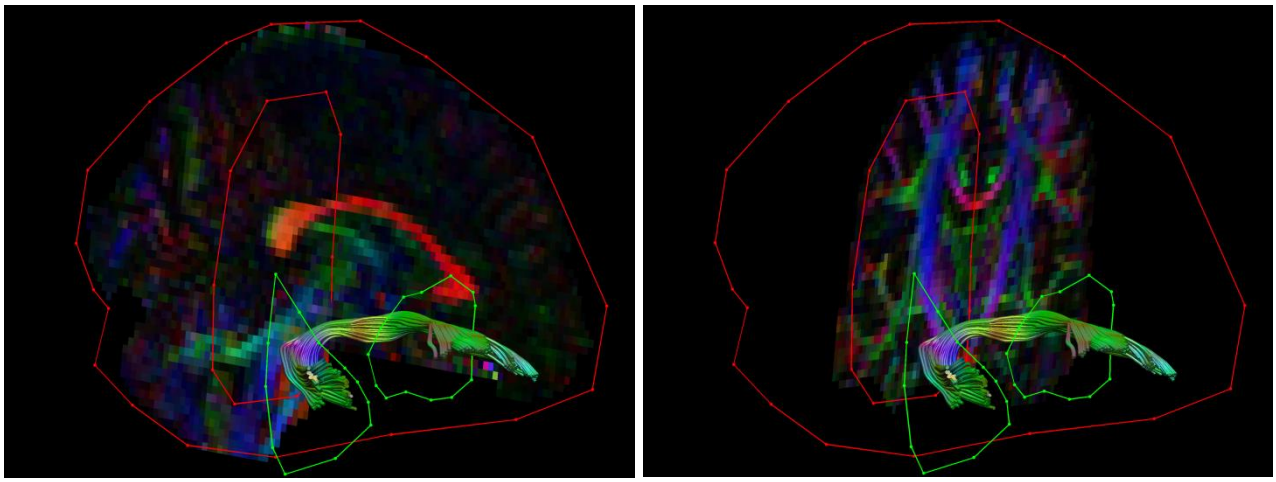
So if I draw two AND ROIs (as in the image above), the only tracts that would remain are these 'fronto-temporal' connections. This is shown below (the yellow ROI above corresponds with the green 'temporal' AND ROI below on the left; the white ROI above corresponds with the green AND ROI below on the right):



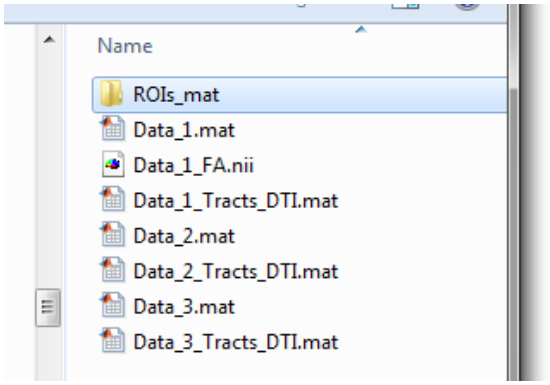
If you click ‘Analyse tracts’ (in the ‘Draw Tools’ palette) or in the menu item ‘Calculate’ → ‘Analyze tracts from ROIs’, and then (after deleting the previous ‘whole-brain’ tracts display), click ‘Draw tracts’, you should see the image below:

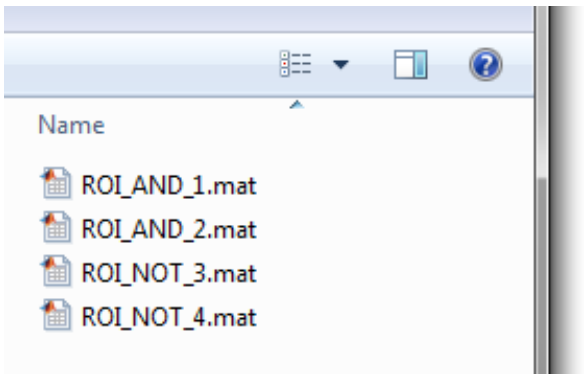


Ideally, only the tracts of interest (R-UNC) are left. In practice, however, there will probably be ‘spurious’ tract pathways that are not part of the structure of interest, but still connect both ROIs (as is the case in this example). These unwanted pathways can be removed by including NOT gates. Try to be as conservative as possible to minimize excluding the ones that are ‘correct’. Below, two NOT ROIs are drawn: one covering the entire mid-sagittal slice, and a second one on a coronal slice posterior (with some margin) to the temporal ROI (after deleting the previous tract display, click ‘Analyse’ again and ‘Draw tracts’):

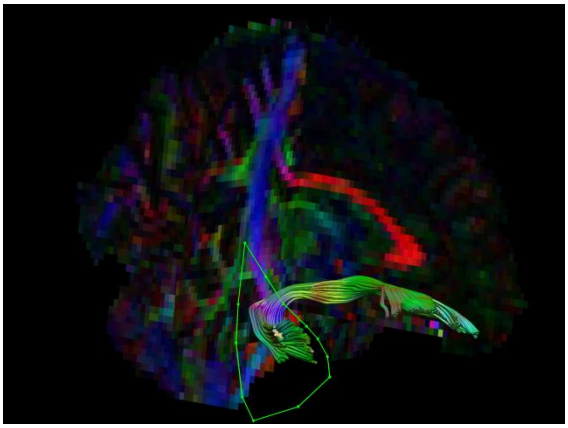


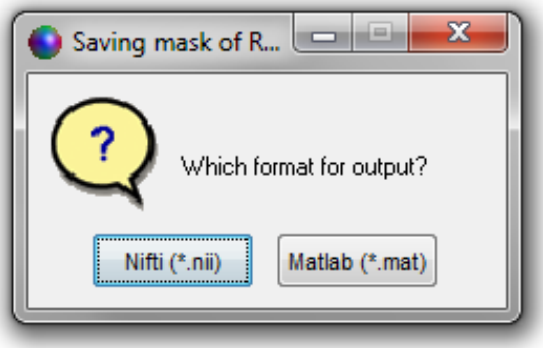
If you are happy with the end result, save each of the ROIs as a separate 3D *.nii mask (using ‘Data’ → ‘Save mask of ROIs’). In this example, I did not save each ROI yet, so I will save all the ROIs in one go with ‘Data’ → ‘Save multiple ROIs’. In doing so, I created a folder ‘ROIs_mat’ (below on the left):



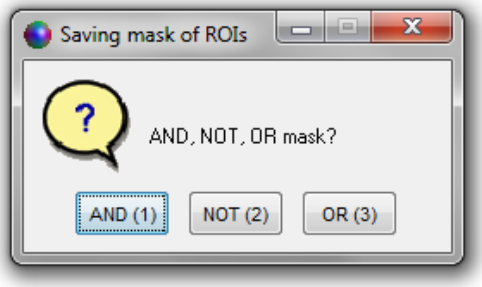


In this folder (above right), there are now 4 ROIs. The following step consists of creating 3D *.nii files from each of these ROIs. The resulting 3D *.nii masks have values '0' representing the background and non-zero values representing the actual ROI. First, delete all ROIs in the current display, and load "ROI_AND_1.mat" via 'Data' → 'Load single ROI' (below left) and then save it via 'Data' → 'Save mask of ROI(s)'. A popup will appear (below right). Here, you should choose 'Nifti' format.

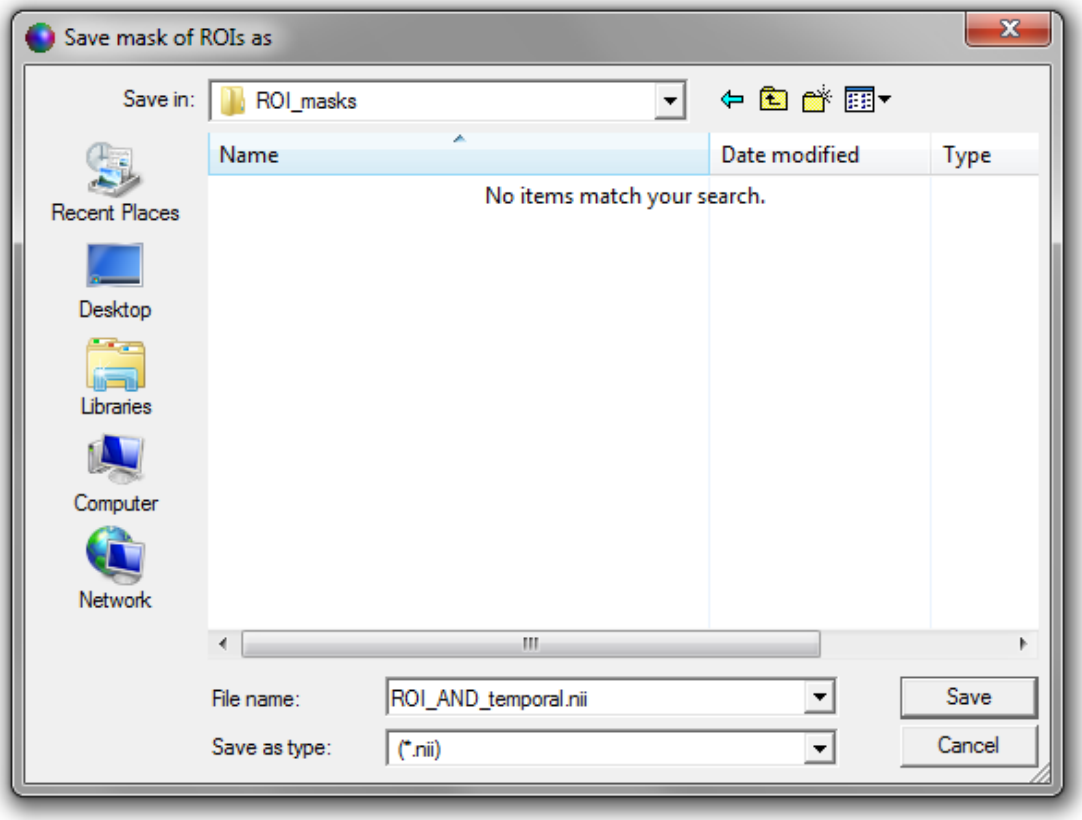




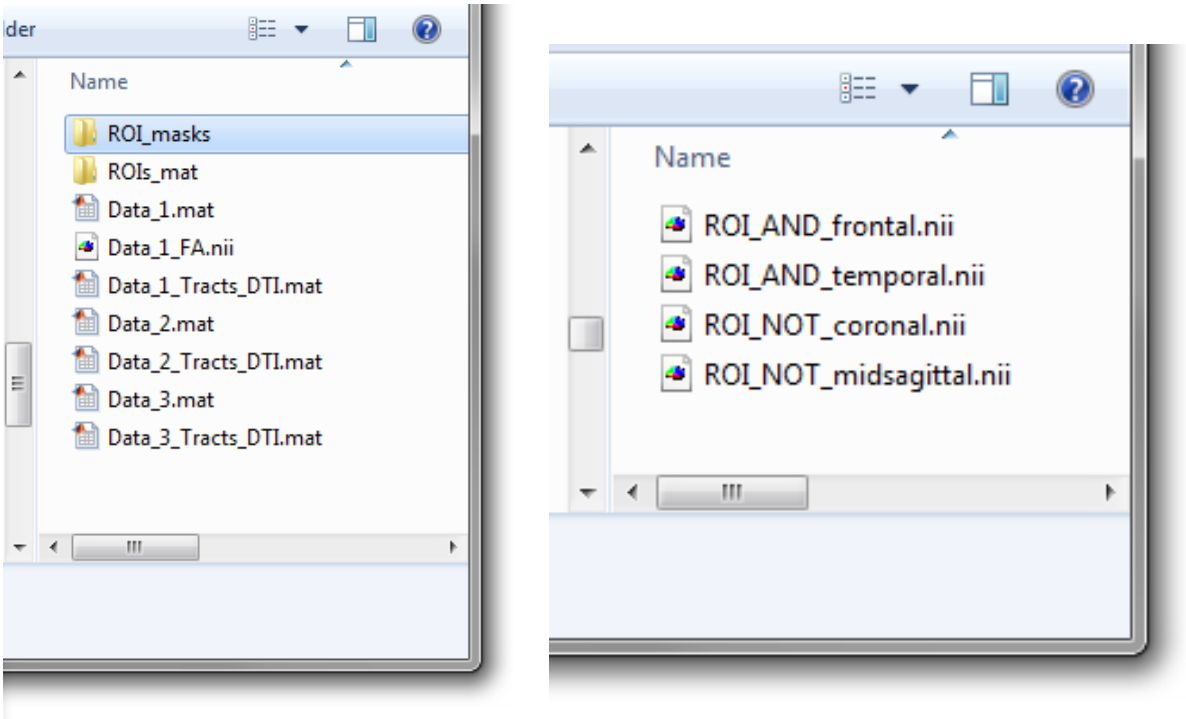
Then, another popup appears, asking you:



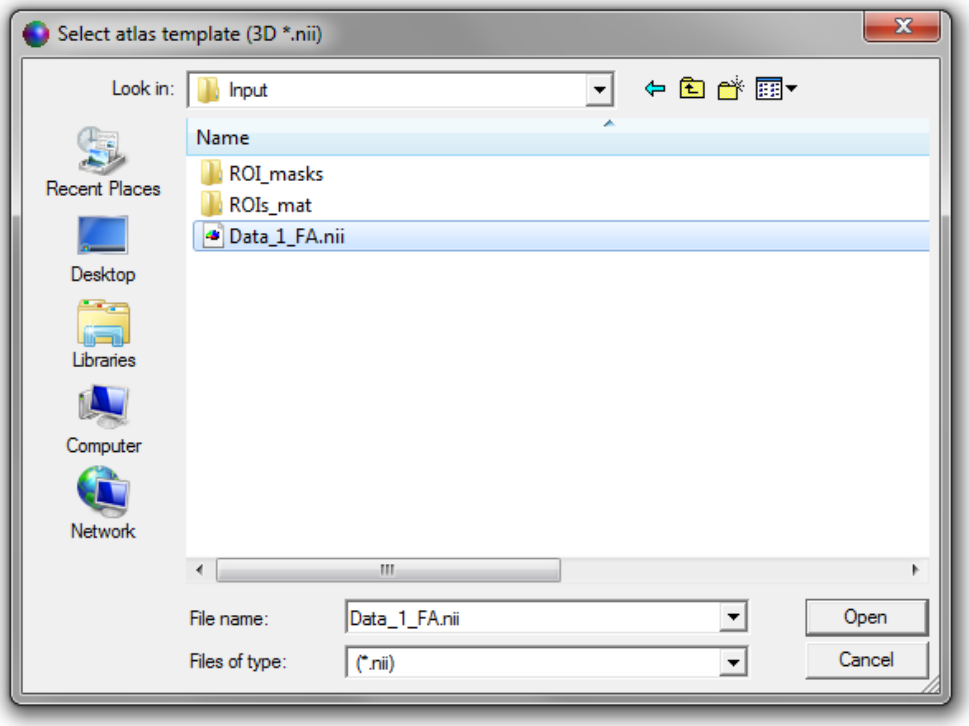
The choice here is not too important to be honest, as you just want to create a mask with '0' values in the background and non-zero values in the ROI. So just select 'AND (1)'. The next step, however, which is giving a name to this *.nii file, is crucial as it is the name itself (or at least part of it) that will reflect what type of ROI mask it represents! So in this case, being an AND ROI, we name it: 'ROI_AND_temporal.nii' and put it in a new folder that I created called 'ROI_masks' (see below).



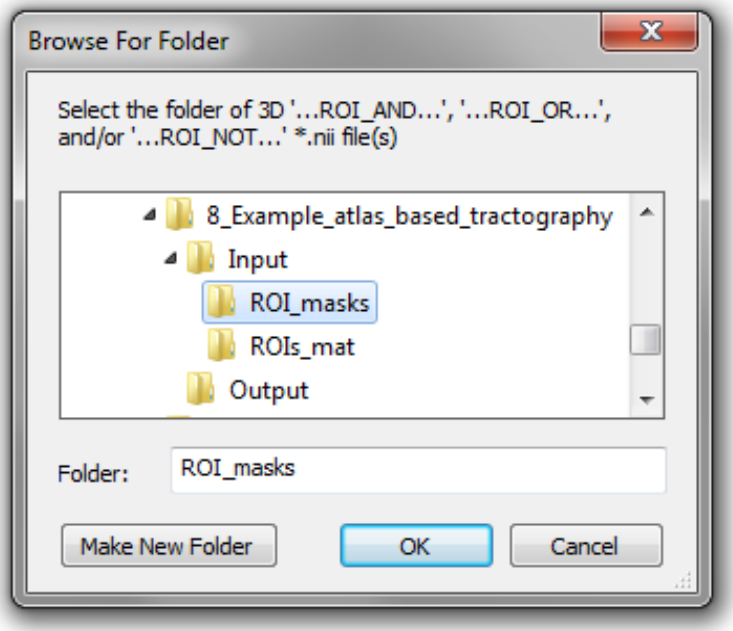
Again: there *should* be a part of the name that is called 'ROI_AND' – the rest (before or after 'ROI_AND') can be anything (in this case '_temporal'). Now, you can do the same for the other ROIs. Analogously, for the NOT ROIs, there should be a part of the file name that is 'ROI_NOT' (see below).



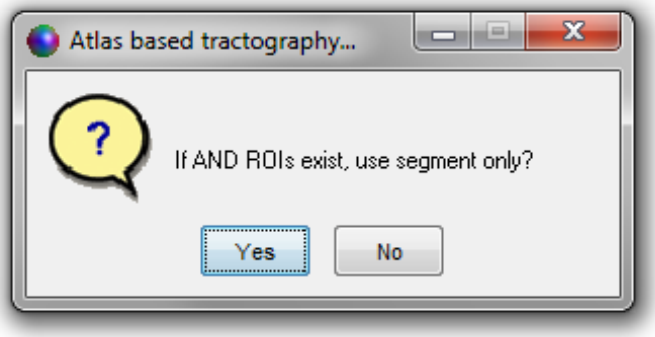
Finally – now we can actually use the tool! So go to 'Plugins' → 'Atlas based tractography segmentation'! If you click it, there will be the request to select the 'atlas template'. Here, you will select the FA map of our 'representative' subject, i.e., 'Data_1_FA.nii' (see below).



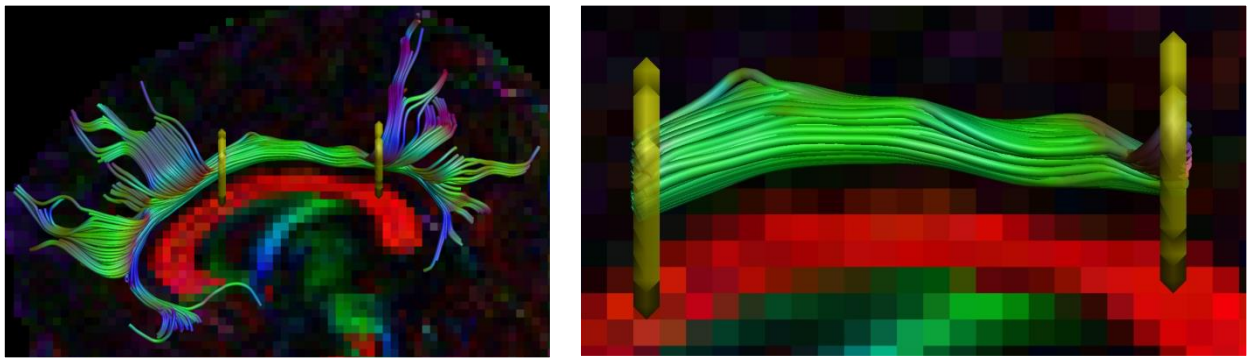
In the next step, select the folder of 3D *.nii masks:



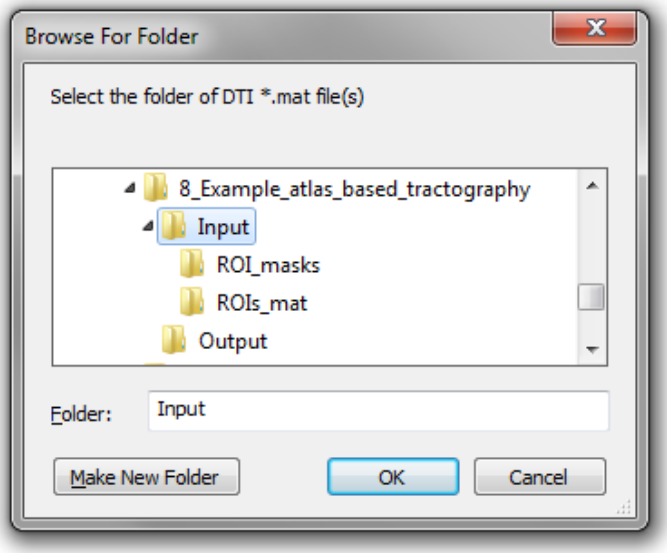
The next popup states:



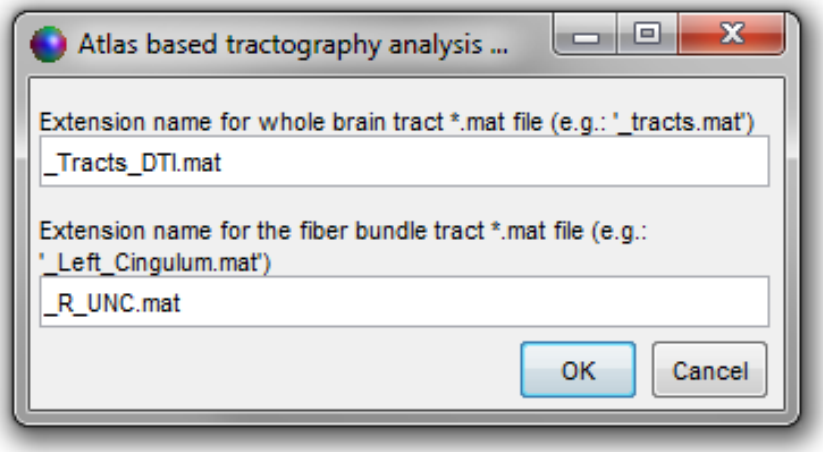
Quick intermezzo! By ‘segment’, I refer to the piece that is between the two outer ROIs. As an example, consider the two tract representations of (part of) the cingulum below. The right one is what I mean with ‘segment’ – so just a section, defined by the two ROIs.



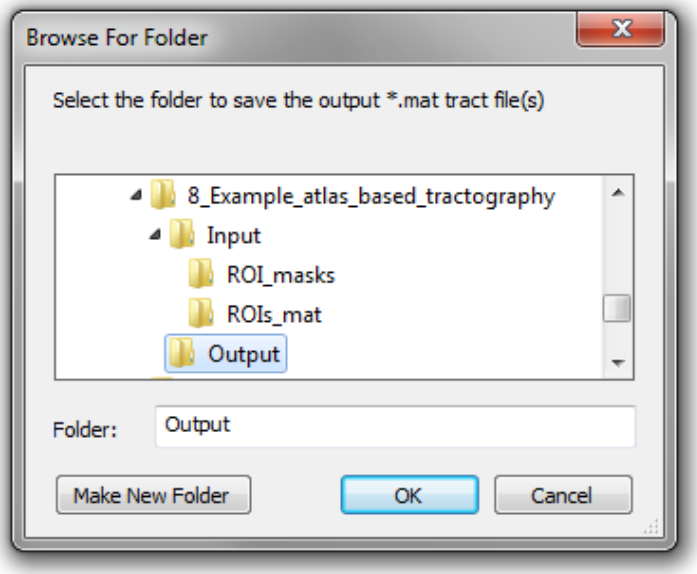
Back to the R-UNC… So now, let us take the entire bundle of pathways (so click ‘No’ in the popup window). Then, we need to select the folder that contains the DTI (and tract) *.mat files (see below).



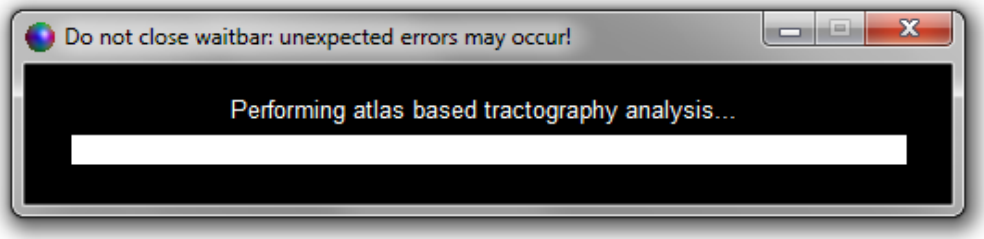
After you select this folder, you get yet another popup asking you to fill in the suffix part of the tract *.mat file and the suffix part for the name of the output tract *.mat name (see below – note: “it’s case sensitive”).



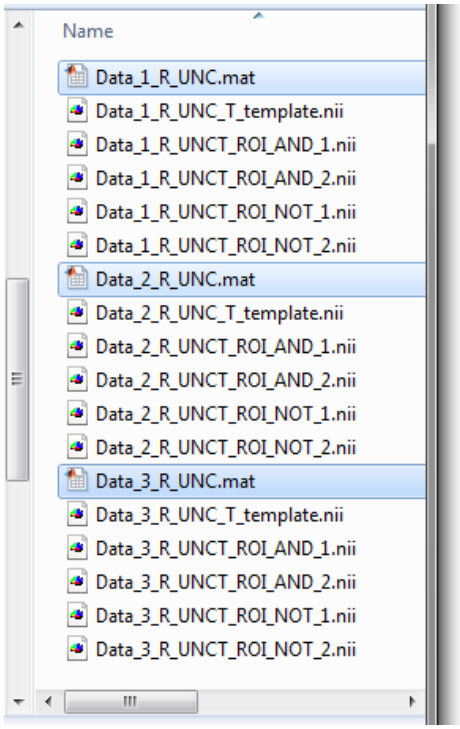
The final popup appears asking to select a folder to put the output files:



And now … have a cup of coffee (see below)!



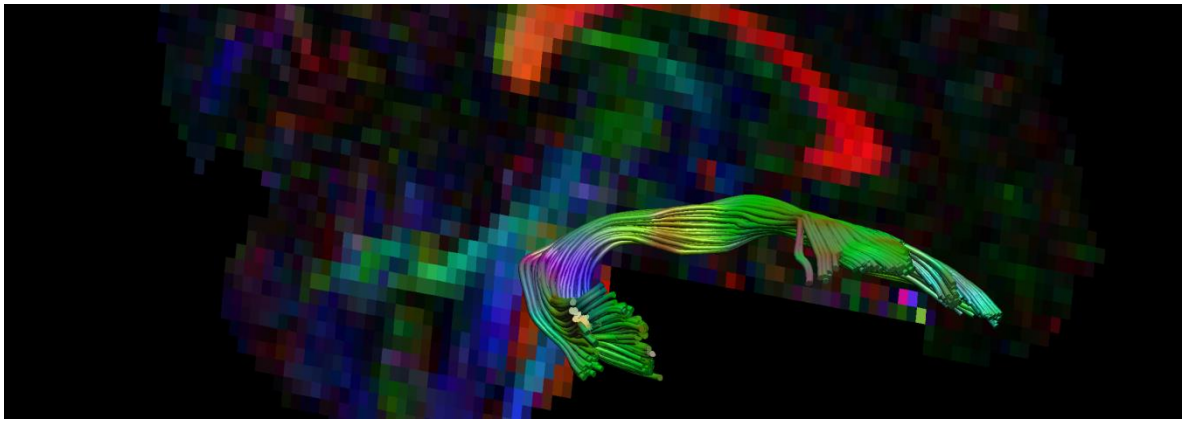
When this “wait bar” disappears, the analysis has finished and you should see output files like the ones below:



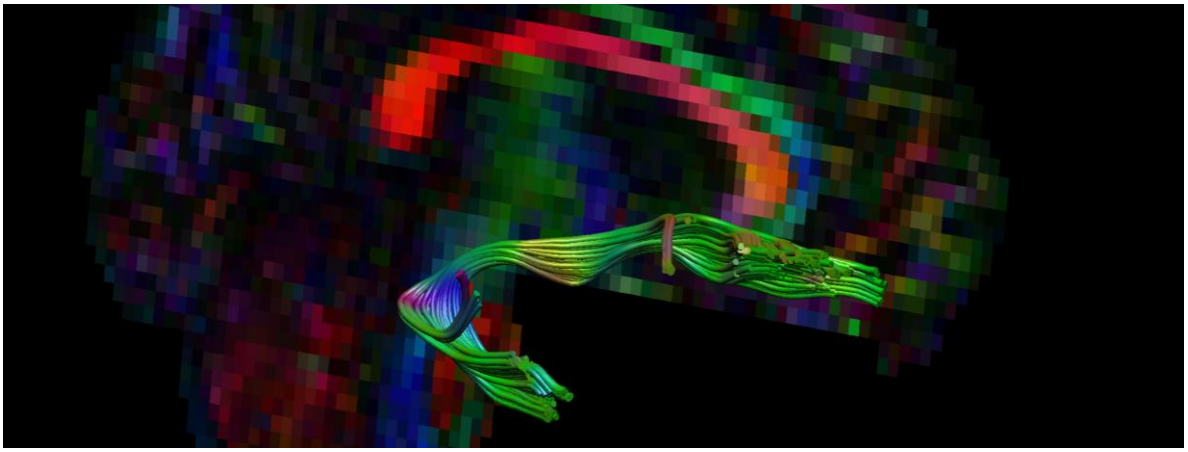
I highlighted the tract *.mat files (above): these are the results! The other data sets are the warped templates and ROIs for each data set (just a post-hoc sanity check – or to see what went wrong if it did go wrong).

So let's just look at the R_UNC tract results for each data set below:

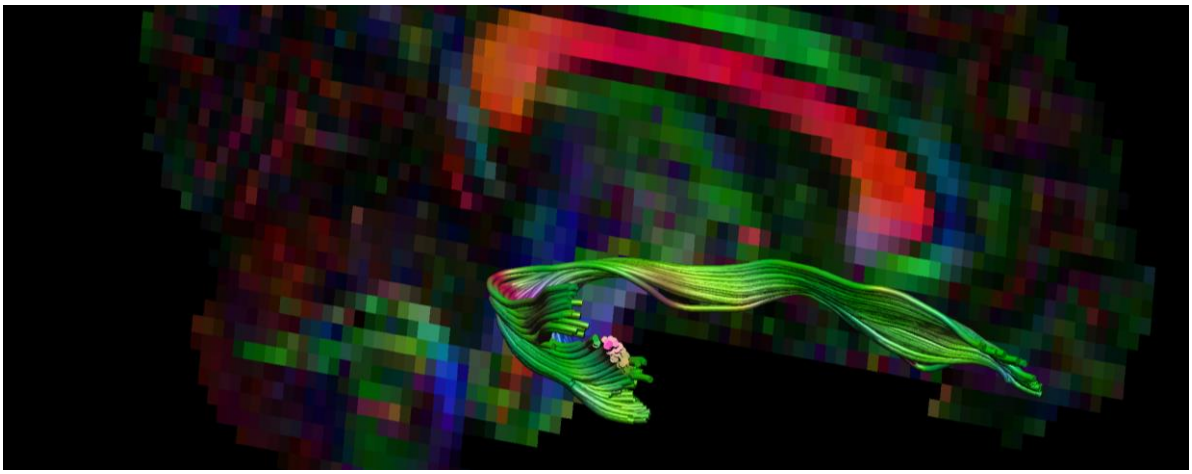
"Data_1_R_UNC.mat" (as expected – we used this one to define the R-UNC in the first place!)



"Data_2_R_UNC.mat"



"Data_3_R_UNC.mat"



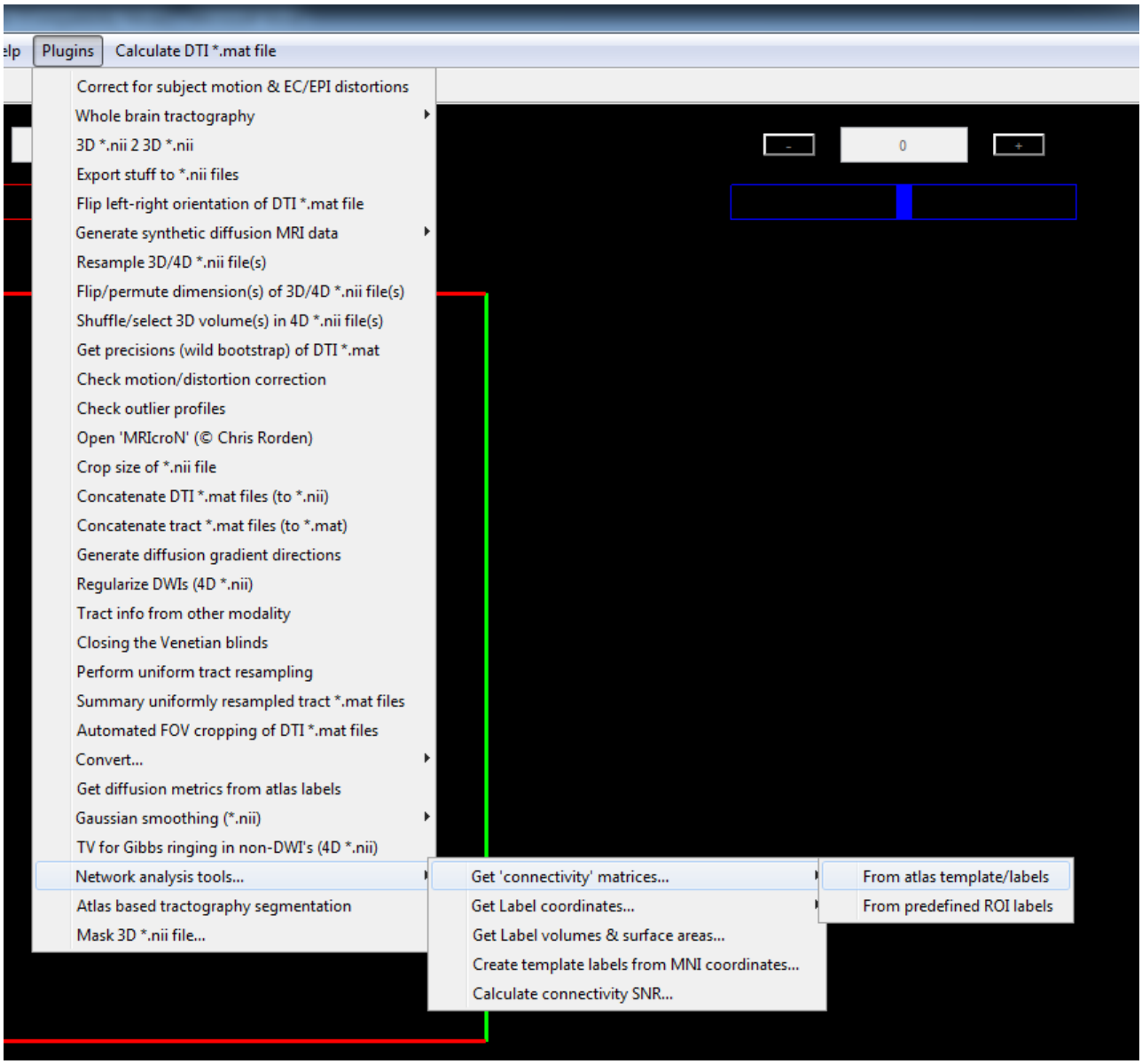
Not bad ;-)! Finally, to get the statistics of these tracts, use the tool 'Plugins' → 'Convert' → ' Info of tract *.mat file(s) to *.txt'. One more thing: with 'Settings' → 'Atlas based tractography' → 'Tract length range', you can further constrain the output tracts to be within a specific length range (again – just an extra way of using prior knowledge in the analysis).

14. Network analysis tools…

If you think of the brain as a network of connections between brain regions, you may be interested in investigating the properties of this network. With *ExploreDTI*, you can obtain the so-called "connectivity matrices" (CMs), from which you can derive network properties, such as "efficiency", "small-worldness", etc. In addition, with *ExploreDTI* you can create the typical network visualizations.

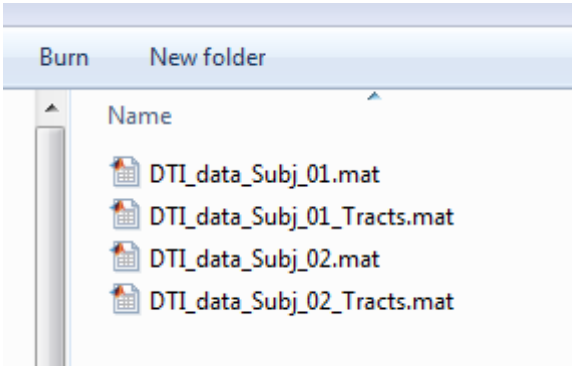
For more background information and example studies in which the network analysis tools of *ExploreDTI* have been used, see the work of Reijmer YD et al (Neurology 2013, Diabetes 2013, Brain 2015) and Caeyenberghs K et al (Brain Struct Funct 2014, Hum Brain Mapp 2014, Neuroimage Clin 2012, Front Hum Neurosci 2013).

In the following, I will first explain how to compute the CMs. Note: there are two ways in *ExploreDTI* to obtain these CMs: "From atlas template/labels" and "From predefined ROI labels" (see below). After that, I will show you how to make network visualizations.

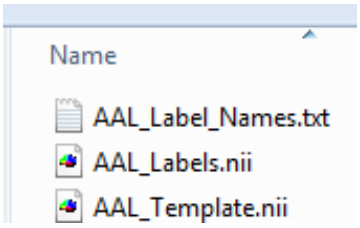


The difference, in a nutshell, is that with the "From atlas template/labels" tool, you will need an atlas template with associated labels (ROIs), which will be registered to each subject's data set. With the other tool ("From predefined ROI labels"), the ROIs are already derived with another approach (for instance, a T1-based cortical segmentation) and located in the same image space as the diffusion data (see example p. 44).

Let's have a look at the "From atlas template/labels" tool. For each subject, you will need the following data sets: a DTI *.mat file and its associated whole-brain tracts data set (see example below for 2 subjects):

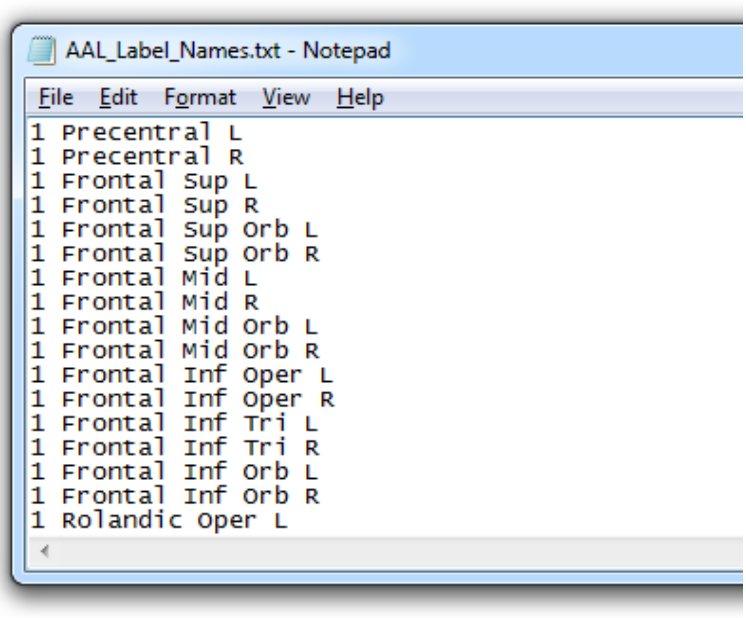


The other files that you will need to run this tool are related to the atlas that you want to use. As an example, let's consider the AAL atlas (see: Tzourio-Mazoyer N et al, NeuroImage 2002). Converted for *ExploreDTI* purposes, the relevant AAL atlas information is now represented in three files:

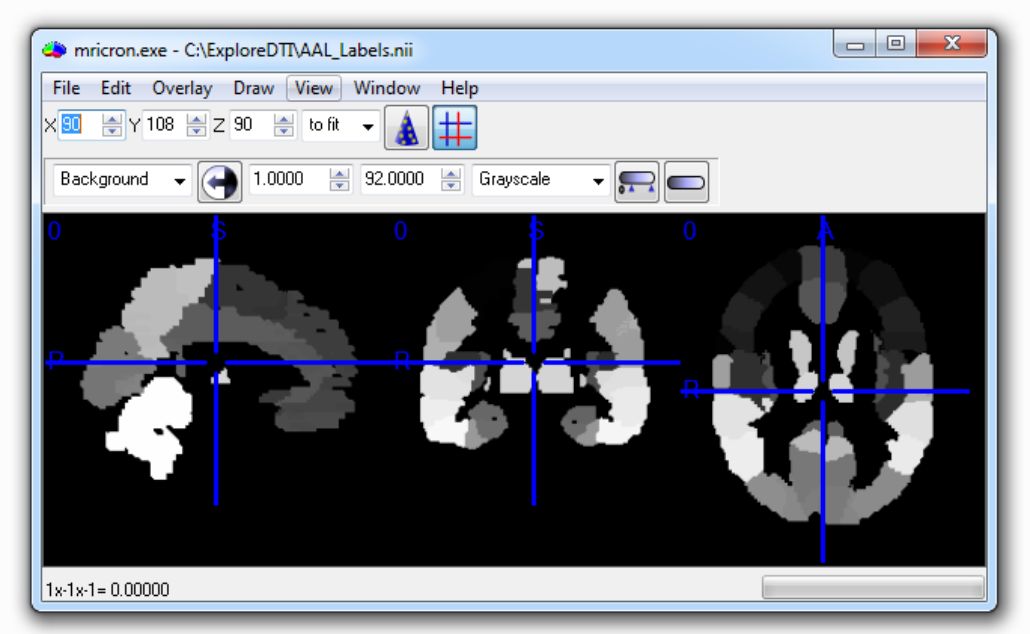


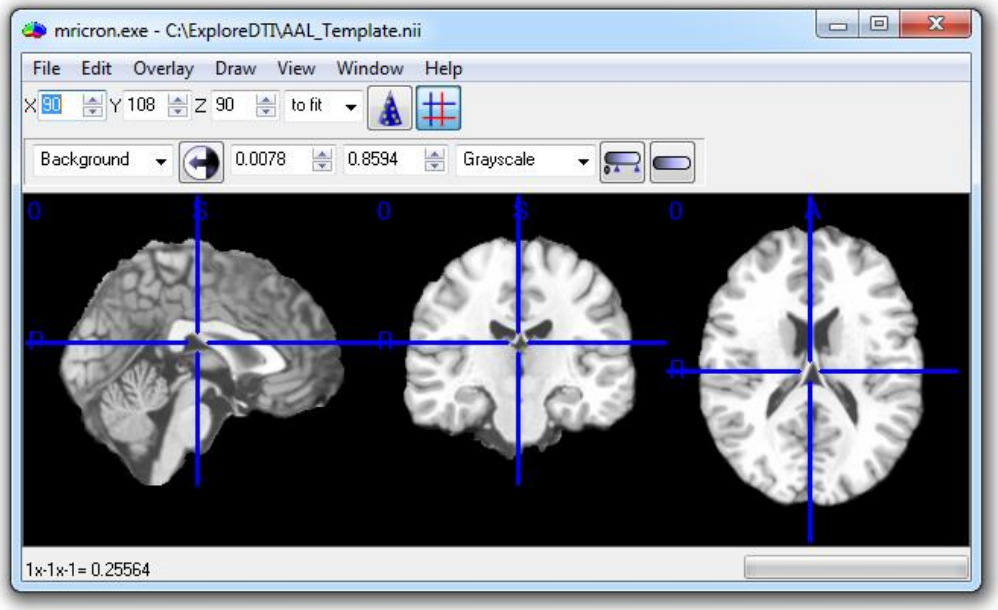
You can download these files from: http://www.exploredti.com/templates/AAL_Template.zip. This one and other examples of such atlas template/label files are provided in the folder "~/Source/Templates".

The *.txt file consists of two columns: the first column contains "0" or "1" values (meaning "0": do not use this ROI in the analysis; "1": do use it for analysis) and in the second column, the label name of the ROI is given. Each ROI is given on a separate row:

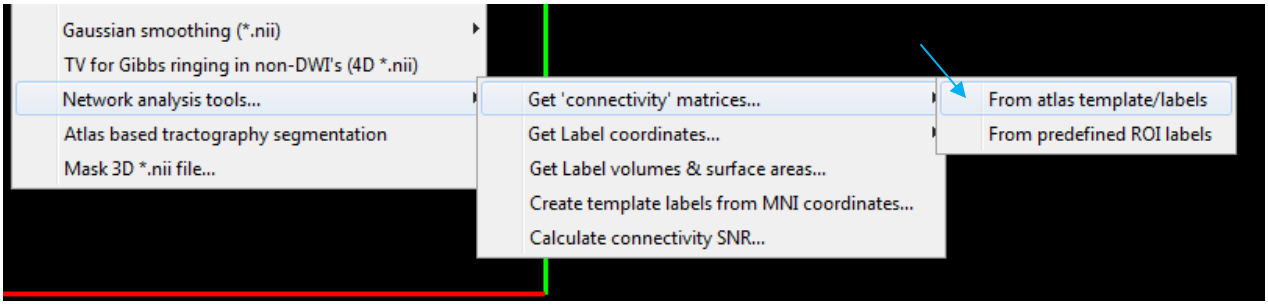


The actual row number *is* important as it represents the intensity value in the associated "Labels" file. So in this case, the 3D nifti file "AAL_Labels.nii" contains the ROIs described in the *.txt file above. More specifically, the region with "0" values is background, the region with intensity values of "1" represents "Precentral L", the region with intensity values of "2" represents "Precentral R", etc. The corresponding AAL atlas template (here, "AAL_Template.nii") is needed to perform the registration. Both files are shown below with MRIcron:

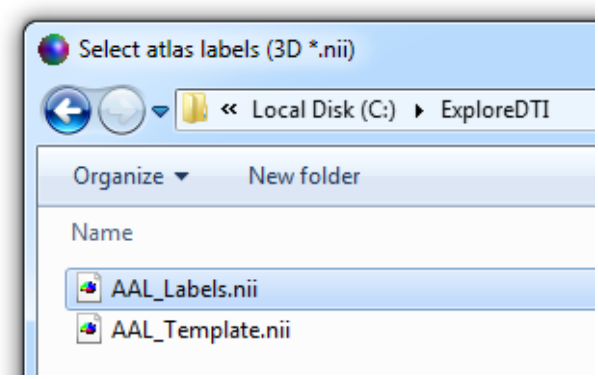




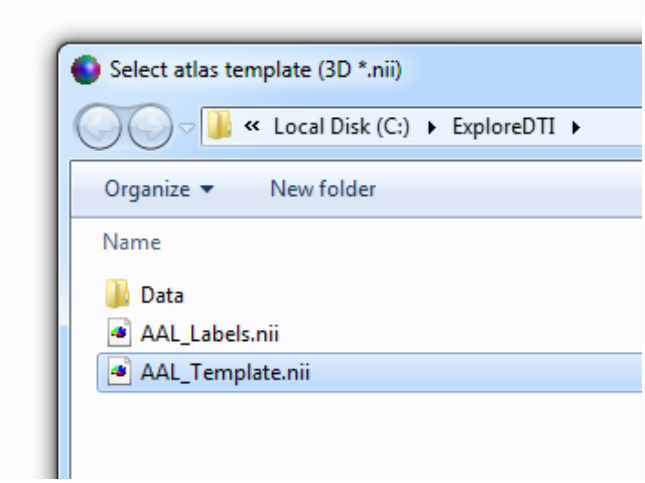
Let's actually run this tool now... Go to "Plugins" and then (see below) "From atlas template/labels" (btw, to speed up things, you may want to use the "Use parallel computing…" option in "Settings" – make sure you have enough memory… memory usage will be proportional with the number of cores that you define and can be roughly in the order of 2 GB per core).



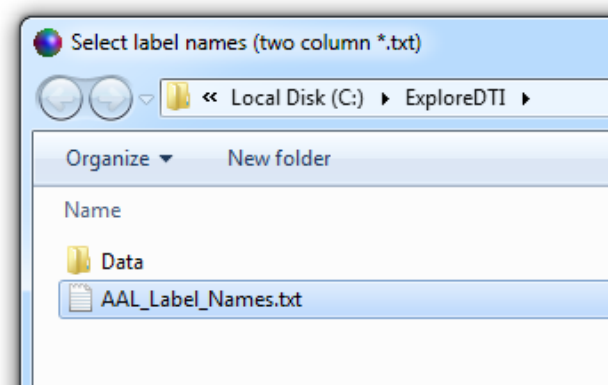
The steps to follow are first:



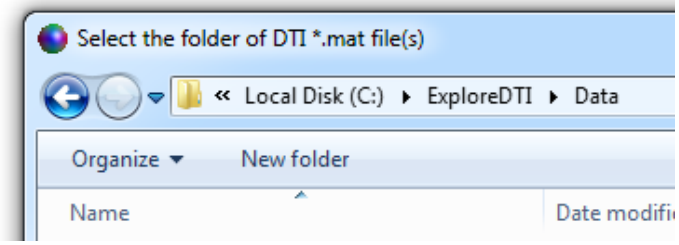
Then:



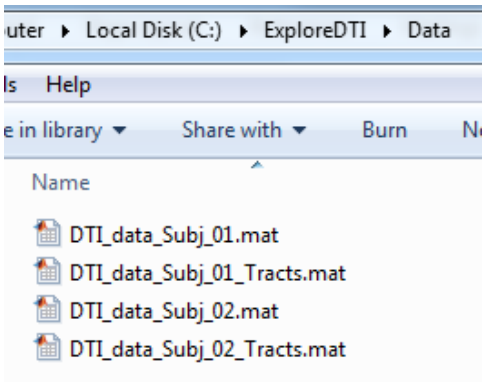
Then:



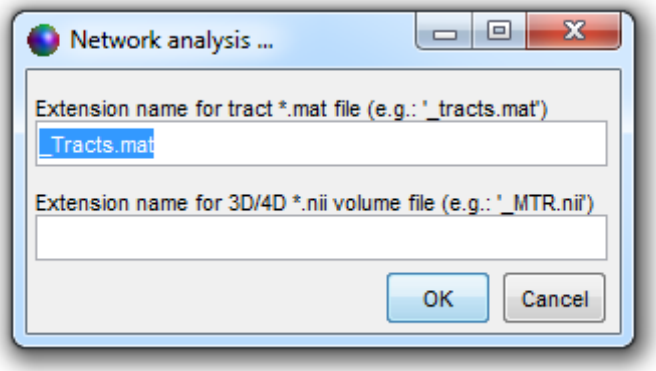
Then:



Recall… this is the folder that also contains the *.mat tract files (see below - this is important for the next step).

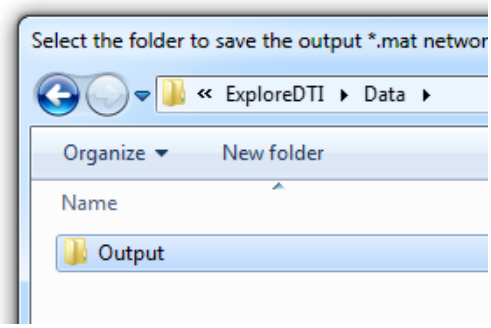


Then you need to enter the end-part of the tract file name – simply to find the corresponding files (note: this is case-sensitive!):

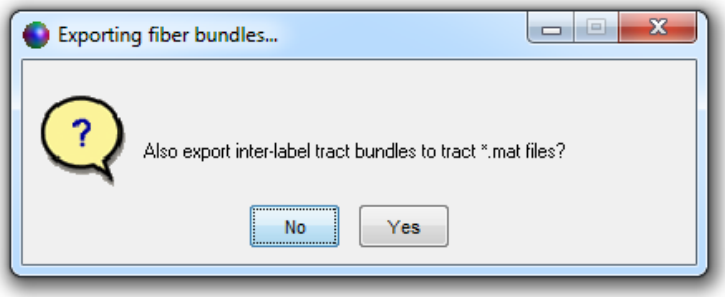


Keep the second line empty (unless you have a *.nii file that is registered to the diffusion data that you want to integrate in this pipeline… anyway, I'll explain this particular feature on another occasion).

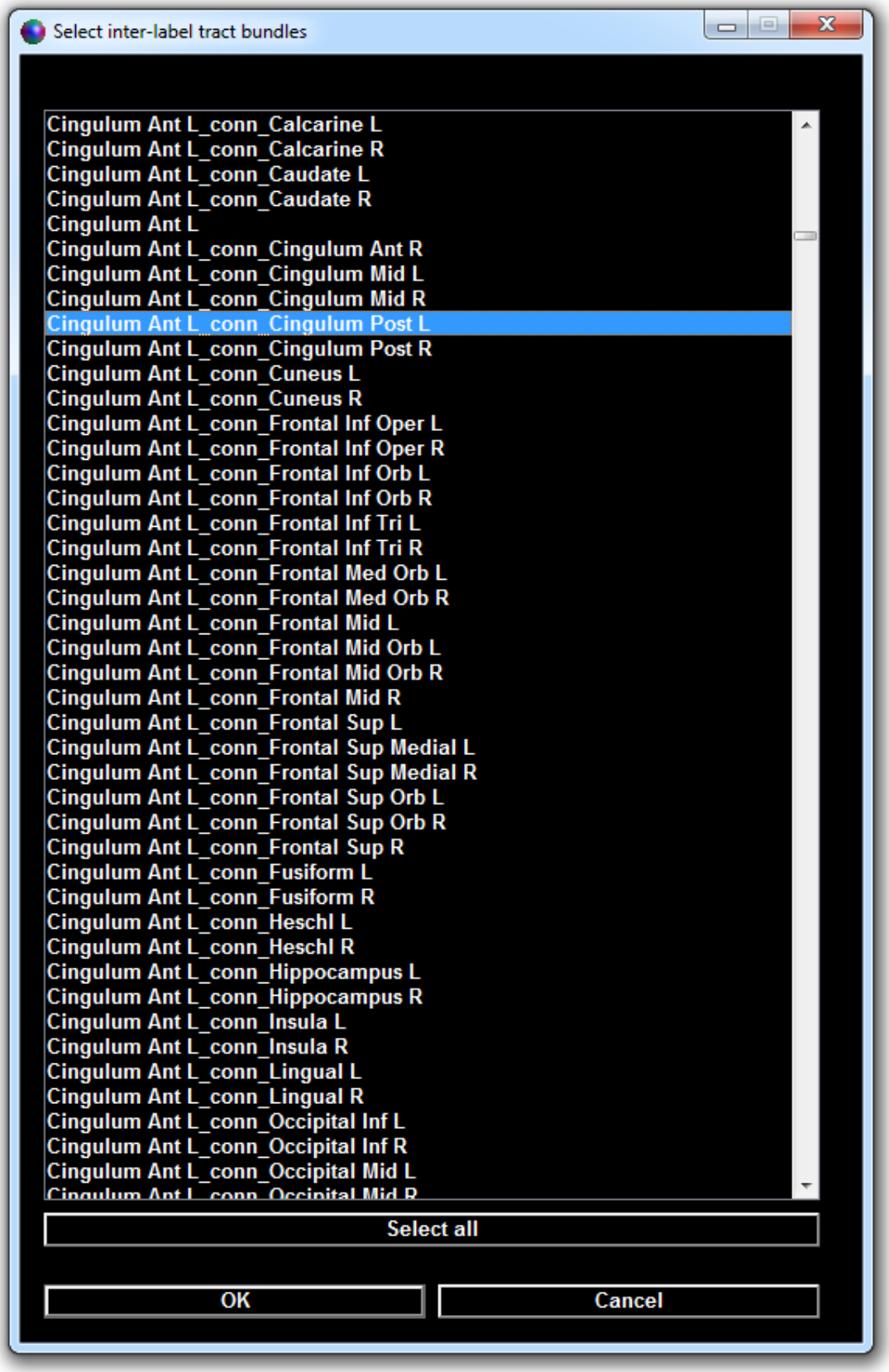
The next step is then to select the folder to save all the output (here, I made a folder called "Output" – just because I can):



Then the next pop-up appears:

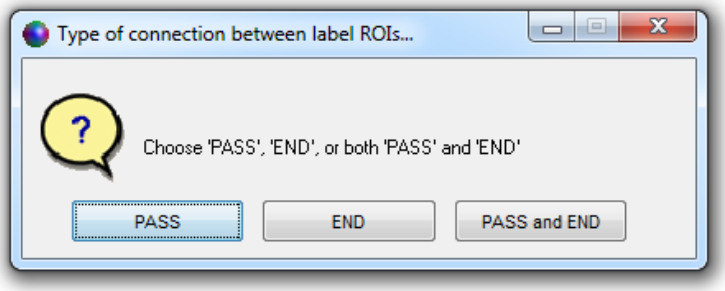


Well – as it says, you can choose to save tracts (if there are any) between the different combinations of the ROIs defined in the template. Let us proceed with yes, then the following pop-up shows all these potential ROI combinations (as an example, I'll choose the tracts that connect the left anterior and posterior cingulate cortices):

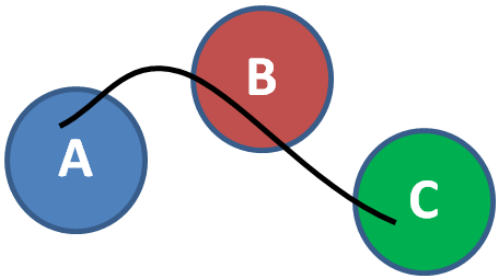


These files can be loaded into *ExploreDTI* afterwards (sanity checks!) or used for other analysis purposes…

The final pop-up before the processing actually starts is:



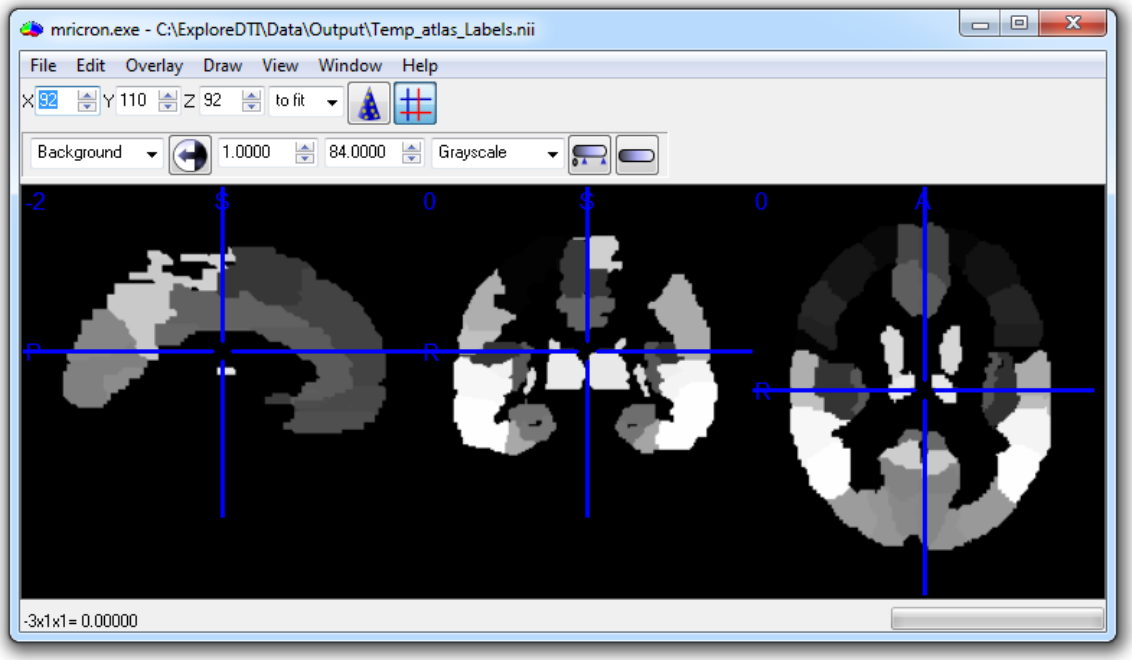
In short, with *ExploreDTI* you can differentiate between tract pathways "passing through" two ROIs and those that actually "end" in these ROIs (see sketch below). In current literature, this differentiation is generally not made – the "pass" definition is implicitly taken then. FYI – there is hardly any extra computational cost if you compute both ("pass" and "end"), hence the third option.



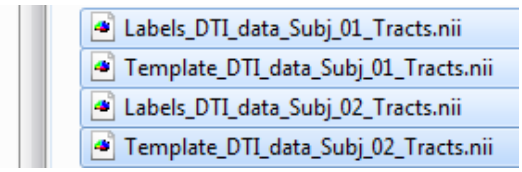
"PASS" definition: A is connected to B, B is connected to C, A is connected to C
"END" definition: A is connected to C (the pathway terminates in regions A and C, but not in B)

Processing time per data set is approximately 10 min to 2 h depending on your hardware specs, data size, etc. The output files are (FYI, I chose the "PASS and END" option):

(1) "Temp_atlas_labels.nii": this is simply the "template labels" file of the atlas, but then only with the ROIs that you have selected for analysis (recall: the first column in the text file being "0" or "1"). In this example, notice that the cerebellar ROIs are not included. This data set will be useful for visualization purposes (see later).

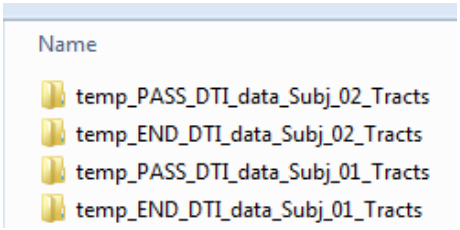


(2) For each subject, there will be a “Template_X.nii” and a “Labels_X.nii” file with the “X” representing the tract file name (so in this example, the “X” is “DTI_data_Subj_01_Tracts” for subject 1 and “_DTI_data_Subj_02_Tracts” for subject 2):

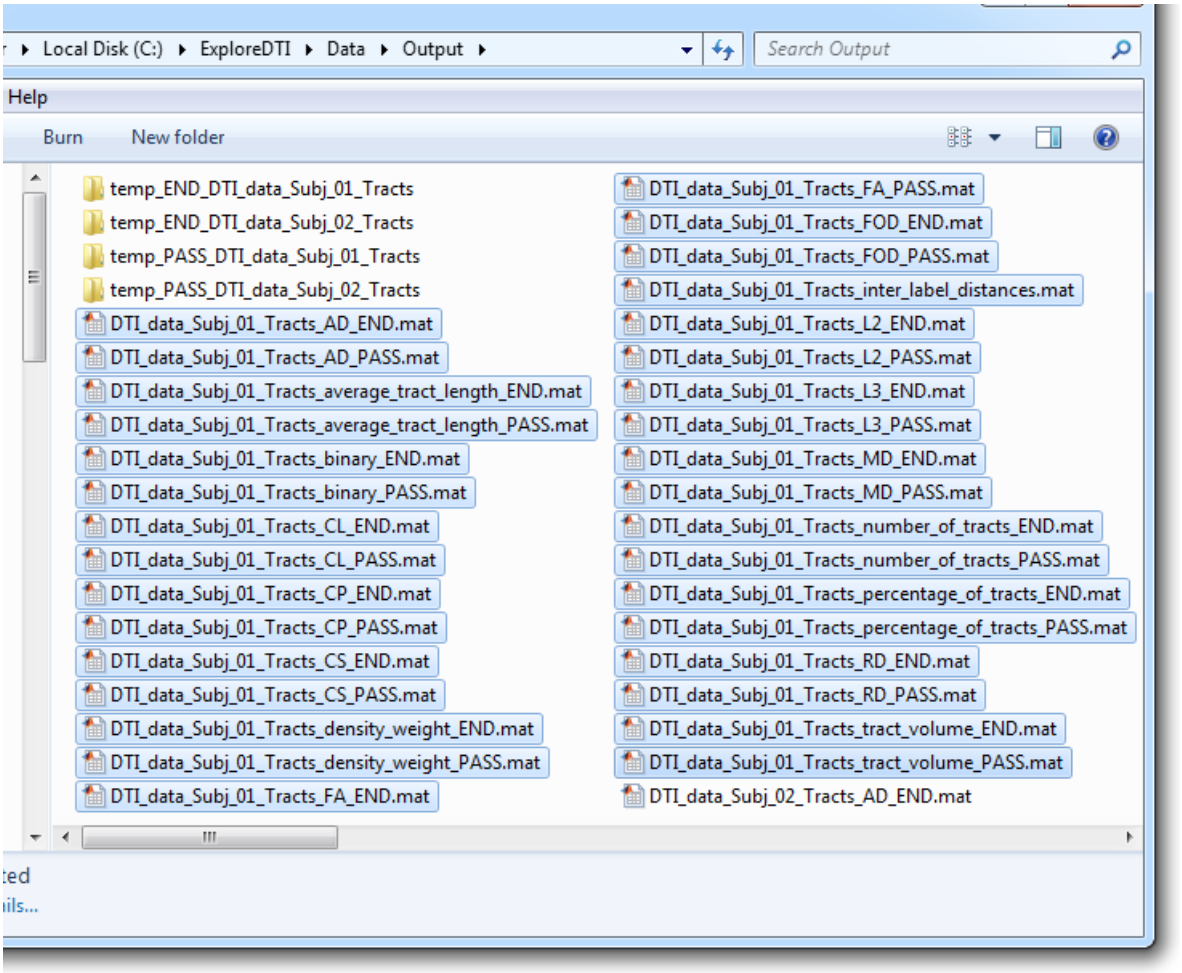


The “Template_X.nii” and “Labels_X.nii” files are just the transformed “template” versions for each subject (this output serves for quality control).

(3) If you chose to export “inter-label” tract bundles, these will be located in “temp_X” folders with the X reflecting the subject tract file names (with “END” / “PASS” specification). In this example:



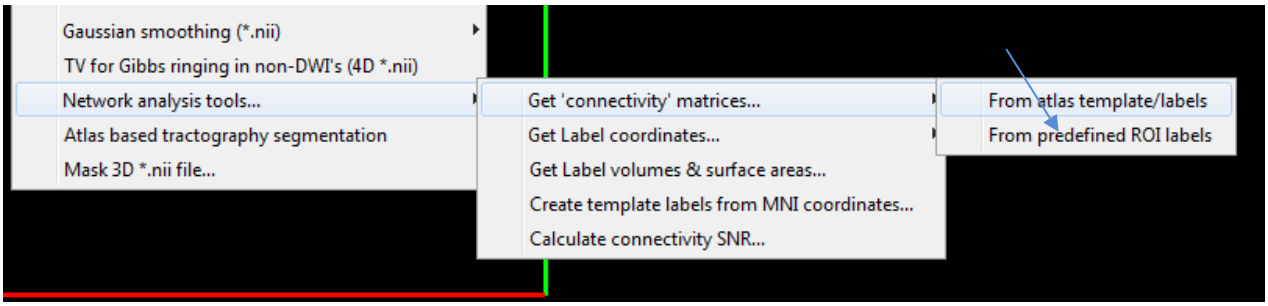
(4) The actual “connectivity matrices” (CMs)! For each subject, the following *.mat files are generated (I highlighted the files for subject 1):



As you can see, there are CMs for “PASS” and “END” definitions. Each *.mat file contains one variable, called “CM” and is simply an N x N matrix with N the number of ROIs in the analysis.

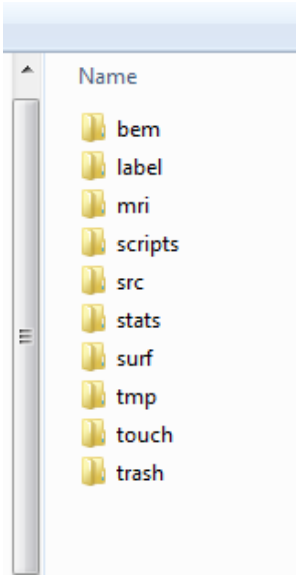
There are many ways to construct a CM. The value at position (i,j) of CM either represents the “number of tracts” (*_number_of_tracts_*), fractional anisotropy (*_FA_*), mean/axial/radial diffusivity (*_MD/AD/RD_*), second/third eigenvalue (*_L2/L3_*), the linear/planar/spherical (“Westin”) anisotropy measures (*_CL/CP/CS_*), the average tract length (*_average_tract_length_*), the binary value: connection or not (*_binary_*), the density weight (see definition in P. Hagmann et al PLoS Biol 2008, *_density_weight_*), percentage of tracts (*_percentage_of_tracts_*), tract volume (*_tract_volume_*), the distance between the ROI centers of gravity (*_inter_label_distances_*), and in case of CSD: the amplitude of the fiber orientation distribution (*_FOD_*) of the fiber bundle connecting ROI “i” with ROI “j”.

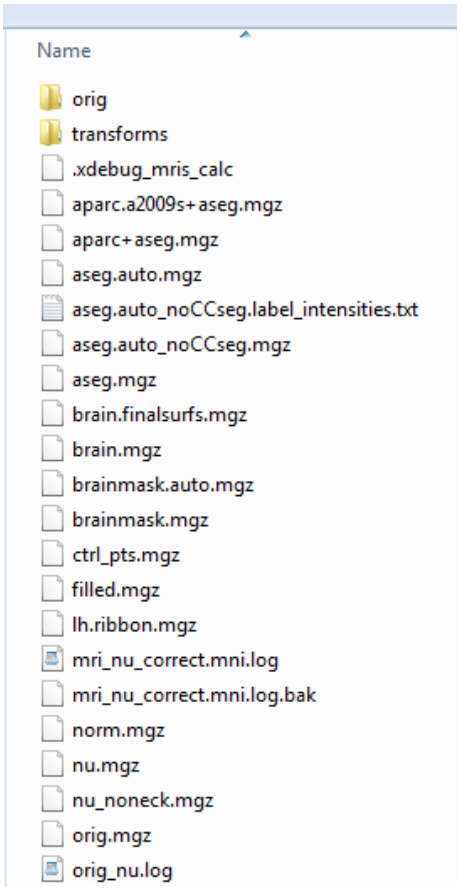
For the other option, “From predefined ROI labels” (see below), the only difference is that for each subject, you’ll need a separate 3D *.nii file (same name, but with suffix added, such as “_labels.nii”) with ROIs in a similar format to the “atlas labels” *.nii file from above. Note that these “label” files must be in the same space of the diffusion data (computation times are in the order of a few minutes to a few hours per data set, depending on size).



I will explain the “From predefined ROI labels” plugin with an example. Assume you want to compute the ROIs with FreeSurfer.

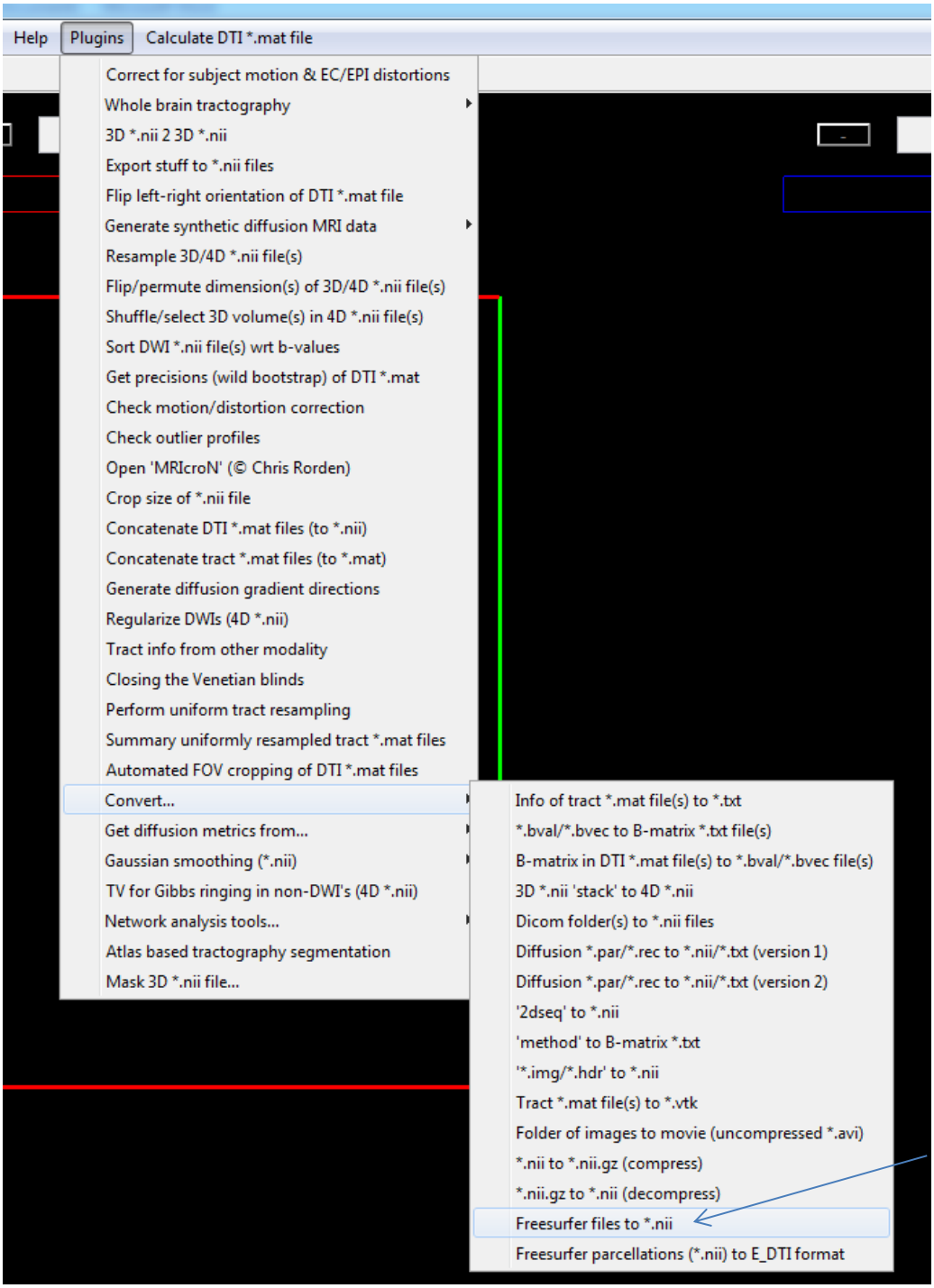
After using FreeSurfer (“recon -all”), the result should be something like the folder structure shown on the left below:





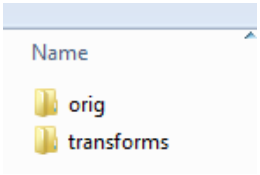
The important folder for now is “mri” and looks something like the right above.

First, convert the FreeSurfer stuff to “ExploreDTI compatible *.nii files”:

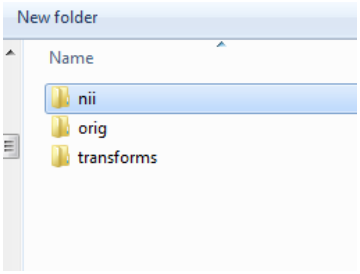


Select multiple:

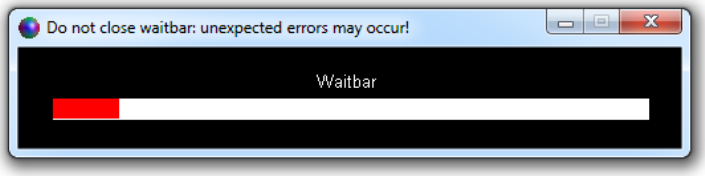
Then select that “mri” folder:



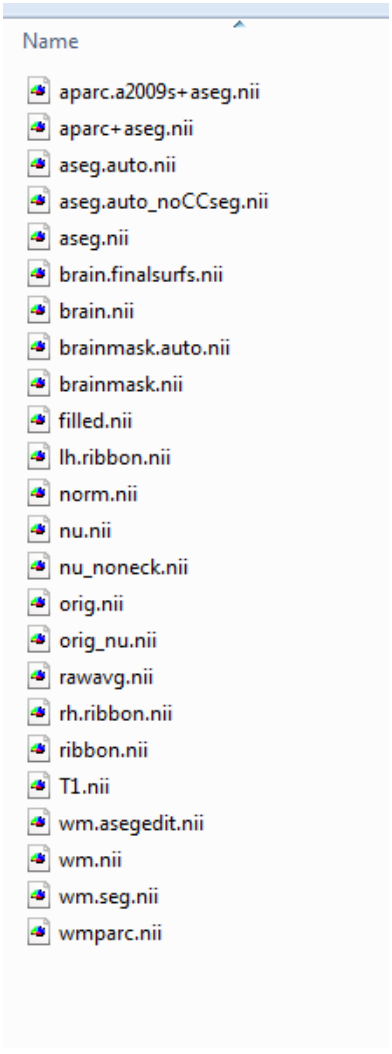
Just make a new folder “nii” and select it as the output folder



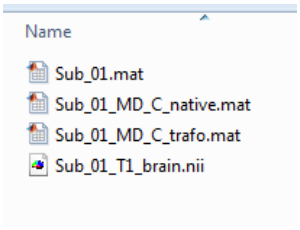
Then wait a bit:



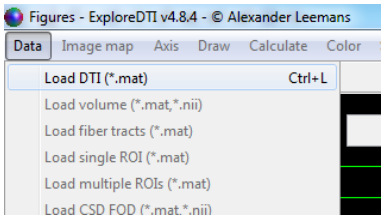
The result looks something like:

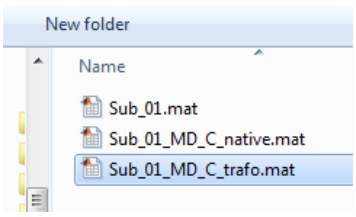


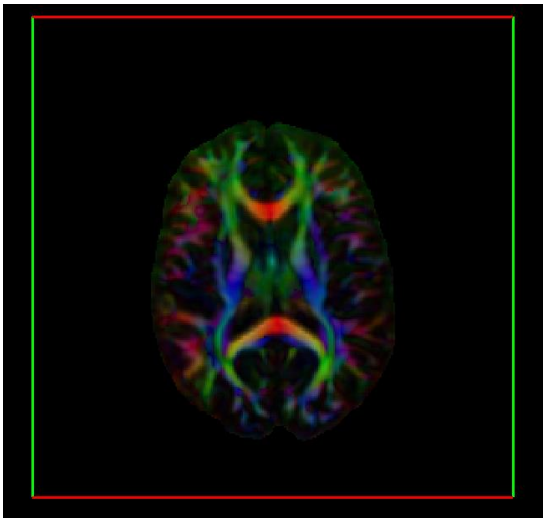
The “brain.nii” file is now actually useful! If your DTI *.mat file (the “uncorrected data”!) is called: “Sub_01.mat”, then rename this “brain.nii” file to “Sub_01_T1_brain.nii”. Now you can use “Sub_01_T1_brain.nii” for doing the SM/EC/EPI distortion correction step (see p. 24/25). Make sure you change the deformation axes settings to [1 0 0] (assuming you have the PE along the AP axis). The result after applying the “Correct for subject…” plugin should be something like:

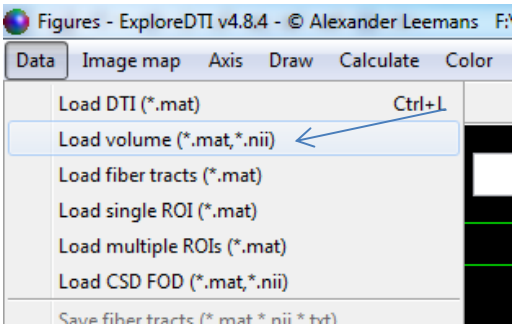


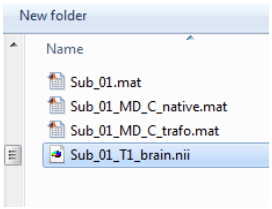
Quick intermezzo: You can check the result by overlaying the color-coded FA with the T1:

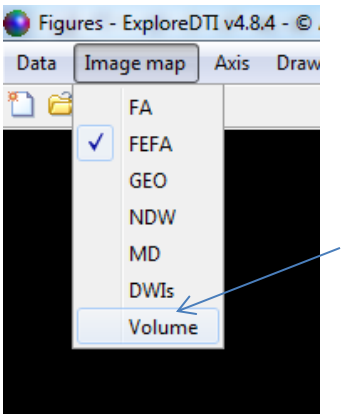


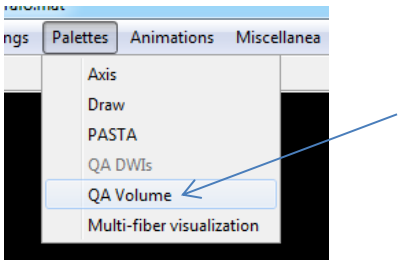


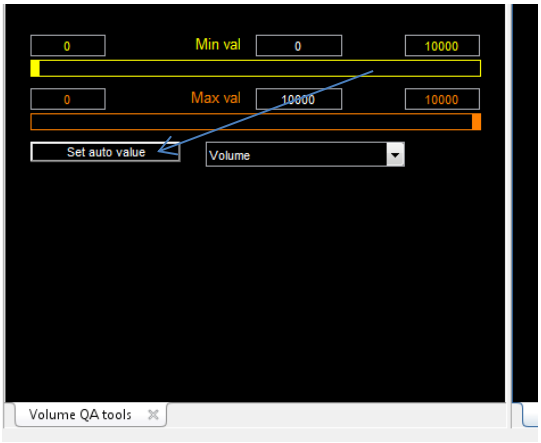


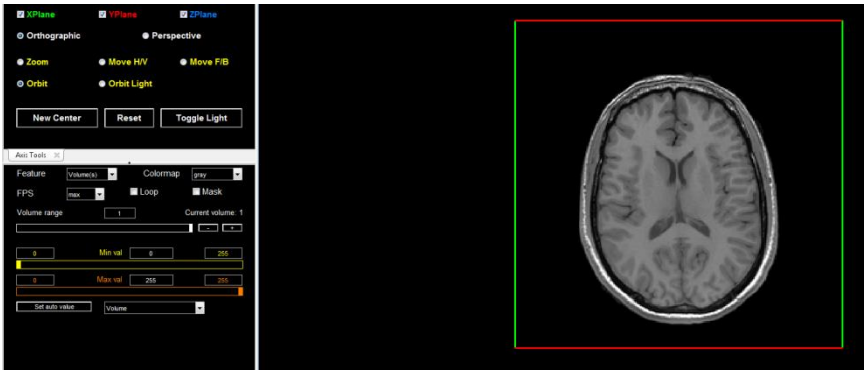


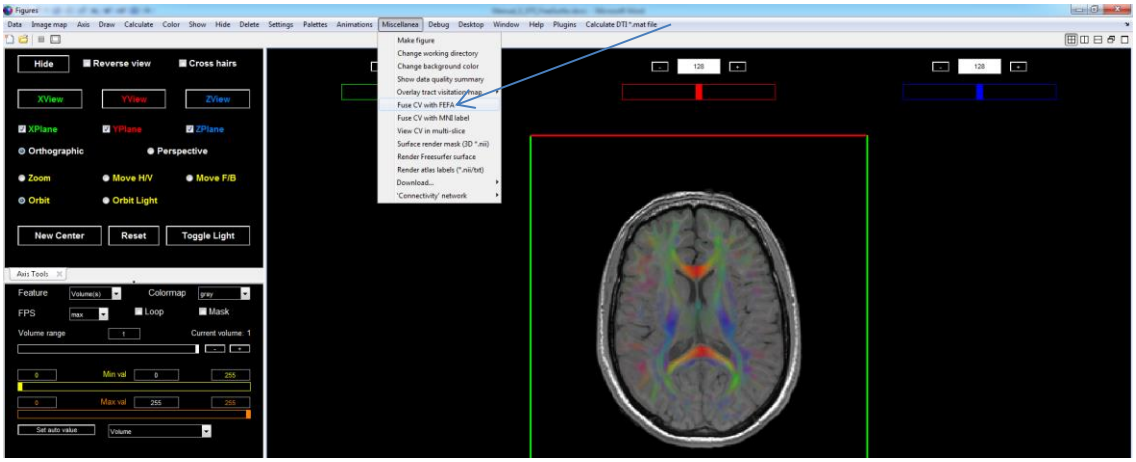

New folder
Name
Sub_01.mat
Sub_01_MD_C_native.mat
Sub_01_MD_C_trafo.mat
Sub_01_T1_brain.nii


Figures - ExploreDTI v4.8.4 - ©
Data
Image map
Axis
Draw
FA
FEFA
GEO
NDW
MD
DWIs
Volume


Palettes
Animations
Miscellanea
Axis
Draw
PASTA
QA DWIs
QA Volume
Multi-fiber visualization

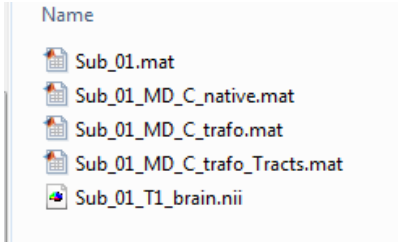

0
Min val
0
10000
0
Max val
10000
10000
Set auto value
Volume
Volume QA tools


XPlane
YPlane
ZPlane
Orthographic
Perspective
Zoom
Move H/V
Move F/B
Orbit
Orbit Light
New Center
Reset
Toggle Light
Axis Tools
Feature
Colormap
gray
FPS
Loop
Mask
Volume range
Current volume: 1
Min val
Max val
Set auto value
Volume



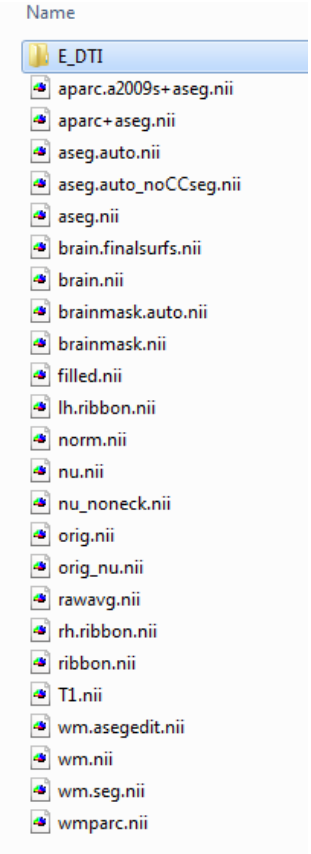
"Miscellanea → Fuse CV with FEFA"… Looks good!

Now do some fiber tractography: DTI or CSD based – whatever you like ;-)! FYI, I named the resulting tract file: "Sub_01_MD_C_trafo_Tracts.mat".

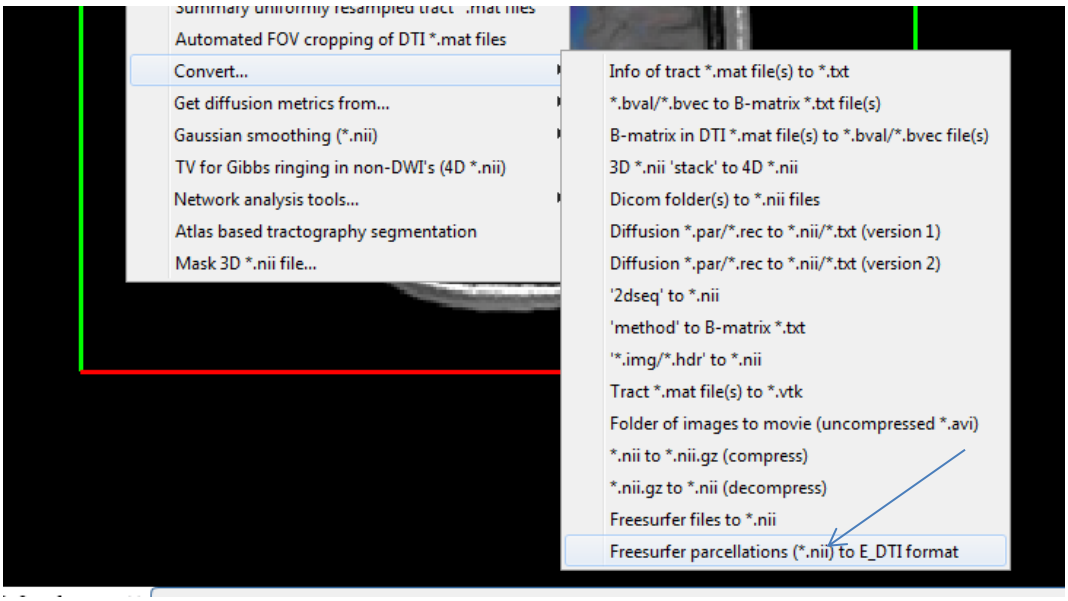


Now, we're going to convert the freesurfer *.nii parcellations to *ExploreDTI* compatible *.nii parcellations (this produces the label *.txt files as well):

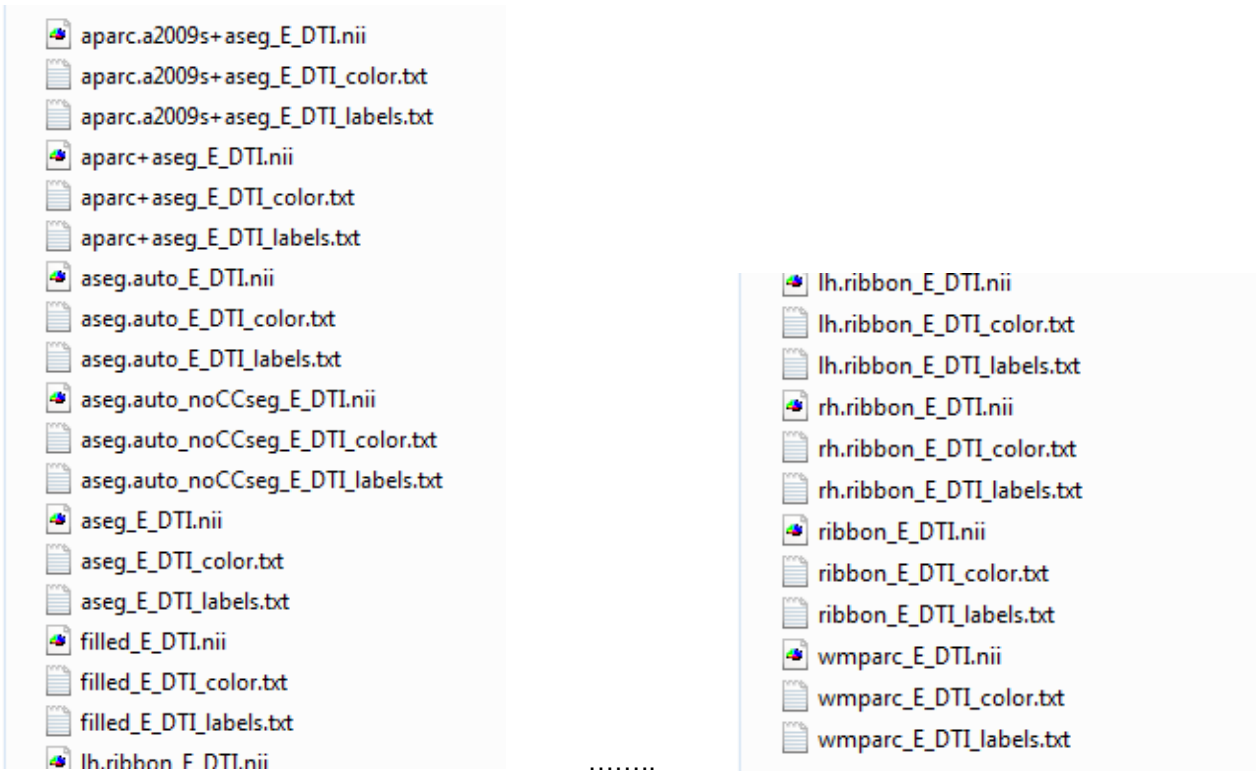
Make a new folder "E_DTI" in the "nii" folder:



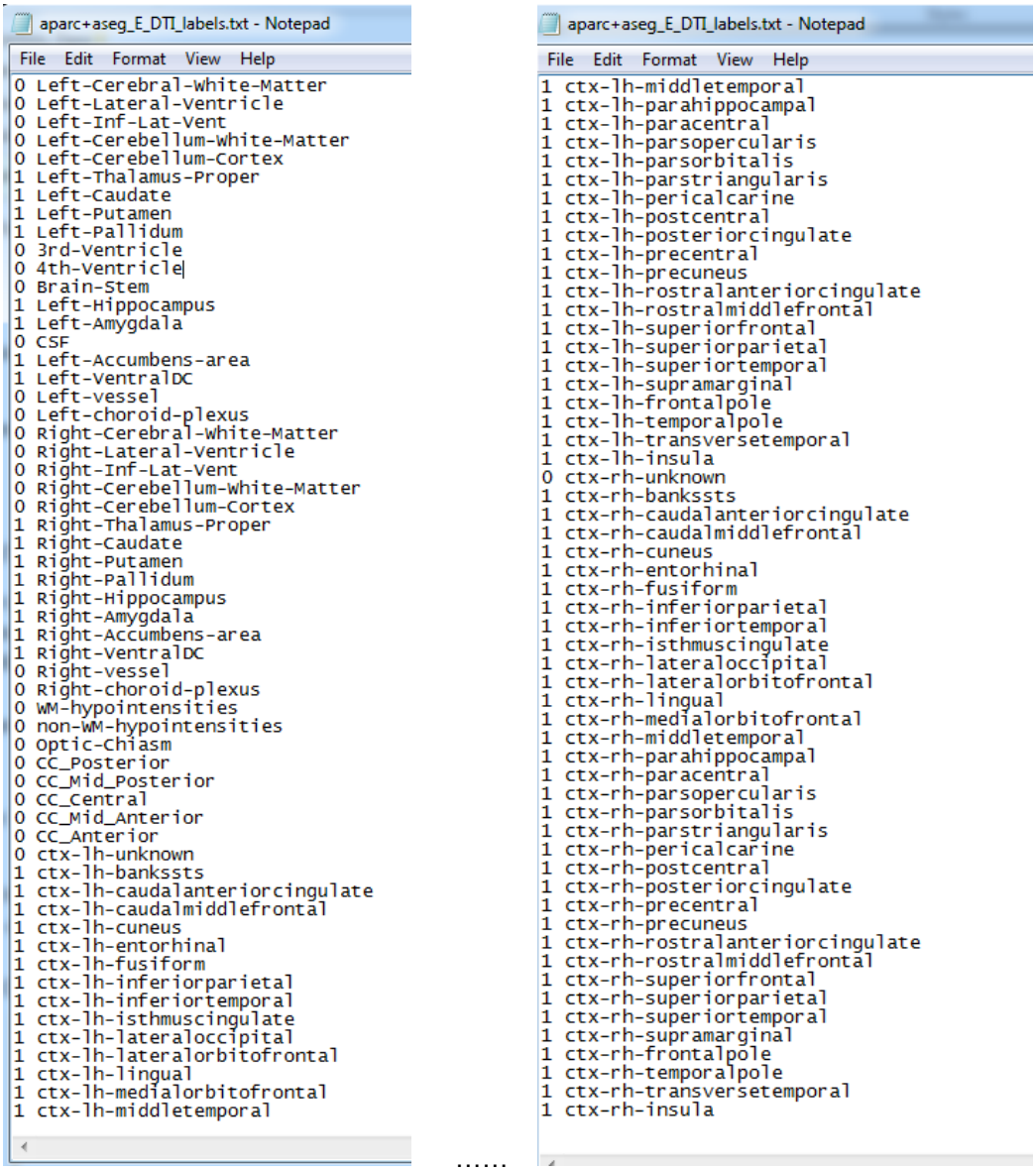
In Plugins, go to:



And select the “nii” folder as input and the “E_DTI” folder as output. Then wait a few seconds…
The output is now something like:



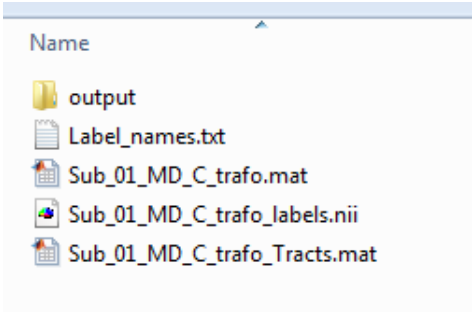
Use for instance the “Desikan-Killiany” atlas (“aparc+aseg_E_DTI”) and only the relevant labels (change “1” values to “0” if you do not want to use them):



So, here, a total of 84 nodes… Just as an example.

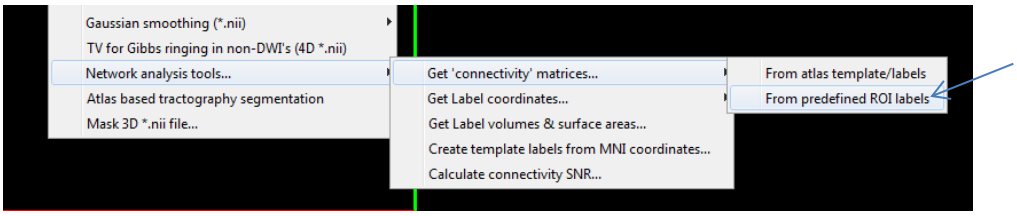
But feel free to use another parcellation scheme!

Now rename the label file “aparc+aseg_E_DTI.nii” to “Sub_01_MD_C_trafo_labels.nii” and put it in the folder with the *.mat DTI file (move non-relevant files to another folder and make a folder “output”):

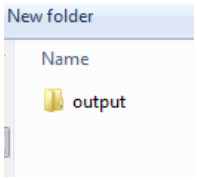


BTW – I just renamed “aparc+aseg_E_DTI_labels.txt” to “Label_names.txt”, but not really necessary (and only one such *.txt file is needed for all subjects when using the network tool below).

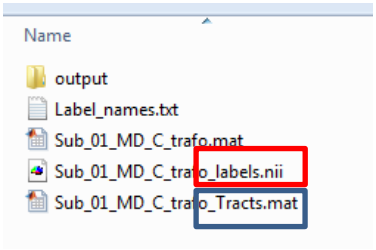
Now – finally ready to use the network tool (make sure you have enough RAM on your PC!):



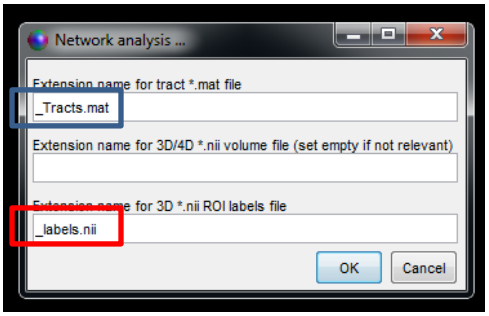
Select the folder with the DTI *.mat files (in this example only 1 subject is in it, but it works for multiple subjects too):



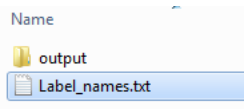
Given this below:



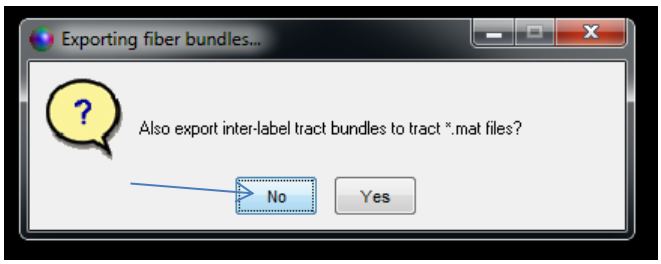
It should be clear that (set the “volume *.nii file extension” to “empty”):



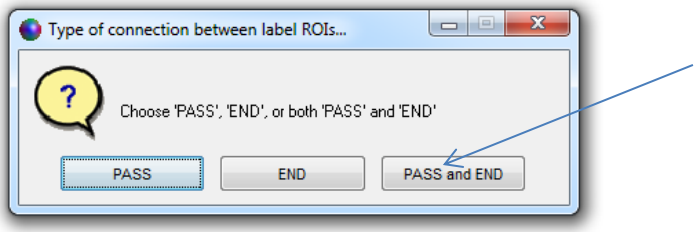
Then, select the text file:



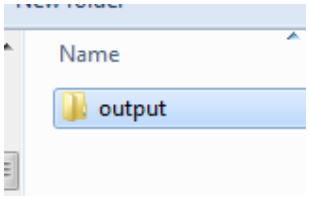
Then… similar as before… I say “No” here for sake of simplicity (see below):



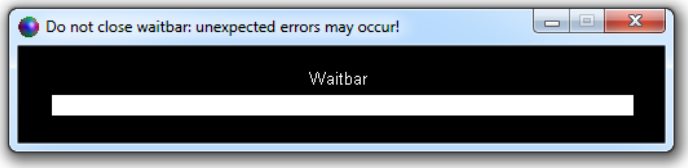
Then, choose the “type of connection” (END/PASS) – see above for explanation… Just select both for now (not too much additional computational cost anyway):



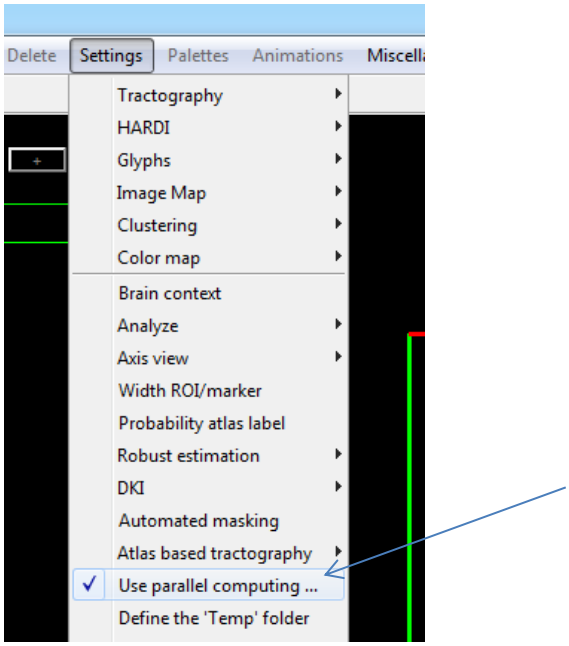
And finally, select the output folder:



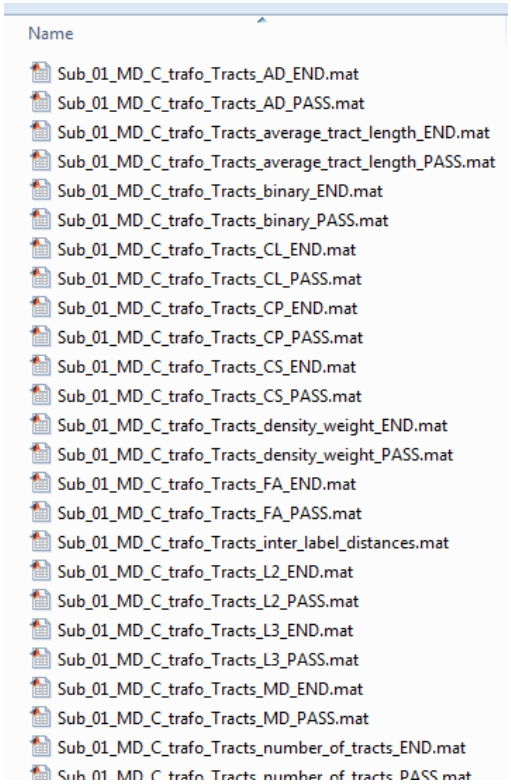
And then wait…



Could be a few hours per subject… depending on PC specs! But it can be parallelized with the setting (again: check RAM!!!):



The output is as before (see p. 42-44):

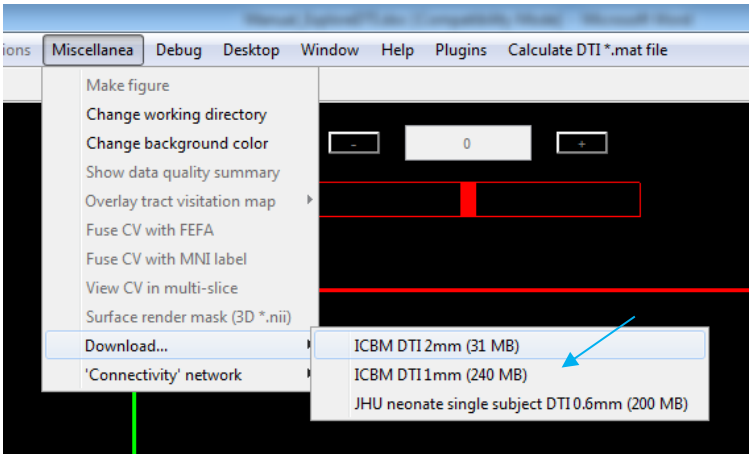


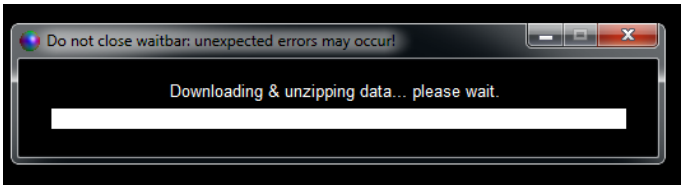
Note: to speed up calculations, you could resample the relevant files to a lower resolution (e.g., [2 2 2] mm instead of [1 1 1] mm results in a speed-up factor of 8!) – before starting the SM/EC/EPI correction step.

Hopefully, you can now manage to get these CMs… So what's next? Well, now you need another software tool that allows you to compute the "graph theory" measures and perform group analyses – no need for me to reinvent the wheel here ;-)! Based on a quick Google search, BCT (http://www.brain-connectivity-toolbox.net) and GAT (http://ncnl.stanford.edu/tools.html) seem to be good candidates – but don't quote me on it… I have not tried these tools myself.

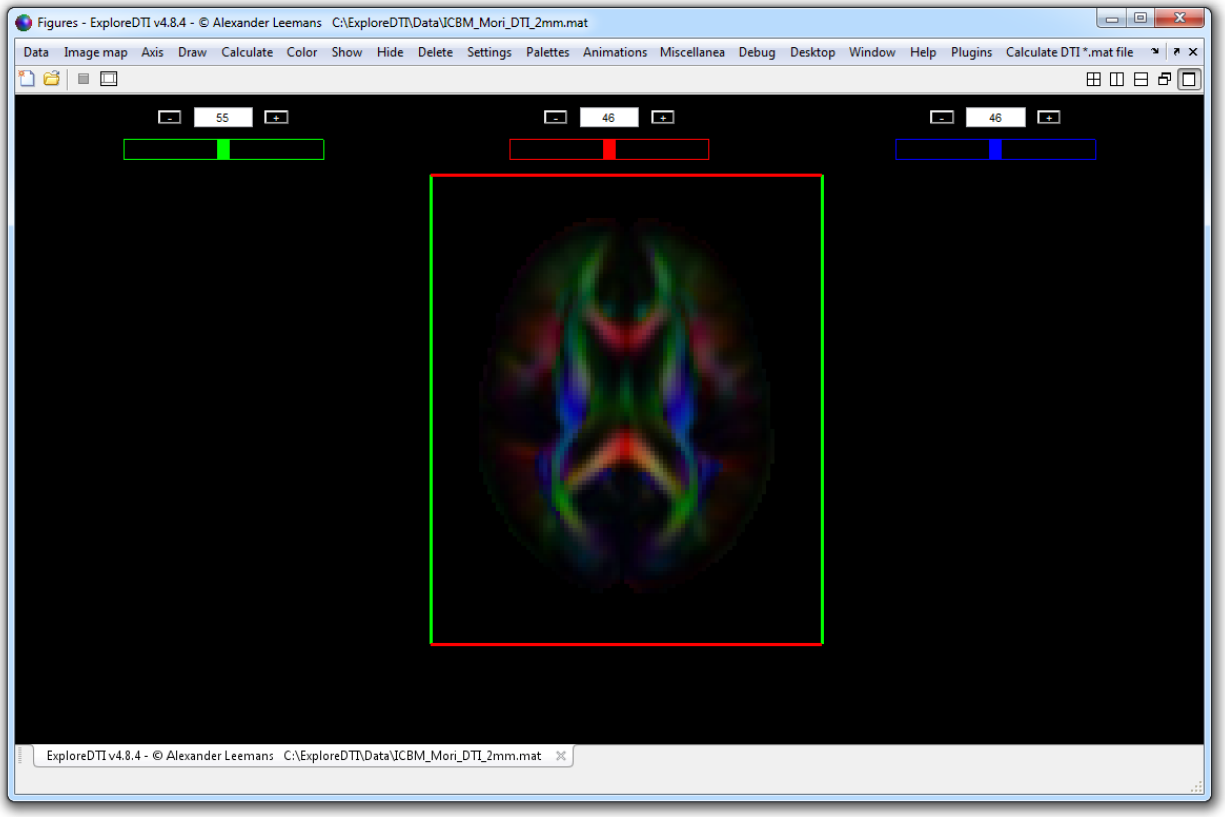
And now… the visuals! Perhaps the most complicated part… and it takes a lot of work, but then again, it may be worth it (think: cover of journal)! There are many options and different ways to create these connectivity visualizations (aka: "connectography"), but for now, I will just provide a "quick" example.

Step 1: download the DTI 2mm template (see below) and wait a few seconds (or minutes):

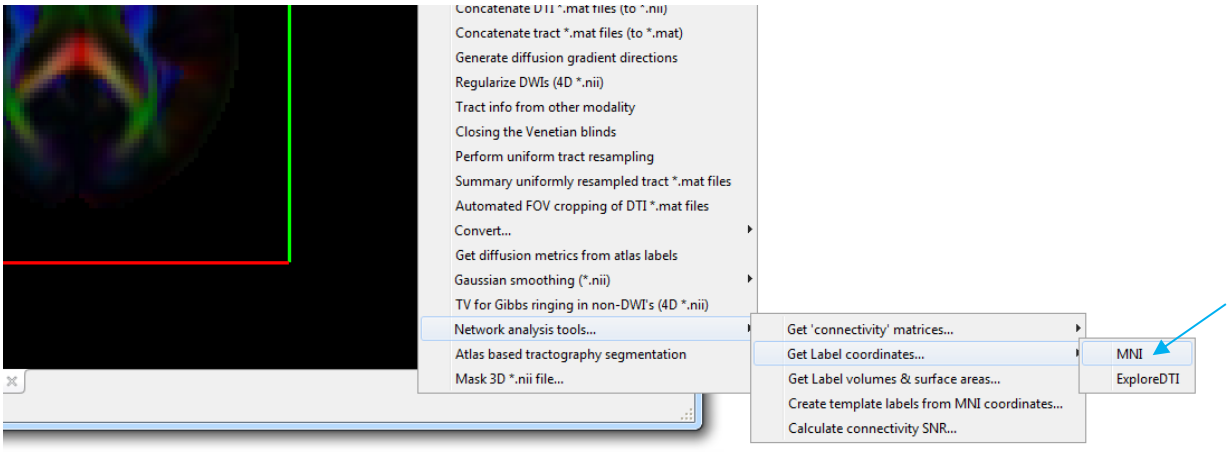




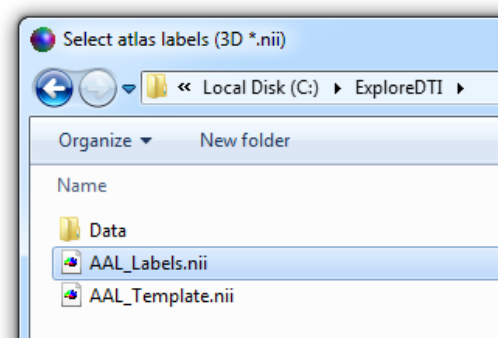
Step 2: Load the resulting “ICBM_Mori_DTI_2mm.mat” file in *ExploreDTI* (see below):



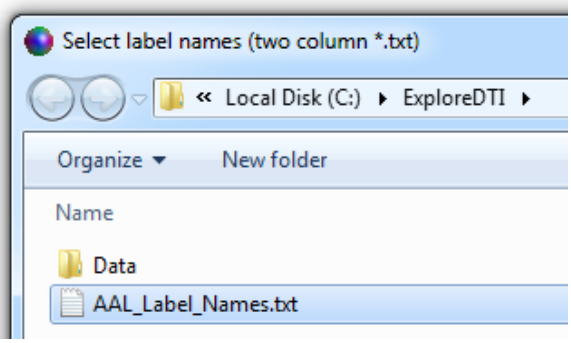
Step 3: “Get label coordinates… → MNI” (see below):



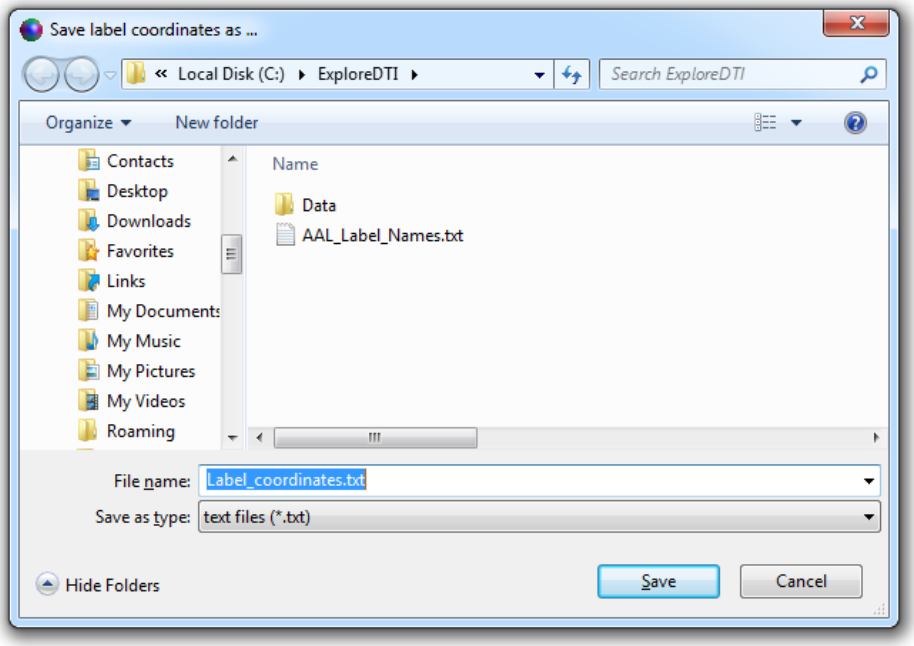
Select the “atlas labels” *.nii and the *.txt “label names” files… Here:



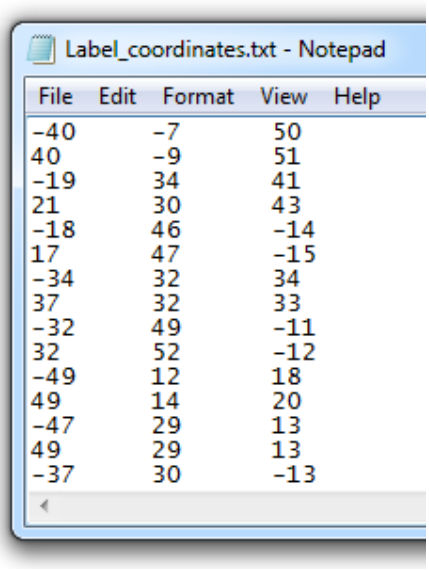
and



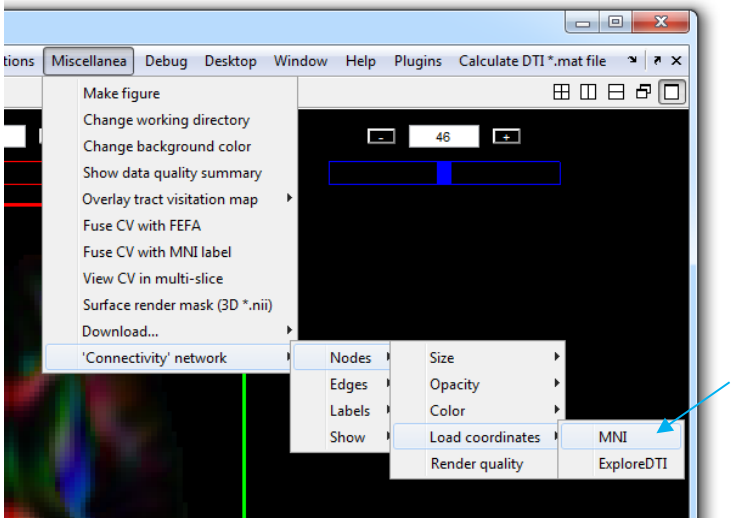
And save the output as:

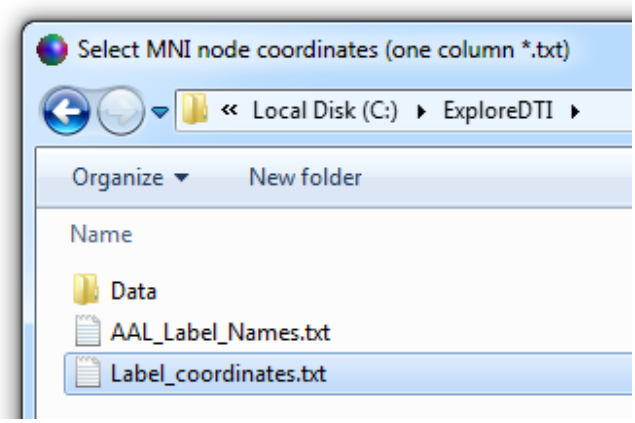


In this step, the MNI coordinates of the center of gravity of the ROIs were computed:

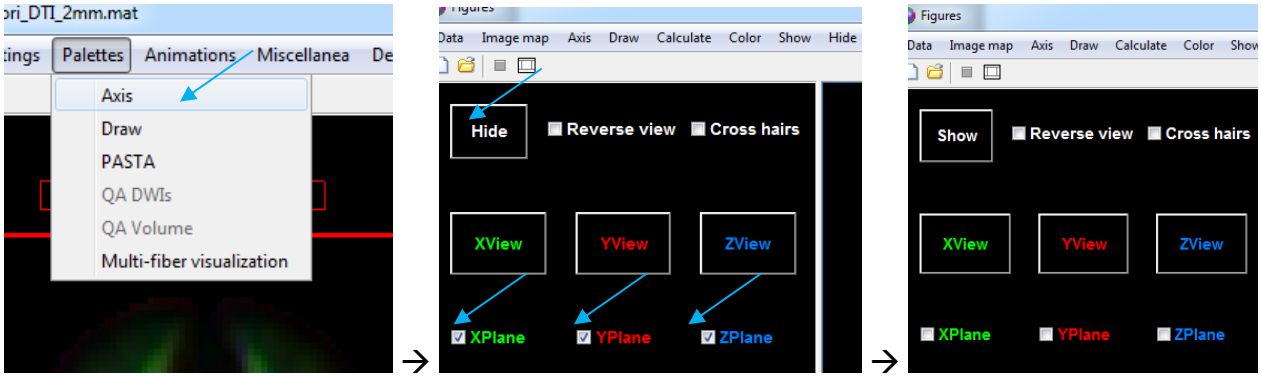


Step 4: Load these “node” coordinates (see below):

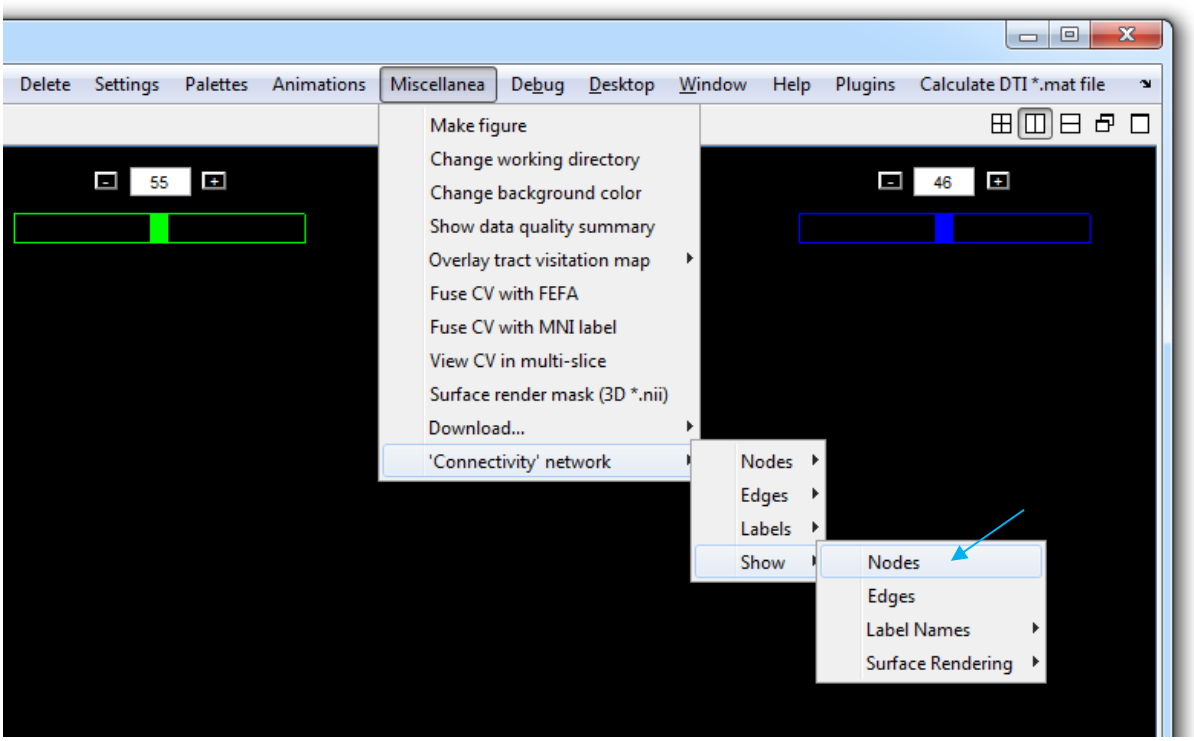




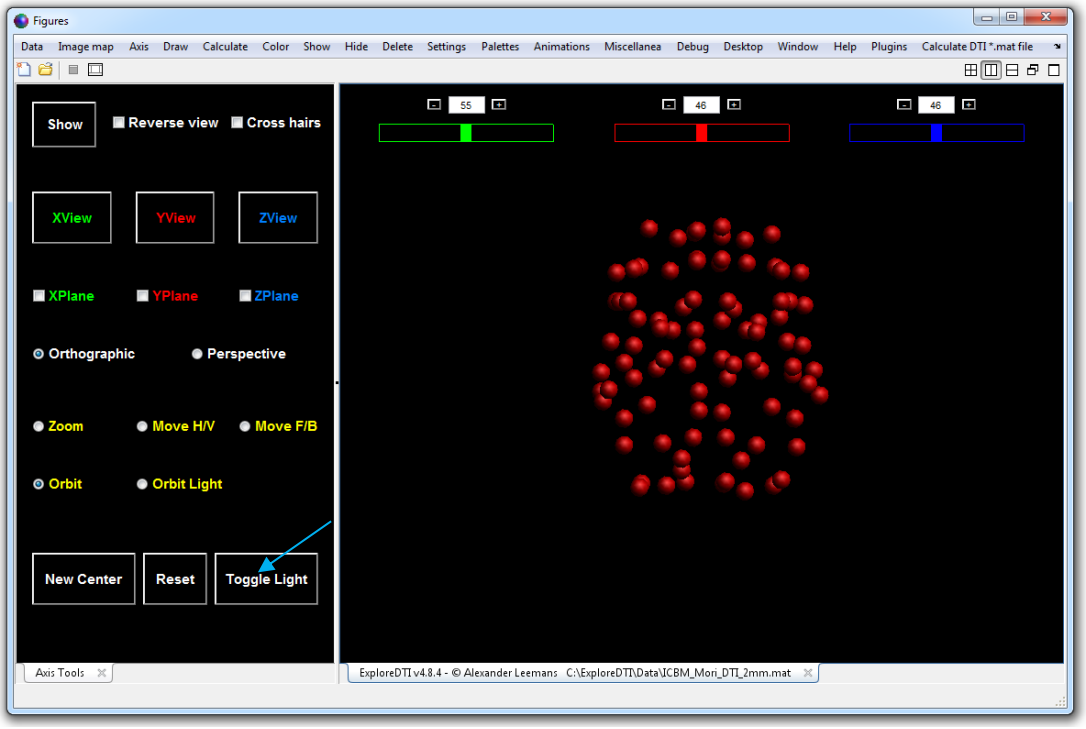
Now, use the “Axis Palette” to uncheck the image planes and “hide the axis”:



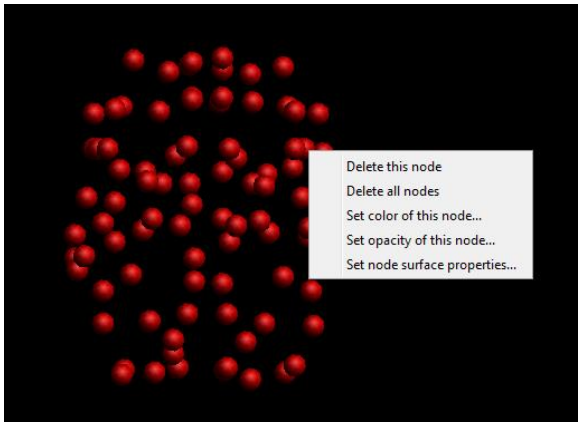
Step 5: now, you will actually see nothing, but then click “Show → Nodes” (see below):



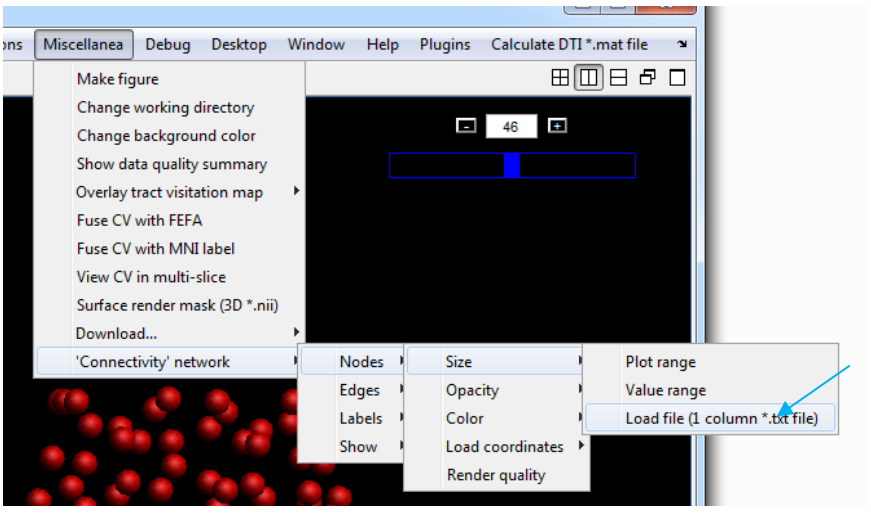
Also click “Toggle light” to get shading effects (see below):



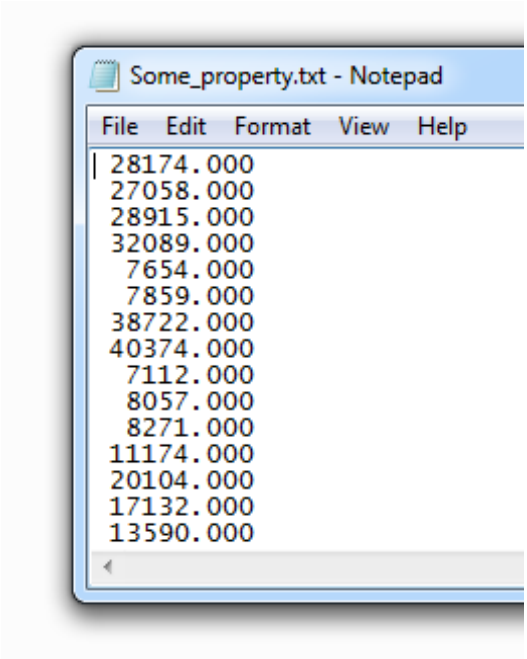
You can right-click these “nodes” to change their properties:



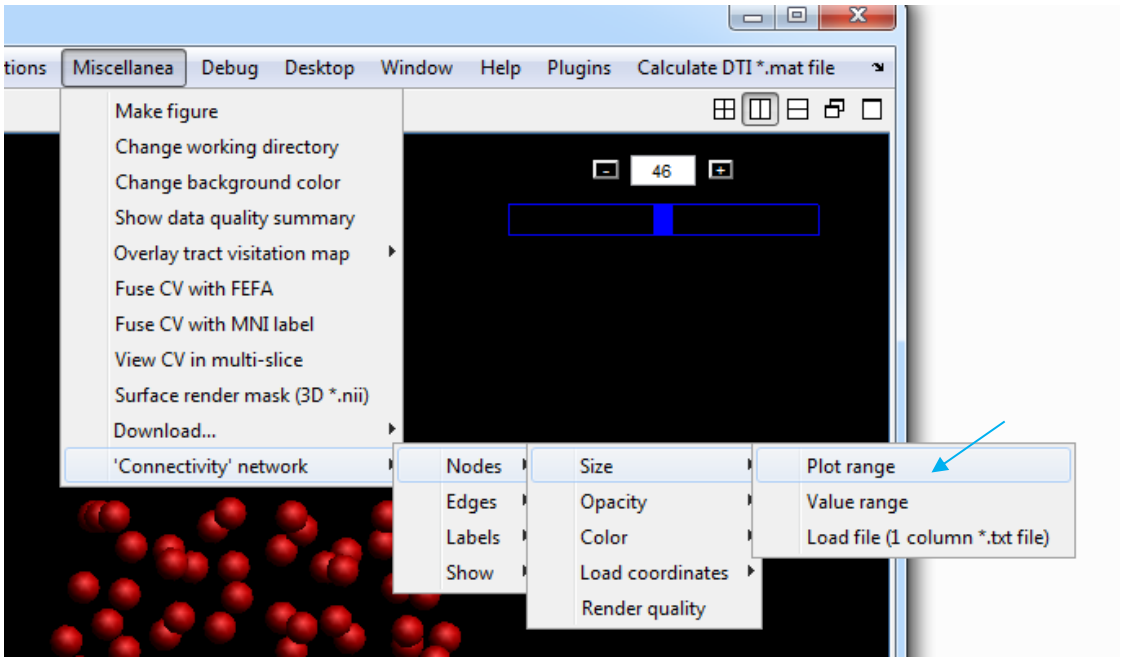
Step 6: Now let’s change the color, size, and opacity of these nodes (starting with “size”):

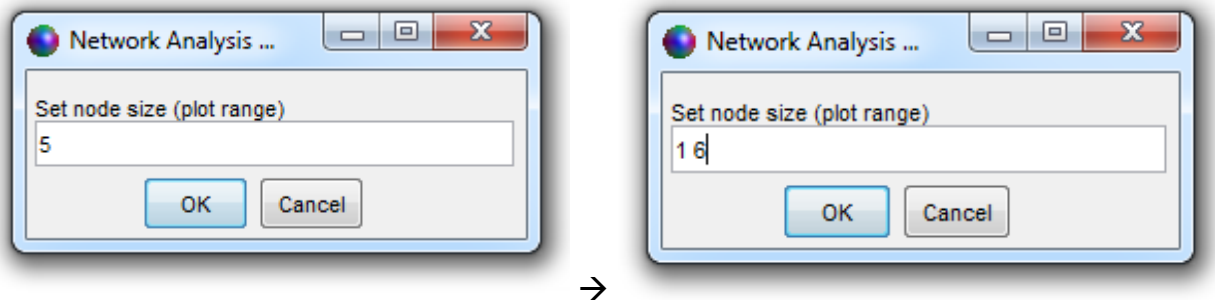


The idea is now to select a text file that contains “numbers” reflecting relative size differences. In this example, I just generated some numbers (N = 90 rows)…

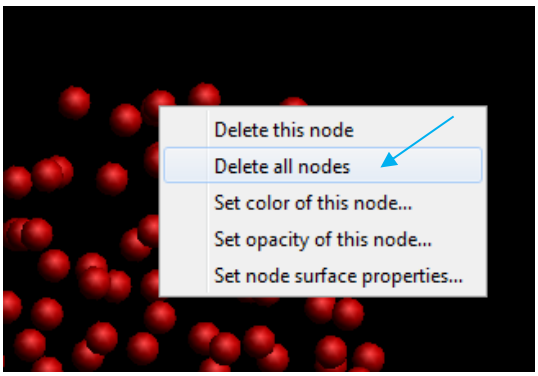


Now go to “Plot range” and adjust “5” to something like [1 6] (see below):

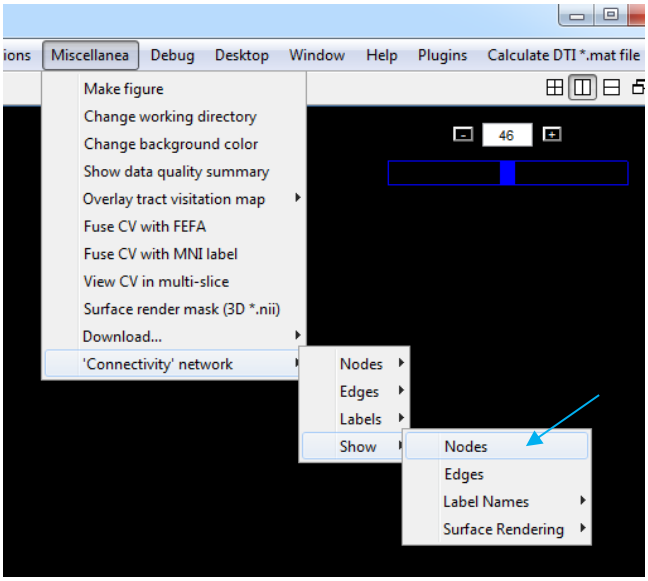


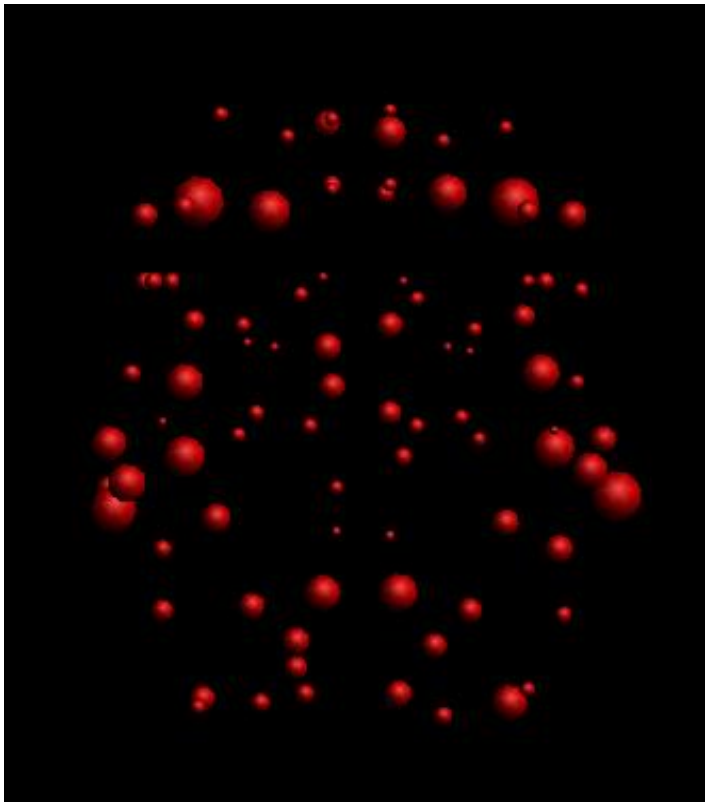


Now “right-click” a node and choose “Delete all nodes” (see below):

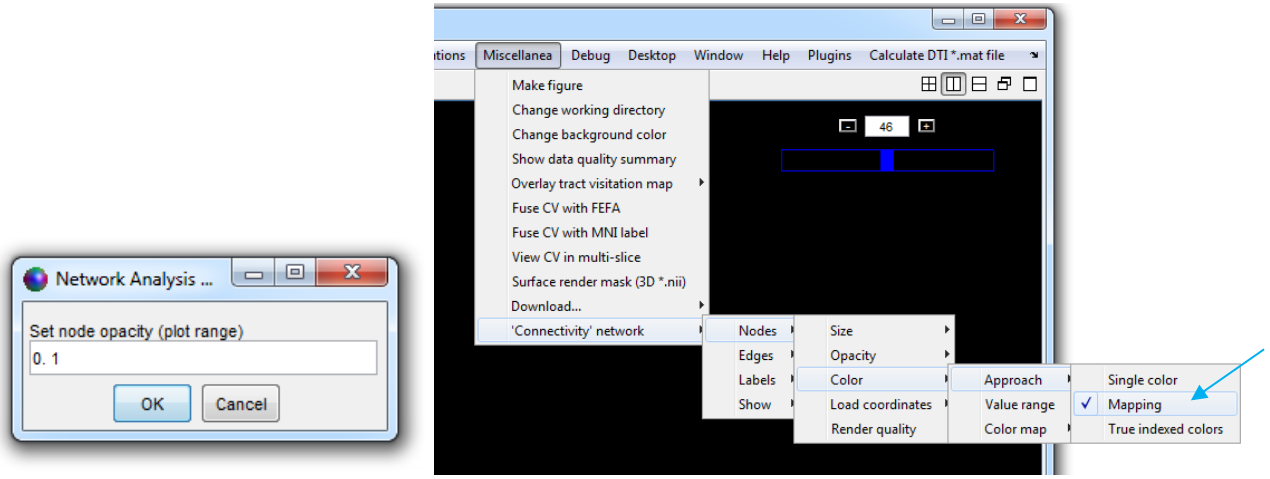


And then, draw the nodes again (see below) – with the new settings for size, you will see:

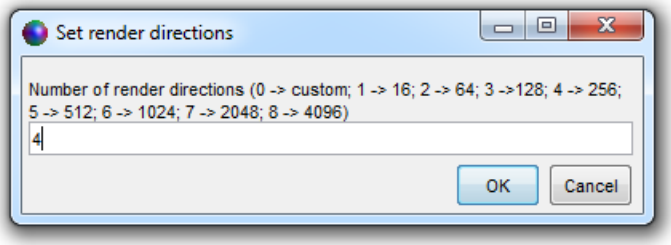




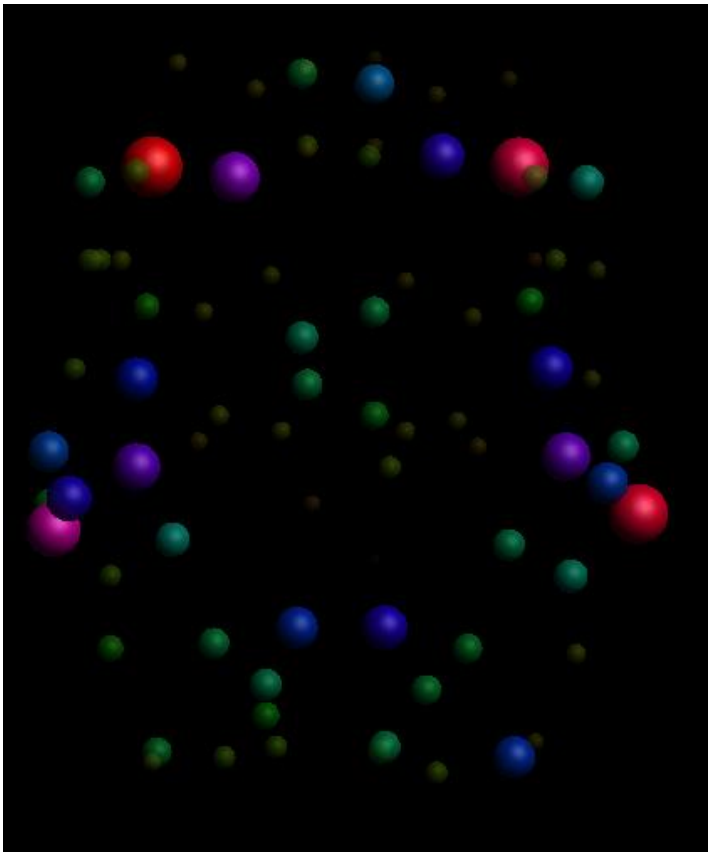
Similarly, you can adjust the node opacity and color… For opacity, use, for instance, a “plot range” of about [0.1 1] (“0” is invisible and “1” is fully opaque) with min-max color “Mapping” (see below):



You can increase the “Nodes” → “Render quality” to get higher-resolution images (at the cost of slower interaction):

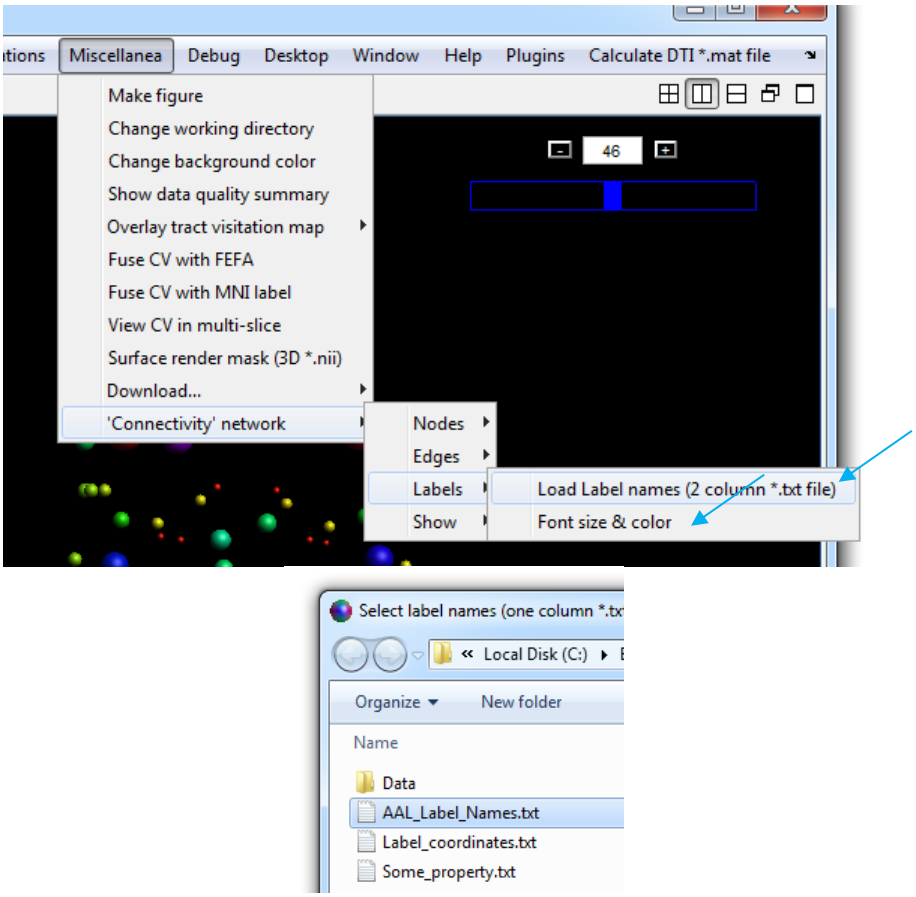


The result (with “hsv” color map):

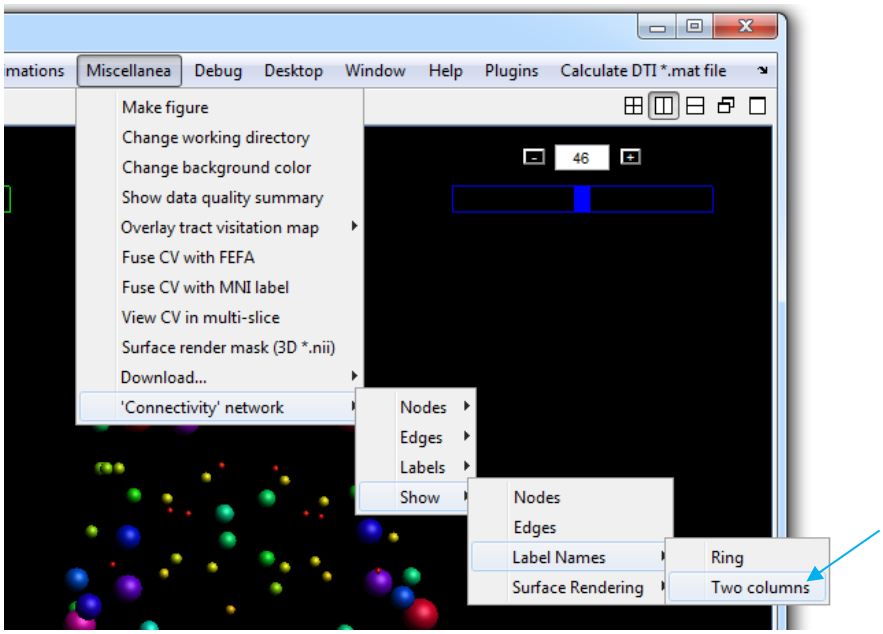


The “Value range” setting is useful for visualizing differences in node (and edge) properties, for instance, between populations. By defining such a fixed value range, the same “dynamic range” for display will be chosen. Otherwise, due to the min-max mapping, you accidently may be comparing apples with oranges. Note that with this setting, you can also perform “value clipping” to get a more “efficient” visualization (e.g., think of outliers). Anyway… moving on!

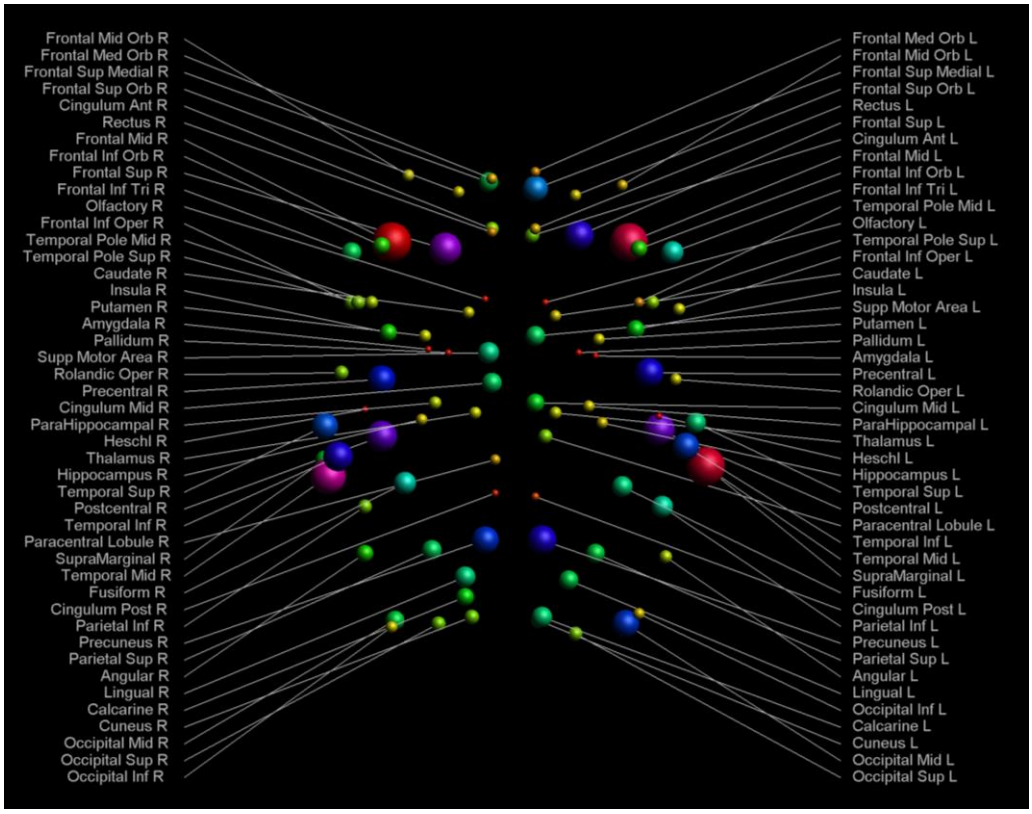
Step 7: Let’s add some label names… this was a fun step to script (the nerdy stuff)! Simply “Load Label names…” and if need be change “Font size & color” (btw, I’ve quickly set the node opacity back to “1” for better visibility of *all* nodes):



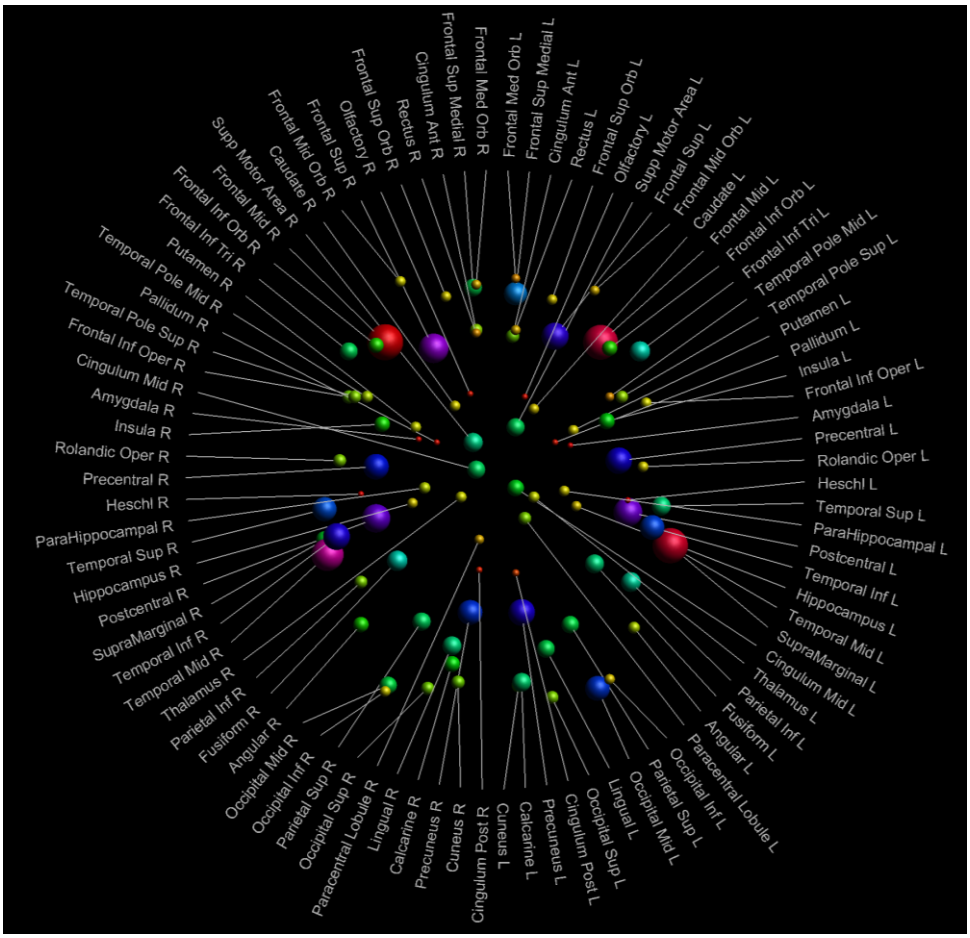
Now, do the following (click “Two columns”):



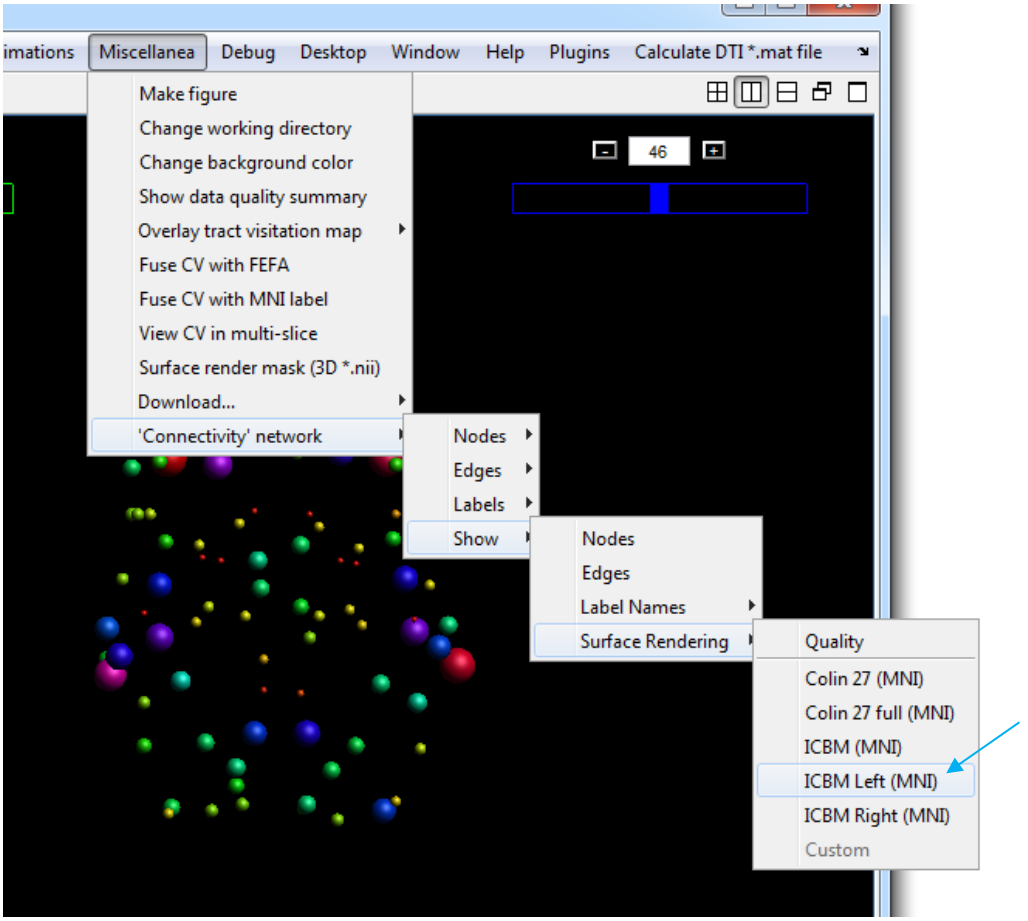
And the result is:



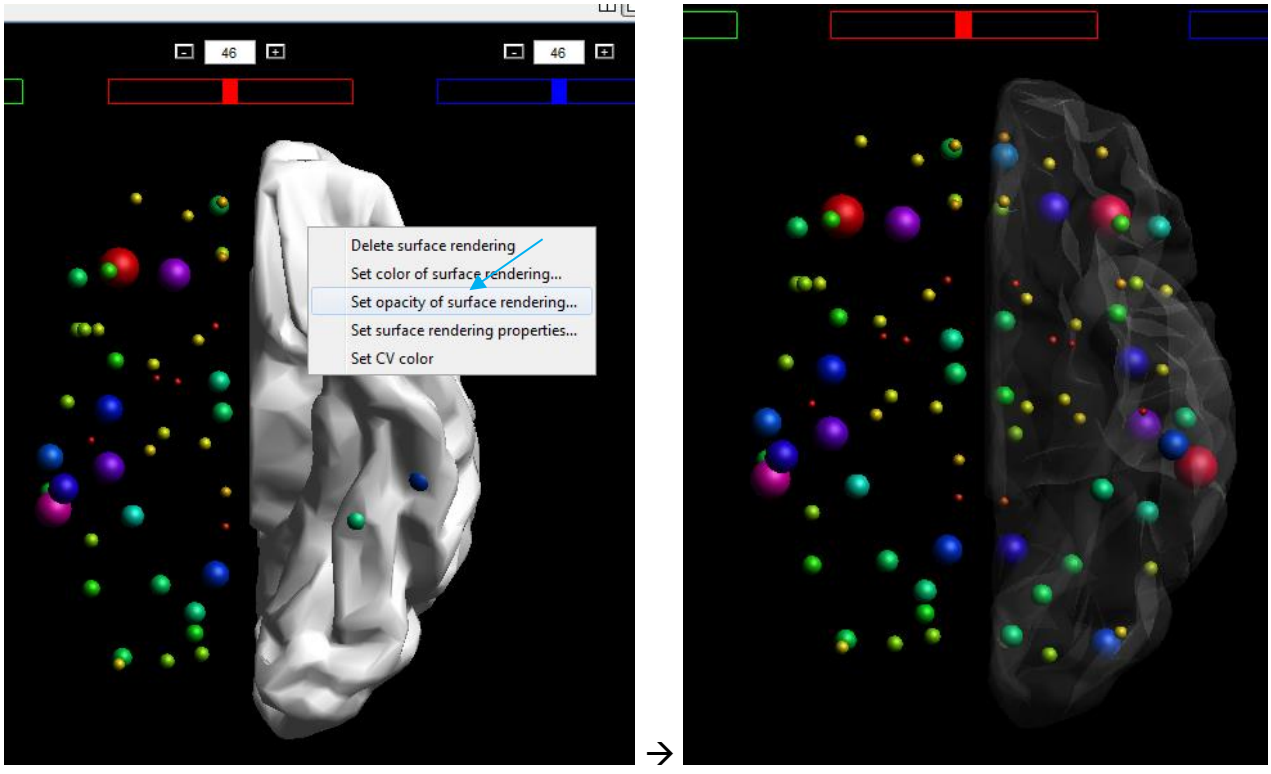
Or if you take the more *artistic* “Ring” option (btw, use the “Miscellanea” → “Make figure” tool to make high-res images):



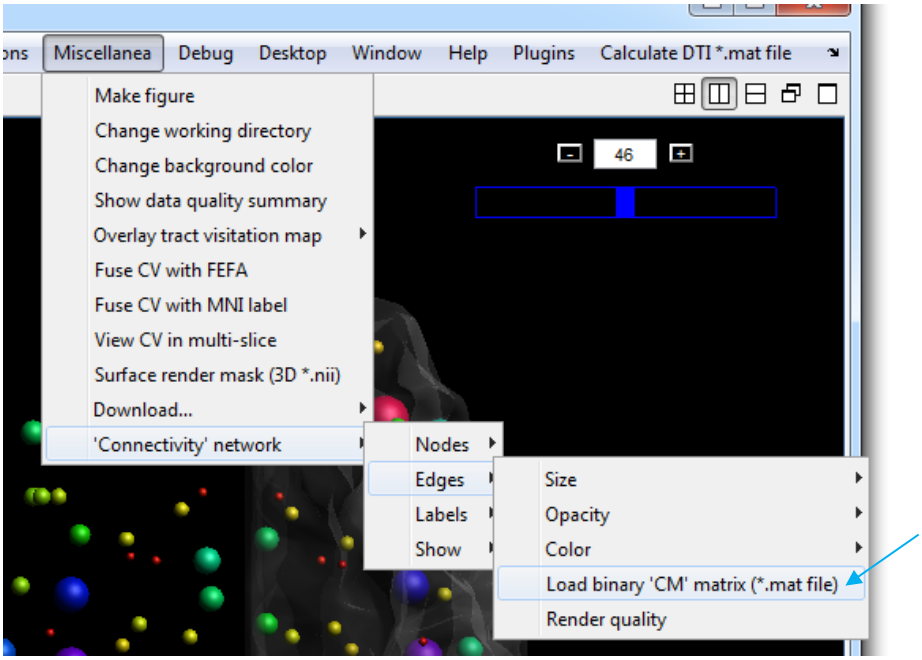
Step 8: Now, let's remove the label names and add some brain context (e.g., left hemisphere – see below):



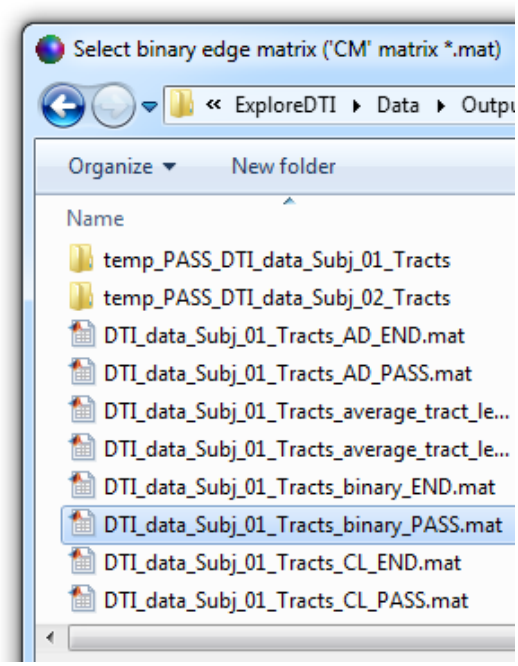
And adjust some settings – e.g. set opacity to 0.1 (right-click, see below):



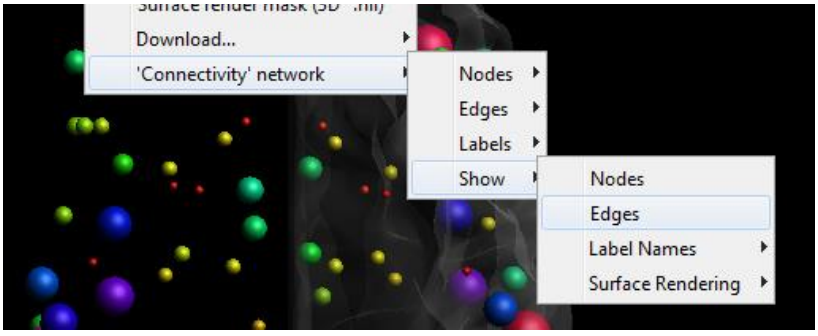
Step 9: finally, let's add some "edges"… First define which edges exist:



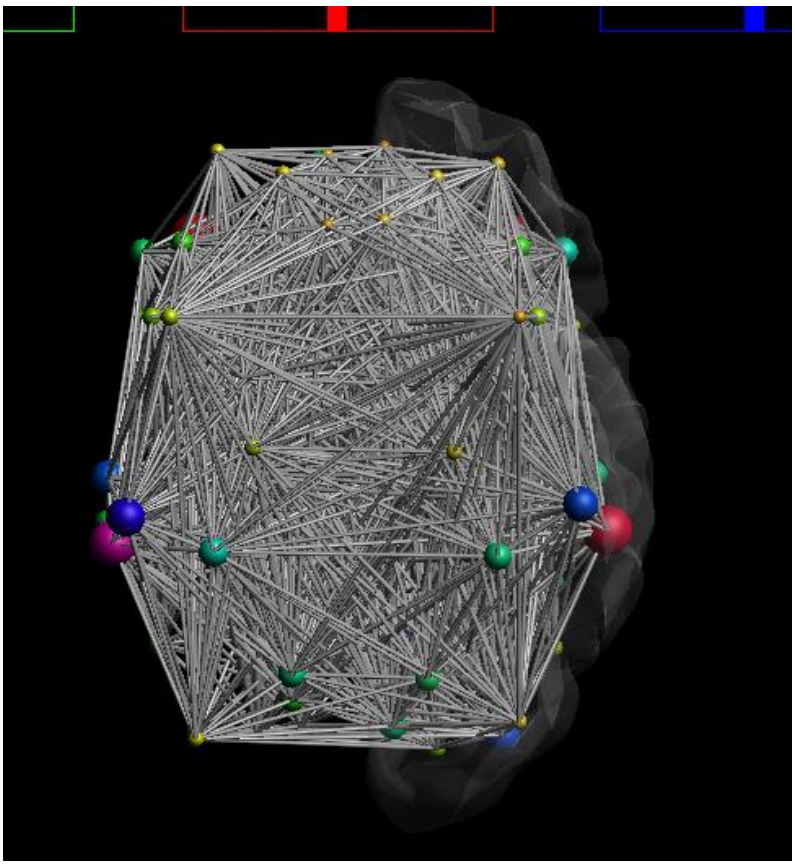
As an example, I'm picking a result from subject 1 here (just for sake of simplicity):



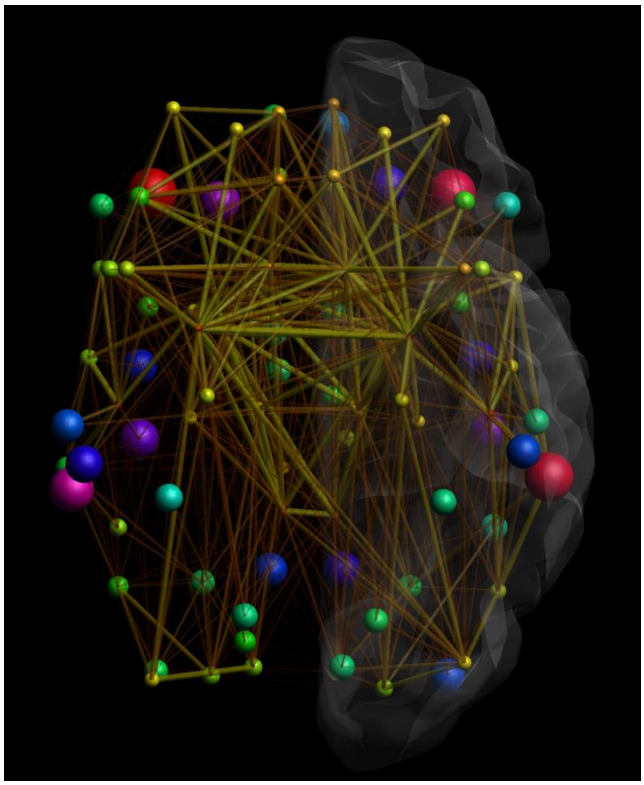
Now, you *could* already show the edges (see below):



But likely, it will not be very informative (with default settings – see below):



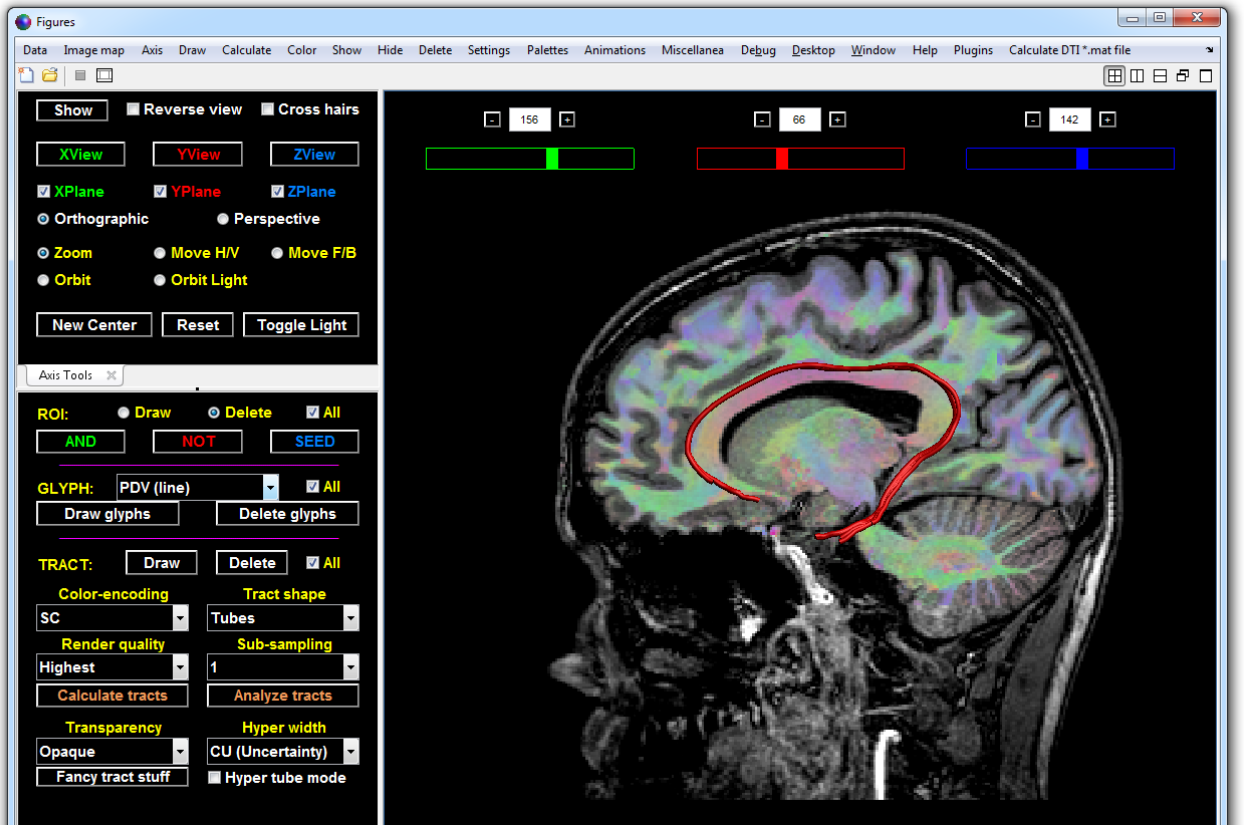
Right-click to delete all the edges and adjust settings of color/size/opacity – in a similar way as with the nodes to get something like this:



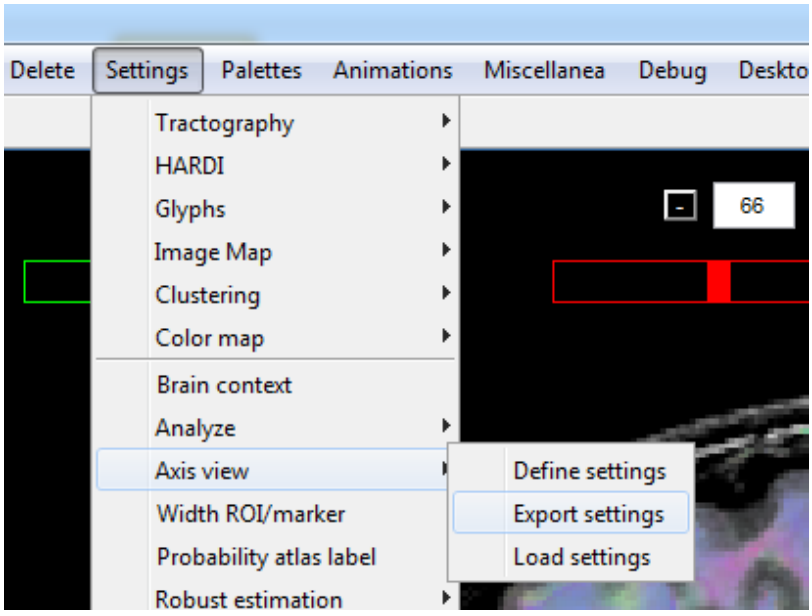
That's all for now ;-)!

15. Making animations

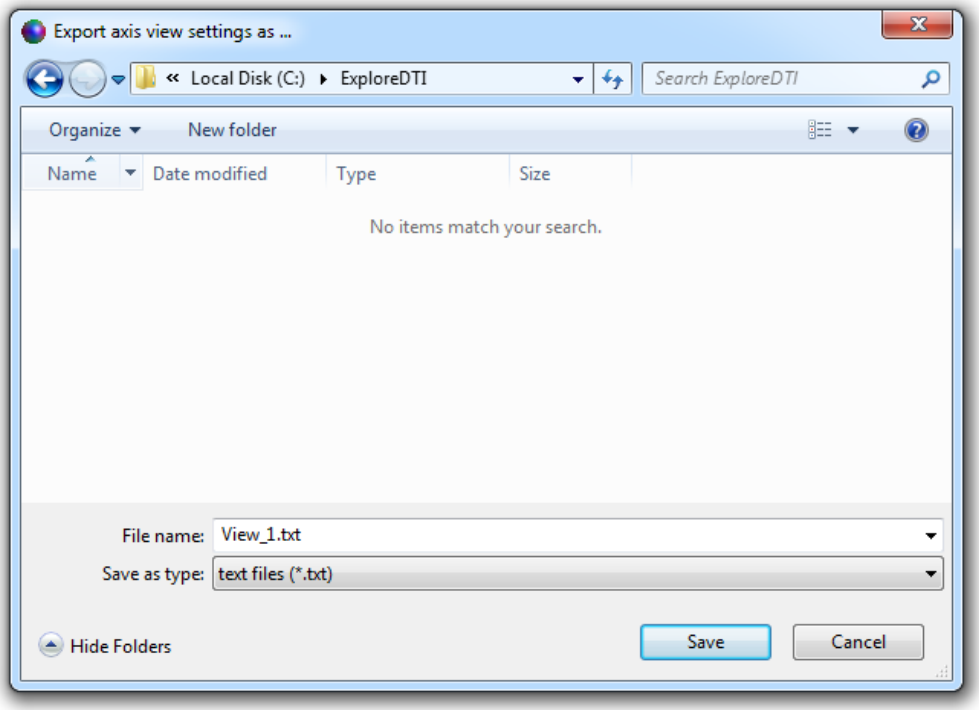
The fun stuff ;-)! First, you need to define a set of “axis views”, which will form the basis of the “path” of the animation. Let’s start off with the first “axis view”, which will then be the start of the animation (see below).



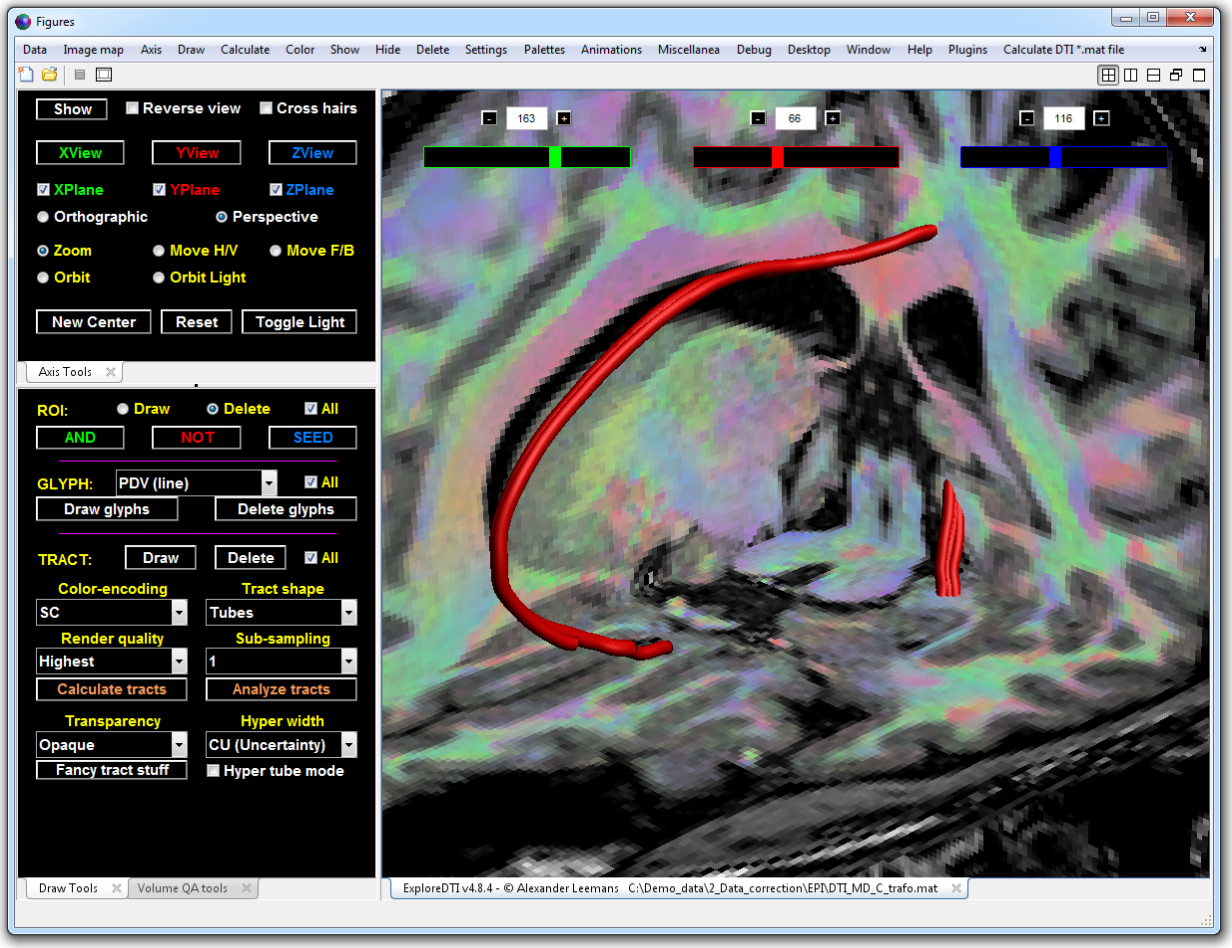
Do the following: “Settings → Axis view → Export settings”.



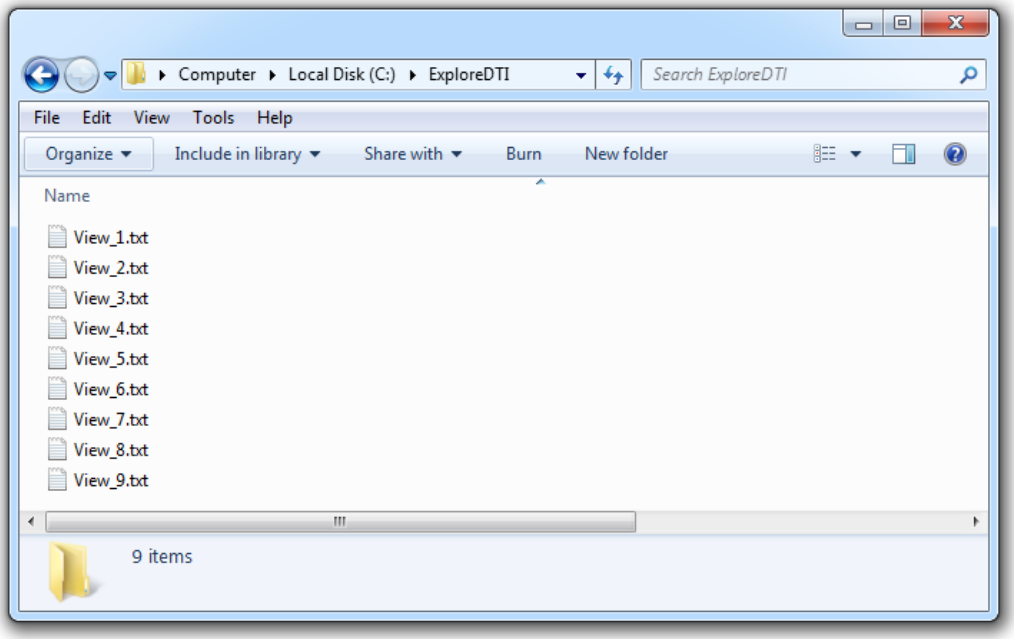
Now save the *.txt file in an empty folder. Here, I’ve named it “View_1.txt” and I’ve put it in the folder “C:\ExploreDTI” (see below).



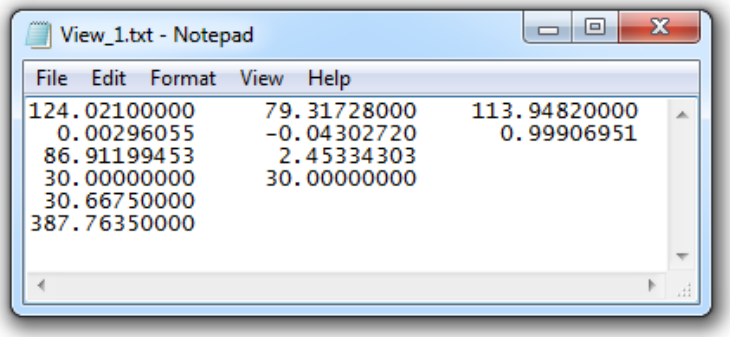
Now, we need to define the next “axis view”. So “manipulate” (e.g., zoom in, move, orbit, etc.) the figure (i.e., the camera axis settings) to another “axis view”. I rotated and zoomed in to the “axis view” shown below:



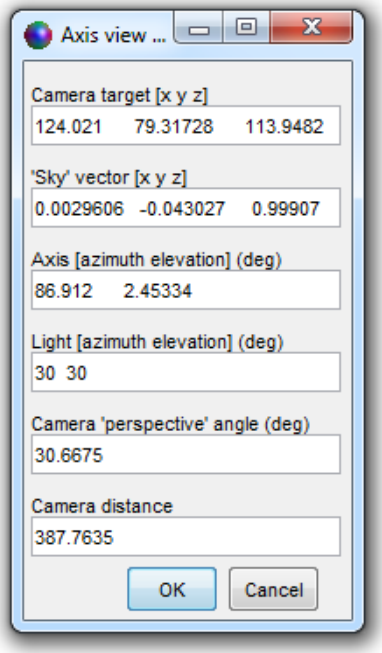
Save this axis view (just like before: “Settings → Axis view → Export settings”), but now name it “View_2.txt”. The numbering is important, as it will determine the order of the “axis views” in the animation. You can repeat these steps as many times as you want. As shown below, I saved nine “axis views”.



In summary, these text files contain “camera settings” that form the “key frames” in the animation. Here’s what such a file looks like:

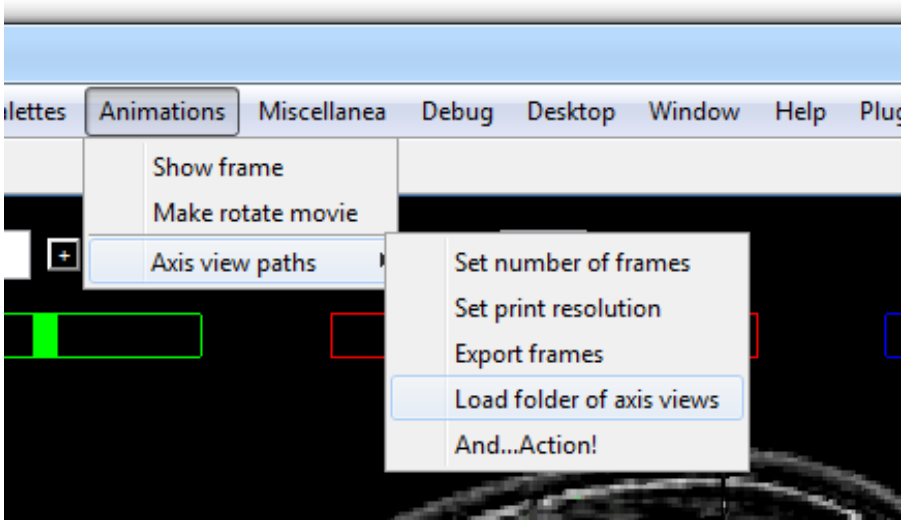


These numbers correspond with the axis properties as shown in “Settings → Axis view → Define settings” (see below). Btw, if you want, you can edit these values… think outside the box ;-)!

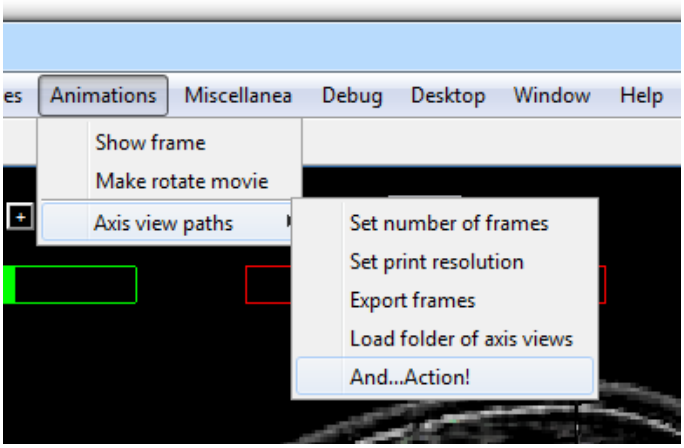


If you want your animation to loop in a natural “cyclic” way, then copy the file “View_1.txt” to “View_9.txt” (that’s what I did here… so the last frame of the animation is equal to the first frame).

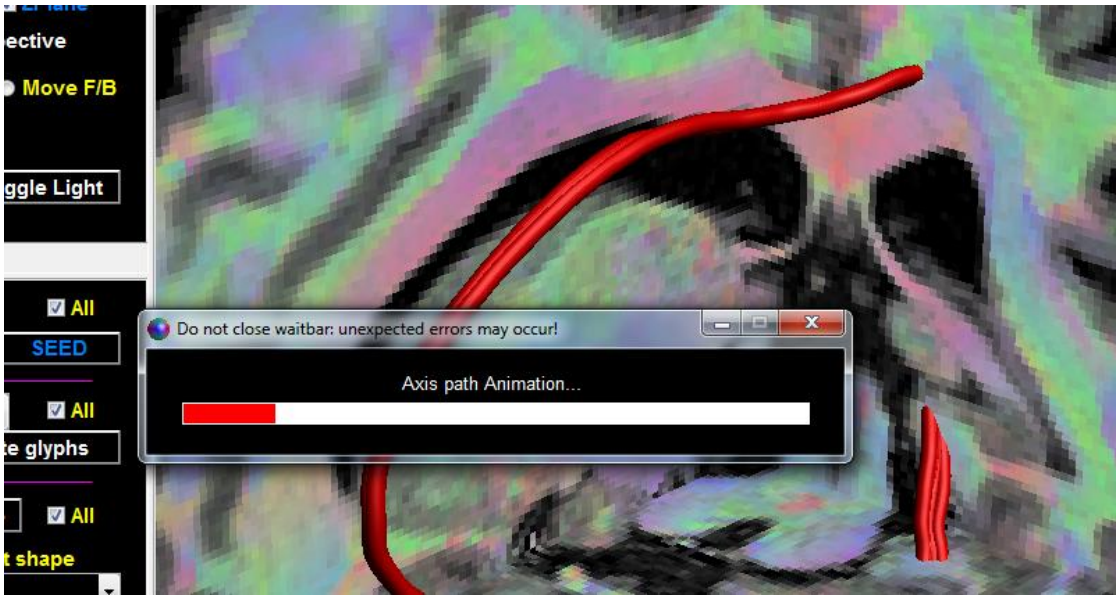
The next step is to actually have a look at the animation. To do this, you need to “Load the folder of axis views” in the “Animations → Axis view paths” menu item (see below).



Here, I have selected this folder: “C:\ExploreDTI”. Then, just click “And… Action!” in the “Animations → Axis view paths” menu item (see below):

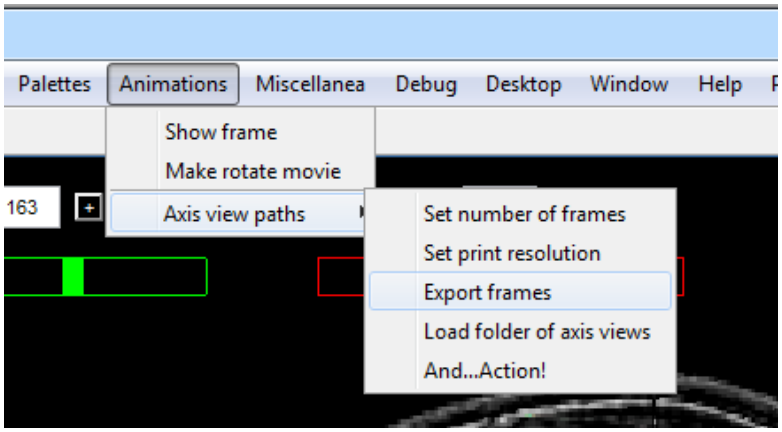


Hopefully, you’ll see something happening as shown below:

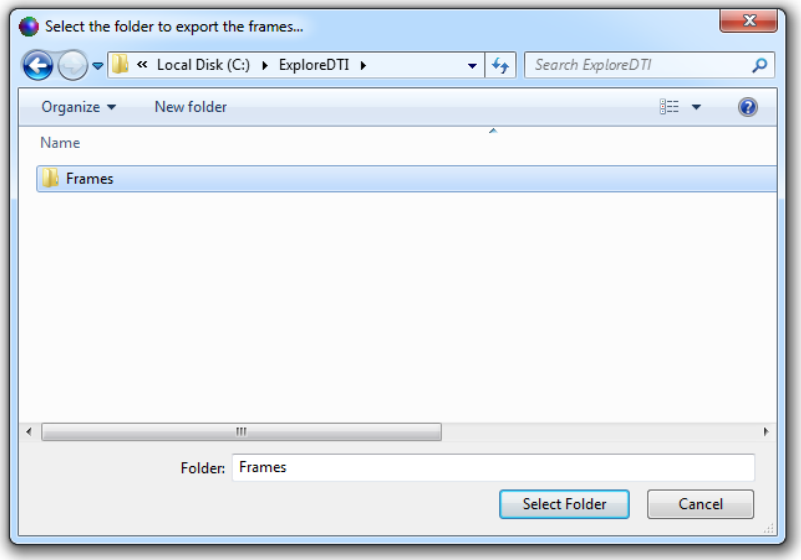


At this stage, you’re simply “checking” your animation… If you are happy with the “path” of animation, then you can adjust the “number of frames” (the temporal resolution). Also the spatial resolution can be adjusted

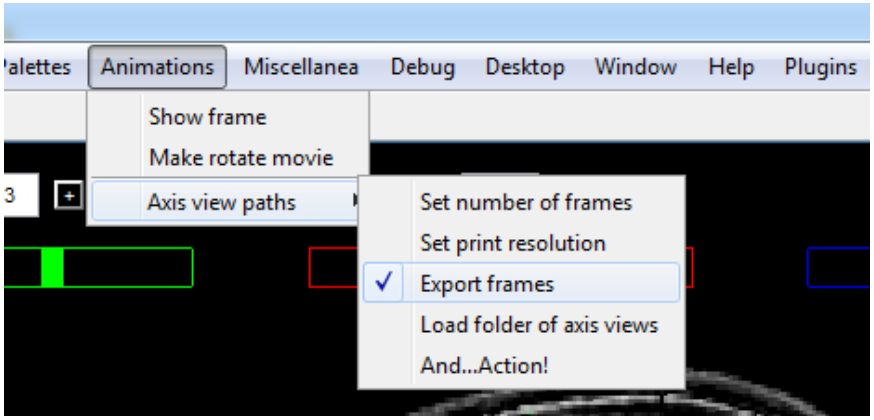
(think “full HD” or even “4K” resolution of the frames). My suggestion, start with low values of “number of frames” (e.g., 100) and “print resolution” (e.g., 100) for sake of efficiency and then increase these values to whatever you want. The final step is to create all the frames (the “computational expensive” part). So define an output folder (click “Export frames”):



I created a new subfolder “Frames” (see below) and selected that one.

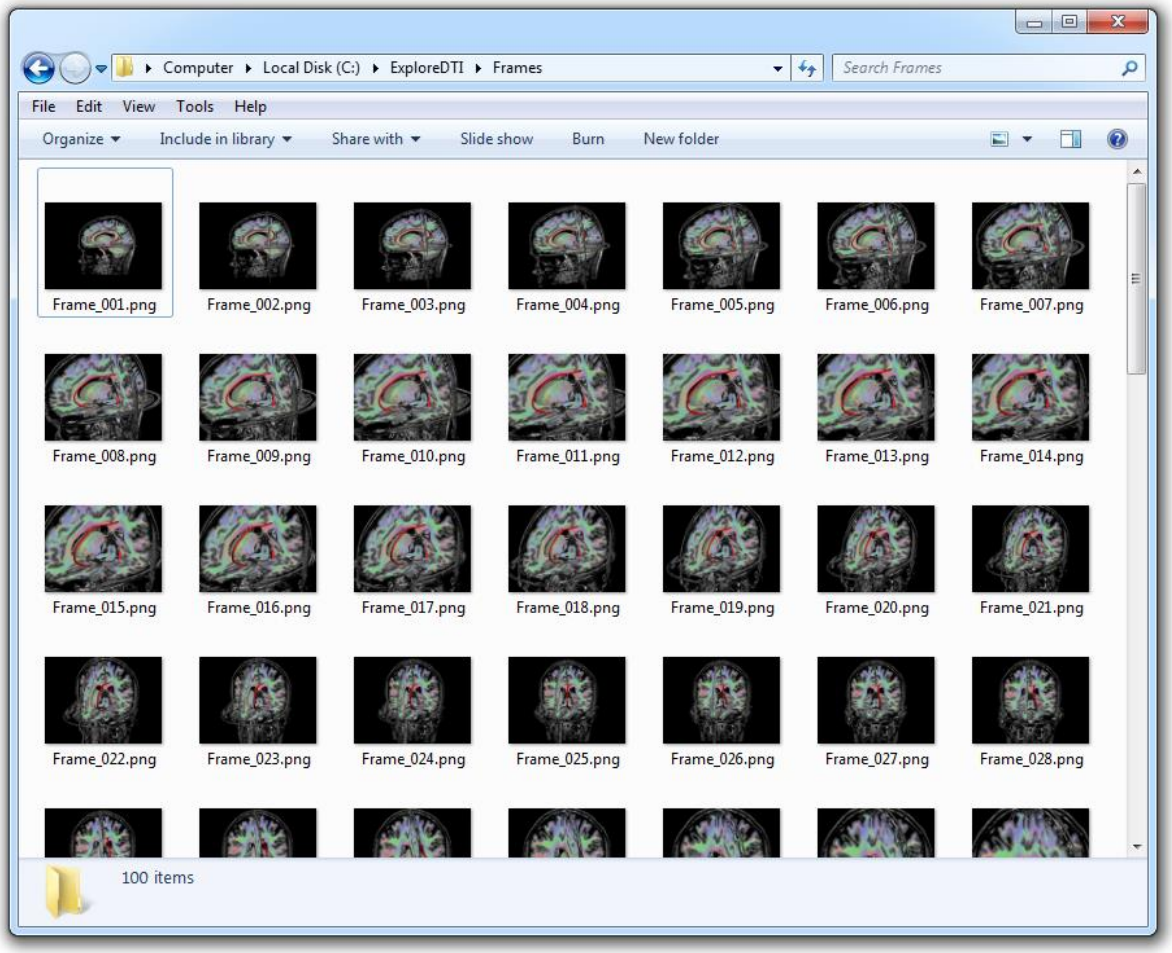


Notice that “Export frames” is now “checked”…



If you click “And…Action!” now (while “Export frames” is “checked”), it will export the frames of your animation to that folder. Note that printing a single frame may take a few seconds (depending on the object rendering and the “print resolution”).

If you made the animation with, for instance, 100 frames, the folder will contain 100 images (“png” format – see below).



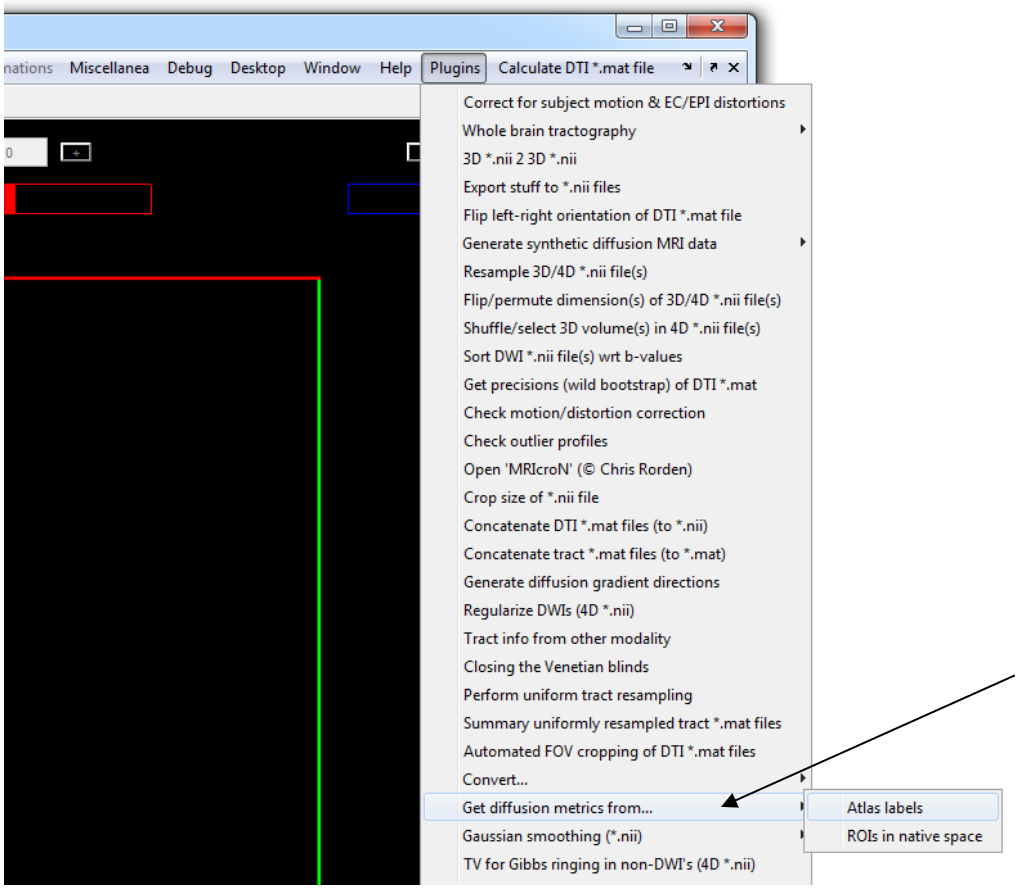
The last step is now to create a movie from all these “frames”. There are many (free) tools out there to do this… Here’s one: “VirtualDub” (http://www.virtualdub.org).

As an example, I ran the animation with 1080 frames and 240 dpi resolution. I used 27 frames per second and cropped the size to 1920 by 1080 (HD) resolution…

The result can be seen here: http://youtu.be/X4HSHE4S6ng

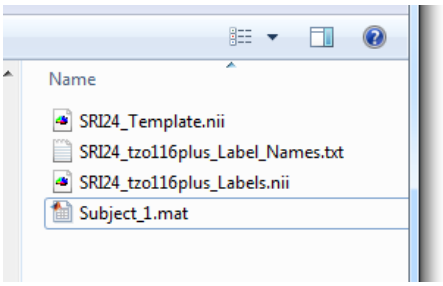
16. Automated / atlas based ROI analysis

There are two ways to do this:

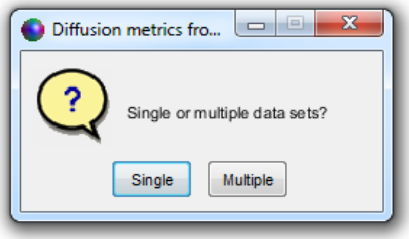


The top one ('Atlas labels') has been used, for instance, in "Kersbergen KJ et al, NeuroImage, 2014" and uses an atlas template, its labels (i.e. the ROIs), the label names, and the DTI *.mat file(s) as input. In short, by warping the atlas template (and transforming the associated labels) to each individual data set, diffusion metrics are calculated in each ROI. Here's how it works:

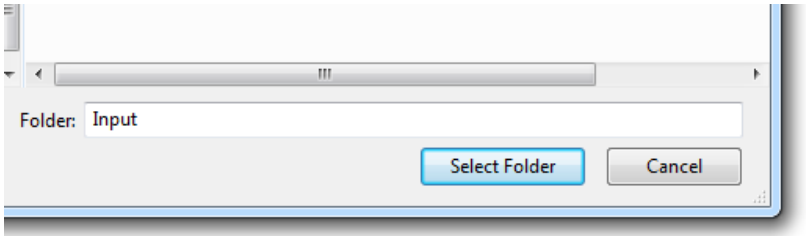
Assume you have the following input data set ("Subject_1.mat" is the DTI file; the other three files are the template files… you can find them in folder ~\ExploreDTI\Source\Templates):



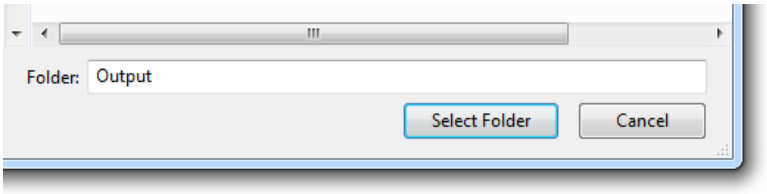
Then choose what applies for your data:



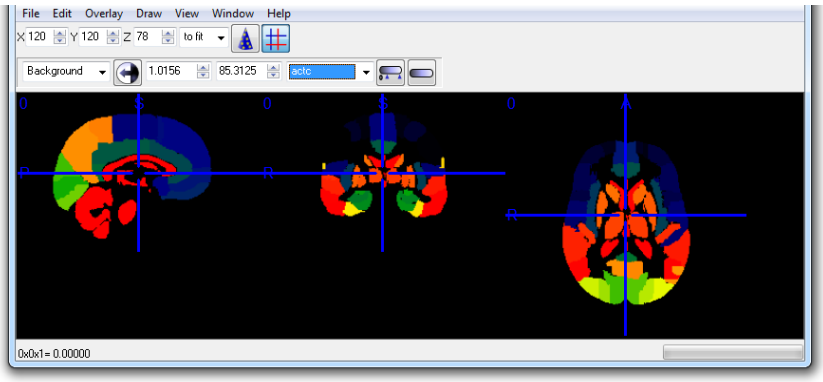
Here, I'll choose "Multiple"… I select my "Input" folder which has the DTI files in it:

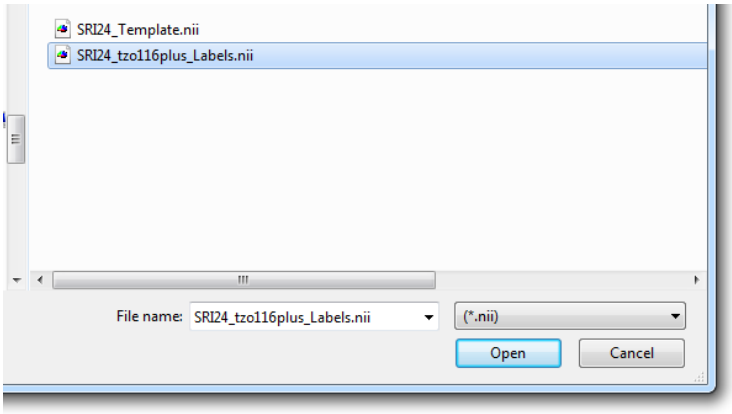


Then, select the output folder:

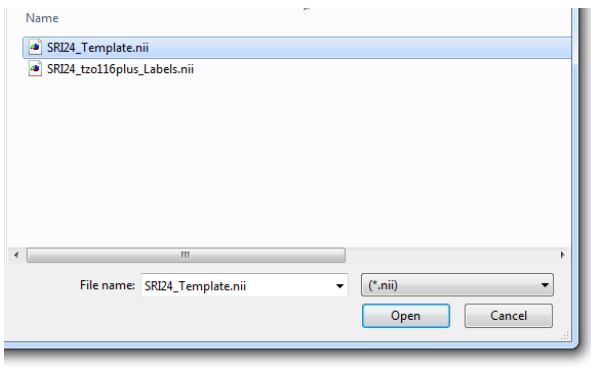


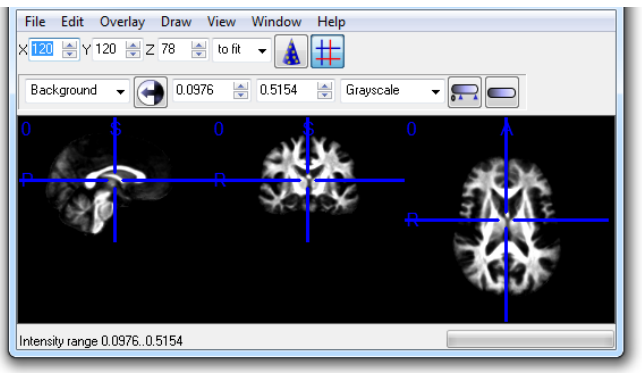
Then, select the atlas labels (screenshot below included simply for illustration purpose):



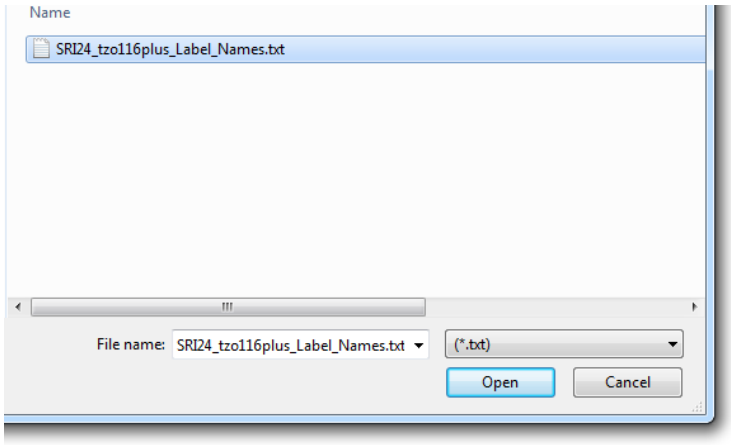


Then, select the atlas template:

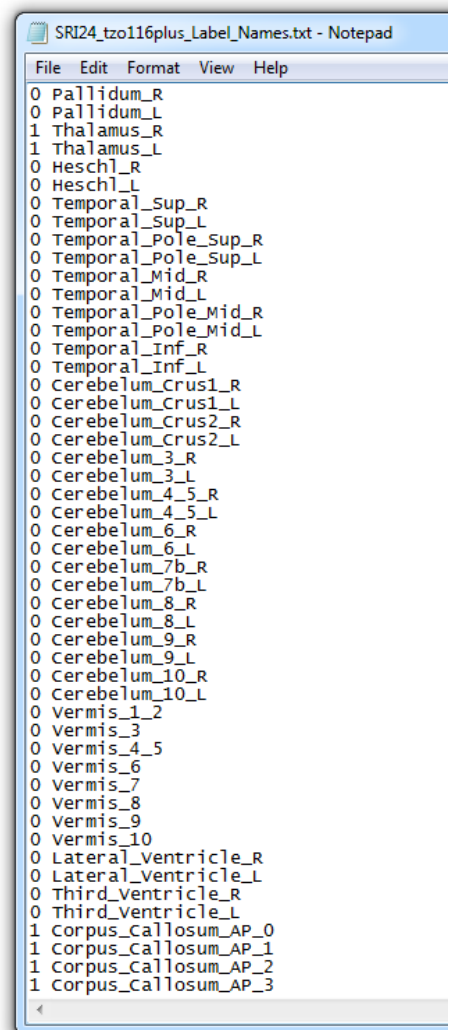




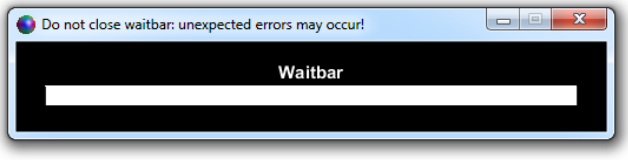
And finally, the label names:



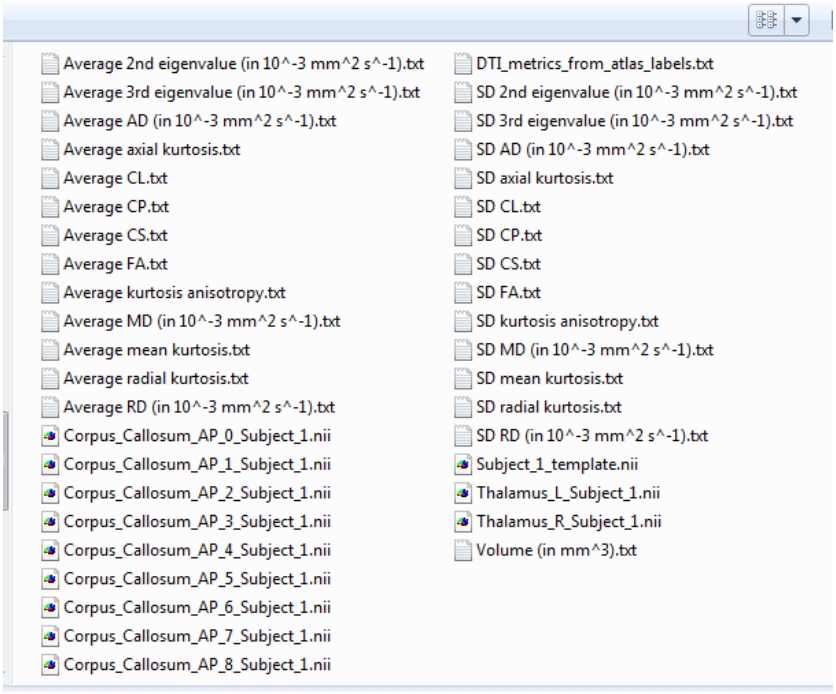
Note that you can choose your labels by setting “1” in the first column… “0” values indicate that you will not include that ROI/label in the analysis (FYI, only a part of the file content is shown below):



And then wait a few minutes (up to a few hours depending on data size and number of atlas labels)



The output looks like this (the *.nii files are for quality control):



As an example, I'll open the "Average FA.txt" file in Excel:

	A	B	C	D	E	F	G	
1		Thalamus_R	Thalamus_L	Corpus_Callosum_AP_0	Corpus_Callosum_AP_1	Corpus_Callosum_AP_2	Corpus_Callosum_AP_3	Cor
2	Subject_1	0.34982246	0.33281156	0.47112158	0.65627211	0.58885288	0.53871077	
3								
4								

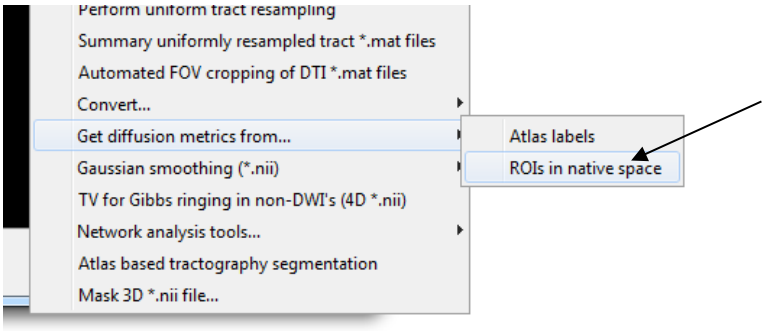
…

H	I	J	K	L
pus_Callosum_AP_4	Corpus_Callosum_AP_5	Corpus_Callosum_AP_6	Corpus_Callosum_AP_7	Corpus_Callosum_AP_8
0.60971981	0.58315009	0.61916775	0.68645954	0.81332505

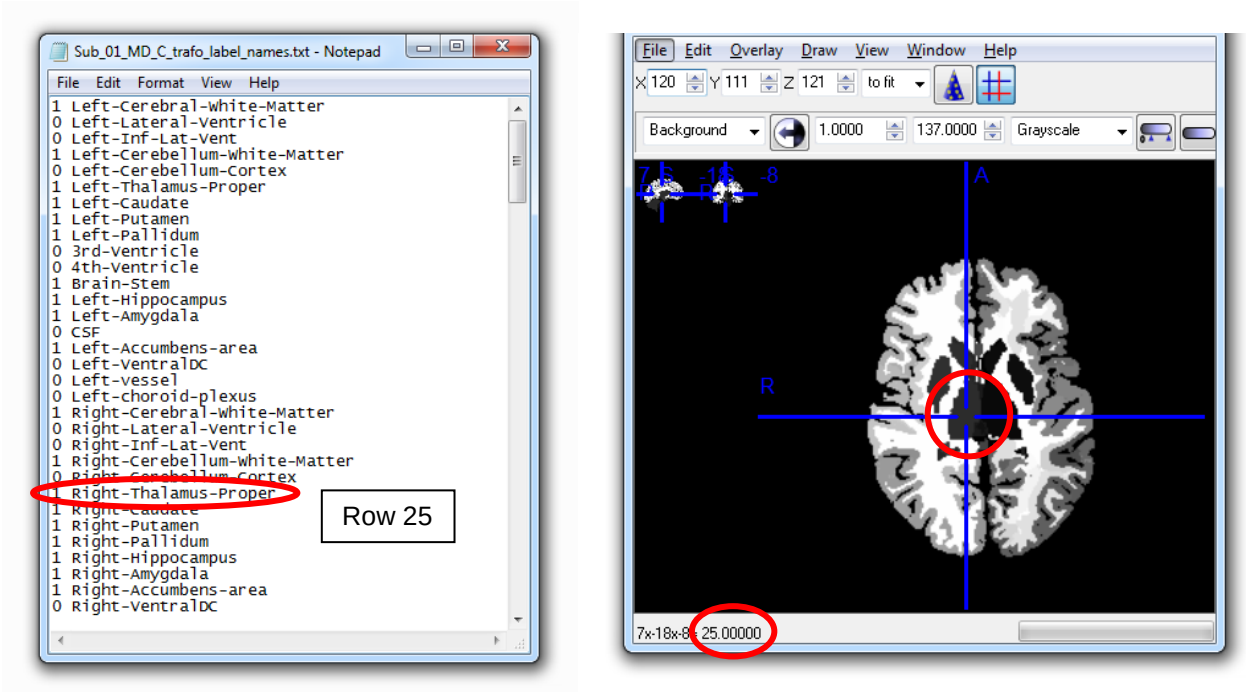
…

If you had more subject data sets, then the file above would have had more rows… That's it!

Now let's look at the other option ('ROIs in native space'):

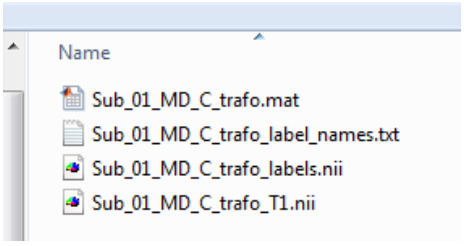


In this case, we assume that the ROIs are already in the same space of the diffusion data. The format of the ROIs is as usual: a 3D *.nii file with "0" values reflecting the background and values 1, 2, 3, etc. representing ROI 1, ROI 2, ROI 3, etc. The *.txt file with label names corresponds to the ROIs where the row number denotes the ROI number (see below).

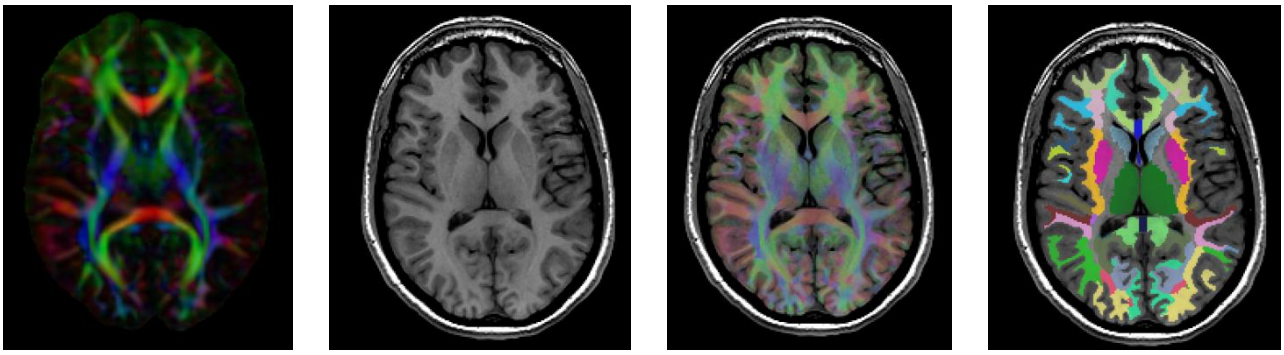


As usual, “0” values in the *.txt file mean that these ROIs are not considered for analysis.

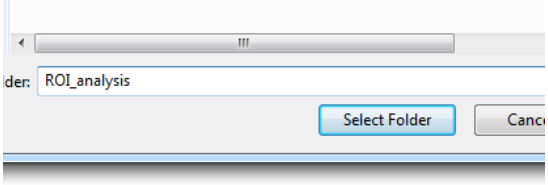
The input typically looks something like (here, folder name was “ROI_analysis” – see later):



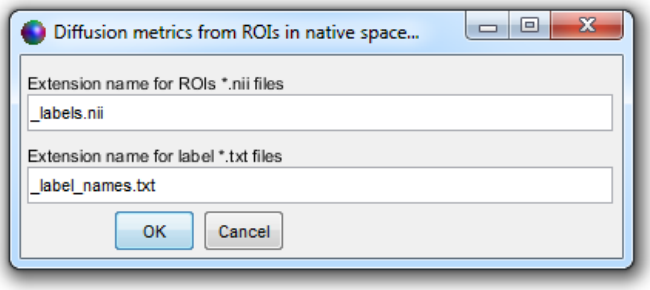
“Sub_01_MD_C_trafo.mat” is the DTI file. Note that it was transformed to the T1-weighted image, i.e. “Sub_01_MD_C_trafo_T1.nii”, during SM/EC/EPI distortion correction. “Sub_01_MD_C_trafo_labels.nii” has the labels and “Sub_01_MD_C_trafo_label_names.txt” the corresponding label names:



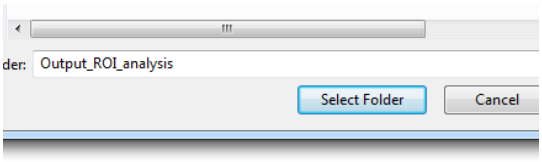
OK – let’s run the ‘ROIs in native space’ tool … First, select the input folder of DTI *.mat files:



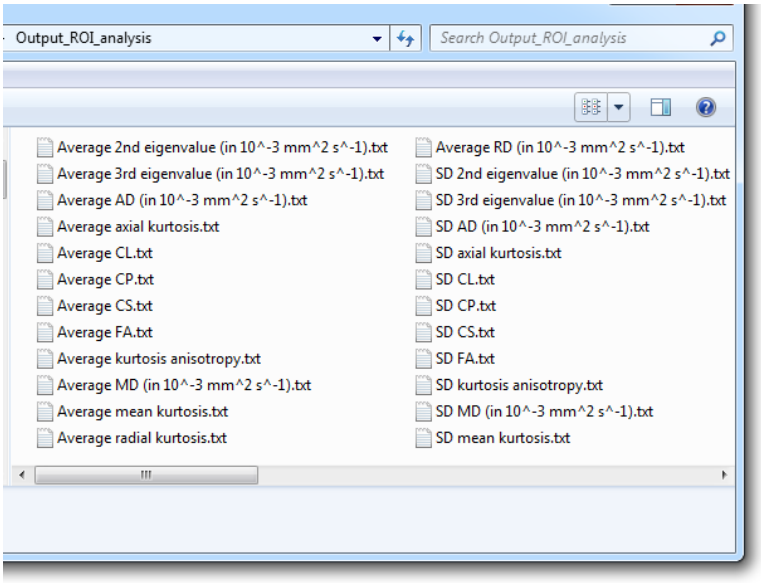
Then fill in the suffix parts corresponding to the labels/ROI *.nii file and label names *.txt file:



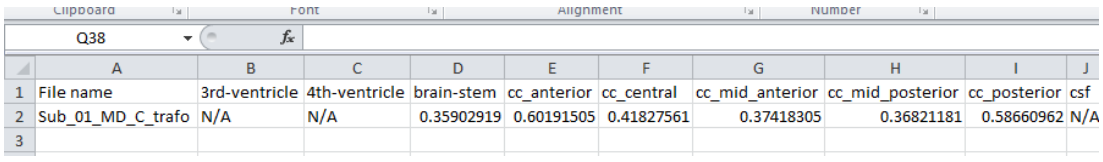
Finally, select an output folder:



Typically, this only takes a few minutes… The output looks like:



If we open for instance “Average FA.txt” in Excel, we get:



Q38

	A	B	C	D	E	F	G	H	I	J
1	File name	3rd-ventricle	4th-ventricle	brain-stem	cc_anterior	cc_central	cc_mid_anterior	cc_mid_posterior	cc_posterior	csf
2	Sub_01_MD_C_trafo	N/A	N/A	0.35902919	0.60191505	0.41827561	0.37418305	0.36821181	0.58660962	N/A
3										

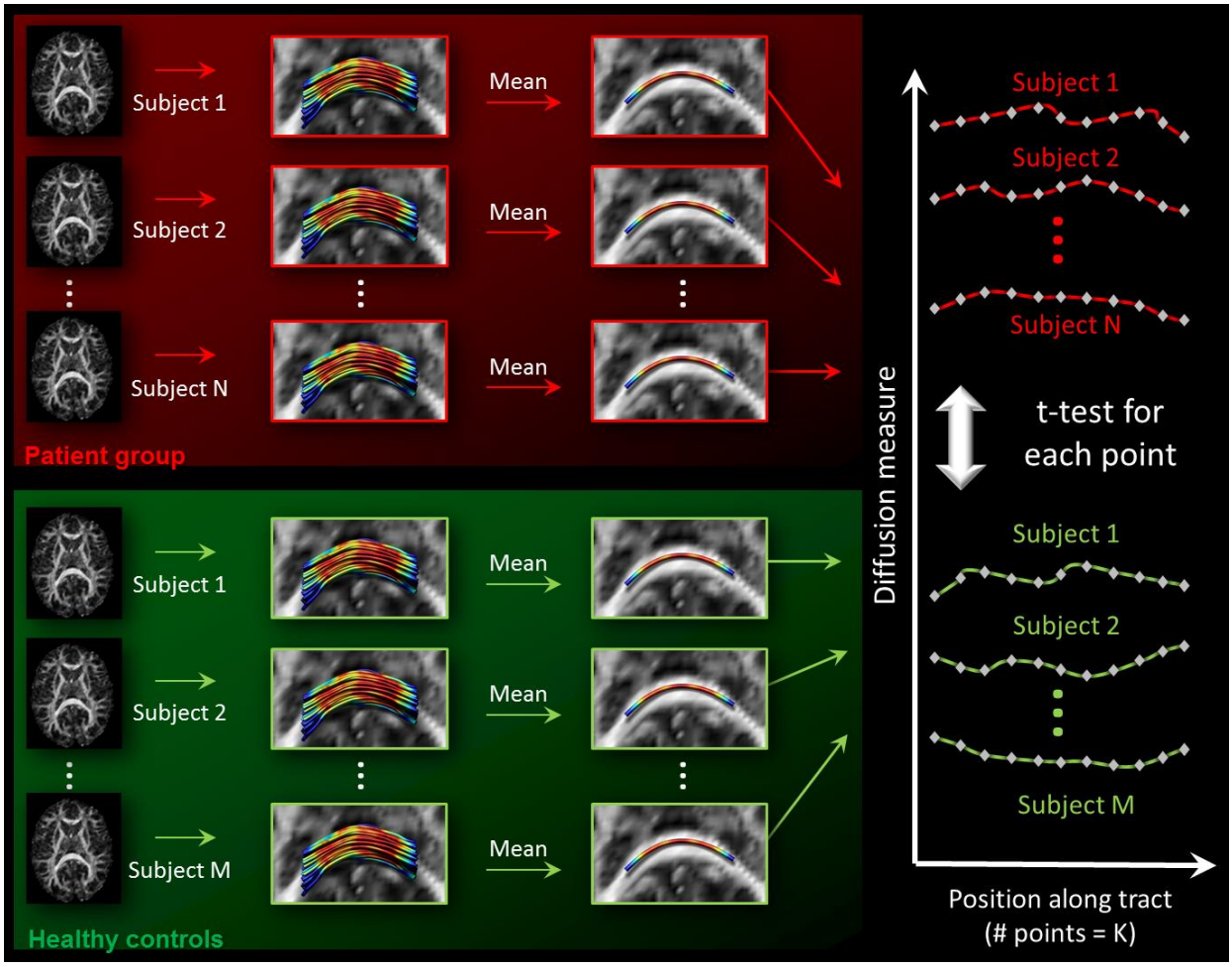
Again, each row represents a subject’s data set. The “N/A” indicates that for that ROI, there was a “0” value in the label names *.txt file.

That’s it!

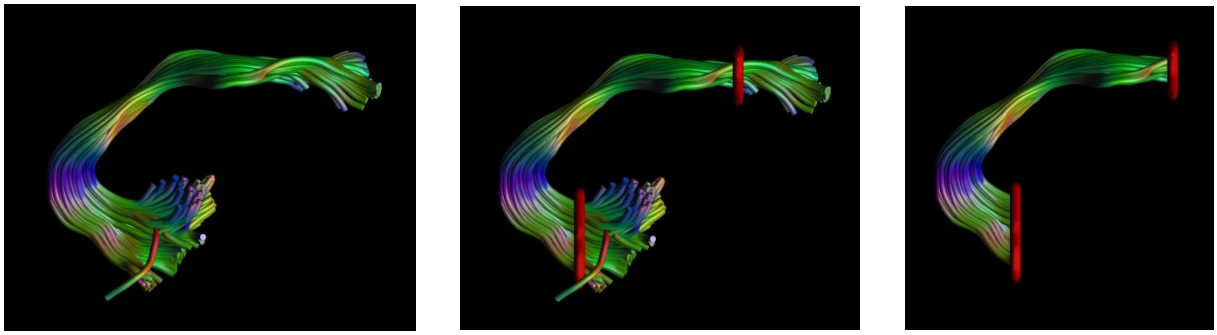
17. Along-tract analysis

For research papers where this kind of analysis has been used, see for instance O'Hanlon E. et al, JAMA Psychiatry 2015, Szczepankiewicz F. et al, NeuroImage 2013, and Reijmer Y.D. et al, Stroke 2013. The methodological details are based on Colby J.B. et al, NeuroImage 2012.

In summary, the idea is to investigate properties of a fiber bundle along its trajectory instead of averaging these properties across its entire path length. The following figure (adjusted from Perrone D. et al, NeuroImage 2015) should clarify this concept. As an example, 'K' points along a mid-sagittal segment of the forceps minor are compared (e.g., with a t-test) between 'M' healthy controls and 'N' patients.



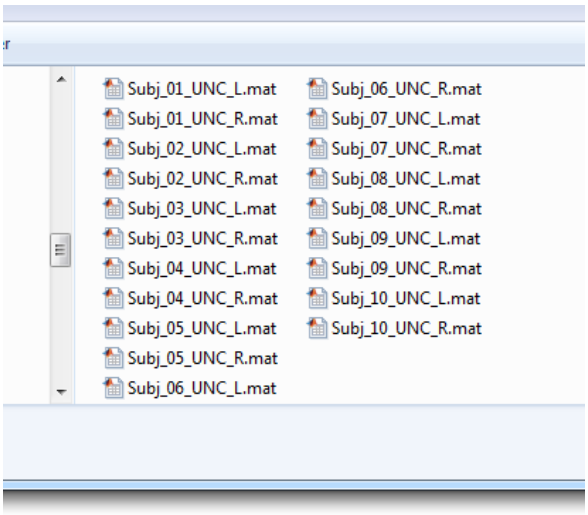
Some considerations… First, for each subject a fiber bundle (segment) needs to be computed. This can, for instance, be achieved with the tool described in Section 13 (or simply manually on a subject-per-subject basis). My personal preference is to use a specific "segment" to make sure the "endings" of a fiber tract are defined in a better way (see below):



Entire fiber bundle → with "segment ROIs" → Segment only

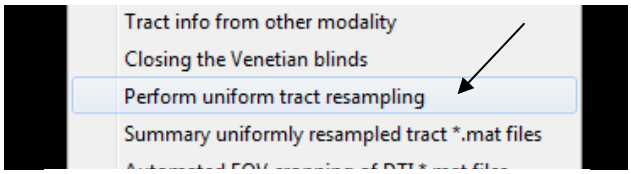
Second, the fiber bundles need to be resampled with a predefined number of points (i.e., “K” – as in the figure above), which should be same for each subject (to ensure correspondence). A reasonable value for “K” is to divide the average (across subjects) tract length of the fiber bundle of interest by the length scale of the voxel size. For example, if the voxel size is 2 x 2 x 2 mm^3 and the average length of the reconstructed uncinate fasciculus segments approximately equal to 70 mm, then K = 70/2 = 35 points should be fine. Taking more points will not add any new information and by sampling fewer points, you may miss information.

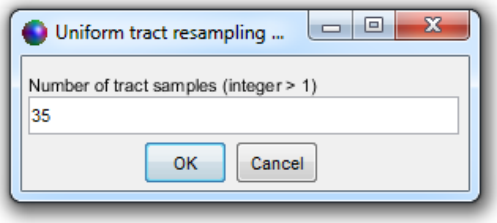
The input for the ExploreDTI tool consists of the tract (segment) *.mat files – all together in a single folder:



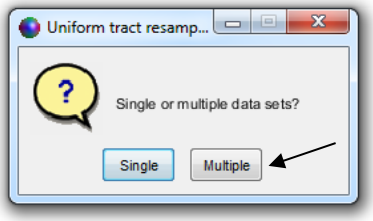
For this toy example, the idea is now to compare diffusion properties between the left and the right uncinate fasciculi for 10 subjects.

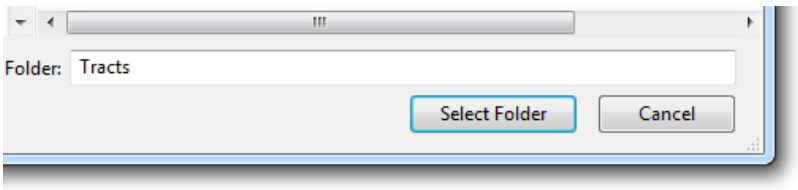
Step 1: Resample all the tract files with “K = 35” points. Go to ‘Plugins’ → ‘Perform uniform…’



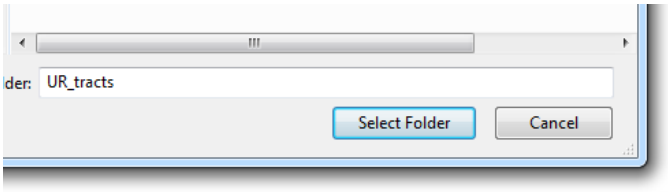


Then choose multiple data and select the folder of the tract *.mat files:

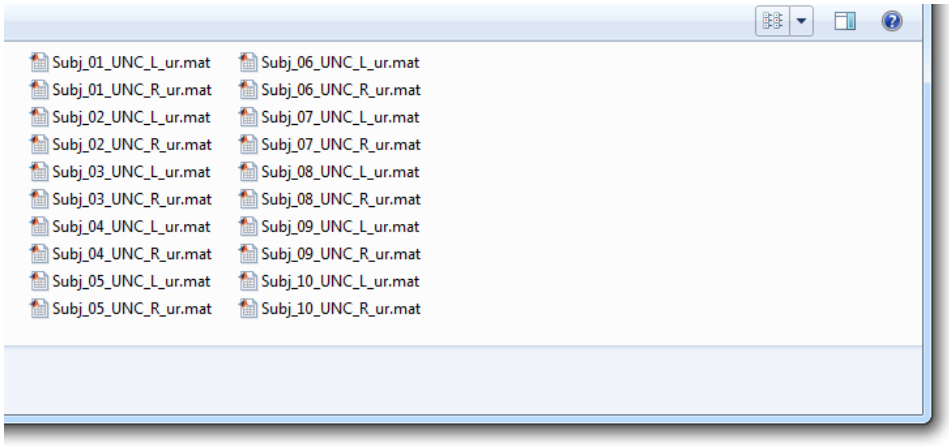




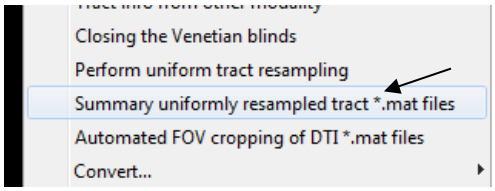
Save the uniformly resampled (UR) tracts in an output folder (here: “UR_tracts”):



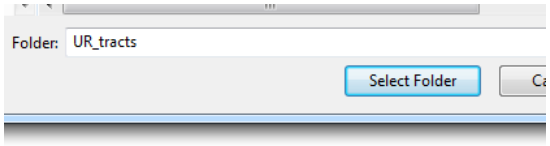
The output of this step is simply another set of tract *.mat files and can be used/visualized as any other tract *.mat file (notice the suffix “_ur” being added):



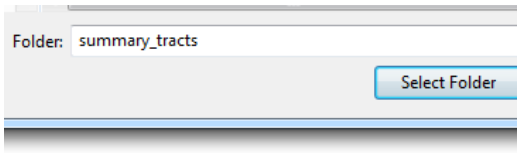
Step 2: for each tract *.mat file, the fiber bundle (set of trajectories) will be reduced to a single “averaged” pathway. This is achieved with the plugin:



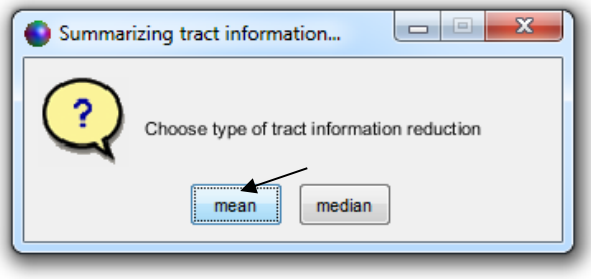
Select that folder of UR tract *.mat files:



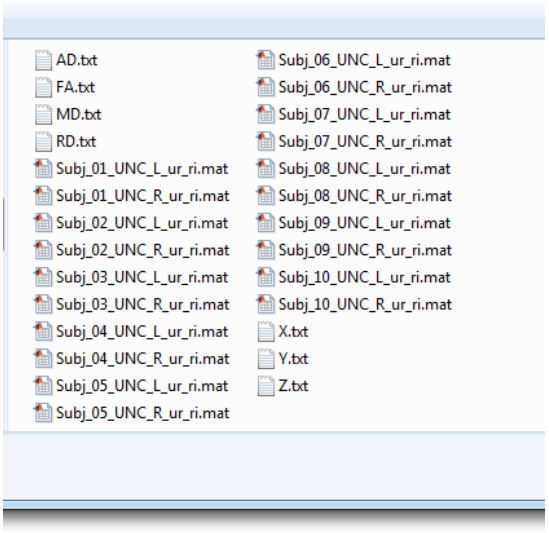
Save the summary of the UR tract *.mat files in some output folder (e.g., summary_tracts):



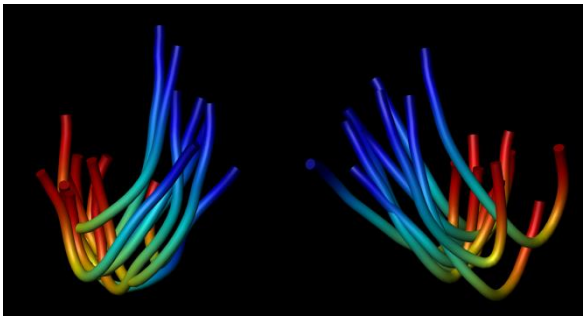
Then use “mean” or “median” to do the averaging of the diffusion metrics (I only included the option to check whether it would make a difference… well, it appears it doesn’t – so simply take the mean):



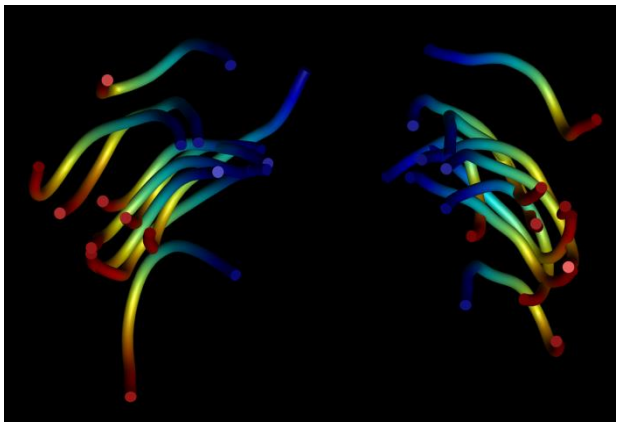
The end result looks like:



Now, each tract *.mat file simply consists of a single trajectory. After making sure that the endings of the pathways correspond between the subjects (and left/right sides), the figure below shows the 10 left and 10 right tracts with color-encoding reflecting this correspondence across subjects/sides:



Top view

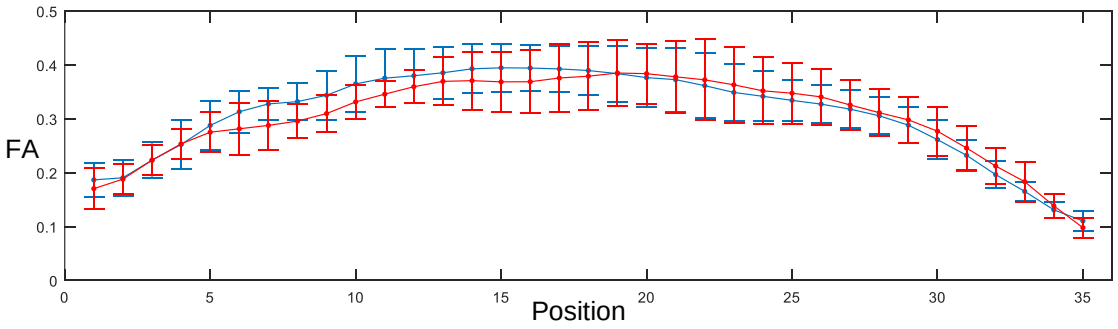


Frontal view

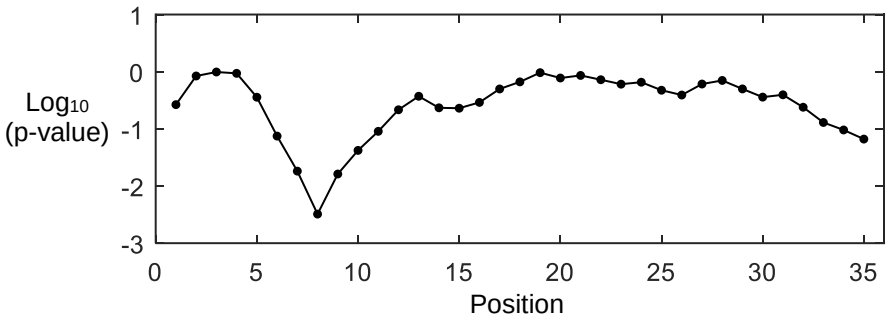
The *.txt files contain the diffusion measures (or world coordinates) along the pathways (columns) for each data set (rows). For instance, for the FA:

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Subj_01_UNC_L_ur	0.1978476	0.22410513	0.2455098	0.27902072	0.30362992	0.29933797	0.30122764	0.30527867	0.31157305	0.34083669	0.3560173	0.3678282
2	Subj_01_UNC_R_ur	0.11055904	0.15978238	0.23502203	0.23698129	0.26954632	0.24962346	0.23676009	0.25822902	0.28972203	0.3209552	0.34120971	0.352996
3	Subj_02_UNC_L_ur	0.20158867	0.19965639	0.23552328	0.29726335	0.34429403	0.34226217	0.3506436	0.3589523	0.37879391	0.40932784	0.40323674	0.4032865
4	Subj_02_UNC_R_ur	0.1765241	0.19477864	0.22247859	0.26083997	0.30781773	0.33403664	0.33748694	0.33021006	0.35423194	0.37576671	0.36522666	0.3652608
5	Subj_03_UNC_L_ur	0.19127484	0.17575671	0.20828917	0.23658924	0.29802598	0.35943015	0.35779146	0.3439793	0.36541418	0.40783358	0.42191944	0.4155703
6	Subj_03_UNC_R_ur	0.19147925	0.17908318	0.23166281	0.27487564	0.25862883	0.24843832	0.26600498	0.29443066	0.33786838	0.37346705	0.38445964	0.4054504
7	Subj_04_UNC_L_ur	0.1905824	0.17127414	0.1837973	0.20057766	0.24413764	0.28880355	0.30610703	0.30908539	0.31823764	0.33898552	0.36140458	0.3859752
8	Subj_04_UNC_R_ur	0.17405451	0.2097058	0.24863955	0.24997392	0.27101473	0.26741215	0.27704688	0.29827911	0.31022659	0.32790035	0.34713296	0.3666902
9	Subj_05_UNC_L_ur	0.13664559	0.15025762	0.21736874	0.24490603	0.27221143	0.26830379	0.28086123	0.28896291	0.30132772	0.33091689	0.37051946	0.383306
10	Subj_05_UNC_R_ur	0.11808045	0.15467317	0.18753063	0.20965887	0.23808695	0.25501636	0.26953916	0.26650675	0.2678526	0.30538511	0.35020485	0.3739258
11	Subj_06_UNC_L_ur	0.16642318	0.15367524	0.16371438	0.16622704	0.19131964	0.24309958	0.29620403	0.28209809	0.26062938	0.25497078	0.25091611	0.258653
12	Subj_06_UNC_R_ur	0.21171651	0.19861943	0.20237974	0.24363264	0.23383232	0.24241234	0.26169646	0.26316012	0.26830285	0.29336377	0.31923048	0.3291594
13	Subj_07_UNC_L_ur	0.15631274	0.18008358	0.21962814	0.25606323	0.2919215	0.30409192	0.3179618	0.33296327	0.35105555	0.37207939	0.37981124	0.374865
14	Subj_07_UNC_R_ur	0.13472333	0.14371899	0.17163569	0.21561202	0.24857433	0.27469067	0.28772675	0.2881444	0.28266783	0.29386919	0.30111358	0.2998688
15	Subj_08_UNC_L_ur	0.22718161	0.24060717	0.28001925	0.32101761	0.33841739	0.35196553	0.35207963	0.35601877	0.36165745	0.35485697	0.34946282	0.3566723
16	Subj_08_UNC_R_ur	0.21355266	0.22388502	0.25485338	0.28713744	0.32507106	0.35151734	0.36047656	0.35025877	0.35200782	0.3595545	0.35920834	0.3707059
17	Subj_09_UNC_L_ur	0.23606914	0.23613196	0.24934666	0.27026541	0.31566818	0.35292476	0.36597455	0.38389892	0.41378311	0.4327728	0.44255176	0.4503983
18	Subj_09_UNC_R_ur	0.16175105	0.1993354	0.23409755	0.26175589	0.25954058	0.23257936	0.23484776	0.27780853	0.29362327	0.30860555	0.32880112	0.3403587
19	Subj_10_UNC_L_ur	0.1628739	0.17451604	0.23039423	0.25174676	0.27804873	0.32202658	0.34790749	0.35836527	0.3746804	0.40557555	0.41904306	0.4035664
20	Subj_10_UNC_R_ur	0.21180424	0.2181099	0.24345584	0.29529049	0.33967236	0.35978288	0.34684397	0.33096345	0.34154448	0.35541155	0.36118051	0.391724
21													

Or when plotting the left (blue) versus the right (red) side FA values (error bars are simply the standard deviation across subjects):



While there are some significant (uncorrected p<0.05) differences (lowest p-value was roughly 0.003 – see below), one may argue that a correction factor of 35 (Bonferroni – number of t-tests) needs to be included… No significant difference then for any point along the tract.



Using the corresponding “X” (front-back), “Y” (left-right), and “Z” (bottom-top) coordinate *.txt files, it’s trivial to understand what is the beginning/ending of the plot (for instance for the uncinate fasciculus, the temporal pole has typically lower values of the z-coordinates than the orbitofrontal part).

That’s all for now!