**Supporting Information**

**S1 File. Analysis pipeline - TBSS analysis of DTI and NODDI data**

1. **Check data quality etc**
* Visual inspection of raw data (using *fslview* time series for instance)

1. **Pre-processing**
* Estimation of susceptibility induced distortions using pairs of b0’s acquired in opposing phase encoding directions (fsl’s *topup*). Use *fslsplit* and *fslmerge* if necessary to create a nii file with all b0’s.

*fsl5.0-topup --imain=IngTim\_001\_b0s --datain=acqparams.txt --config=b0.cnf --iout=IngTim\_001\_topup\_iout --out=IngTim\_001\_topup\_out -- fout=IngTim\_001\_topup\_fout --logout=IngTim\_001\_topup\_logout --verbose*

* Estimation of eddy current-induced distortions and head motion, plus correction of all distortions (fsl’s *eddy*). Combine data again and create a brain mask (fsl’s *bet* and fsl’s *fslmaths*).

*fsl5.0-eddy --imain=IngTim\_001\_all --mask=IngTim\_001\_brainmask --acqp=acqp\_all\_eddy.txt --index=indexfile\_all.txt --bvecs=IngTim\_001\_all.bvec -- bvals=IngTim\_001\_all.bval --out=IngTim\_001\_all\_eddy --topup=IngTim\_001\_topup\_out –verbose*

* Rotate bvecs to account for the corrections
1. **DTI fitting using fsl dtifit**
* Fit the DTI model to the data (fsl’s *dtifit*). Note that this should be done on one shell (b1000 is more common than b2000); split the data (*fslsplit*, *fslmerge* or *fslroi*) and also adapt bvec and bval files accordingly.

*fsl5.0-dtifit --data=IngTim\_001\_b1k\_eddy --out=IngTim\_001\_b1k\_eddy --mask=IngTim\_001\_brainmask --bvecs=IngTim\_001\_b1k\_rotated.bvec --bvals=IngTim\_001\_b1k.bval –verbose*

1. **Registration using DTI-TK (tensor-based registriation)**
* Download and install DTI-TK - http://dti-tk.sourceforge.net/pmwiki/pmwiki.php
* Transform data to DTI-TK format (DTI-TK’s *fsl\_to\_dtitk*).

*fsl\_to\_dtitk IngTim\_001\_b1k\_eddy*

* Check the data using *TVtool* and *TVglyphs* (check orientation of glyphs)
* Spatial normalization in 3 steps (1. bootstrapping, 2. affine alignment, 3. diffeo alignment)

*1. Create txt file with all subject names (IngTim\_001\_b1k\_eddy\_dtitk.nii.gz etc)*

*dti\_template\_bootstrap /home/brain/Desktop/Tools/dtitk/ixi\_aging\_template\_v3.0/template/ixi\_aging\_template.nii.gz subjs.txt*

*2. dti\_affine\_population mean\_initial.nii.gz subjs.txt EDS 3*

*3. Create a mask - TVtool -in mean\_affine3.nii.gz –tr; BinaryThresholdImageFilter mean\_affine3\_tr.nii.gz mask.nii.gz 0.01 100 1 0*

*dti\_diffeomorphic\_population mean\_affine3.nii.gz subjs\_aff.txt mask.nii.gz 0.002 (this last one takes a lot of time, try using an advanced computer or server)*

* Check registration by a) creating FA maps (DTI-TK’s *TVtool*) and inspecting these; b) create grids, apply transformations to the grids and inspect them using *MRIcron* for instance
1. **Post registration steps – preparation for TBSS**
* Compute the combined transformation field, and warp the data using this single transformation (one transformation is always better than multiple).

*dti\_warp\_to\_template\_group subjs.txt mean\_diffeomorphic\_initial6.nii 1 1 1*

* Create high resolution template from the individually warped data (*TVMean*) and generate a mean FA map from this (*TVTool*)

*TVMean -in subjs\_normalized.txt -out mean\_final\_high\_res.nii.gz (requires txt file with all normalized file names)*

*TVtool -in mean\_final\_high\_res\_masked.nii.gz –fa (rename to be consistent with TBSS: mv mean\_final\_high\_res\_masked\_fa.nii.gz mean\_FA.nii.gz)*

* Create FA maps for each individual and merge them into one file (*fslmerge*)

*TVtool -in IngTim\_001\_b1k\_dtitk\_diffeo.nii -fa, etc*

*fsl5.0-fslmerge -t all\_FA IngTim\_001\_b1k\_dtitk\_diffeo\_fa.nii ……*

* Generate the mean FA skeleton

*fsl5.0-tbss\_skeleton -i mean\_FA -o mean\_FA\_skeleton*

* Put everything in folder named in agreement with TBSS (move all\_FA, mean\_FA, mean\_FA\_mask and mean\_FA\_skeleton to a folder called ‘stats’)
1. **TBSS - Pre-statistics**
* Check the skeleton and see whether FA of .2 works as threshold

*fslview all\_FA -b 0,0.8 mean\_FA\_skeleton -b 0.2,0.8 -l Green*

* Create binary skeleton mask, create distance map for projection of FA to skeleton, and create the 4D file with all skeletonized FA data – TBSS step 4

*fsl tbss\_4\_prestats 0.2*

1. **(For AD, RD and TR)**
* Extract the data from the normalized tensor data using TVtool

*TVtool -in IngTim\_001\_b1kl\_dtitk\_diffeo.nii.gz -ad, etc*

* Merge all data into one 4D file (*fslmerge*)
* Create mean output maps (requires txt file with file names)

*SVMean -in subjs\_AD.txt -outMean mean\_AD.nii.gz -outStd meanstd\_AD.nii.gz*

* Copy following files to ‘stats’ folder: all\_AD, mean\_AD, all\_FA, mean\_FA, mean\_FA\_mask, mean\_FA\_skeleton, mean\_FA\_skeleton\_mask\_dst
* Project the data onto the FA skeleton using the distance map created using the FA data

*fsl5.0-tbss\_skeleton -i mean\_FA -p 0.2 mean\_FA\_skeleton\_mask\_dst /usr/share/fsl/5.0/data/standard/LowerCingulum\_1mm all\_FA all\_AD\_skeletonised -a all\_AD -s mean\_FA\_skeleton*

1. **NODDI fitting (in parallel, after step 1)**
* Matlab toolbox - http://mig.cs.ucl.ac.uk/mig/mig/index.php/?n=Tutorial.NODDImatlab/
* Create a region of interest (ROI) for the NODDI fitting (usually whole brain)

*CreateROI('IngTim\_001\_all\_eddy.nii', 'IngTim\_001\_brainmask.nii', 'IngTim\_001\_NODDI\_roi.mat');*

* Convert the bval/bvec files into the required format

*protocol = FSL2Protocol('IngTim\_001\_all.bval', 'IngTim\_001\_all\_rotated.bvec');*

* Create the NODDI model structure

*noddi = MakeModel('WatsonSHStickTortIsoV\_B0');*

* Run the NODDI fitting (using either parallel computing toolbox or without)

*batch\_fitting('NODDI\_roi.mat', protocol, noddi, 'FittedParams.mat', 8); or:*

*batch\_fitting\_single('NODDI\_roi.mat', protocol, noddi, 'FittedParams.mat');*

* Convert the estimated NODDI parameters into scalar output maps. This steps give you the following files: \_ficvf.nii (NDI), \_odi.nii (ODI), \_fiso.nii (CSF volume fraction), \_fibredirs\_{x,y,z}vec.nii (fiber orientation), \_fmin.nii (fitting objective function values), \_kappa.nii (concentration parameter of Watson distribution used to compute ODI), and \_error\_code.nii (non-zero values indicate fitting errors)

*SaveParamsAsNIfTI(FittedParams, ROIFile, MaskFile, 'IngTim\_001\_NODDI')*

1. **Registration of the NODDI data (ficvf/NDI, ODI, fiso) to the template (after step 4)**
* Prepare NODDI output maps for DTI-TK: set origin of the scalar maps to 0 0 0

*SVAdjustVoxelspace -in IngTim\_001\_NODDI\_ficvf.nii -origin 0 0 0 -out IngTim\_001\_NODDI\_ficvf\_origin.nii*

* Warp the data using combined transformation file (from DTI-TK) to the high resolution template

*deformationScalarVolume -in IngTim\_001\_NODDI\_ficvf\_origin.nii -trans IngTim\_001\_b1k\_dtitk\_combined.df.nii.gz -target mean\_final\_high\_res.nii.gz -out IngTim\_001\_NODDI\_ficvf\_normalized.nii*

1. **Continue with TBSS for the NODDI data**
* Create mean output maps

*SVMean -in subjs\_normalized\_ficvf.txt -outMean mean\_ficvf.nii.gz -outStd meanstd\_ficvf.nii.gz*

* Merge the aligned data into a 4D dataset (*fslmerge*)
* Put everything in folders named in agreement with TBSS - copy files to 'stat' folder in TBSS folder, including FA files: all\_NODDI\_xxx, mean\_NODDI\_xxx, all\_FA, mean\_FA, mean\_FA\_mask, mean\_FA\_skeleton, mean\_FA\_skeleton\_mask, mean\_FA\_skeleton\_mask\_dst
1. **TBSS - Pre-statistics for NODDI data**
* Skeletonise data using existing skeleton and distance map (adapted from tbss\_non\_FA)

*fsl5.0-tbss\_skeleton -i mean\_FA -p 0.2 mean\_FA\_skeleton\_mask\_dst /usr/share/fsl/5.0/data/standard/LowerCingulum\_1mm all\_FA all\_ficvf\_skeletonised -a all\_ficvf -s mean\_FA\_skeleton*

1. **Permutation based statistics**
* Voxel-wise statistics on entire skeleton (both DTI-derived and NODDI data)

*fsl5.0-randomise -i all\_ficvf\_skeletonised -m mean\_FA\_skeleton\_mask -d design.mat -t design.con -n 5000 --T2 -o stats\_ficvf\_tbss*