**ROSSEZ Arno, DELOULME Jean-Christophe, MATHIEU Hervé**

**GIN, Grenoble Institut des Neurosciences**

05/06/2025

**SpasTract Tutorial Step-by-step - V1.0**

**A. Prerequisites :**

* Have a station equipped with a screen, a keyboard and a mouse.
* Know how to navigate session files, how to use a terminal (Here Linux, Ubuntu).
* Have the SpasTract tutorial folder on your station.
* Python installed (recommended 3.8+)
* SpasTractRequirements.txt conditions met
* [Optional for external masks] MATLAB installed
* Note for the reader: The terminal location is shown in blue, and the commands used are shown in ***red italics***.
* The '*\'*  show the line return in this protocol but are not part of the command lines.

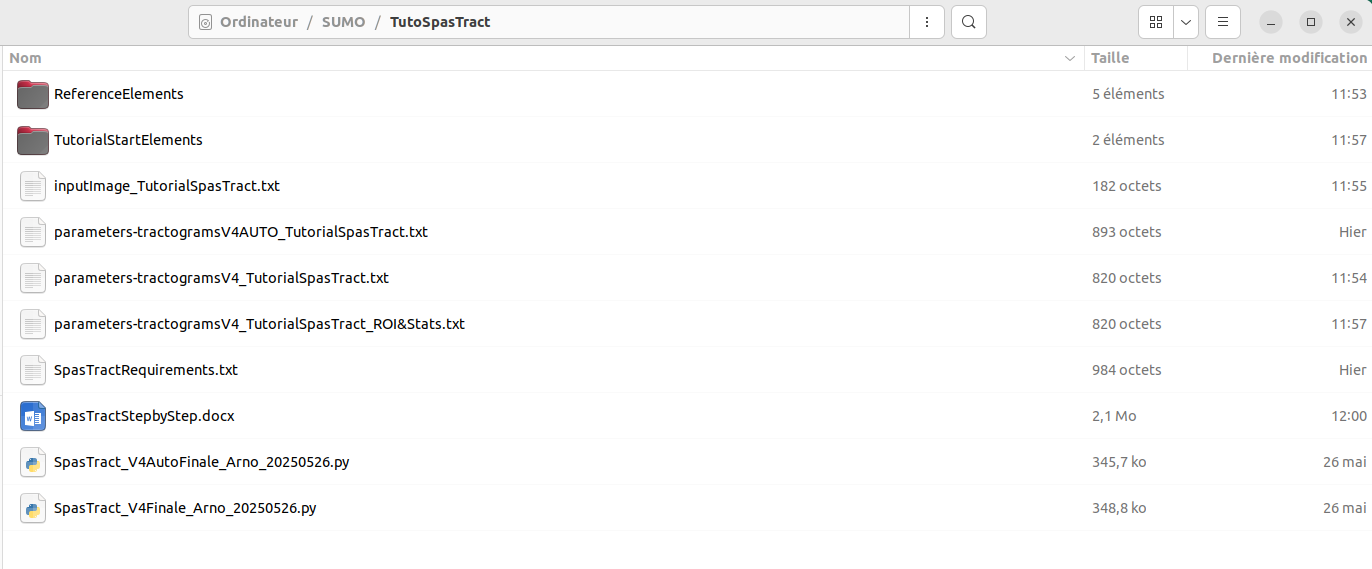
**B. Method :**

* *Running Python pipeline*
  + Make sure to have the Python script "SpasTract\_V4(Auto)Finale\_Arno\_20250526.py". With this script, also remember to prepare an "inputImages\_TutorialSpasTract.txt" file (or equivalent, the name is not under any constraints) containing the paths to the .nii data (unless it's an automated code) and prepare a sample "parameters-tractogramsV4\_TutorialSpasTract.txt" to follow and modify for configuration (here the name needs to contain the string 'parameters-tractograms' somewhere in the name).
  + Edit "parameters-tractograms.txt" to include the parameters of the tractography(ies) to be performed (number of samples, voxel dimensions, FA cutoff, etc.). As well as for the different script modes: generating tractography(ies), performing a SIFT1 or SIFT2 analysis on tractograms, performing an analysis of specified ROIs on tractograms, performing a statistical compilation of tractograms, etc.

*Note : The directions will often be of 6,15,28,… to match the thresholds described below :* Determine the maximum value for lmax that is supported by the number of DWI volumes in the shell being processed (or the total number of non-[\*](https://mrtrix.readthedocs.io/en/3.0_rc1/constrained_spherical_deconvolution/lmax.html" \l "id1)b\*=0 volumes in a single-shell acquisition). This is the number of coefficients required to store an anitipodally-symmetric spherical harmonic function:

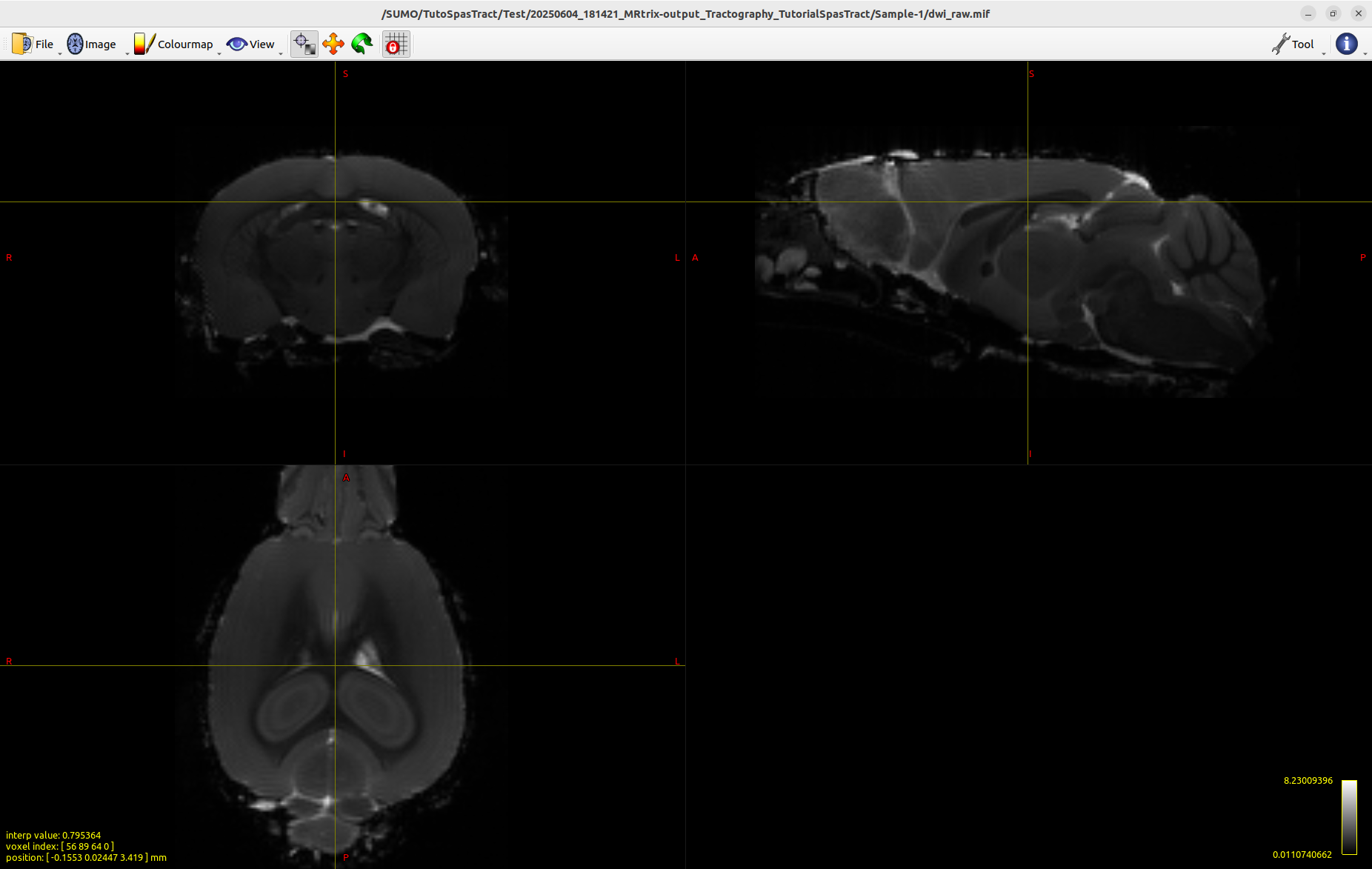
|  |  |
| --- | --- |
| l\_max | Required volumes |
| 2 | 6 |
| 4 | 15 |
| 6 | 28 |
| 8 | 45 |
| 10 | 66 |
| 12 | 91 |
| ... | ... |

* + It is recommended to organize the folder so that the different files are not confused: mask.nii//bvecsbvals.txt//dMRI-TutorialSpasTract.nii//parameters-tractograms.txt//inputImages.txt. And in the case of an automatic script launch, have next to the script : the parameter file containing at least 'parameters-tractograms' in .txt format and a subfolder like follows dMRI/[mask.nii,bvecsbvals.txt,dMRI-TutorialSpasTract.nii] where we ensure that the subfolder name is contained in the name of dMRI-TutorialSpasTract.nii data.



*Figure 1 : How your SpasTract tutorial folder should look like*

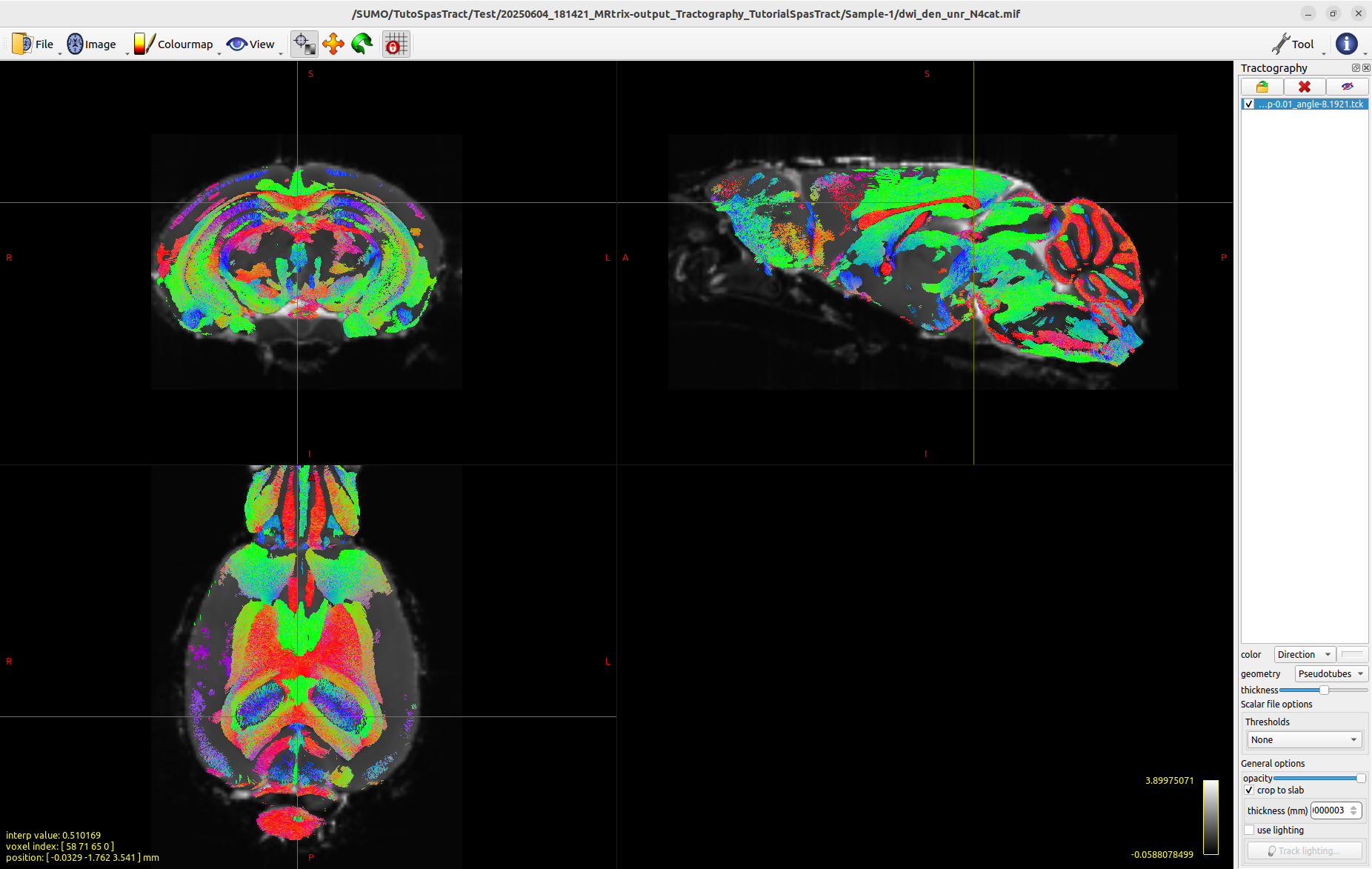
* + Open a terminal in the organized tractography folder then enter: *python SpasTract\_V4Finale\_Arno\_20250526.py* . Then follow the instructions given by the script dialog boxes. **NB:** Normally, you don't need to specify a Python version when launching the script, as the script is compatible with all 3.X versions of Python. However, it is possible to specify *python3.12*, for example, when ordering.
  + View the tractography results with for example the command: *mrview dwi\_den\_unr\_N4cat.mif*, then in the MRtrix GUI, go to Tools > Tractography > and select one of the .tck files. Below is a showcase of expected results for Sample-1 a.k.a F1023 but you could also check in the ReferenceElements folder the other results for Sample-2.

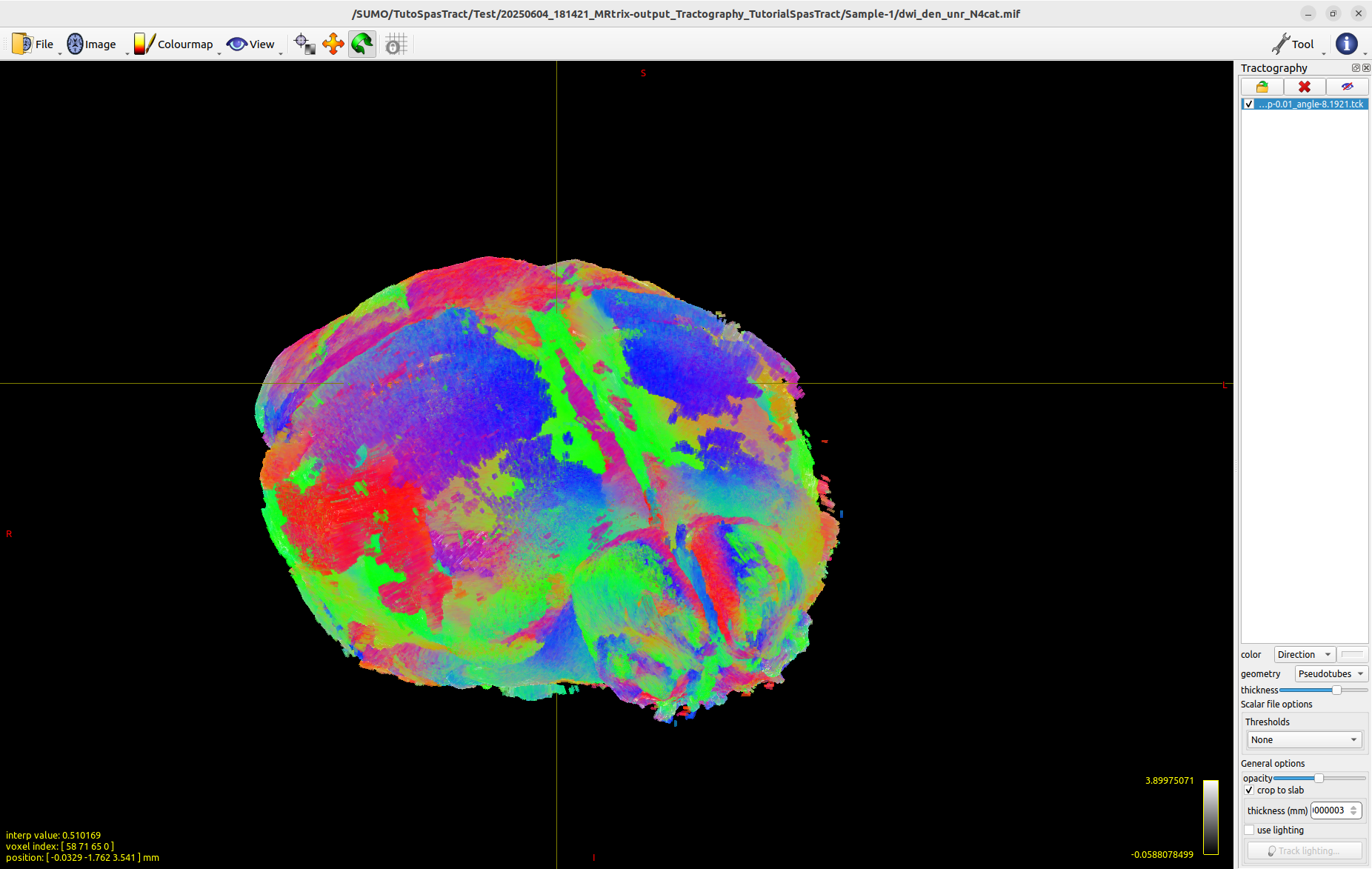


*Figure 2 : Raw DWI image of Sample-1*



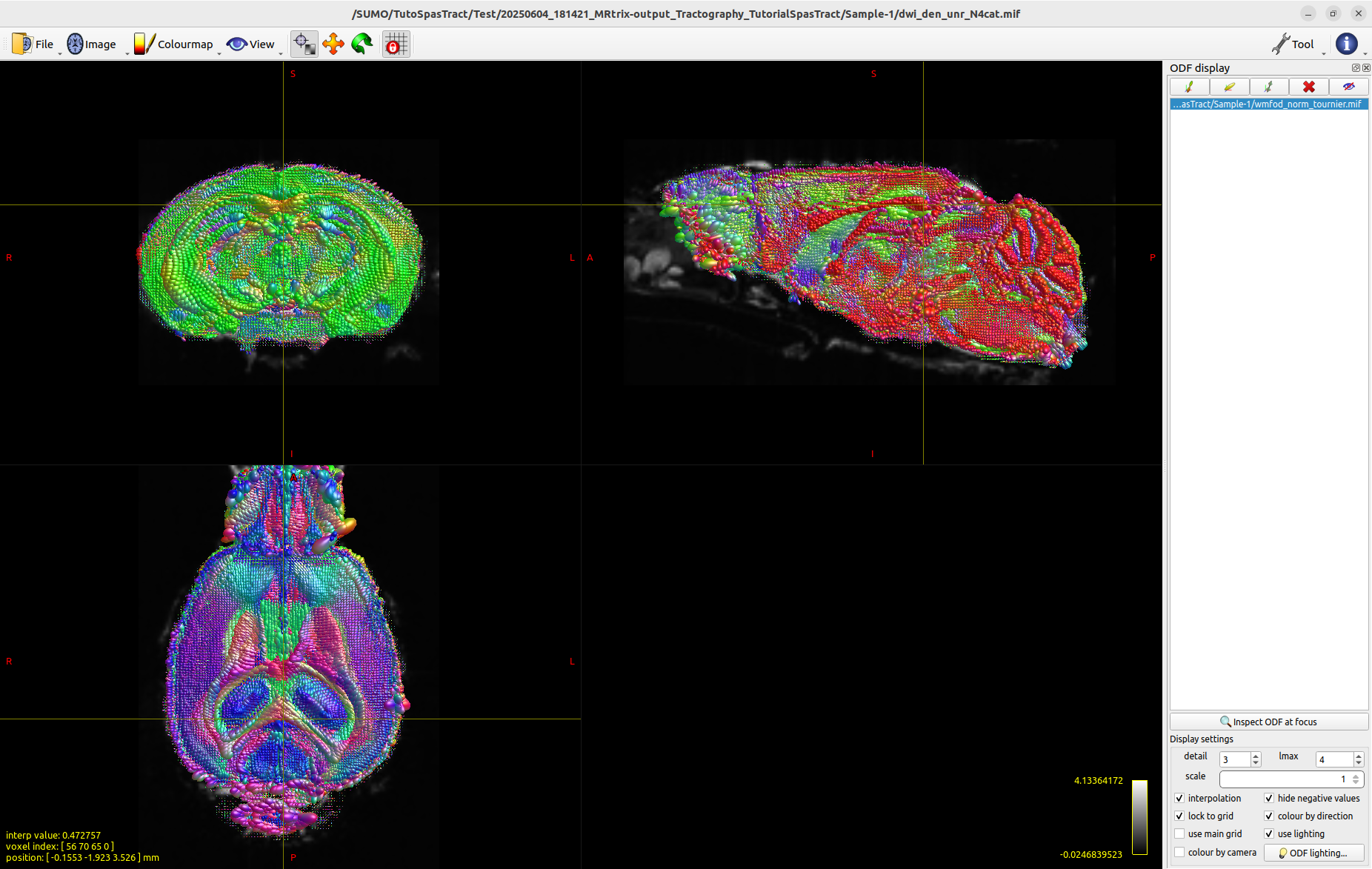
*Figure 3 : preprocessed DWI image of Sample-1*





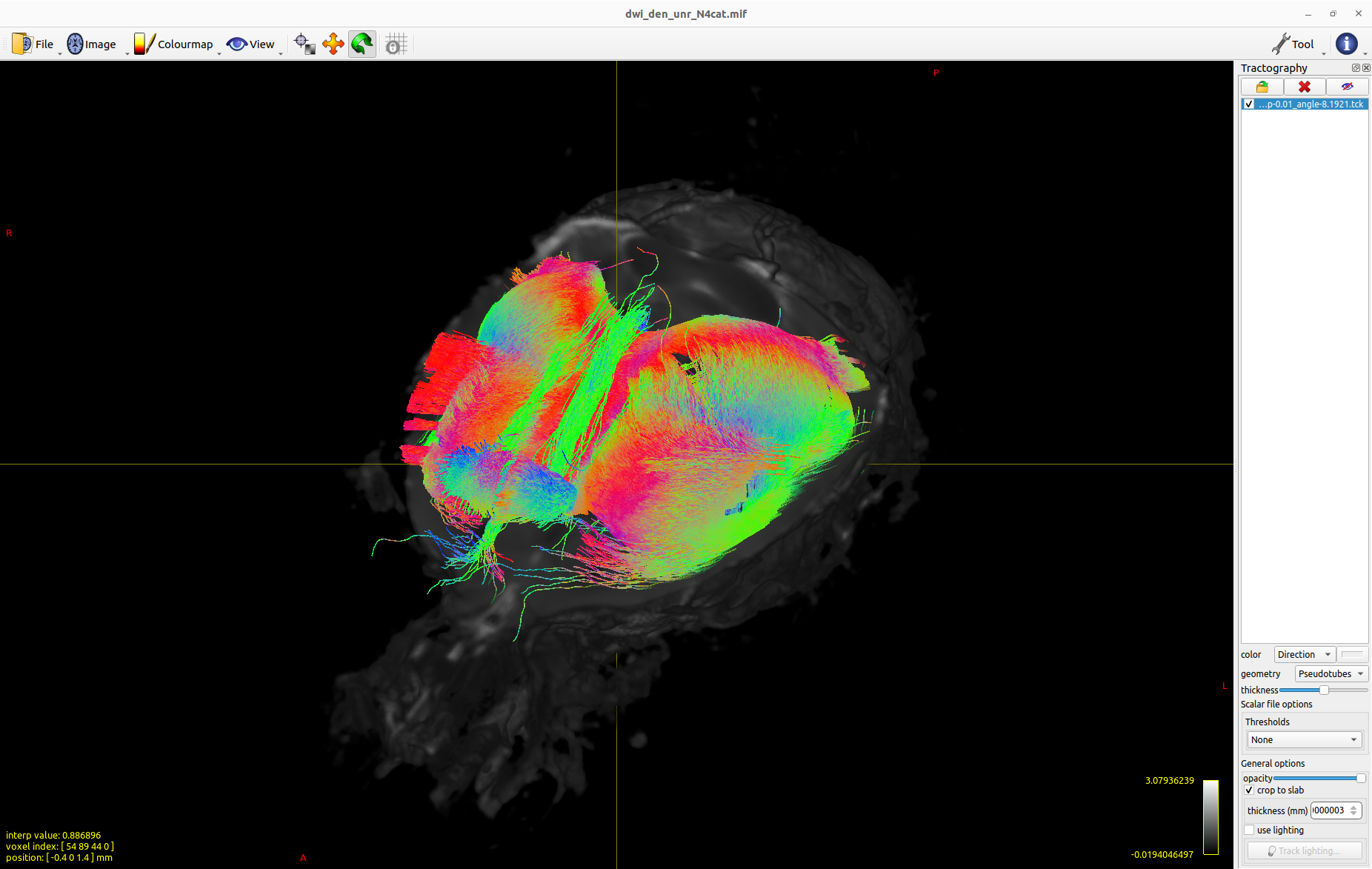
*Figure 4.a,b : Example of obtained result after tractography for Sample-1*

**NB :** It may also be a good idea to check the generated FODs, whether normalized (or not), by going to Tools>ODF Display>Open SH image and then selecting, for example, wmfod\_norm.mif, the generated FODs will be visible over the brain. You can then also check specific ODFs using 'Inspect ODF at focus', which will open a separate window with the FOD.



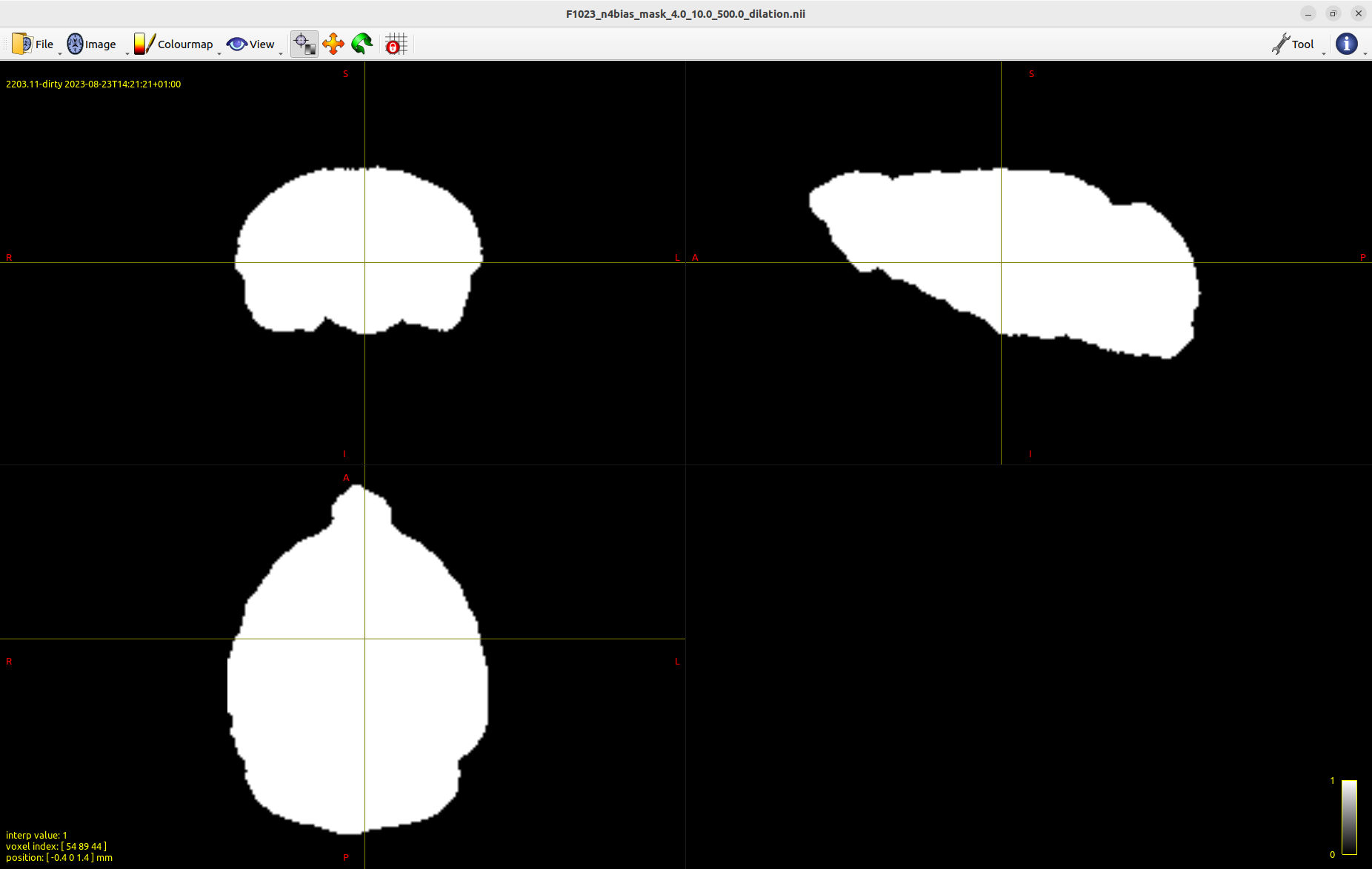
*Figure 5 : Example of ODFs visualisation for Sample-1*

* *Note on ROIFiltering and Statistics :*
  + You can set the parameter file to Yes on the different modes like ROIFiltering, ROIStatistics and WholeBrainStatistics… Each time it will trigger the code once you run it to ask you where you stored your output tractography, you have to point where the 'Sample-X' folder(s) appear(s). For ROI however you need also to place the ROI files into those 'Sample-X' folders (while being careful not misplace them !).
  + These modes will generate files in the 'Sample-X' folders but also a separate folder next to them with the specific files generated rather than mixed with the rest.
* *Note on expected results :*
  + You will find every expected results in the ReferenceElements folder, this is a way to quickly verify if your results have significant issues with a glance.
  + Finally another detail to check is that after ROIFiltering of ac,cc,py (ROIs are provided in ReferenceElements), you have damaged tracts for Sample-2 and intact tracts for Sample-1 as they are respectively M974 KO-SPASTIN and F1023 WT. Most blatant example is corpus callosum showing holes or not :



*Figure 6.a,b : Sample-1 F1023 WT corpus callosum (6.a) vs Sample-2 M974 KO-SPASTIN corpus callosum (6.b)*

* Note on creating ROIs [In case you are preparing an ROI analysis based on tractograms]:
  + In the MRtrix GUI, select Tools>ROI Editor>Use the "brush" and "fill" tools to place ROIs. For example, on the fornix at +3 slices and -3 slices of the parasagittal slice (where the fornix is at its highest, middle of the coronal and axial slices), save these ROIs then enter the following command in a terminal located in the folder containing the .tck: "*tckedit \ -include ROIsag\_AC\_1\_and.mif -include ROIsag\_AC\_2\_and.mif \ tracks\_3\_cutoff-0.2800\_minl-0.1\_maxl-50\_step-0.01\_angle-8.1921.tck \ AC\_reconstructiontest\_028.tck*" then go to Tools > Tractography > select the .tck generated from the fornix "AC\_reconstructiontest\_028.tck" and check the correct orientation of this tract (normally red indicating a left-right orientation).
  + Finally, if a compilation of statistical parameters has been launched on the tractograms or on the ROI tractograms, check that the Results\_WholeBrainStatistics.txt and/or Results\_roi.txt files are present and filled in. Similarly, histograms listing the number of fibers for each length in mm had to be generated for each sample. (Examples of ROIs for ac,cc,py are available for both samples in ReferenceElements and a parameter file has been prepared in ReferenceElements too for ROI&Stats)
* *Note on the creation of external mask via PCNN3D [Optional] :*
  + Retrieve the data in .nii format, then open a terminal at their location.
  + Enter the command: *dwiextract dwi.nii - -no\_bzero | mrmath – mean \ mean\_Xdirsvolumes.nii -axis 3* , this will extract the direction volumes and then average them together to obtain an image with the best SNR while keeping a proper header (this also allows you to ignore the agarose present otherwise in the volume(s) b=0).
  + Once this is done, correct the surface antenna bias using the N4BiasFieldCorrection function from ANTs: *N4BiasFieldCorrection -i mean\_Xdirsvolumes.nii -o N4mean\_Xdirsvolumes.nii*
  + Finally, launch MATLAB and select the PCNN3D script (for example, "KHC\_SimpleSelectPCNN3D\_singlerun"). Check the parameters, then run the script by clicking or placing it upstream on N4mean\_Xdirsvolumes.nii. At the end of the script, a mask resulting from the segmentation will be produced on N4mean\_Xdirsvolumes.nii in .nii.gz format. Unzipping it produces our mask (whose name includes the various parameters used). Correct the mask produced if necessary..



*Figure 7 : Example of obtained mask via PCNN3D (or automatically by dwi2mask in the script)*